

# UACM

Universidad Autónoma  
de la Ciudad de México

*Nada humano me es ajeno*

COLEGIO DE CIENCIA Y TECNOLOGÍA

LICENCIATURA EN INGENIERÍA EN SISTEMAS ELECTRÓNICOS  
INDUSTRIALES

## **Monitoreo y Automatización Inalámbrica de un Sistema Tinaco-Cisterna**

TESIS

QUE PARA OPTAR POR EL TÍTULO DE  
LICENCIADOS EN INGENIERÍA EN SISTEMAS ELECTRÓNICOS  
INDUSTRIALES

PRESENTA:

**EDGAR MOISES AGUILAR CURIEL**

**JOSÉ ALBERTO LEÓN LÓPEZ**

DIRECTOR

**M. EN I. CHRISTIAN AGUSTÍN VÁZQUEZ VILLANUEVA**

Ciudad de México, noviembre de 2023.

## SISTEMA BIBLIOTECARIO DE INFORMACIÓN Y DOCUMENTACIÓN



## UNIVERSIDAD AUTÓNOMA DE LA CIUDAD DE MÉXICO COORDINACIÓN ACADÉMICA

### RESTRICCIONES DE USO PARA LAS TESIS DIGITALES

### DERECHOS RESERVADOS<sup>©</sup>

La presente obra y cada uno de sus elementos está protegido por la Ley Federal del Derecho de Autor; por la Ley de la Universidad Autónoma de la Ciudad de México, así como lo dispuesto por el Estatuto General Orgánico de la Universidad Autónoma de la Ciudad de México; del mismo modo por lo establecido en el Acuerdo por el cual se aprueba la Norma mediante la que se Modifican, Adicionan y Derogan Diversas Disposiciones del Estatuto Orgánico de la Universidad de la Ciudad de México, aprobado por el Consejo de Gobierno el 29 de enero de 2002, con el objeto de definir las atribuciones de las diferentes unidades que forman la estructura de la Universidad Autónoma de la Ciudad de México como organismo público autónomo y lo establecido en el Reglamento de Titulación de la Universidad Autónoma de la Ciudad de México.

Por lo que el uso de su contenido, así como cada una de las partes que lo integran y que están bajo la tutela de la Ley Federal de Derecho de Autor, obliga a quien haga uso de la presente obra a considerar que solo lo realizará si es para fines educativos, académicos, de investigación o informativos y se compromete a citar esta fuente, así como a su autor ó autores. Por lo tanto, queda prohibida su reproducción total o parcial y cualquier uso diferente a los ya mencionados, los cuales serán reclamados por el titular de los derechos y sancionados conforme a la legislación aplicable.



# UACM

Universidad Autónoma  
de la Ciudad de México

*Nada humano me es ajeno*

COLEGIO DE CIENCIA Y TECNOLOGÍA

LICENCIATURA EN INGENIERÍA EN SISTEMAS ELECTRÓNICOS  
INDUSTRIALES

## **Monitoreo y Automatización Inalámbrica de un Sistema Tinaco-Cisterna**

TESIS

QUE PARA OPTAR POR EL TÍTULO DE  
LICENCIADOS EN INGENIERÍA EN SISTEMAS ELECTRÓNICOS  
INDUSTRIALES

PRESENTA:

**EDGAR MOISES AGUILAR CURIEL**  
**JOSÉ ALBERTO LEÓN LÓPEZ**

DIRECTOR

**M. EN I. CHRISTIAN AGUSTÍN VÁZQUEZ VILLANUEVA**

Ciudad de México, noviembre de 2023.



# **Monitoreo y automatización inalámbrica de un sistema Tinaco-Cisterna**

**Edgar Moises Aguilar Curiel**

**José Alberto León López**

**noviembre 2023**

**Agradecimientos.**

En agradecimiento a toda la comunidad de la UACM, que me rodeó a lo largo de estos años, e hizo posible mi paso por esta gran institución; y que ahora culmina con este último trabajo de titulación. Un sueño que se veía muy lejano, pero ahora también gracias a la familia, amigos, y al gran grupo de docentes de la UACM, que me asesoraron en gran manera positiva siempre, hoy les doy las gracias, y que la vida siempre les sonría.

**Pon en manos del Señor todas tus obras y tus proyectos se cumplirán.**

[Proverbios 16:3]Biblia Nueva Versión Internacional 2022

Edgar Moises Aguilar Curiel

José Alberto León López

## Índice de Tablas y Figuras

1.1: Tabla N°1. Productos actuales en el mercado. ....	5
1.2: Figura N°1. Diagrama de una Instalación típica de un sistema tinaco-cisterna. Donde se aprecia el alambrado desde los depósitos de agua hacia el tablero de control. (Tomada de [5]).....	7
2.1: Figura N°2. Diagrama general del prototipo. (Tomada de [5]) .....	8
2.1: Figura N°3. Comunicación ESP-NOW entre las tarjetas del prototipo. (Tomada de [6]).....	9
2.2: Figura N°4. Tarjeta de desarrollo ESP32 de 30 pines. (Tomada de [7]) .....	10
2.2: Figura N°5. Modulo relevador electrónico de 2 canales. (Tomada de [8]) .....	10
2.2: Figura N°6. Tarjeta de desarrollo ESP8266 de 30 pines. (Tomada de [9]).....	11
2.2: Figura N°7. Cubo de vidrio tipo precera de 29cm x 22cm x 18cm. ....	11
2.2: Figura N°8. Electroválvula de 12 v CC a 1 ampere. (Tomada de [10]) .....	11
2.2: Figura N°9. Shield o escudo utilizado para ESP8266. (Tomada de [11]).....	12
2.2: Figura N°10. Shield o escudo utilizado para ESP32. (Tomada de [12]) .....	12
2.2: Figura N°11. Electrobomba de agua eléctrica de 24v CC. (Tomada de [13]).....	13
2.2: Figura N°12. Sensor XKC-Y25-V. (Tomada de [14]).....	13
2.2: Figura N°13. Sensores de nivel flotador horizontal. (Tomada de [15]) .....	13
2.2: Figura N°14. Contactor de CA doméstico 2P 2NO 40A 220v/230v contactor modular. (Tomada de [16])..	14
2.3: Figura N°15. Circuito lector de niveles con leds indicadores de altura. (Tomada de [17]) .....	14
2.4: Tabla N°2. Materiales del prototipo. ....	15
2.4: Figura N°16. Foto del prototipo de sistema tinaco-cisterna. ....	16
3.1: Tabla N°3. Seudocódigo maestro-esclavo utilizando con protocolo ESP-NOW. (Tomada de [18]).....	18
3.2: Figura N°17. Posibles comunicaciones entre tarjetas Espressif. (Tomada de [19]).....	20
3.2: Figura N°18. Comunicación bidireccional. (Tomada de [20]) .....	20
3.2: Figura N°19. Comunicación multidireccional. (Tomada de [21]).....	20
3.3: Figura N°20. Diagrama esquemático, emisor y receptor, electrobomba y control. (Tomado de [22]) .....	21
3.3: Figura N°21. Diagrama esquemático, emisor-receptor de circuito de control remoto. (Tomada de [22]).....	22
3.3: Figura N°22. Diagrama esquemático receptor de circuito de control. (Tomada de [22]) .....	23
3.3: Figura N°23. Foto tomada al prototipo en prueba modo manual. ....	24
3.3: Figura N°24. Foto tomada al prototipo en prueba modo automático. ....	25
3.4: Figura N°25. Diagrama de flujo, (programación de la electrobomba en ESP8266) .....	26

3.4: Figura N°26. Diagrama esquemático del sistema de electrobomba. (Tomada de [22]) .....	27
3.5: Figura N°27. Diagrama de flujo (programación tarjeta electroválvula) .....	29
3.5: Figura N°28. Diagrama esquemático para el Sistema de válvula y de control. (Tomada de [22]) .....	30
3.6: Figura N°29. Diagrama de flujo del control remoto, para el sistema tinaco-cisterna.....	31
3.6: Figura N°30. Diagrama esquemático del control de monitoreo con pantalla OLED. (Tomada de [22]) .....	32
4.1: Figura N°31. Medición de señal de alcance desde laboratorio C-005 a el patio de lado oriente en la UACM SLT. ....	33
4.1: Figura N°32. Foto de la altura a las pruebas del dispositivo. ....	34
4.2: Figura N°33. Diagrama esquemático, Control Remoto con pantalla OLED para el sistema tinaco-cisterna. (Tomada de[22]) .....	35
4.2: Figura N°34. Foto del Control Remoto con pantalla OLED. ....	36
4.3: Figura N°35a. Foto al Funcionamiento del prototipo. ....	37
4.3: Figura N°35b. Foto a los sensores externos, funcionando en el prototipo. ....	38
4.3: Figura N°36a. Foto al Encendido y pagado de la bomba y electroválvula.....	38
4.3: Figura N°36b. Foto al Encendido y pagado de electrobomba y electroválvula. ....	39
4.4: Tabla N°4. Ejemplo de presupuesto para la instalación en una casa de una planta baja. ....	40
4.4: Tabla N°5. Ejemplo de presupuesto para la instalación en un edificio de 5 plantas. ....	41
4.4: Tabla N°6. Ejemplo de montos totales de instalación y comparación. ....	41
Figura N°37. Sistema solar propuesto. (Tomada de [23]) .....	44
Figura N°38. Foto al funcionamiento de los sensores externos.....	44
Figura N°39: Fotografía de la Activación y desactivación de relevador utilizando un canal.....	45
Figura N°40. Fotografía del funcionamiento del detector de niveles. ....	45
Figura N°41. Sensor ultrasónico resistente al agua. (Tomada de [24]) .....	46
Figura N°42. Diagrama para el volumen de un depósito de agua. (Tomada de [25]) .....	46

## Índice General

Introducción.....	1
Motivación y justificación.....	1
Objetivos generales y particulares.....	1
Metodología.....	2
Organización del trabajo.....	3
Capítulo 1. Antecedentes.....	4
1.1 Productos actuales en mercado.....	4
1.2 Diagrama tipo de conexión de electrobombas.....	7
Capítulo 2. Desarrollo del hardware sistema y construcción del prototipo.....	8
2.1 Diagrama general del prototipo.....	8
2.2 Materiales de control, sensores y actuadores usados en la construcción del prototipo.....	10
2.3. Circuito adicional para medir nivel.....	14
2.4. Lista de costo de materiales y prototipo.....	15
Capítulo 3. Desarrollo del software.....	17
3.1. Protocolo de comunicación ESP-NOW.....	17
3.2. Estructuras de conexiones usando el protocolo ESP32, ESP8266.....	20
3.3. Diagramas generales del circuito electrónico del sistema.....	21
3.4. Programación de la tarjeta conectada a la bomba eléctrica.....	26
3.5. Programación de la tarjeta conectada a la válvula eléctrica.....	27
3.5: Figura N°28. El sistema de válvula y de control.....	30
3.6. Programación para el control remoto del sistema tinaco-cisterna.....	30
Capítulo 4. Resultados.....	33
4.1. Prueba inalámbrica alcance de la comunicación ESP-NOW.....	33
4.2. Prueba manual.....	35
4.3. Prueba automática.....	36
4.4. Simulación del monto en pesos mexicanos del prototipo para una casa o edificio.....	40
Capítulo 5. Conclusiones.....	42
Anexos.....	44
Referencias.....	64

## Introducción.

### *Motivación y justificación.*

A manera de introducción para este trabajo hablaremos del diseño de un prototipo de monitoreo para un sistema tinaco y cisterna. Un sistema tinaco-cisterna, es muy importante para el almacenamiento de agua principalmente, en la Ciudad de México, ya que existen diversos diseños de edificaciones destinadas a viviendas, desde hogares individuales hasta edificios con múltiples departamentos. En estas estructuras, es común que una cisterna sea capaz de abastecer de agua tanto a una vivienda como a un gran número de departamentos. Por lo tanto, resulta fundamental monitorear el proceso de llenado del tinaco y de la cisterna para prevenir insuficiencias en el suministro de agua a los departamentos, y así asegurar que se conserve una reserva suficiente para un servicio adecuado durante el tiempo requerido por los usuarios.

En una vivienda, el suministro de agua es de vital importancia, especialmente para aquellos que trabajan en entornos donde la higiene es primordial, como la industria, el campo de la medicina y otros lugares con altos riesgos sanitarios. Además, cada vez es más común encontrar micro invernaderos en balcones, muros y otros espacios donde se cultivan vegetales comestibles para uso doméstico. Durante horarios especiales, como en el inicio del día, el suministro de agua no puede fallar para garantizar el aseo personal. Además, durante remodelaciones en una vivienda, el nivel de agua en los tanques puede ser reducido dependiendo de la complejidad de los trabajos realizados.

Ahora con las nuevas tecnologías y el uso moderado del **Wi-Fi**, donde ya existen una gran variedad de microcontroladores los cuales permiten diversas comunicaciones como de tipo unidireccional, bidireccional y multidireccional, las cuales nos permiten comunicar desde una tarjeta de desarrollo o múltiples de ellas con otra u otras.

Estamos muy interesados en el monitoreo y control del nivel de agua en los tanques de almacenamiento, principalmente tinacos porque hemos notado que en la ciudad de México existe un gran desperdicio innecesario de agua, pudiéndonos permitir con el desarrollo de nuestro primer prototipo, la posibilidad de retener el agua necesaria, detectar fugas, proponer sistemas de alerta manual o automatizados para llevar a cabo sin problemas un consumo de agua más responsable y eficiente.

Nuestro trabajo consiste en este primer prototipo para el **Monitoreo y Automatización Inalámbrica de un Sistema Tinaco-Cisterna**. En un modelo real en viviendas o unidades habitacionales.

### *Objetivos generales y particulares.*

#### Objetivos generales

Utilizar las tarjetas de microcontroladores **ESP32**, **ESP8266** y el protocolo **ESP-NOW** de comunicación para:

- Realizar una rutina programación en lenguaje de programación “c” que lleva a cabo el monitoreo de un sistema tinaco-cisterna que ayude a medir los niveles de agua y emita alertas en caso de estado vacío o con suficiente agua en el tanque tinaco.
- Implementar un sistema de control automático para el llenado del tinaco y el cierre de la electroválvula de vaciado, con activación inalámbrica de la electrobomba.
- Activar y desactivar una electrobomba de agua y una electroválvula de manera manual mediante un control inalámbrico

## Objetivos particulares

- Eliminar el cableado que va de los electrodos a la superficie alta o baja del sistema de bombeo de un edificio o casa convencional o de construcción promedio, utilizando **Wi-Fi** y microcontroladores ya existentes en el mercado.
- Diseñar un sistema de tinaco-cisterna que sea compatible con múltiples edificios y casas para detectar los niveles de agua de manera precisa y confiable por medio de una pantalla **OLED** de 1" x 0.7" suficiente para leer el estado del tinaco.
- Desarrollar un Sistema de Tinaco-Cisterna de fácil instalación en edificios y viviendas, con componentes y conexiones inalámbricas que simplifiquen el proceso para el usuario. La instalación de este equipo no debe rebasar las 24 horas de lo contrario es un caso de difícil acceso.
- Establecer la capacidad de monitorear el sistema de tinaco-cisterna de forma remota, permitiendo el acceso y control a una distancia de hasta 100 metros.

Estos objetivos generales y particulares buscan implementar un sistema eficiente, versátil y de fácil uso para el control y monitoreo del nivel de agua en un sistema de tinaco-cisterna, aprovechando las capacidades de las tarjetas **ESP32** y **ESP8266**, así como el protocolo **ESP-NOW** para la comunicación inalámbrica. El objetivo final es brindar una solución integral que facilite la gestión del agua y optimice su uso en edificios y viviendas.

### *Metodología.*

Para el desarrollo de este proyecto, utilizamos técnicas de programación de microcontroladores en este caso en el lenguaje de programación "c". Empleamos el software IDE Arduino en PC para realizar la compilación de diversos programas relacionados con un sistema de tinaco-cisterna. El software IDE Arduino permite descargas suficientes para hacer múltiples tareas con microcontroladores **Espressif**.

La metodología seguida para el diseño y desarrollo del sistema electrónico se resume en los siguientes pasos:

- Diseño del sistema de medición de nivel de agua y control de la electrobomba eléctrica contando con agua en el tinaco el mayor tiempo posible, cuando no exista consumo simplemente se detiene el sistema.
- Programación en lenguaje "c" utilizando el IDE Arduino, sin hacer uso de bibliotecas de **Wi-Fi**. Se implementa una rutina de llenado y vaciado del tinaco, detectando su altura del agua. Para el vaciado del tinaco, se utiliza una electroválvula en este proyecto de manera interactiva y con fines de control automático.
- Utilización de bibliotecas **Wi-Fi** en lenguaje "c" para **ESP32** y **ESP8266**, junto con el protocolo **ESP-NOW**, descargas y configuraciones correspondientes al trabajo empleado. En este caso se utiliza una rutina en el software de configuración permitiendo emitir a los controladores utilizados.

Es importante destacar que, al utilizar un sistema inalámbrico, se omitió el cableado convencional que normalmente se realiza para conectar los electrodos de censado vacío y lleno en los dos tanques, en un edificio de 4 pisos. Esto evita tener que reponer el cable en las cajas asignadas durante el mantenimiento. En algunos casos, el cableado debe pasar por donde se encuentra el cableado general de los 20 departamentos, que suele ser de 127 Volts CA y puede transportar un alto potencial eléctrico, al omitir este cableado, se reduce el riesgo de daños a la instalación eléctrica en caso de perturbaciones o fallas. Aunque las conexiones de este tipo suelen estar bien conectadas y no presentar problemas en el día a día, en algunos casos ha pasado mucho tiempo desde su instalación original y el aislante puede haberse deteriorado.

Con esta metodología, buscamos desarrollar un sistema eficiente y seguro para el control y monitoreo del nivel de agua en un tinaco-cisterna, utilizando tecnología inalámbrica y aprovechando las plataformas para desarrollar el hardware necesario.

### ***Organización del trabajo.***

El trabajo se organizará de esta manera en los siguientes capítulos:

En el capítulo 1, abordaremos el tema con productos existentes en el mercado, veremos sus ventajas y desventajas, se analizarán algunos productos disponibles relacionados a un sistema de bombeo y control inalámbrico y, se presentarán las características, ventajas y desventajas de estos productos, con el objetivo de comparar su aplicabilidad en este proyecto.

En el capítulo 2, sobre la construcción y el funcionamiento del prototipo, se detallará la elaboración del prototipo desarrollado para este trabajo final. Donde se irá describiendo el cómo se construyó este sistema, cuáles son los dispositivos que lo componen y cómo se simulan. El sistema tinaco-cisterna, se explicará el funcionamiento general del prototipo y cómo interactúan los diferentes dispositivos electrónicos. Así como materiales, el prototipo construido y su esquema de conexiones.

Para el capítulo 3 ya es donde se describe la programación de las tarjetas **ESP32** y la comunicación inalámbrica con el protocolo **ESP-NOW**; se abordará la programación de las tarjetas de desarrollo **ESP32** para establecer la comunicación inalámbrica utilizando el protocolo **ESP-NOW**. Se proporcionará también una introducción sobre cómo funciona este protocolo y se describirá el diagrama de flujo o pseudocódigo de cada uno de los programas puestos en marcha en los microcontroladores, y el **Wi-Fi** que está integrado en el mismo microcontrolador, también se muestran las diferentes posibilidades de comunicación entre tarjetas **ESP32** y **ESP8266**.

En este Capítulo 4, se presentarán los mayores resultados obtenidos del prototipo, el cual fue hecho dentro del laboratorio de potencia de la UACM SLT. Ahora bien los resultados obtenidos a partir de las pruebas realizadas en dicho laboratorio de potencia, se mostrarán datos, mediciones así como las observaciones más relevantes obtenidas durante las pruebas hechas al prototipo, evaluando su desempeño y funcionalidad. Las pruebas del sistema tinaco-cisterna en modo manual, automático y una tabla de una descripción del costo aproximado de la instalación de este trabajo final, contando todos sus componentes simulados en una casa de una planta y un edificio de cinco plantas.

Por último para las conclusiones de este trabajo final; se expondrán las conclusiones generales del trabajo ya realizado. Se recapitularán los objetivos alcanzados, se discutirán los resultados obtenidos y se ofrecerán reflexiones sobre la viabilidad y aplicabilidad del prototipo desarrollado. También se mencionarán posibles mejoras o líneas de trabajo a futuro que podrían abordarse a partir de este gran proyecto. Con esta estructura, se busca presentar de manera organizada y detallada cada aspecto relevante del proyecto, desde el análisis de productos existentes hasta los resultados obtenidos y las conclusiones.

## Capítulo 1. Antecedentes.

### *1.1 Productos actuales en mercado.*

Hay muchos edificios en la Ciudad de México que carecen de personal confiable, para el mantenimiento y técnicos especialistas en electricidad, para actividades cotidianas, como vigilar el uso adecuado de agua, conocer la altura del agua disponible en los tinacos o cisternas del inmueble, y sin tener la experiencia para subir a puntos elevados; algo que solamente realizaría un técnico especializado y equipado con las medidas de seguridad necesarias.

En vista de estas condiciones a las que los usuarios están expuestos de manera constante, el monitoreo y control automático e inalámbrico se convierte en una herramienta útil y conveniente para aquellos que manejan dispositivos con acceso a internet. Aunque la mayoría de los edificios constan de aproximadamente cuatro pisos y 20 departamentos, también existen edificios más grandes en la Ciudad de México que carecen de servicios tecnológicos de uso cotidiano. Esto se debe a diversas razones, como la alta rotación de inquilinos en edificios de alquiler, la presencia de un único personal de mantenimiento para múltiples tareas en los edificios o la falta de conocimiento sobre las ventajas de contar con un sistema automatizado e inalámbrico.





Además, en algunos edificios, solo la mitad de los departamentos disponibles están habitados, y en otros casos, las razones simplemente se desconocen, ya que se abren y cierran los depósitos de almacenamiento de agua y las cisternas sin considerar el riesgo que implica su exposición al interior de estos. Es importante resaltar que los desperfectos en la instalación de tinacos y cisternas surgen debido a la falta de consultar a un técnico y a su vez estos suelen ser más costosos y no es recomendable operar sin las instrucciones técnicas adecuadas.

En vista de estas necesidades y desafíos, surge la motivación para desarrollar un sistema de monitoreo y control automático e inalámbrico para el nivel de agua en los depósitos elevados y cisternas. Este sistema busca garantizar un suministro adecuado de agua en viviendas y edificios, evitando insuficiencias y desperdicios innecesarios. Además, proporciona comodidad y seguridad a los usuarios al permitirles acceder a la información del nivel de agua y controlar el proceso de llenado de manera remota, a través de dispositivos con conexión a internet.

Una de las ventajas de utilizar un sistema inalámbrico es la facilidad de instalación, evitando la necesidad de cableado complejo y la intervención en el interior de los departamentos o registros eléctricos. Además, al tener un alcance de comunicación de hasta 100 metros, es adecuado para edificios de la Ciudad de México, donde los sistemas de canalización y cableados deben cubrir distancias considerables.

En cuanto al consumo eléctrico, se tiene en cuenta la capacidad de la bomba para agua utilizada y seleccionar el relevador adecuado para soportar el amperaje necesario, al igual que un contactor. Es importante destacar que las instalaciones eléctricas varían dependiendo del tamaño de la vivienda o edificio, y se deben considerar las necesidades específicas de cada caso.

En la siguiente tabla se muestran los diferentes productos que existen en el mercado para el control de un sistema cisterna-tinaco y sistema de monitoreo de nivel. El diseño de este trabajo está basado en eliminar los más que se pueda de cableado eléctrico, de un tanque tinaco y cisterna.

Producto.	Características.	Imagen.	MXN\$	Ventajas.	Desventajas.
Marca innovatec Modelo S- CIMBOM- 471/750. (Tomada de [1])	De dos módulos y trabajan como maestro esclavo, según el fabricante, la altura máxima es de 15 metros, se alimenta con celda fotovoltaica de baja potencia.		\$2,072	Buen alcance de señal Wi-Fi cuenta con una fotocelda solar y una batería de baja potencia. Y funciona a una altura de 4 plantas edificadas.	Electrobombas de 127Volts CA hasta 1HP. Altura máxima de 4 plazas para edificio.
Marca Racom CBST Wi-Fi. (Tomada de [2])	Este sistema es alámbrico y funciona con Wi-Fi la altura estimada por el cable disponible, regula el abasto de agua en el tinaco.		\$950	Controlador de tinaco o cisterna en otro caso ambos tanques se controlan. Indica si existe agua en el tanque de operación.	Requiere cableado para conexión de electrodos entre la cisterna y el tinaco, los electrodos en su esquema de conexiones aparecen en párelo de cisterna y tinaco.
KONLEN Modelo Water Level Controller. (Tomada de [3])	El funcionamiento requiere 12volts CC para funcionar y encender y apagar la electrobomba, cuenta con wifi y aplicación para su control.		\$1200	Cuenta con Wi-Fi y aplicación de fácil instalación y alimentación de baja potencia.	Es alámbrico pues debe conectarse a en el lugar del tinaco y hasta la cisterna 10A de consumo eléctrico es para electrobombas pequeñas.
Electro nivel inalámbrico yosuna. (Tomada de [4])	Función como maestro-esclavo en sus componentes, con Wi-Fi y aplicación, ideal para un edificio de hasta 8 pisos, controla el nivel del tinaco.		\$2,127	De fácil instalación y funciona a 12Volts CC el sistema es inalámbrico.	Cuenta con un flotados como sensor para el tinaco y para la cisterna.

1.1: Tabla N°1. Productos actuales en el mercado.

[1] (Electronivel Control Inalámbrico De Bombeo P/Tinaco 1100t En Venta En Tlaxcala Tlaxcala Por Sólo \$ 2,095.00 - OCompra.com México, n.d.)[2] (Racom, n.d.)

[3] (Sensor De Nivel De Agua Para Tinaco Wifi, n.d.)[4] (Yonusa, n.d.-b)

Estos sistemas actuales en mercado cuentan con un alcance de hasta 8 pisos en comunicación entre los más altos en sus emisores y receptores para activar y desactivar la bomba eléctrica.

Si bien estos sistemas son de control inalámbrico, tienen algunas limitaciones para poder operar en bombas eléctricas de 1 ½ HP que es común en la ciudad de México en edificios de 5 plantas, los sistemas expuestos en tabla N°1 están disponibles en la red de internet coincidiendo con el tema de sistema tinaco-cisterna inalámbrico o control vía **Wi-Fi**, manejan un amperaje máximo de 20 amperes o menos siendo relativamente de baja resistencia eléctrica para bombas de agua en la CDMX.

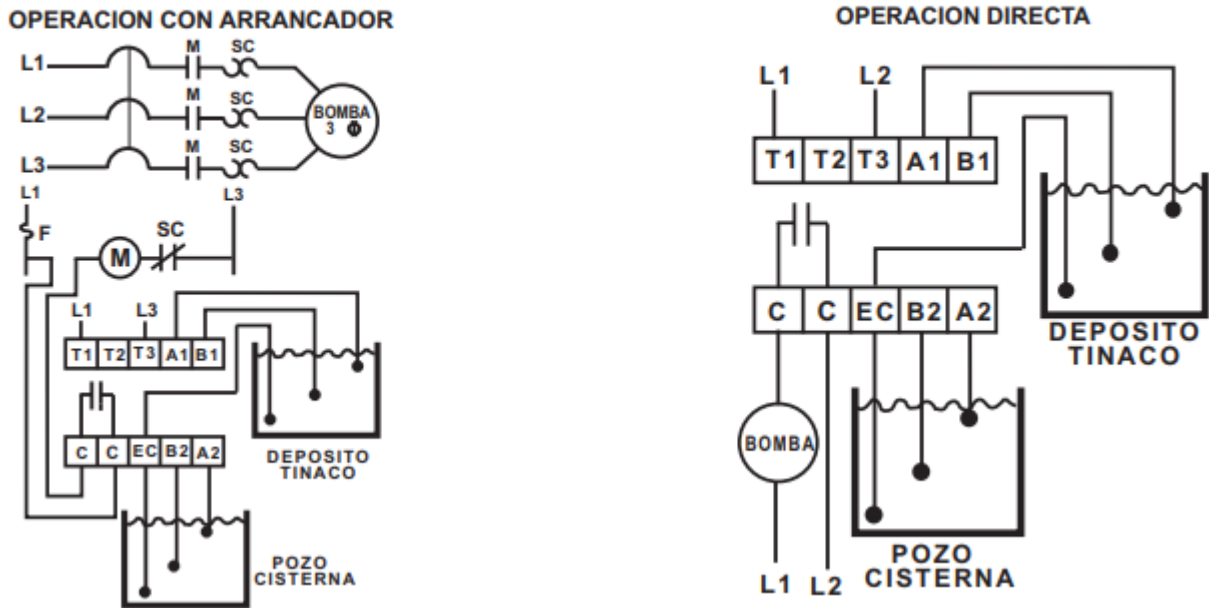
El sistema que se desarrollo tiene un alcance hasta de 100 m que puede servir para un edificio de 20 pisos, tiene dos formas de operación manual y automática, y tiene un control remoto para monitorear el nivel del tinaco y controlar de manera manual inalámbrica la electroválvula en el tinaco y la electrobomba en la cisterna.

En la prueba de señal de comunicación de **ESP32** a **ESP8266** resulto exitosa a 20 metros de altura en el edificio "C" de UACM, SLT en comunicación unidireccional. La comunicación aplicada al prototipo es bidireccional.

## 1.2 Diagrama tipo de conexión de electrobombas.

Estos son dos ejemplos de diagramas de conexión de operación de un sistema tinaco-cisterna, en la Figura N°1, se aprecia una instalación típica de electro niveles con sus electrodos alambrados al circuito electrónico de control

### Ejemplos típicos de conexión



1.2: Figura N°1. Diagrama de una Instalación típica de un sistema tinaco-cisterna. Donde se aprecia el alambrado desde los depósitos de agua hacia el tablero de control. (Tomada de [5])

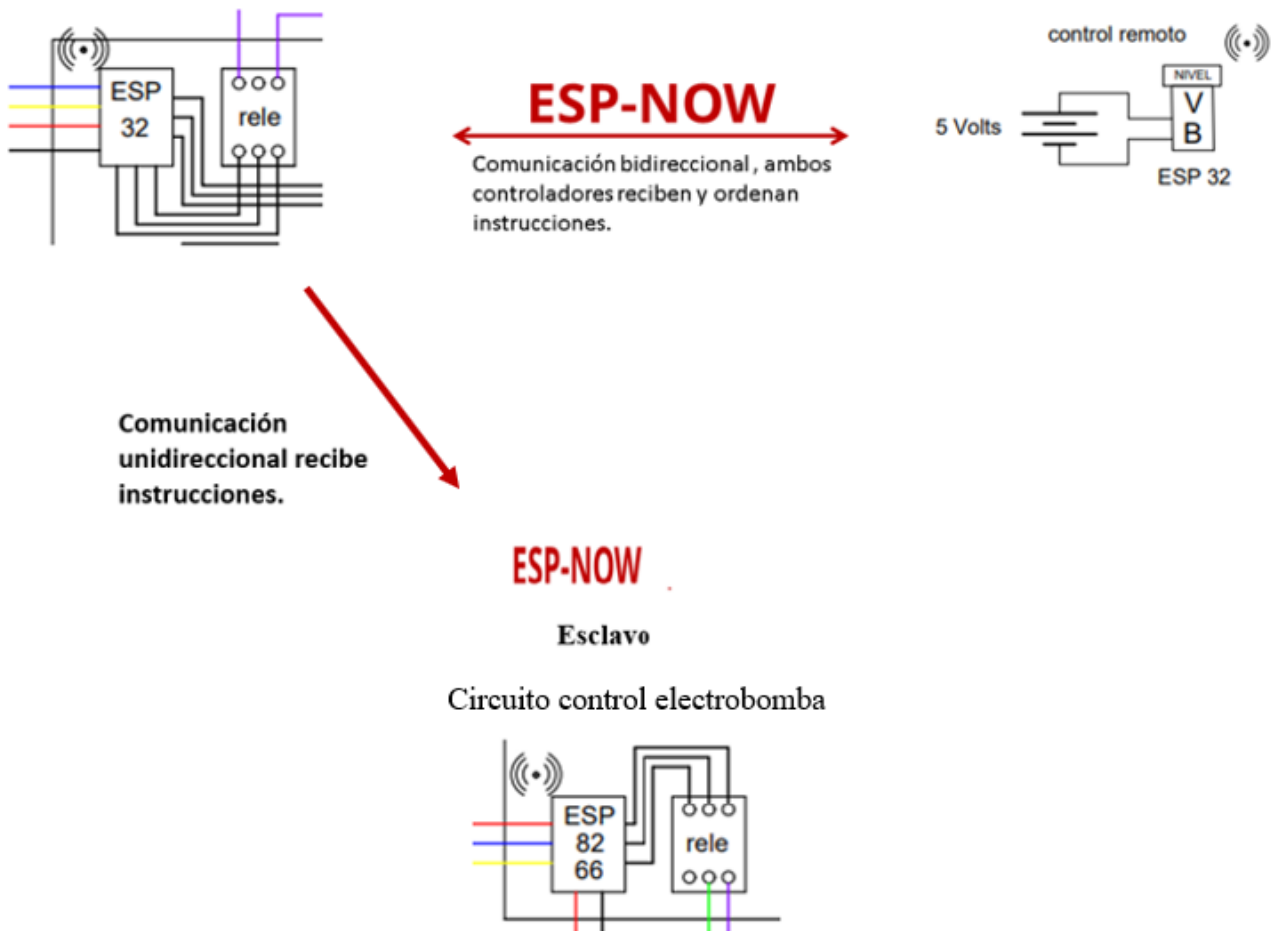
También en esta instalación típica podemos observar a más detalle la colocación del alambrado para los electro niveles para el tinaco-cisterna con cableado en los dos tanques de agua. Regularmente la distancia entre un tinaco y una cisterna en una casa habitación o un edificio de 5 plantas es 8 hasta 20 metros en promedio, aunque puede variar a más distancia.



En este diagrama general de la figura N°2 de nuestro prototipo se muestra lo siguiente:

1. El depósito elevado simula el tinaco de una instalación hidráulica y posee dos sensores de altura bajo y alto.
2. El depósito inferior simula la cisterna de una instalación hidráulica y posee dos sensores de altura bajo y alto.
3. El controlador **ESP32** controla con un relevador el encendido y apagado de la electroválvula, como maestro envía una orden de encender y apagar la electrobomba al esclavo **ESP8266**, receptor y emisor con el control remoto con señal **Wi-Fi**.
4. El controlador **ESP8266** es esclavo de **ESP32** numero 3 recibe ordenes de apagar y encender la electrobomba por medio de un relevador de corriente utilizando su wifi integrado.
5. El controlador **ESP32** funciona como control de mando a distancia utilizando su **Wi-Fi** como emisor y receptor con el controlador de la electroválvula **ESP32** correspondiente a número 3.

El funcionamiento de las tarjetas con protocolo ESP-NOW es el siguiente:



2.1: Figura N°3. Comunicación ESP-NOW entre las tarjetas del prototipo. (Tomada de [6])

La comunicación entre la tarjeta que controla la electroválvula y el control remoto es bidireccional, ya que la tarjeta va a enviar al control remoto los niveles de altura y el control remoto envía la señal para activar la válvula y la electrobomba; donde la comunicación que existe entre la tarjeta de la electroválvula y la tarjeta conectada a la electrobomba es unidireccional según el protocolo **ESP-NOW**, ya que la tarjeta de la válvula envía la activación para la electrobomba en configuración maestro-esclavo modo automático. En el capítulo 4.1 se describe un ejemplo de protocolo **ESP-NOW** aplicado a maestro esclavo. Ver Tabla N°3.

## 2.2 Materiales de control, sensores y actuadores usados en la construcción del prototipo.



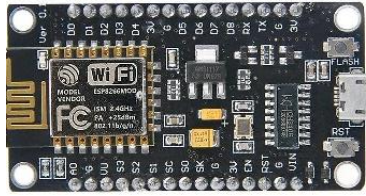
2.2: Figura N°4. Tarjeta de desarrollo ESP32 de 30 pines. (Tomada de [7])

Tarjeta de desarrollo **ESP32** cuenta con sistema wifi integrado de 2,4ghz a baja potencia además de un sistema bluetooth para comunicarse entre sí por medio de protocolo **ESP-NOW** proporcionado por el mismo fabricante **Espressif**. y es amigable con IDE Arduino y lenguaje “c”. **Espressif Systems** es el fabricante de los módulos Esp32, integra un potente microcontrolador con arquitectura de 32 bits, conectividad **Wi-Fi** y **Bluetooth**.



2.2: Figura N°5. Modulo relevador electrónico de 2 canales. (Tomada de [8])

Los relevadores se encargan de cerrar un cable alta potencia desde baja potencia en forma de interruptor en este caso de 5v CC a 120 v CA. Al encender un relevador abre una compuerta y cierra otra con hilo común a las dos salidas en tres bornes. Se utilizan para controlar fuentes de voltaje en corriente alterna desde corriente directa, como se aprecia en la figura N°5.



2.2: Figura N°6. Tarjeta de desarrollo ESP8266 de 30 pines. (Tomada de [9])

Esta tarjeta electrónica cuenta con **Wi-Fi** de 2.4 GHz de baja potencia integrado y se comunica con otros microcontroladores **Espressif** o en tarjetas diseñadas con estos microcontroladores. Software desde IDE Arduino.



2.2: Figura N°7. Cubo de vidrio tipo precera de 29cm x 22cm x 18cm.

- Cubo de vidrio, se utilizaron dos con capacidad para 9 litros .
- Se utilizaron para simular el tinaco y la cisterna en el prototipo.
- Un cubo lleva en una cara de vidrio templado un barreno de 1 pulgada para el desague con una electroválvula.

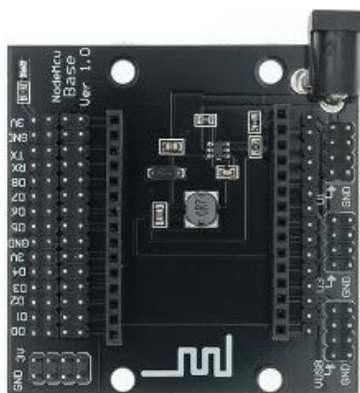


2.2: Figura N°8. Electroválvula de 12 v CC a 1 ampere. (Tomada de [10])

La electroválvula permite, una apertura y cierre al paso de agua, energizando su devanado interno; su instalación no es compleja excepto que, para la baja presión por la altura del tanque utilizado, en este caso se hace una calibración, ya que no se cuenta con la presión suficiente para el flujo de agua durante el desague.

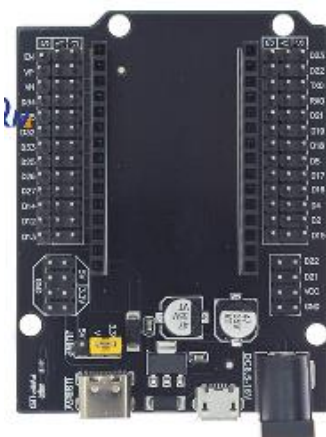
[9] (ESP8266 Wi-Fi MCU | Espressif Systems, n.d.-b)

[10] (Válvula Solenoide 12V DC 1 Pulgada - Aelectronics, n.d.-b)



2.2: Figura N°9. Shield o escudo utilizado para ESP8266. (Tomada de [11])

El diseño de esta tarjeta permite conectar fuentes de alimentación de 5v CC a máximo 2 A. Por otra parte los pines están despejados para varias conexiones tanto de entrada como de salida, puesto que la tarjeta **ESP8266** opera desde 3.3v CC, con este shield o escudo se realizan circuitos electrónicos a 5v CC sin problemas.



2.2: Figura N°10. Shield o escudo utilizado para ESP32. (Tomada de [12])

El diseño de esta tarjeta permite despejar pines de salida y entrada para las tarjetas esp32 de 30 pines, aumentando su potencia de señales de salida. Como recomendación se pueden hacer uso de fuentes de voltaje a 5v CC y a un máximo 2 A, esto es suficiente carga para circuitos y proyectos de baja potencia.

[11] (NodeMCU ESP8266 Base Shield, n.d.-b)

[12] (Base 38 Pines Shield ESP32 ESP32S Wifi Regulador 5V 2A, n.d.-b)



2.2: Figura N°11. Electrobomba de agua eléctrica de 24v CC. (Tomada de [13])

Electroválvula regulable que va desde los 6v CC hasta 24v CC regulando su potencia desde su cargador eléctrico, para este trabajo se utiliza a 12v CC y se activa desde su fuente de 127 v CA.



2.2: Figura N°12. Sensor XKC-Y25-V. (Tomada de [14])

El sensor funciona mandando una señal de la misma intensidad de su fuente hasta 5v CC se activa el led en rojo al momento en que el nivel del agua pasa por el recipiente elegido.



2.2: Figura N°13. Sensores de nivel flotador horizontal. (Tomada de [15])

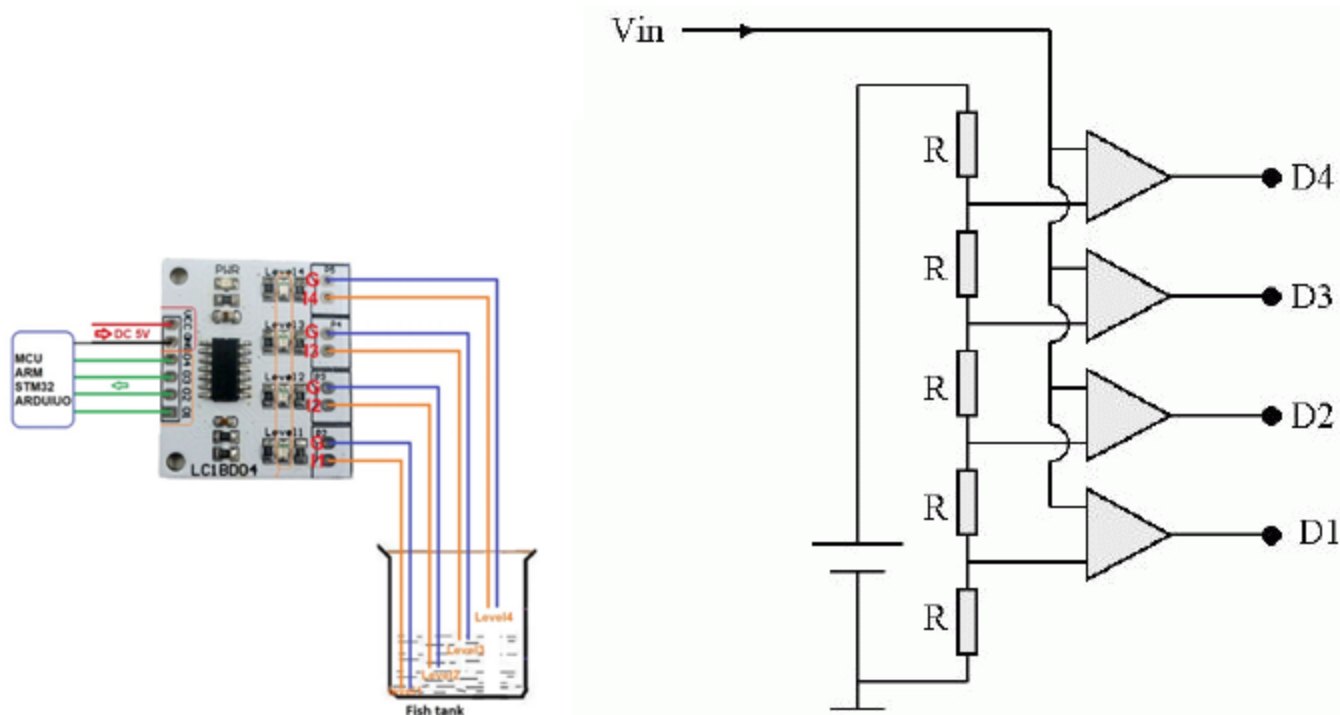
- El aro flotador funciona como interruptor para cerrar sus dos cables moviéndose hacia arriba y hacia abajo sobre el agua.
- Fabricado con PVC hidraulico y facil de instalar.



2.2: Figura N°14. Contactor de CA doméstico 2P 2NO 40A 220v/230v contactor modular. (Tomada de [16])

Montaje en riel DIN de 35 mm, contactor de automatización 50/60HZ, Frecuencia máxima de encendido/apagado 30 veces/h.

### 2.3. Circuito adicional para medir nivel.



2.3: Figura N°15. Circuito lector de niveles con leds indicadores de altura. (Tomada de [17])

Este circuito indica el nivel del tanque ocupado y elegido por su simplicidad, se utiliza a un máximo de 5v CC. en este trabajo se utilizaron 5v CC a 2A en las primeras pruebas en una cisterna de 1,6m de fondo con respuesta óptima. Encendiendo todos sus leds sin problema. Este circuito es anexo para garantizar el nivel sin interrupciones con luces led.

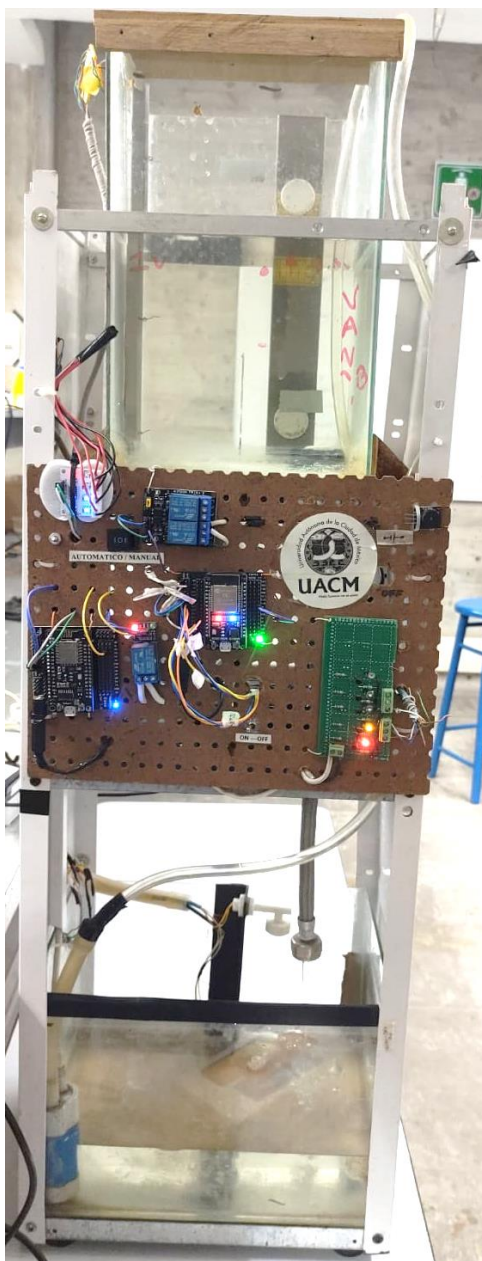
[16] (tameson.es, n.d.) (Contactor De CA Doméstico 2P 2NO 40A 220V/230V Contactor Modular, Montaje En Riel DIN De 35 Mm Contactor De Automatización 50/60HZ, Frecuencia Máxima De Encendido/Apagado 30 Veces/H : Amazon.com.mx: Herramientas Y Mejoras Del Hogar, n.d.-b)[17] (2.82€ 32% De DESCUENTO)Dc 5v 7-12ma 4 Indicador De Nivel De Líquido De Agua Digital Tablero Controlador Sensor Interruptor Módulo Sensor De Nivel De Agua Para Tanque De Peces Torre De Agua Sensor Nivel Agua Indicador Nivel De Agua - Sensores De Flujo - AliExpress, n.d.)

#### 2.4. Lista de costo de materiales y prototipo.

En la siguiente tabla se muestra la lista del material usada en el proyecto:

Cantidad	Componente	Precio Unitario	Total
2	ESP32	\$80	\$160
1	Pantalla OLED Azul 128×64 1.3" I2C SH1106	\$100	\$100
1	Esp8266	\$150	\$150
1	Escudo o Shield para ESP32 5v CC hasta 2	\$275	\$275
1	Escudo o Shield para ESP8266 5v CC hasta 2 A	\$275	\$275
2	Interruptor de tres pines dos canales	\$17.50	\$35
1	Bomba de 24v CC	\$375	\$375
1	Electroválvula de 12v CC	\$105	\$105
2	Cubo de vidrio de 29cm x 22cm x 18cm	\$130	\$260
1	Barreno de 1 pulgada de diámetro	\$80	\$80
2	Módulo relevador de 5v CC	\$35	\$70
2	Circuito integrado ULN2803	\$22.50	\$45
10	Resistencias de 1k ohmios	\$1.50	\$15
10	Resistencias de 330 ohmios	\$1.50	\$15
10	Transistores 2n2222	\$6	\$60
1	Buse de 3 pines	\$50	\$50
4	Borneras de 3 canales	\$15	\$60
2	Borneras de 2 canales	\$15	\$30
2	Placas perforadas de 5cm x 10 cm	\$25	\$50
2	Sensores XKC-Y25	\$170	\$340
2	Electrodos de flotador de 90°	\$47	\$94
10	Leds	\$1.50	\$15
3	Paquete de headers, M-M, H-H y H-M	\$45	\$135
1	Despachador de agua reciclado para estructura	\$150	\$0
1	Contactador de CA doméstico 2P 2NO 40A 220V/230V Contactador modular	\$277	\$277
#	Monto total	\$2,944	\$2,971

2.4: Tabla N°2. Materiales del prototipo.



2.4: Figura N°16. Foto del prototipo de sistema tinaco-cisterna.

Foto tomada en el laboratorio de potencia C-005 planta baja, de la UACM SLT, este prototipo es un diseño típico para simular un sistema tinaco-cisterna, cabe mencionar que este diseño lo construimos, a partir de reciclar la base de un despachador de agua, el sistema de control se compone de relevadore, tarjetas **ESP32** y **ESP8266**, circuitos detectores de niveles, sensores de nivel externos, flotadores y los cuales ya hemos mencionado anteriormente en este mismo capítulo.

## Capítulo 3. Desarrollo del software.

### 3.1. Protocolo de comunicación ESP-NOW.

El protocolo **Espnow** es un protocolo de comunicación inalámbrica basada en una capa de datos definida por la empresa fabricante de tarjetas de desarrollo **Espressif**, la comunicación entre tarjetas es vía **Wi-Fi** o vía **Bluetooth**, cada una de las tarjetas **ESP32** o **ESP8266** posee su propia dirección MAC o MAC Address para su emparejamiento o comunicación.

La distancia esperada de comunicación es de unos 70 metros el tipo de **Wi-Fi** es 802.11 b/g/n según el instituto San palomera de España es la distancia esperada y con obstáculos 40 metros; en pruebas dentro da la UACM SLT se han registrado distancias de 130mts sin obstáculos.

El protocolo **ESP-NOW** permite comunicación de forma unidireccional, bidireccional y multidireccional o de forma emisor-receptor para unidireccional, los dos emiten y reciben órdenes para bidireccional, hasta 20 tarjetas emiten y reciben órdenes.

Existen algunos lenguajes de programación y plataformas para usar tarjetas **ESP32** o **ESP8266** para ver su correcto funcionamiento de estas tarjetas la plataforma de IDE Arduino es la más práctica de utilizar debido a que los códigos de prueba y funcionamiento son sencillos, se descargan de la página oficial de Arduino las bibliotecas o datos necesarios para utilizar el software dependiendo del código y tareas que deseemos crear en lenguaje de programación “c”, donde en la siguiente página web <https://www.arduino.cc/> es posible hacer una donación a beneficio de esta plataforma para descargar su software.

Los proyectos que podemos realizar con tarjetas de la línea **ESP32** y **ESP8266** son como las otras tarjetas que se han utilizado y comercializado años antes a diferencia que contamos con **Wi-Fi** y **Bluetooth**, esto quiere decir que es posible trabajar con muchos sensores y actuadores, componentes de audio, seguridad, etc.

Para utilizar protocolo **ESP-NOW** en IDE Arduino:

Una vez abierto el IDE Arduino en la consola de comandos deberá seleccionar preferencias en introducir;

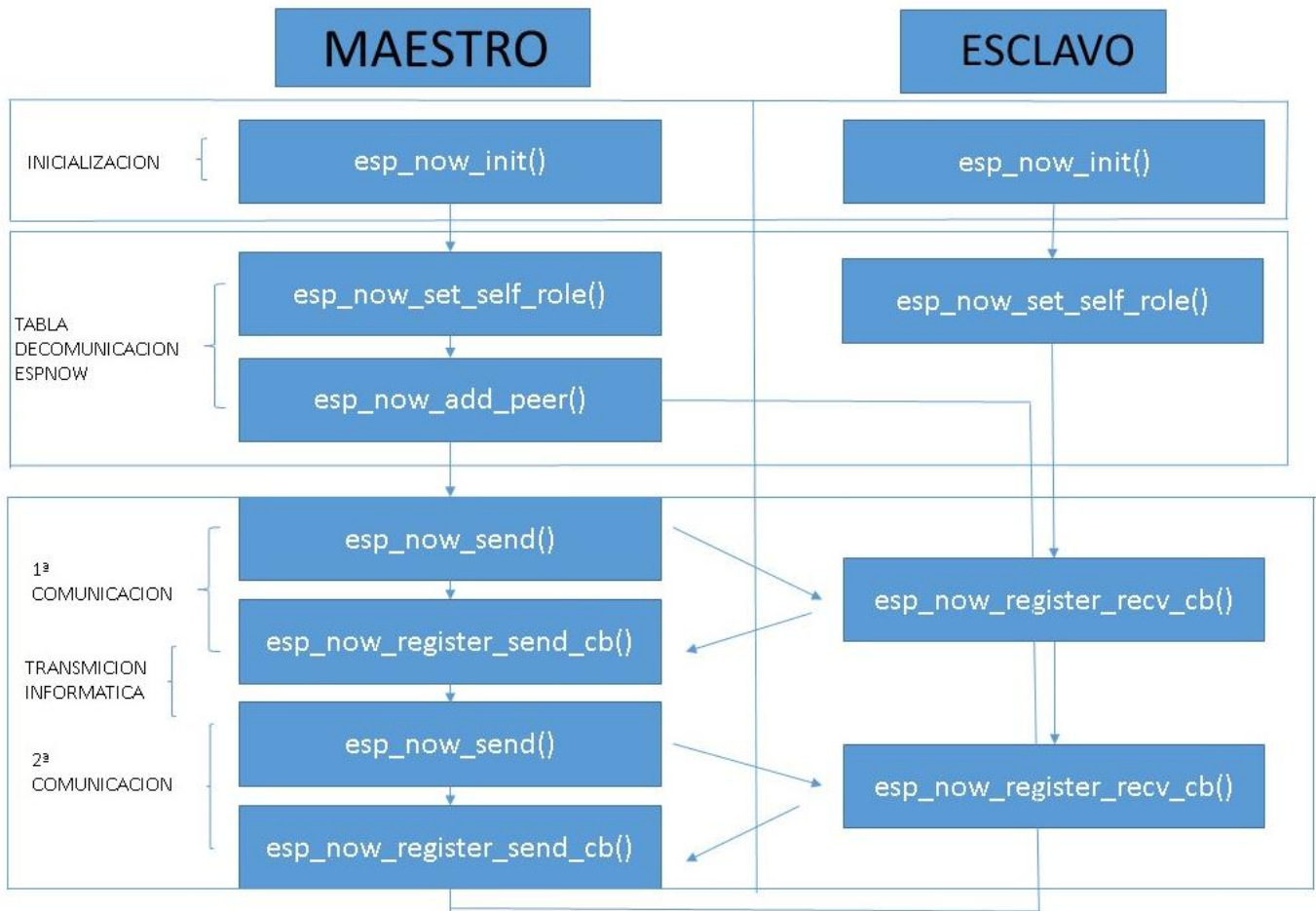
[http://arduino.esp8266.com/stable/package\\_esp8266com\\_index.json](http://arduino.esp8266.com/stable/package_esp8266com_index.json)

[https://dl.espressif.com/dl/package\\_esp32\\_index.json](https://dl.espressif.com/dl/package_esp32_index.json)

Después en la consola de comandos en herramientas seleccionar administrador de bibliotecas y elegir la que se va a utilizar de acuerdo con el trabajo prescrito y bibliotecas generales.

Existen diferentes arquitecturas de la comunicación **ESP-NOW**.

Aplicando protocolo **ESP-NOW** en un programa maestro-esclavo en la tabla N°3 se muestra el seudocódigo de la comunicación entre dos **ESP32**.



3.1: Tabla N°3. Seudocódigo maestro-esclavo utilizando con protocolo ESP-NOW. (Tomada de [18])

Describiendo las líneas de código maestro.

**esp\_now\_init()**

**esp\_now\_set\_self\_role()**

El dispositivo receptor, se configura como un dispositivo esclavo. Registramos la función de devolución de llamada del receptor.

```
esp_now_set_self_role(role) // Algunas opciones para utilizar el comando:
ESP_NOW_ROLE_IDLE = 0,
ESP_NOW_ROLE_CONTROLLER,
ESP_NOW_ROLE_SLAVE,
ESP_NOW_ROLE_COMBO,
ESP_NOW_ROLE_MAX
```

Esta función sirve para inicializar el protocolo **ESP-NOW**. Se utiliza biblioteca **Wi-Fi** en inicio de código, **esp\_now\_add\_peer()**

Esta función permite añadir un cliente utilizando su dirección MAC, lo que lo pondrá en la lista de microcontroladores que recibirán los mensajes. Se emparejan las placas de otro modo no se enlazan **esp\_now\_send()**

Esta función sirve para enviar un mensaje desde la placa maestra a la placa esclava. Se le adjuntará la variable que contiene los datos que queremos enviar, y esta variable podrá ser de cualquier tipo conocido, incluyendo estructuras personalizadas. si enviamos una estructura, dicha estructura está también definida en la placa esclava para que sea capaz de guardar el mensaje en una variable de este tipo.

Estructura de lectores de tanque tinaco.

### **esp\_now\_register\_send\_cb()**

Con esta función registramos una función «callback» que será llamada cada vez que se mande un mensaje en la placa maestra. En la placa esclava no funcionará, principalmente porque esta no enviará mensajes. Esta función debe aceptar los argumentos `const uint8_t *mac_addr` y `esp_now_send_status_t status`, los cuales son la dirección MAC del destinatario, y el estado del envío respectivamente.

Verificamos el envío de datos si es con éxito.

### **esp\_now\_register\_rcv\_cb()**

Esta función registra una función «callback», pero que esta vez será llamada cuando se reciban mensajes. Esta función está destinada a las placas esclavas que son las que van a recibir los mensajes de las placas maestras. En este caso esta función deberá ser capaz de aceptar los argumentos `const uint8_t * mac`, `const uint8_t *incomingData` e `int len`, que son la dirección MAC del emisor, los datos que envía y la longitud de estos respectivamente.

Describiendo las líneas de código esclavo.

### **esp\_now\_init()**

Esta función sirve para inicializar el protocolo **ESP-NOW**. Se utiliza biblioteca **Wi-Fi** en inicio de código.

### **Esp\_now\_register\_rcv\_cb()**

Para registrarse recibiendo a la función de devolución de llamada, es importante que primero llamemos a la función de devolución de llamada de recepción, cuando esta reciba el protocolo **ESP-NOW**.

### 3.2. Estructuras de conexiones usando el protocolo ESP32, ESP8266.

Para la comunicación unidireccional tenemos maestro esclavo, un esclavo varios maestros y un maestro múltiples esclavos.



3.2: Figura N°17. Posibles comunicaciones entre tarjetas Espressif. (Tomada de [19])

Comunicación bidireccional como esclavo/maestro y esclavo/maestro en sus códigos previamente diseñados.



3.2: Figura N°18. Comunicación bidireccional. (Tomada de [20])

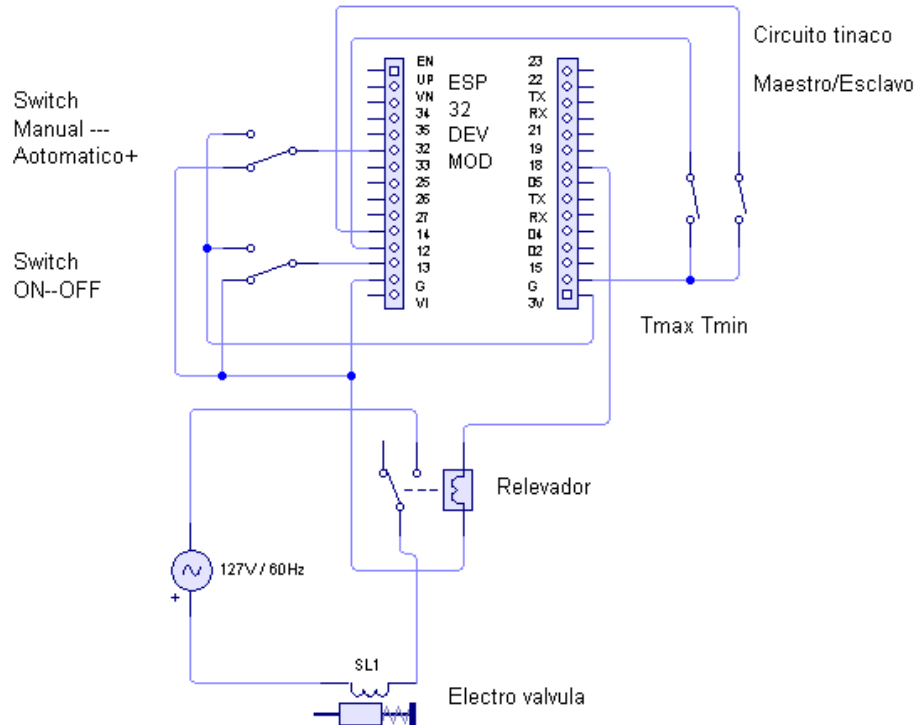
Comunicación multidireccional emisor1/emisor2 en tres múltiples controladores emisor1/emisor2/emisor3/emisorN desde tres dispositivos hasta 20 dispositivos.



3.2: Figura N°19. Comunicación multidireccional. (Tomada de [21])

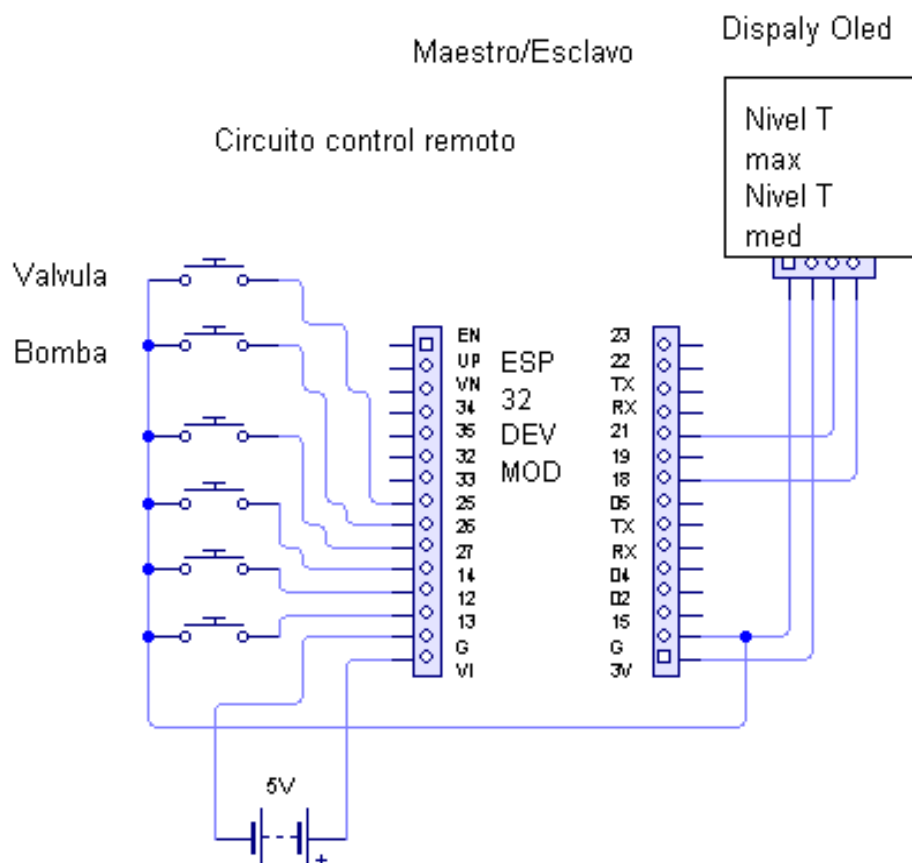
Con estas configuraciones utilizando protocolo **ESP-NOW** es posible hacer múltiples proyectos con sensores y actuadores que nos permitan una vida más cómoda y en la industria mejorar a los sistemas electrónicos.

### 3.3. Diagramas generales del circuito electrónico del sistema.



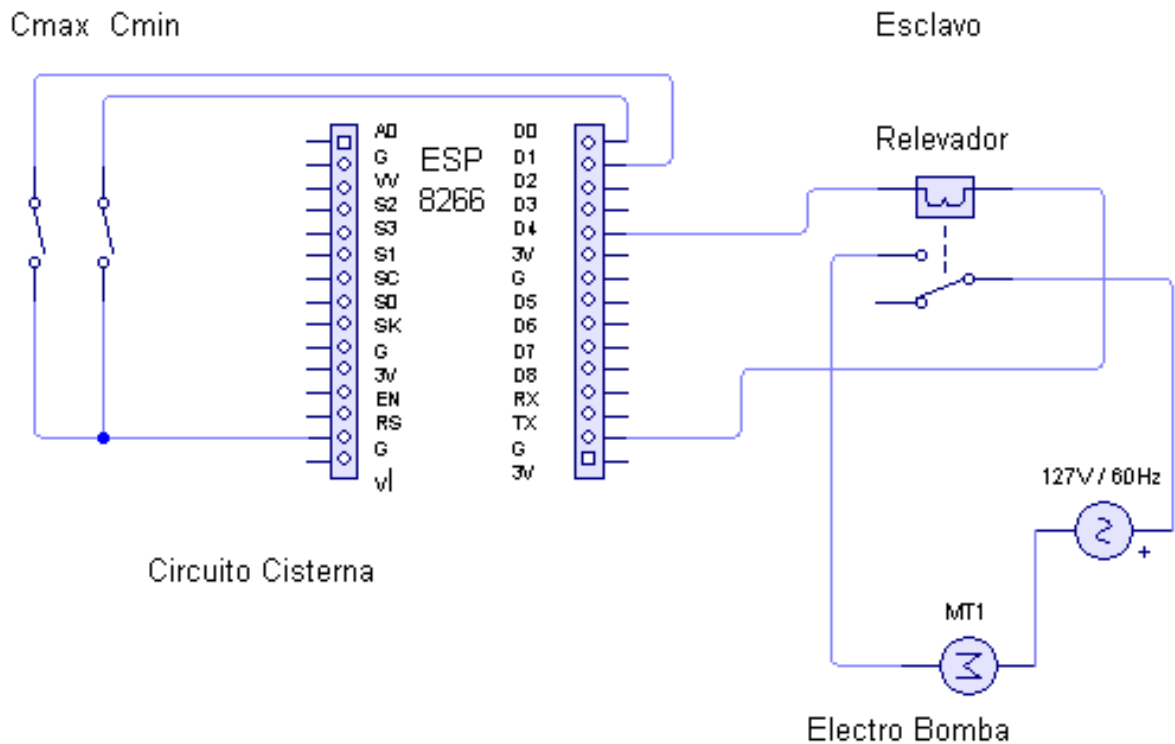
3.3: Figura N°20. Diagrama esquemático, emisor y receptor, electrobomba y control. (Tomado de [22])

Los siguientes circuitos electrónicos que hemos editado en el software PCB Wizard, para el diseño electrónico de PCB y esquemáticos, con la posibilidad de realizar la simulación de dichos circuitos. En este caso con la electroválvula se encarga de encender y apagar el sistema desde un interruptor, que cambia de modo automático a modo manual y con otro interruptor, va a monitorear el llenado del tinaco mediante sensores de nivel sin contacto emitiendo Tmin. y Tmax. que son las etiquetas de llenado máximo y mínimo en el tinaco. Ahora sincronizadamente enciende y apaga la electroválvula por medio del relevador de corriente utilizado de acuerdo con la altura del agua en el tinaco, y emite esta señal de altura del agua a el control remoto donde se despliega el nivel de llenado. El funcionamiento es en forma de maestro-esclavo.



3.3: Figura N°21. Diagrama esquemático, emisor-receptor de circuito de control remoto. (Tomada de [22])

El circuito electrónico del control remoto muestra una pantalla **OLED** y una serie de botones los cuales se utilizan dos para el modo manual que permite encender y a apagar la electrobomba y la electroválvula en cualquier momento del modo manual, la pantalla **OLED** despliega en modo automático la altura del tinaco en el momento actual y su alimentación es de 5 volts CC. El funcionamiento es en forma de maestro-esclavo.



3.3: Figura N°22. Diagrama esquemático receptor de circuito de control. (Tomada de [22])

En el circuito electrónico de la figura N°22, se muestra un controlador **ESP8266**, un relevador que opera a 5 volts DC, una electrobomba y dos sensores de nivel en este caso interruptores en aplicación de flotadores, los cuales reciben la orden de encender y apagar el bombeo de agua en forma de esclavo.

Este sistema está diseñado para que opere de dos modos en el **modo manual** y **modo automático**. (En los subtítulos siguientes se describe cada uno de los tres circuitos por separado.)

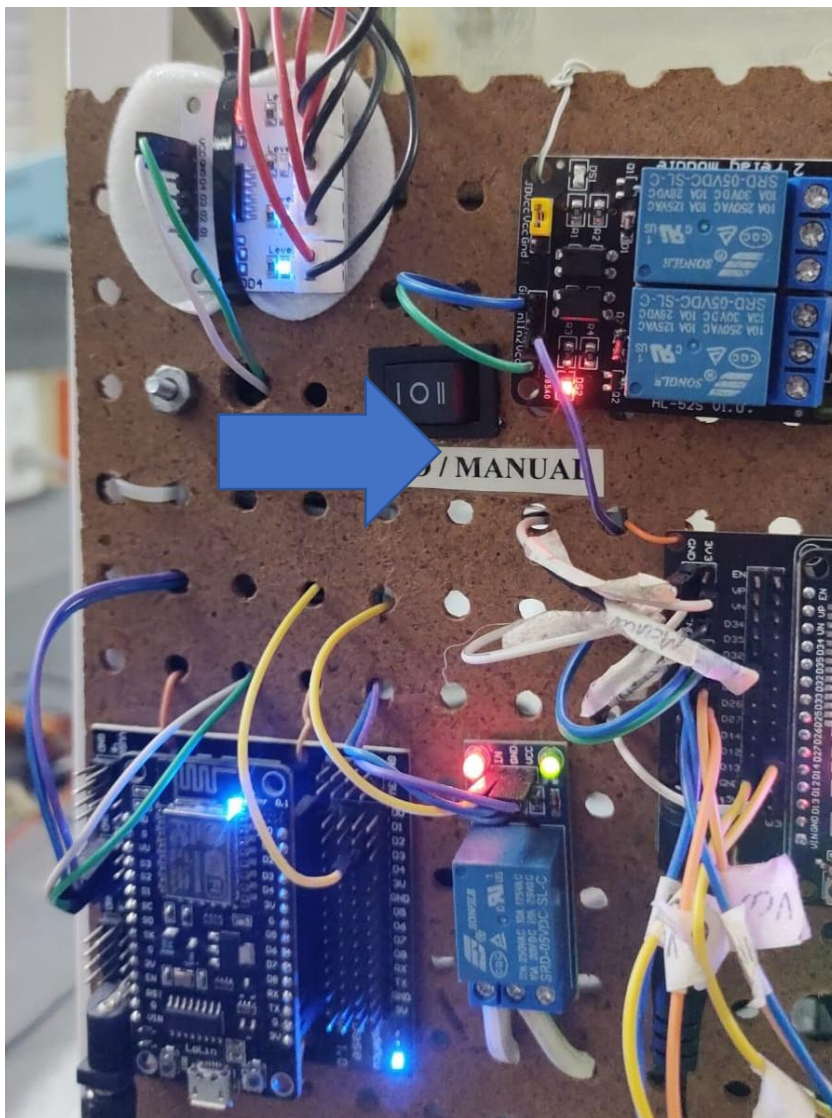
La programación de los tres circuitos electrónicos empieza una vez que la cisterna está llena y el tinaco se encuentra vacío inmediatamente se enciende la electrobomba y los sensores de nivel indican las posiciones del tinaco y la cisterna, el tinaco se llena y se apaga la electrobomba y la electroválvula se enciende hasta que se vacía por completo en modo automático el circuito tinaco se conjuga con el circuito cisterna.

Para modo manual la electrobomba y la electroválvula se pueden encender y apagar en cualquier momento.

### Modo manual.

Consiste en activar un interruptor justo en la dirección como lo indica la flecha azul en la figura N°23, que indica la opción de elegir entre manual y automático una vez seleccionado el modo manual, con el control de mando a distancia, es posible apagar y encender la electrobomba, y activar a la electroválvula, independientemente del estado en el que se encuentre actualmente el sistema ya sea llenando o vaciando.

En el control se despliega el mensaje del estado del sistema indicando tinaco vacío, tinaco medio o tinaco lleno.

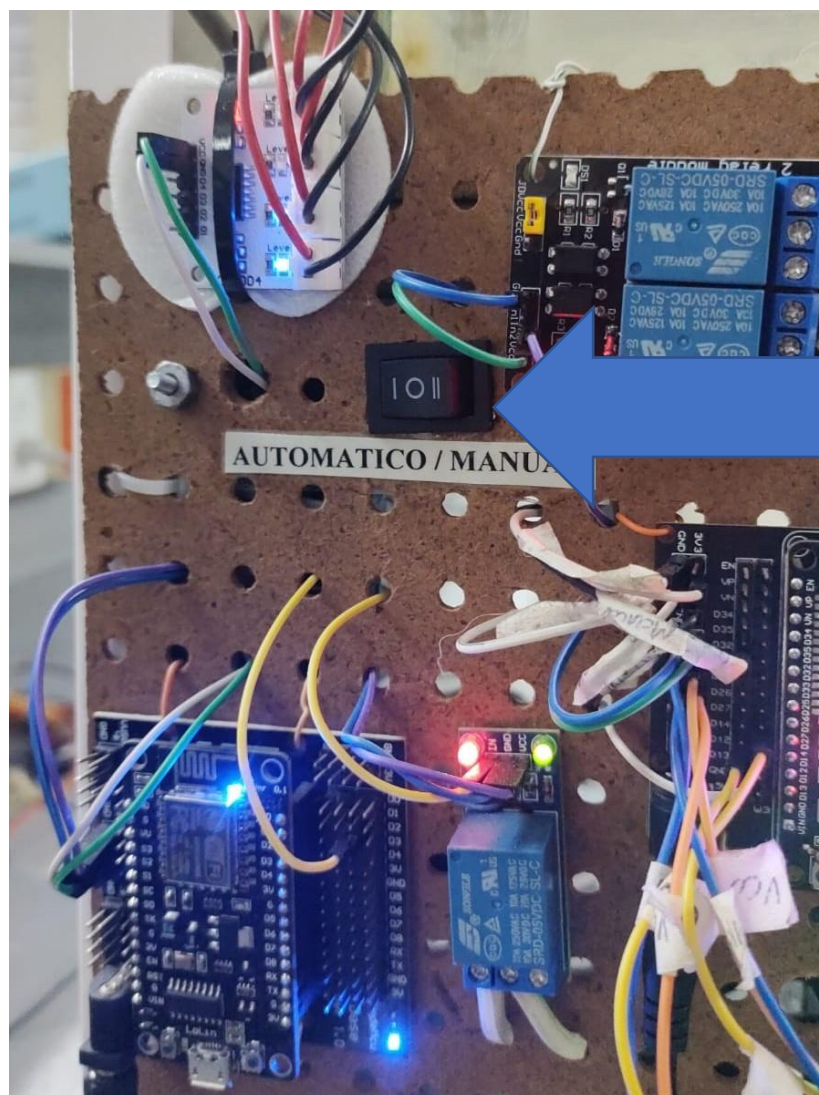


3.3: Figura N°23. Foto tomada al prototipo en prueba modo manual.

### Modo automático.

Ahora hemos puesto a el interruptor de mando de nuestro sistema en modo automático, justo como lo indica la flecha azul de abajo en la figura N°24, donde en esta selección, tenemos dos estados el de Tinaco lleno o Tinaco vacío entonces dependiendo el estado de operación que está actualmente el sistema podrá activar la electrobomba de agua para el llenado del tinaco o la electroválvula para el desagüe del tinaco, por medio de los sensores XKC-Y25-V, que están censando el tinaco.

Una vez que el sistema detecte Tinaco lleno la electrobomba se apagará y cuando el tinaco este vacío se encenderá de nuevo. Para vaciar el tinaco se acciona la electroválvula en Tinaco lleno y se desactiva en Tinaco vacío. La pantalla de nuestro control indicara el estado del tinaco en cualquier estado.

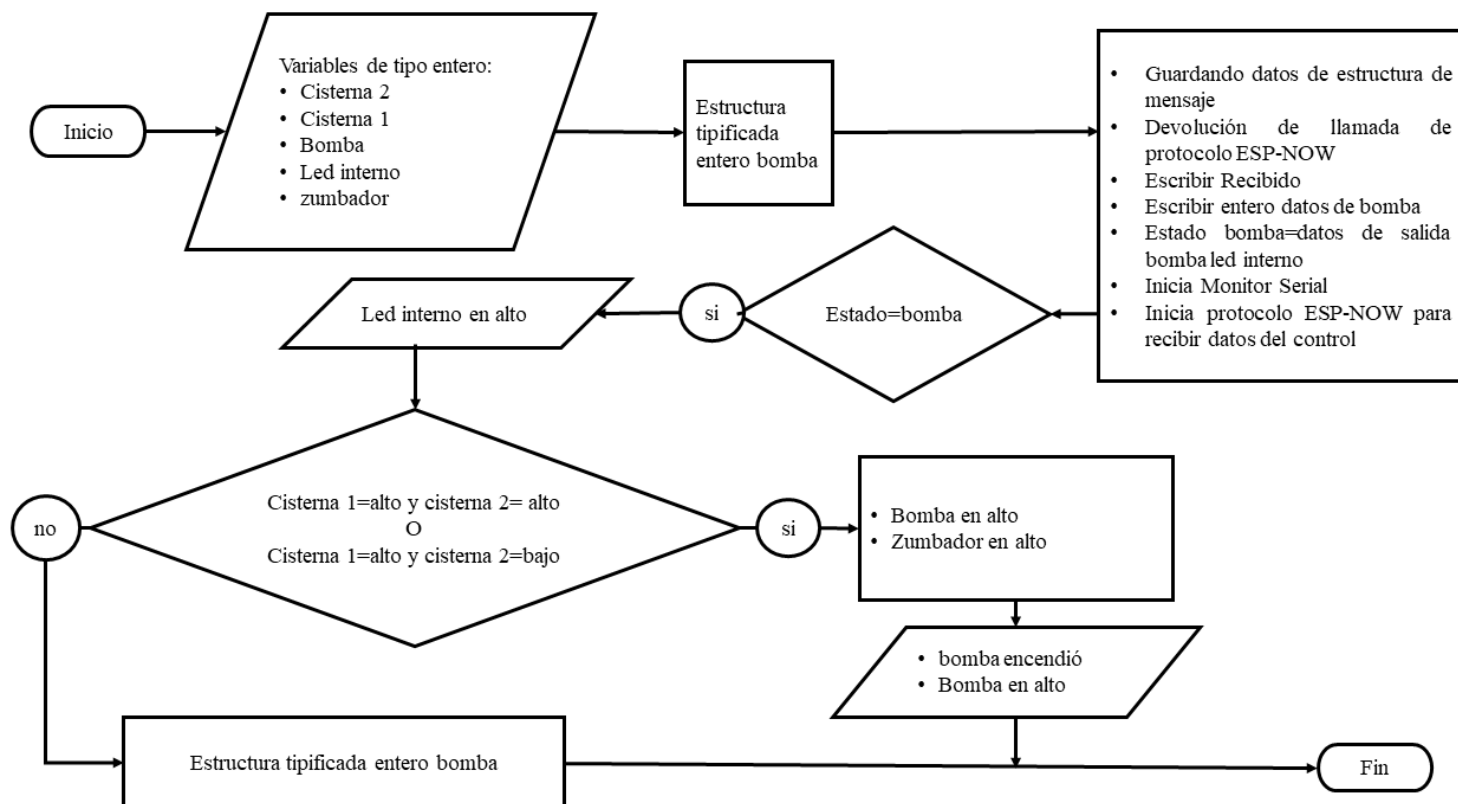


3.3: Figura N°24. Foto tomada al prototipo en prueba modo automático.

### 3.4. Programación de la tarjeta conectada a la bomba eléctrica.

Esta tarjeta **ESP8266** cumple las siguientes funciones.

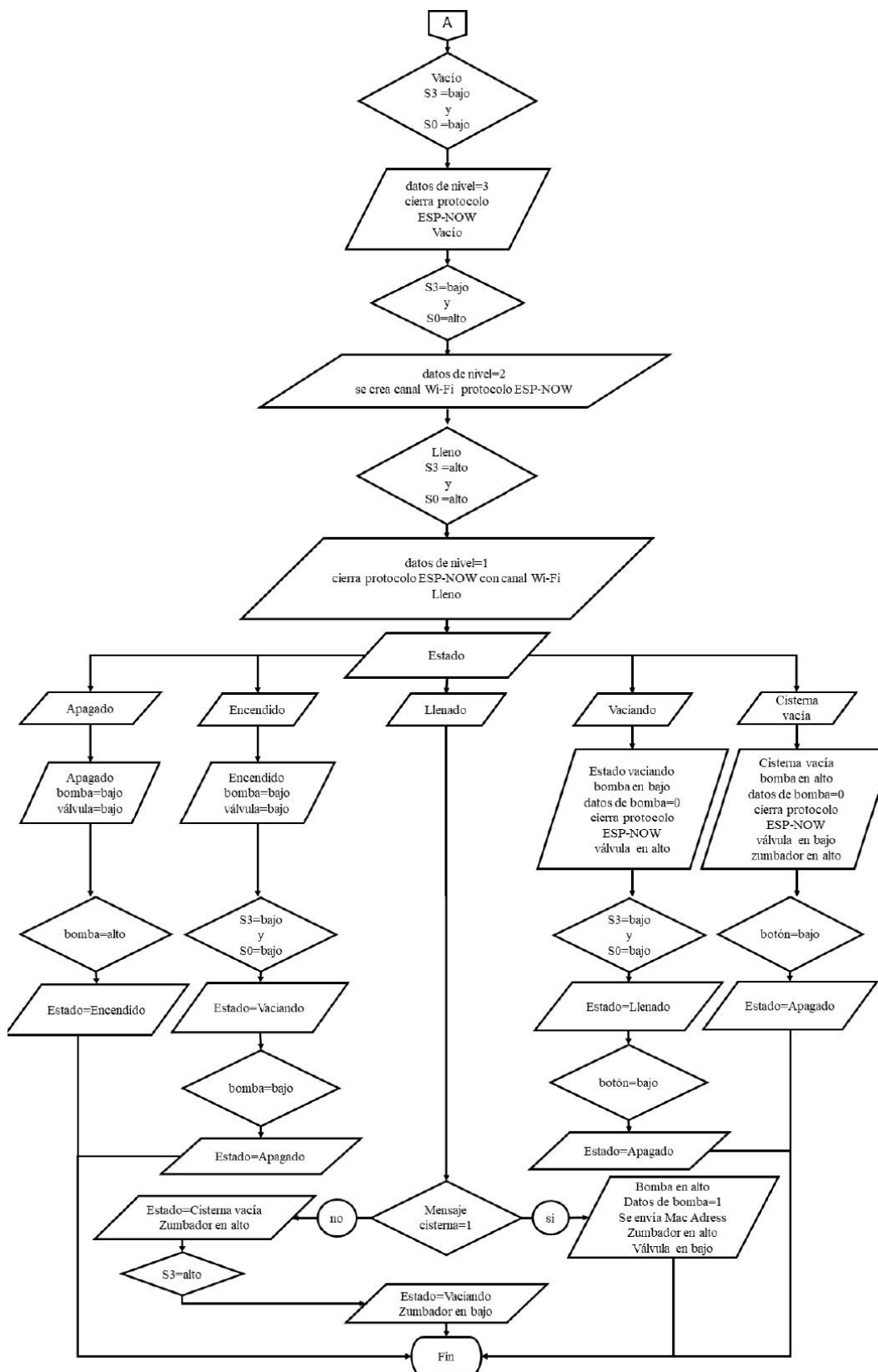
- Debe medir los niveles de altura de la cisterna.
- Activar la electrobomba de manera manual.



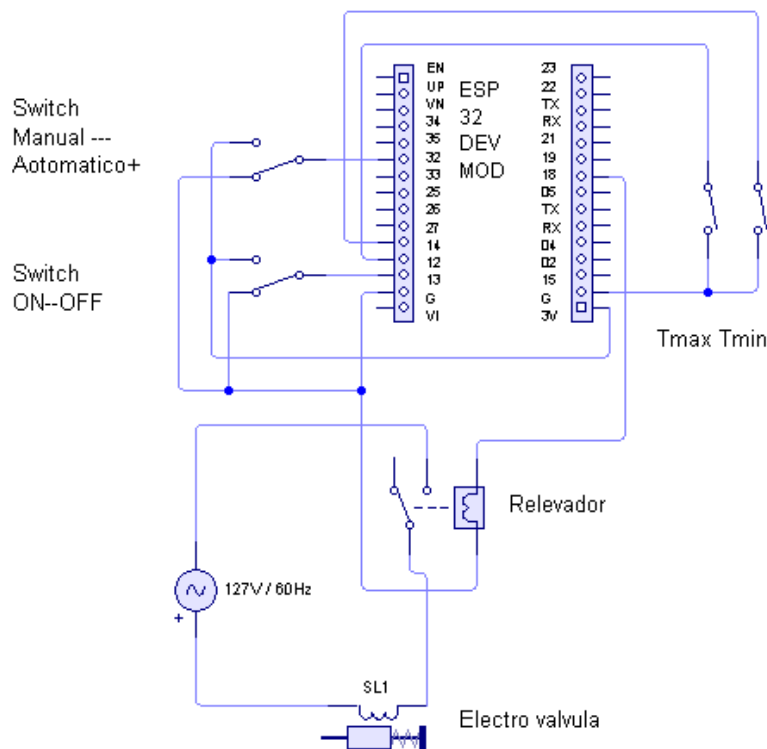
3.4: Figura N°25. Diagrama de flujo, (programación de la electrobomba en ESP8266)







3.5: Figura N°27. Diagrama de flujo (programación tarjeta electroválvula)



3.5: Figura N°28. Diagrama esquemático para el Sistema de válvula y de control. (Tomada de [22])

El circuito muestra Tmax simulando el cierre y censado de nivel alto del tinaco, Tmin cierre de nivel bajo del tinaco que están conectados a una tarjeta de desarrollo **ESP32**, con el código inicializado que hace funcionar la electroválvula.

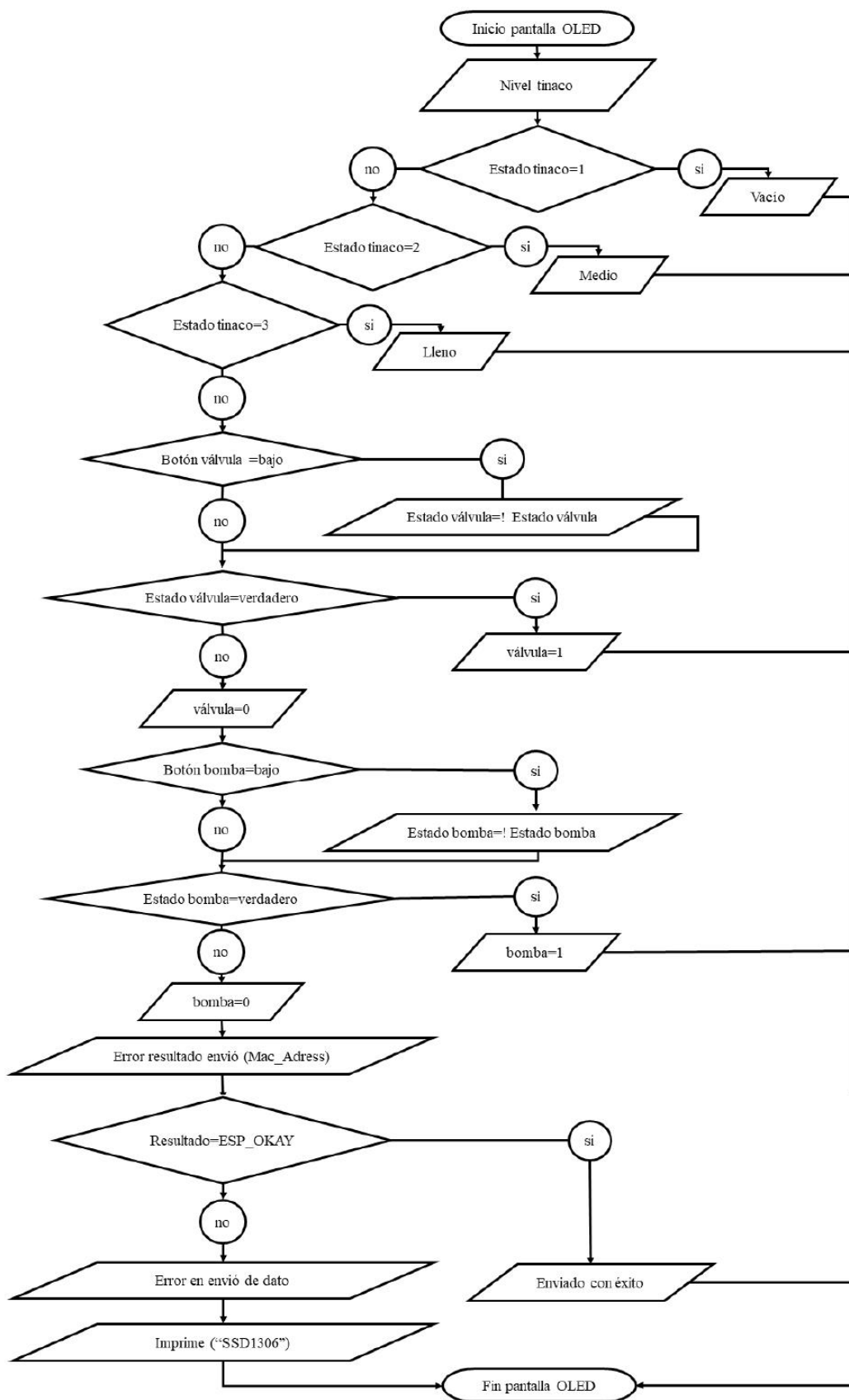
Muestra dos interruptores, uno para encender o apagar el sistema completo y el otro interruptor es para controlar y cambiar de modo automático a modo manual y viceversa.

El relevador y la electroválvula solo están conectados a los 127v CA.

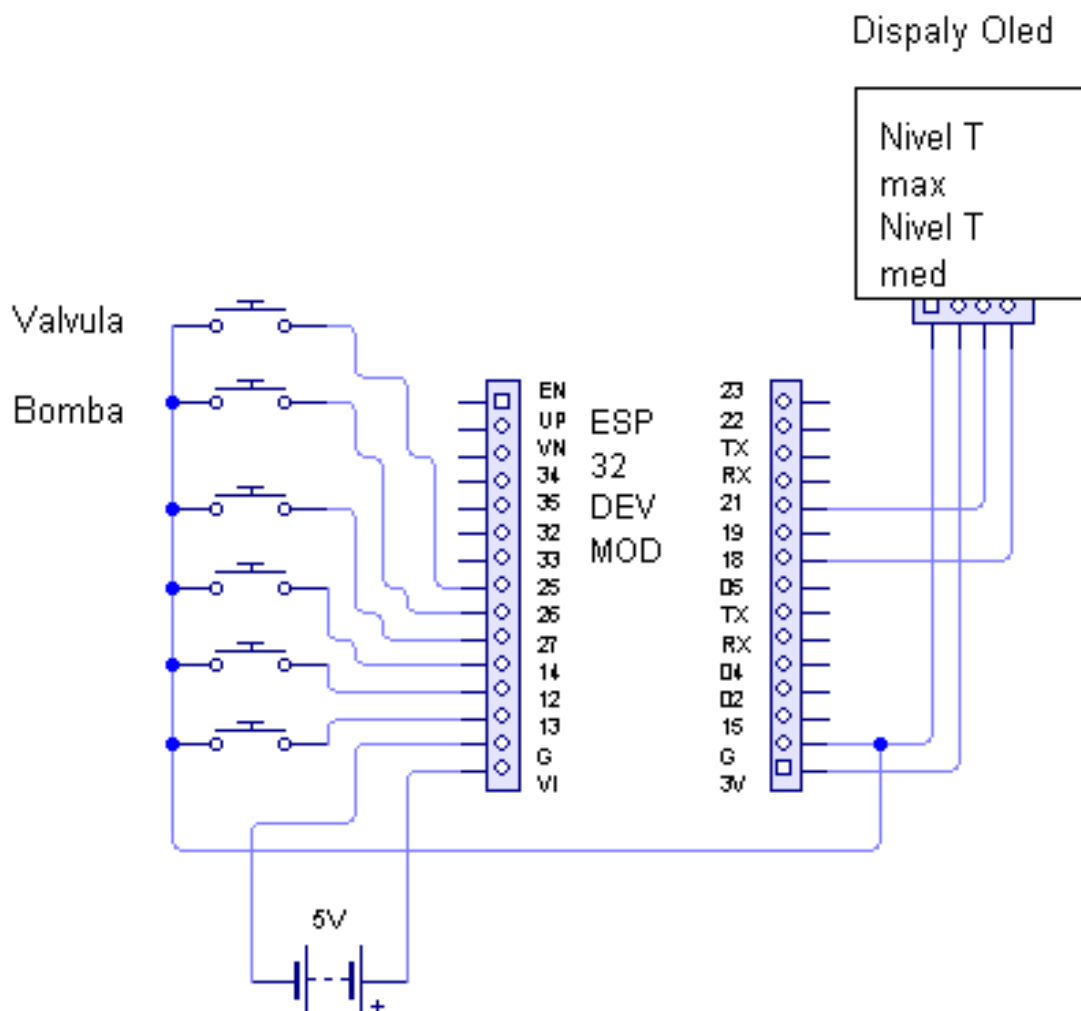
### 3.6. Programación para el control remoto del sistema tinaco-cisterna.

Esta tarjeta **ESP32** cumple las siguientes funciones:

- Recibir estado actual de los niveles y desplegarlos en la pantalla **OLED**.



3.6: Figura N°29. Diagrama de flujo del control remoto, para el sistema tinaco-cisterna.



3.6: Figura N°30. Diagrama esquemático del control de monitoreo con pantalla OLED. (Tomada de [22])

El diagrama muestra que están habilitados dos botones o pulsadores, estos funcionarían en caso de elegir en el sistema por medio de un interruptor, el modo manual, entonces se activa y desactiva la bomba y la electroválvula en el momento elegido por el usuario.

Elijiendo modo automático, solo desplegará en la pantalla **OLED**, el estado del tinaco ya sea tanque lleno o tanque vacío.

El software y su **Wi-Fi** integrado permiten comunicación con las otras dos tarjetas que son las que controlan el tinaco y la cisterna o la electroválvula o la electrobomba eléctrica, utilizando 5v CC para su funcionamiento es suficiente, para activar la rutina previamente programada y por su sencilla construcción y eficacia.

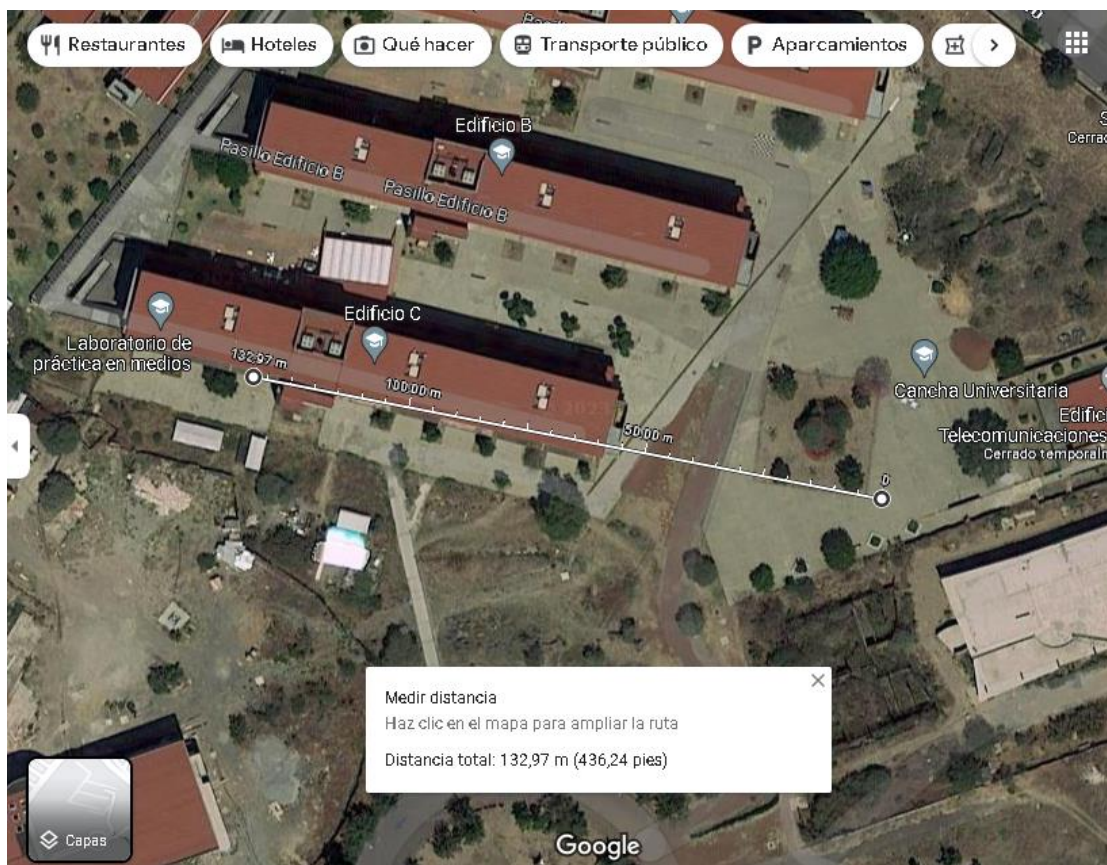
## Capítulo 4. Resultados.

### 4.1. Prueba inalámbrica alcance de la comunicación ESP-NOW.

Monitoreo inalámbrico de nivel de líquido de un tanque elevado y cisterna.

En la actualidad la tecnología wifi nos permite hacer diferentes proyectos, las características mencionadas en metodología de este trabajo, permiten transmisión de datos a más de 60 metros de altura suficiente para utilizarse en un edificio con dimensiones promedio en la ciudad de México.

Con el código de prueba de señal de Esp32 en la biblioteca de IDE Arduino es posible verificar el alcance de esta señal de 2.4GHz a baja potencia con un máximo de 20 canales en su banda ancha, los códigos de enlace los proporciona **Espressif** empresa fabricante de **ESP32** y **ESP8266**, con estas herramientas las pruebas resultaron óptimas. Funcionamiento a 134 metros del edificio “C” salón C-005 al patio cerca del SAIT en la UACM SLT sin problemas de señal de maestro esclavo.



4.1: Figura N°31. Medición de señal de alcance desde laboratorio C-005 a el patio de lado oriente en la UACM SLT.

### Pruebas del proyecto:

- En la prueba de altura con éxito a una altura de 20 metros medido con una cinta desde el balcón del edificio “C” de las escaleras de lado oriente.
- En la prueba de altura con éxito a una altura de 20 metros medido con una cinta desde el balcón del edificio “C” de las escaleras de lado oriente.



4.1: Figura N°32. Foto de la altura a las pruebas del dispositivo.





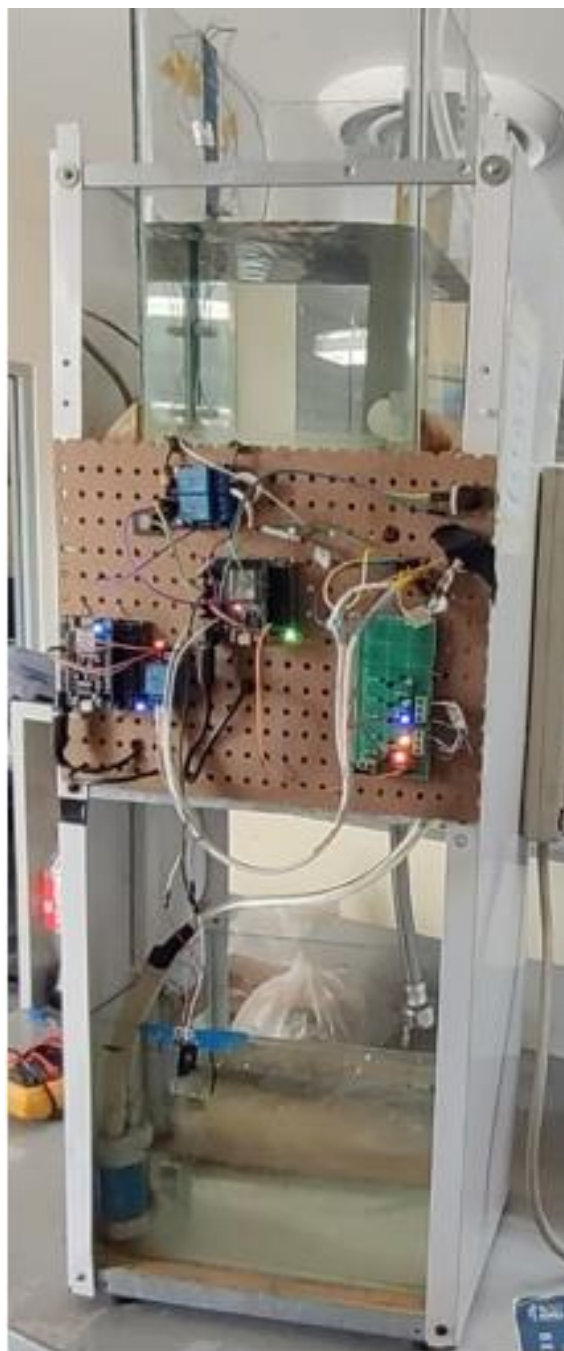
4.2: Figura N°34. Foto del Control Remoto con pantalla OLED.

En la figura N°34, observamos el control de mando a distancia con los interruptores principales, “V” para la válvula y “B” para la activación de la bomba, para prender y apagar el sistema en algún caso de emergencia.

### ***4.3. Prueba automática.***

El modo automático funciona al detectar los niveles de tinaco-cisterna de modo que el consumo de agua lo enciende y apaga la bomba y la electroválvula, donde la válvula eléctrica simula el consumo de agua gastado.

El nivel más bajo de tinaco suponiendo que la cisterna está llena, por abasto de aguas locales en la ciudad de México, el agua que llega a las cisternas no se interrumpe de manera continua solo si es por mantenimiento o reparaciones. A hora bien tomando esto en cuenta, el inicio del sistema tinaco-cisterna en modo automático es necesario considerar que la cisterna este llena siempre solo a si se cumplirá la rutina con éxito de llenado de tinaco y vaciado, simulando el consumo o gasto de agua potable y donde las tres tarjetas muestran su funcionamiento estable.



4.3: Figura N°35a. Foto al Funcionamiento del prototipo.

En el primer funcionamiento se enciende la electrobomba, partiendo de que tenemos vacío nuestro sistema tinaco-cisterna, en esta figura N°35 observamos que al encender la bomba en el prototipo, ya luego de varios minutos se puede ver como el agua está a la mitad tanto en el tinaco como en la cisterna.



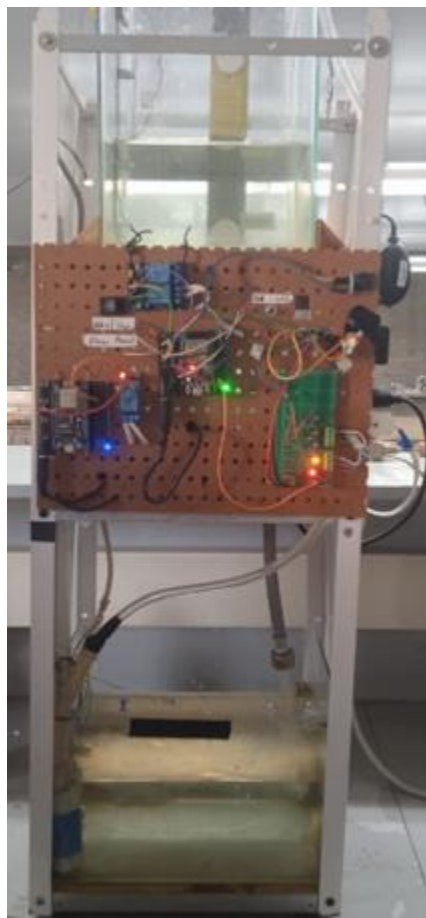
4.3: Figura N°35b. Foto a los sensores externos, funcionando en el prototipo.

En este segundo funcionamiento para el prototipo de sistema tinaco-cisterna, podemos observar en la figura N°35b como se da la activación de los sensores de nivel externos por medio de un led que ha encendido en rojo, esto al detectar la presencia del agua. Se apaga la bomba una vez que el sensor que se encuentra en la parte superior detecta agua este prende en rojo y procede en el sistema a activar a la electroválvula, para que esta permita el desagüe en el tinaco, en el cual el flujo del agua bajara por la manguera plateada que se muestra en la figura N°35a.



4.3: Figura N°36a. Foto al Encendido y pagado de la bomba y electroválvula.

En la fotografía de la figura N°36a podemos observar el buen funcionamiento de nuestro sistema, el led azul es el control para la bomba de agua el led rojo es el que está indicando la desactivación del relevador, usado para la bomba. El led verde es el led que indica el buen funcionamiento del controlador que activa y desactiva la electroválvula.



4.3: Figura N°36b. Foto al Encendido y pagado de electrobomba y electroválvula.

En la figura N°36b, observamos en esta fotografía, que se ha desactivado la bomba que suministra agua al tinaco, y ahora esta activada la electroválvula, y por la manguera plateada se está llevando a cabo desagüe del tinaco hacia la cisterna. (si el sistema está en modo automático, el sensor inferior del tinaco ya una vacío, desactivara la válvula y activara la electrobomba para llenar el tinaco nuevamente).

#### 4.4. Simulación del monto en pesos mexicanos del prototipo para una casa o edificio.

Presupuesto para una casa de 5 habitantes, con 4.5 metros de alto desde el nivel de suelo hasta la parte más alta y tomando en cuenta que se utilizan 25 metros de tubería de PVC hidráulica.

Cantidad	Descripción	Precio Unitario + IVA	Total
1	Electrobomba Centrífuga Turbo Light/WEG 1 HP	\$3,099.00	\$3,099.00
1	Tinaco Rotoplas para 1100 LITROS	\$4,779.00	\$4,779.00
9	Tubo de PVC hidráulico de 3 metros de 1 pulgada	\$123.00	\$1,107
1	Válvula esférica de PVC para 1 pulgada	\$92.00	\$92.00
5	Conexión T de PVC para 1 pulgada	\$14.40	\$72.00
10	Codo de 90° de PVC para 1 pulgada	\$11.20	\$120.00
5	Codo de 45° de PVC para 1 pulgada	\$15.68	\$78.40
3	Cinta teflón	\$24.00	\$72.00
2	Pegamento para tubería PVC	\$59.00	\$118.00
2	ESP32	\$80	\$160
1	ESP8266	\$150	\$150
1	Escudo o Shield para ESP32 5v CC hasta 2 A	\$275	\$275
1	Escudo o Shield para ESP8266 5v CC hasta 2 A	\$275	\$275
2	Interruptor de tres pines dos canales	\$17.50	\$35
2	Módulo relevador de 5v CC a 120v CA	\$35	\$70
2	Circuito integrado ULN2803	\$22.50	\$45
10	Resistencias de 1k ohm	\$1.50	\$15
10	Resistencias de 330 ohm	\$1.50	\$15
10	Transistores 2n2222	\$6	\$60
1	Buse de 3 pines	\$50	\$50
4	Borneras de 3 canales	\$15	\$60
2	Borneras de 2 canales	\$15	\$30
2	Placas perforadas de 5cm x 10 cm	\$25	\$50
2	Sensores XKC-Y25	\$170	\$340
2	Electrodos de flotador de 90°	\$47	\$94
10	Leds.	\$1.50	\$15
3	Paquete de conectores tipo jumper, M-M, H-H y H-M	\$45	\$135
#	Mano de obra	\$2,500	\$2,500
#		Monto Total	\$13,776.40

4.4: Tabla N°4. Ejemplo de presupuesto para la instalación en una casa de una planta baja.

Presupuesto para un edificio de más de 17 metros, tomando en cuenta que se utilizan 35 metros de tubería.

Cantidad	Descripción	Precio + IVA	Total
1	Electrobomba Centrífuga Turbo Light/WEG 1 -1/2HP.	\$3,445.10	\$3,445.10
2	Tinaco Rotoplas para 1100 LITROS.	\$4,779.00	\$9,558.00
12	Tubo de PVC hidráulico de 3 metros de 1 “.	\$123.00	\$1,476.00
1	Válvula esférica de PVC para 1”	\$92.00	\$92.00
5	Conexión T de PVC para 1”	\$14.40	\$72.00
10	Codo de 90° de PVC para 1”	\$11.20	\$120.00
5	Codo de 45° de PVC para 1”	\$15.68	\$78.40
3	Cinta teflón	\$24.00	\$72.00
2	Pegamento para tubería PVC	\$59.00	\$118.00
2	ESP32	\$80	\$160
1	ESP8266	\$150	\$150
1	Escudo o Shield para esp32 5v hasta 2 A.	\$275	\$275
1	Escudo o Shield para esp8266 5v hasta 2 A.	\$275	\$275
2	Interruptor de tres pines dos canales.	\$17.50	\$35
2	Módulo relevador de 5vdc a 120vac	\$35	\$70
2	Circuito integrado ULN2803.	\$22.50	\$45
10	Resistencias de 1kohms.	\$1.50	\$15
10	Resistencias de 330ohms.	\$1.50	\$15
10	Transistores 2n2222.	\$6	\$60
1	Buse de 3 pines.	\$50	\$50
4	Borneras de 3 canales	\$15	\$60
2	Borneras de 2 canales.	\$15	\$30
2	Placas perforadas de 5cm x 10 cm.	\$25	\$50
2	Sensores XKC-Y25.	\$170	\$340
2	Electrodos de flotador de 90°.	\$47	\$94
10	Leds.	\$1.50	\$15
3	Paquete de Jumpers, MM HH y HM.	\$45	\$135
	Mano de obra	\$6,000	\$6,000
#		Monto Total	\$22,905.50

4.4: Tabla N°5. Ejemplo de presupuesto para la instalación en un edificio de 5 plantas.

Costo aplicado a un edificio de cinco plantas.	Monto Total	\$13,776.40	Costo aplicado típico con cableado a una casa habitación.	\$12,337
Costo aplicado a una vivienda de una planta.	Monto Total	\$22,905.50	Costo aplicado típico con cableado sobre el edificio.	\$21,061

4.4: Tabla N°6. Ejemplo de montos totales de instalación y comparación.

## Capítulo 5. Conclusiones.

En relación con los dispositivos existentes en el mercado, se han identificado algunas desventajas, como el aislamiento de los sensores utilizados, la limitación de controlar solo un tanque y la falta de información detallada sobre el consumo eléctrico. En este proyecto, se busca superar estas limitaciones al utilizar tarjetas de desarrollo precisas y programables que permiten realizar múltiples tareas, tanto en el monitoreo de sensores como en la activación de actuadores, ofreciendo una solución más completa y adaptada a las necesidades del proyecto.

El sistema de llenado de los depósitos de agua se diseña para integrarse con el sistema de bombeo de agua existente en viviendas, el cual consta de un circuito eléctrico de bombeo convencional. Este circuito está compuesto principalmente por una electrobomba de agua de 127v AC con diferentes capacidades de HP ( $\frac{1}{4}$ ,  $\frac{1}{2}$ , 1 y  $1 \frac{1}{2}$ ). El control del sistema se realizará a través de un dispositivo digital con pantalla que permite activar la electrobomba manualmente o de manera automática.

Algunos productos de sistemas de bombeo que se pueden conseguir en página web de internet con características como **Wi-Fi** integrado, en la tabla N°1 de los productos actuales en el mercado se describen ventajas y desventajas.

Se elaboró un diagrama general de conexiones del prototipo armado donde se puede observar la conexión electrónica, el lenguaje de programación es de tipo “c” en la Plataforma de IDE Arduino con bibliotecas de código y programación de **Espressif**, el fabricante **Espressif** proporciona un protocolo llamado protocolo **ESP-NOW** que permite comunicación entre 20 dispositivos ya sea en red, o maestro esclavo.

El sistema de llenado de tanques de agua monitorea el porcentaje de agua en la cisterna, considerando las dimensiones máximas y mínimas de dicho depósito. Además, la activación de la electrobomba puede realizarse de forma remota, sin necesidad de exponerse a condiciones climáticas desfavorables o horarios poco convenientes.

El monitoreo del tanque elevado garantiza un suministro adecuado de agua al usuario, mientras que el trabajo de la electrobomba se ajusta según la demanda de agua de cada departamento. Esto contribuye a reducir el desgaste del motor eléctrico utilizado en cada ciclo de llenado.

Uno de los objetivos de este trabajo a partir de una instalación típica con una cisterna, y electrobomba un tinaco es eliminar el cableado eléctrico de una instalación tinaco-cisterna para casas o edificios simplificando su mantenimiento y su instalación.

En el caso de las casas habitación, donde es menos común contar con una cisterna, el sistema de llenado se adapta para solicitar el servicio de agua local y así funcionar de manera más eficiente, proporcionando un mejor servicio de agua sin afectar el sistema eléctrico existente.

Es importante destacar que el proyecto de monitoreo remoto a través de **Wi-Fi** no afecta la instalación eléctrica existente ni el encendido y apagado de la electrobomba. El monitoreo se encuentra completamente aislado de la potencia eléctrica del equipo, lo que garantiza un proceso seguro y confiable de llenado del tanque y monitoreo de la cisterna.

El horario de encendido y apagado de la electrobomba queda a decisión del usuario, lo que permite mayor comodidad y seguridad al controlar el abastecimiento de agua desde el propio departamento, ya sea de forma manual o automática.

En caso de ser necesario, el agua contenida en el tanque elevado o la cisterna puede ser vaciada para realizar mantenimiento en la instalación hidráulica. Para ello, se recomienda contar con personal certificado especializado en la limpieza de tanques de agua, garantizando así la calidad y el correcto funcionamiento del sistema.

El abastecimiento individual por departamento depende de la tubería proporcionada por el sistema de aguas y drenajes de la Ciudad de México. El monitoreo del nivel del agua en el tanque (tinaco) para consumo final es una de las salidas del sistema de monitoreo de tanques y cisternas para viviendas en la Ciudad de México. Esto permite tener un control más preciso y eficiente del abastecimiento de agua en cada departamento.

**De acuerdo con lo anterior con los resultados obtenidos en el diseño, desarrollo y armado, para el monitoreo, control y automatización de un sistema cisterna-tinaco podemos concluir lo siguiente:**

- **Se logró con éxito el control inalámbrico encendido o apagado de la válvula en el tinaco y la electrobomba de la cisterna**
- **Se logró con éxito monitorear de manera inalámbrica el nivel del tinaco sin necesidad de subir a revisar**
- **Se logró con éxito la automatización del llenado del tinaco con la activación inalámbrica de la electrobomba**
- **El protocolo de comunicación ESP-NOW embebido en las placas EPS32 y ESP8266 es ideal para la comunicación inalámbrica ya que es persistente y de un buen alcance.**

El sistema de monitoreo y automatización inalámbrica de un sistema tinaco-cisterna permita activar una electrobomba eléctrica de hasta 250v a 2HP que es una corriente relativamente alta para un edificio convencional en una electrobomba de 1HP no habrá mayor problema.

Al ser inalámbrico el sistema tinaco-cisterna simplifica mucho una instalación típica de una vivienda o edificio y disminuye el costo de instalación que tiene que ver con la red eléctrica utilizada

En un futuro las cosas que se podrían mejorar son la siguientes:

Implementación del prototipo en un edificio o casa real.

Mejor el monitoreo por sensor ultrasónico impermeable.

Hacer un diseño más ergonómico e industrial de la placa y el control.

Hacer un producto que se pudiera comercializar.

El costo del prototipo fue un total de \$ 2,694.00 MXN.

## Anexos.

### Control con celda solar.

De esta manera quedaría el Sistema propuesto a futuro con alimentación eléctrica por medio de una celda solar de 12v CC a una potencia de 20w y un amperaje a 1,6 A para la tarea del censado del tanque tinaco y encendido de la electroválvula que en este caso se agregó, para el sistema fotovoltaico se debe buscar dar 5v CC y una corriente entre los 1.5A y 2A para garantizar su correcto funcionamiento.

La señal **Wi-Fi** se activa óptimamente a 5v CC y a 1A.

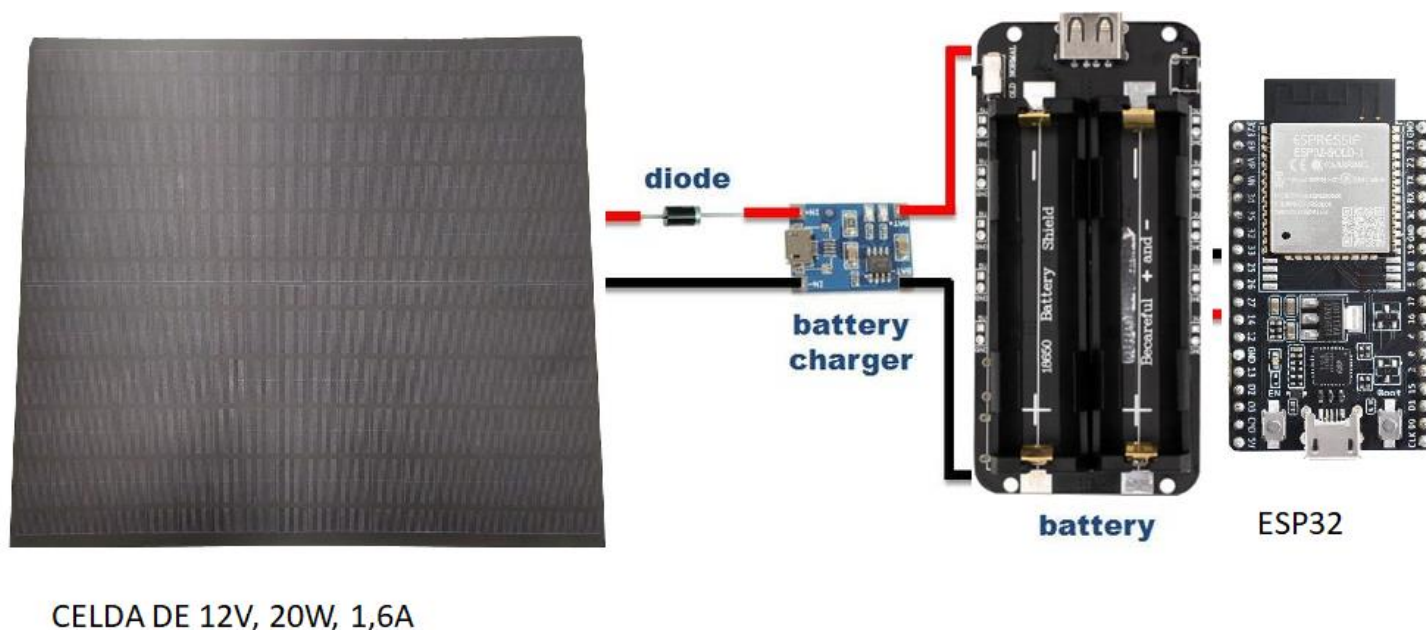


Figura N°37. Sistema solar propuesto. (Tomada de [23])



Figura N°38. Foto al funcionamiento de los sensores externos.

Los sensores xkc-y25 detector de altura se activa desde 200mA.

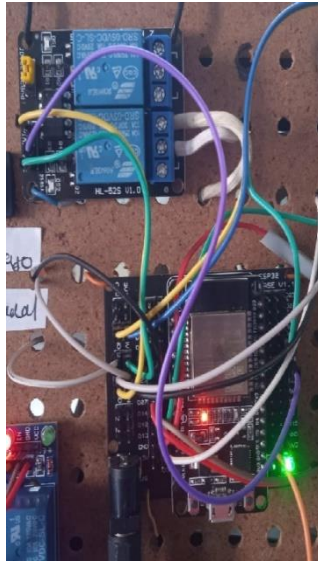


Figura N°39: Fotografía de la Activación y desactivación de relevador utilizando un canal.

El módulo relevador activándose desde 250mA.

Así mismo recibe y envía señales de **Wi-Fi** pues trabaja de modo bidireccional con el control automático y a la vez unidireccional con el módulo ESP8266 para encender la electrobomba.

Los sensores anexos para medir el nivel al sistema tinaco-cisterna funcionan óptimamente a 5v CC y 750mA.



Figura N°40. Fotografía del funcionamiento del detector de niveles.

Utilizando estos materiales el costo de inversión se incrementa de manera considerable pues; la celda solar de 20W tiene un costo de \$300 MXN, el Cargador de dos pilas \$200 MXN, una fuente de 12 volts CA a 2 A tiene un costo de \$180 MXN. Ahora bien podemos concluir que se incrementa en \$680 nuevos pesos mexicanos.

Monitoreo con sensor ultrasónico resistente al agua.

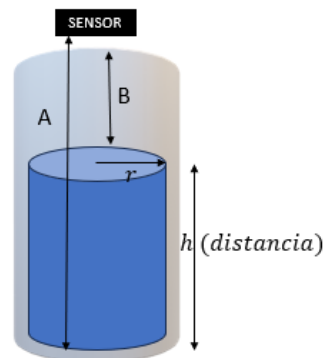


Figura N°41. Sensor ultrasónico resistente al agua. (Tomada de [24])

Con este tipo de sensores de distancia ultrasónicos resistentes al agua, ya no puede ser un problema para futuras actualizaciones del prototipo. Con un diseño para soportar los elementos al aire libre. Su costo aproximado esta entre los \$120 MXN.

Este es un sensor de distancia ultrasónico con un emisor sellado a prueba de agua. Este sensor es adecuado para exteriores, y aplicaciones como sensores de marcha atrás de automóviles, alarmas de seguridad, inspección industrial, etc. Por lo cual queremos darle uso para el monitoreo del líquido en el tinaco, incluso en la cisterna.

Sensor → distancia → volumen



$$v = \pi r^2 h$$

$$h = A - B$$

Donde  $r, h$  están dados en decímetros (dm)

Figura N°42. Diagrama para el volumen de un depósito de agua. (Tomada de [25])

En este diagrama podemos observar el volumen  $v$ , para un depósito de agua,  $h$  es la diferencia entre la altura  $A$  entre la base del depósito hasta la parte superior donde se encuentra el sensor ultrasónico resistente al agua, y  $B$  es el espacio vacío entre la parte superior de la superficie del líquido a nuevamente nuestro sensor.

Códigos en lenguaje “c” utilizados en este proyecto.

**Código utilizado para el control remoto una vez seleccionado modo manual.**

```
#include <esp_now.h> // Biblioteca para utilizar protocolo Esp-Now
#include <WiFi.h> //biblioteca para utilizar el Wi-Fi de las placas Esp32 y Esp8266
#define botonValvula 25 //botón ON OFF válvula
#define botonBomba 26 //botón ON OFF bomba

//botones disponibles en PCV actualmente

//#define boton3 27
//#define boton4 14
//#define boton5 12
//#define boton6 13//BOTON DE PAGADO

bool estadoValvula=false; //on-of en variable booleana leemos, on-off
bool estadoBomba=false; // on-of en variable booleana leemos, on-off
#define __DEBUG__

#include <SPI.h> // Interfaz periférica serial funciona bien a cortas distancias
#include <Wire.h> // Circuito Inter integrado o dos líneas SDA y SCL
#include <Adafruit_GFX.h> // Biblioteca para sintaxis común y un conjunto de funciones gráficas la pantalla OLED.
#include <Adafruit_SSD1306.h> //Controlador para ancho de pixeles oled.

// Definir constantes
#define ANCHO_PANTALLA 128 // Ancho pantalla OLED
#define ALTO_PANTALLA 64 // Alto pantalla OLED

// Objeto de la clase Adafruit_SSD1306
Adafruit_SSD1306 display(ANCHO_PANTALLA, ALTO_PANTALLA, &Wire, -1);
int estadoTinaco; //Variable de tipo entero

// REPLACE WITH YOUR RECEIVER MAC Address
uint8_t broadcastAddress[] = {0xE0, 0x5A, 0x1B, 0x9F, 0xFD, 0x38}; // Macadrrestinaco
//uint8_t broadcastAddress1[] = {0x08, 0x3A, 0xF2, 0x8F, 0x5C, 0x88}; // Macadressbomba

// Ejemplo de estructura para recibir datos
```

```

// Debe coincidir con la estructura del remitente
typedef struct struct_message {
    int a; // Copia del mensaje a recibir variable de tipo entero.
} struct_message;// Estructura del mensaje
typedef struct struct_message_control { //Estructura tipificada del mensaje de control
    int Valvula; //Controlar válvula
    int bomba; //Controlar bomba
} struct_message_control;//Estructura del mensaje del control.
// Crea un struct_message llamado myDataRec recibido
struct_message myDataRec; /// Recibe el nivel del tinaco
// Crea un struct_message llamado myDataSen envió
struct_message_control myDataSen; // Control de válvula y tinaco
esp_now_peer_info_t peerInfo;//
// Devolución de llamada cuando se envían datos
void OnDataSent(const uint8_t *mac_addr, esp_now_send_status_t status) {
    Serial.print("\r\nLast Packet Send Status:\t");
    Serial.println(status == ESP_NOW_SEND_SUCCESS ? "Envio exito al maestro" : "Delivery Fail");//?????
}
// Función de devolución de llamada que se ejecutará cuando se reciban los datos
void OnDataRecv(const uint8_t * mac, const uint8_t *incomingData, int len) {
    memcpy(&myDataRec, incomingData, sizeof(myDataRec));
    Serial.print("Datos recibidos del maestro: ");
    Serial.println(len);
    Serial.print("Int: ");
    Serial.println(myDataRec.a);
    Serial.println();
    estadoTinaco = myDataRec.a;
}
void setup() {
    // Inicialice Serial Monitor

```

```

#ifdef __DEBUG__
  Serial.begin(9600);
  delay(100);
  Serial.println("Iniciando pantalla OLED");
#endif

// Iniciar pantalla OLED en la dirección 0x3C
if (!display.begin(SSD1306_SWITCHCAPVCC, 0x3C)) {
#ifdef __DEBUG__
  Serial.println("No se encuentra la pantalla OLED");
#endif
  while (true);
}
pinMode(botonValvula,INPUT_PULLUP);
pinMode(botonBomba,INPUT_PULLUP);
// Set device as a Wi-Fi Station
WiFi.mode(WIFI_STA);
// Init ESP-NOW
if (esp_now_init() != ESP_OK) {
  Serial.println("Error initializing ESP-NOW");
  return;
}
// Once ESPNow is successfully Init, we will register for recv CB to
// get recv packer info
esp_now_register_recv_cb(OnDataRecv);
// Once ESPNow is successfully Init, we will register for Send CB to
// get the status of Trasnmitted packet
esp_now_register_send_cb(OnDataSent);
// Register peer
memcpy(peerInfo.peer_addr, broadcastAddress, 6);
peerInfo.channel = 0;

```

```

peerInfo.encrypt = false;
// Add peer
if (esp_now_add_peer(&peerInfo) != ESP_OK){
    Serial.println("Failed to add peer");
    return;
}
}
void loop()
{
    // Limpiar buffer
display.clearDisplay();
// Tamaño del texto
display.setTextSize(1);
// Color del texto
display.setTextColor(SSD1306_WHITE);
// Posición del texto
display.setCursor(10, 32);
//Activar página de código 437
display.cp437(true);
    display.write(173);
display.write(173); // Escribir texto
display.println("Nivel tinaco:"); // Escribir el carácter
if(estadoTinaco==1) {
    display.setCursor(10, 40);
    display.write("Vacio");
}
if(estadoTinaco==2) {
display.setCursor(10, 40);
    display.write("Medio");
}
}

```

```

if(estadoTinaco==3) {
display.setCursor(10, 40);
    display.write("Lleno");
} // Enviar a pantalla
display.display();
delay(2000);
if (digitalRead(botonValvula)==LOW){
    delay(200);
    estadoValvula=!estadoValvula;
}
if(estadoValvula==true) {
    myDataSen.Valvula = 1;
} else {
    myDataSen.Valvula = 0;
}
if (digitalRead(botonBomba)==LOW){
    delay(200);
    estadoBomba=!estadoBomba;
}
if(estadoBomba==true) {
    myDataSen.bomba = 1;
} else {
    myDataSen.bomba = 0;
}
esp_err_t result = esp_now_send(broadcastAddress, (uint8_t *) &myDataSen, sizeof(myDataSen));
if (result == ESP_OK) {
    Serial.println("Sent with success");
}
else {
    Serial.println("Error sending the data");
}

```

```

}
display.println("SSD1306");
delay(2000);
}

```

Código en lenguaje “c” utilizado para controlar la electroválvula y sensores de nivel de altura de agua.

```

#include <esp_now.h> //Biblioteca para leer el Código Esp-Now
#include <WiFi.h> //Biblioteca para utilizar el Wi-Fi de ESP32
// Direcciones MAC receptor bomba y control Dirección MAC
uint8_t broadcastAddress[] = {0x84, 0xF3, 0xEB, 0xCA, 0x61, 0x8D}; // Mac Adress de la bomba
uint8_t broadcastAddressNivel[] = {0x08, 0x3A, 0xF2, 0x8F, 0x5C, 0x88}; // Mac Adress del control
int estadoValvula = 0; // Se inicializa en cero apagado la válvula.
int estadoBomba = 0; // Se inicializa en cero apagado la bomba.
// Ejemplo de estructura para enviar datos
// Debe coincidir con la estructura del receptor
typedef struct struct_message { // datos enviados a la bomba
    int bomba; // control bomba modo entero.

} struct_message; // estructura de datos o mensaje a enviar.
typedef struct struct_message_nivel { // datos enviados al control
    int a;

} struct_message_nivel; // estructura de datos o mensaje a enviar.
typedef struct struct_message_control { // datos recibidos del control
    int valvula; // controlar valvula modo entero.
    int bomba; // controlar valvula modo entero.

} struct_message_control; // estructura de datos o mensaje a recibir.
struct_message_control myDataRec; // datos recibidos de válvula y tinaco
// Crea un struct_message llamado myData.
struct_message myData;
// Crea un struct_message llamado myDataNivel.

```

```

struct_message_nivel myDataNivel;

esp_now_peer_info_t peerInfo;//La función peer_Info confirma si encontró el primer módulo asignado con un
ESP_OK.

esp_now_peer_info_t peerInfo2;//La función peer_Info confirma si encontró el segundo modulo asignado con un
ESP_OK.

// devolución de llamada cuando se envían datos.

void OnDataSent(const uint8_t *mac_addr, esp_now_send_status_t status) { /// la función OnDataSent confirma
Mac Adress y tipo de valor.

    Serial.print("\r\nLast Packet Send Status:\t");

    Serial.println(status == ESP_NOW_SEND_SUCCESS ? "Envía con exito" : "Delivery Fail");//Confirmación de
datos si fueron enviados o fallo él envío.

}

/// para recibir datos del control.

void OnDataRecv(const uint8_t * mac, const uint8_t *incomingData, int len) {///confirma MAC Adress del control
esp32 y valor de unidades.

    memcpy(&myDataRec, incomingData, sizeof(myDataRec));//copia el dato recibido previamente creado.

    Serial.print("Recibe datos del control");//Impresión y confirmación de datos recibidos.

    Serial.println(len);//La función LEN devuelve un valor entero que indica la cantidad de caracteres en la cadena
de entrada.

    Serial.print("Int: ");//Imprimir entero recibido.

    Serial.println(myDataRec.bomba);//Imprime y salto de línea entero bomba recibidos.

    Serial.println(myDataRec.valvula);//Imprime y salto de línea entero válvula recibidos.

    Serial.println();//salto de línea, no imprime.

    estadoValvula = myDataRec.valvula;//Variable estado Válvula igual al dato recibido de ESP32 válvula.

    estadoBomba = myDataRec.bomba;//Variable estado Bomba igual al dato recibido de ESP8266 bomba.

}

//Declaramos entradas o los pines constantes en el programa que controla los electro niveles, interruptores y
botones.

const int boton=13;// es el botón on-off D2

const int s3=12; //tinaco máx.

//const int s2=4; // tinaco ¾ agua

//const int s1=5; // tinaco bajo nivel de agua

```

```

const int s0=14; // tinaco min (vacío)
const int c3=27; // cisterna máx.
//const int c2=8; // cisterna 3/4 agua
//const int c1=9; // cisterna bajo nivel de agua
const int c0=26; // cisterna vacía alerta de abasto
const int modo=32; // interruptor para controlar el modo manual

// Declaramos pines de salida para conmutar la bomba, la válvula, zumbador o alarma y Led, on-off de verificación

const int bomba=19; //bomba de agua
const int valvula=18; //válvula
const int zumbador=21;
int ledonoff =2;
int mensaje_cisterna=1;
#define APAGADO 0
#define ENCENDIDO 1
#define LLENADO 2
#define VACIANDO 3
#define CISTERNA_VACIA 4
int estado = APAGADO; //espera el primer Led de tinaco el led de tinaco vació
//con el botón interruptor en on o valor 1 lógico.

void setup() {
  pinMode(boton, INPUT); //declaramos el botón como entrada
  pinMode(s3 , INPUT); //declaramos el sensor de tinaco lleno
  //pinMode(s2, INPUT);
  //pinMode(s1, INPUT);
  pinMode(s0, INPUT); //declaramos el señor de tinaco vació
  //pinMode(c3, INPUT);
  //pinMode(c2, INPUT);
  //pinMode(c1, INPUT);
  //pinMode(c0, INPUT);

```

```

pinMode(mod0, INPUT);
pinMode(bomba, OUTPUT); //declaramos la bomba como salida.
pinMode(valvula, OUTPUT); //declaramos la válvula como salida.
pinMode(zumbador, OUTPUT); //declaramos la válvula como salida.
digitalWrite(bomba, LOW); //inicialmente la bomba apagada.
digitalWrite(valvula, LOW); //inicia válvula apagada.
Serial.begin(115200); //velocidad en baudios por segundo.

// Set device as a Wi-Fi Station
WiFi.mode(WIFI_STA);
// Init ESP-NOW
if (esp_now_init() != ESP_OK) {
  Serial.println("Error initializing ESP-NOW");
  return; }
// Once ESPNow is successfully Init, we will register for Send CB to
// registrar función para enviar datos a la bomba y al control
esp_now_register_send_cb(OnDataSent);
// declarar función para recibir
esp_now_register_recv_cb(OnDataRecv);
// Register bomba
memcpy(peerInfo.peer_addr, broadcastAddress, 6);
peerInfo.channel = 0;
peerInfo.encrypt = false;
// Add peer
if (esp_now_add_peer(&peerInfo) != ESP_OK){
  Serial.println("Failed to add peer");
  return;
}
//***** envía el nivel se registra control

memcpy(peerInfo2.peer_addr, broadcastAddressNivel, 6);

```

```

peerInfo2.channel = 1;
peerInfo2.encrypt = false;

if (esp_now_add_peer(&peerInfo2) != ESP_OK){
  Serial.println("Failed to add peer");
  return;
}
}

void Controlmanual(){
//
  if(estadoBomba==1) {
    digitalWrite(bomba,LOW); //BOMBA ALTO se activa la bomba
    myData.bomba=0;
    esp_now_send(broadcastAddress, (uint8_t *) &myData, sizeof(myData));
    Serial.println("Bomba prendida Manual");
  } else {
    digitalWrite(bomba,HIGH); //BOMBA ALTO se desactiva la bomba
    Serial.println("Bomba apagada Manual");
    myData.bomba=1;
    esp_now_send(broadcastAddress, (uint8_t *) &myData, sizeof(myData));
  }
//// para controlar a la valvula.
  if(estadoValvula==1) {
    digitalWrite(valvula,HIGH);
    Serial.println("Valvula prendida Manual");
  } else {
    digitalWrite(valvula,LOW);
    Serial.println("Valvula apagada Manual");
  } }
void loop() {

```

```

Monitoreo();
/***/ controla la válvula y bomba manual*///
Serial.println(digitalRead(modos));
if (digitalRead(modos)==LOW) {
Serial.println("Modo Manual");
Controlmanual();
} else {
Serial.println("Modo Automático");
Automatico();
}
}
void Monitoreo(){
esp_now_send(broadcastAddressNivel, (uint8_t *) &myDataNivel, sizeof(myDataNivel));
if (digitalRead(s3)==LOW and digitalRead(s0)==LOW) { // vacío

myDataNivel.a=3;
esp_now_send(broadcastAddressNivel, (uint8_t *) &myDataNivel, sizeof(myDataNivel)); // envía al dato
de vacío al display
Serial.println("Vacío");
}
if (digitalRead(s3)==LOW and digitalRead(s0)==HIGH) { // intermedio
myDataNivel.a=2;
esp_now_send(broadcastAddressNivel, (uint8_t *) &myDataNivel, sizeof(myDataNivel)); // envía al dato
de vacío a la pantalla OLED
Serial.println("Intermedio");
}
if (digitalRead(s3)==HIGH and digitalRead(s0)==HIGH) { // lleno
myDataNivel.a=1;
esp_now_send(broadcastAddressNivel, (uint8_t *) &myDataNivel, sizeof(myDataNivel)); // envía al dato
de vacío a la pantalla OLED
Serial.println("Lleno");
}
}
}

```

```

}
}
void Automatico(){
  switch(estado) {
  case APAGADO
    Serial.println("Apagado");
    digitalWrite(bomba,LOW);
    digitalWrite(valvula,LOW);
    if ( digitalRead(boton)==HIGH ) { /// si esta vació el tinaco se va llenado
      estado = ENCENDIDO;
    }
    break;
  case ENCENDIDO:
    Serial.println("Encendido");
    digitalWrite(bomba,LOW);
    digitalWrite(valvula,LOW);
    if (digitalRead(s3)==LOW and digitalRead(s0)==LOW) { /// si esta vació el tinaco se va llenado
      estado = LLENADO;
    }
    if (digitalRead(s3)==LOW and digitalRead(s0)==HIGH) { /// si No esta vació el tinaco se va a vaciando
      estado = VACIANDO;
    }
    if ( digitalRead(boton)==LOW ) { /// si esta vació el tinaco se va llenado
      estado = APAGADO;
    }
    break;
  case LLENADO:
    Serial.println("LLENADO");
    if (mensaje_cisterna==1 ) { //CISTERNA ALTO se verifica que exista agua en el cisterna

```

```

Serial.print("Activa bomba");
digitalWrite(bomba,HIGH); // Prende un led para indicar la bomba se activa
myData.bomba=1; /// activar bomba del esclavo
esp_now_send(broadcastAddress, (uint8_t *) &myData, sizeof(myData));
digitalWrite(zumbador,HIGH); // se activa el zumbador con un mensaje
digitalWrite(valvula,LOW); // se cierra la válvula
} else {    /// si está vacía la cisterna se va al estado CISTERNA_VACIA
estado = CISTERNA_VACIA;
digitalWrite(zumbador,HIGH); // se activa la bomba
}
if ( digitalRead(s3) == HIGH ){ ///TINACO ALTO cuando se llena el tinaco se va al estado vaciando
Serial.print("Ya se llenó");
estado = VACIANDO;
digitalWrite(zumbador,LOW); // se apaga el zumbador
}
    if ( digitalRead(boton)==LOW ) { /// si esta vació el tinaco se va llenado
estado = APAGADO;
}
break;
case VACIANDO: // cuando servicio el tinaco
    Serial.println("VACIANDO");
digitalWrite(bomba,LOW); // se desactiva la bomba
myData.bomba=0;
esp_now_send(broadcastAddress, (uint8_t *) &myData, sizeof(myData));
digitalWrite(valvula,HIGH); // se activa la valvula
if ( digitalRead(s0) == LOW && digitalRead(s3) == LOW){ /// si vació el tinaco se va llenado
    estado = LLENADO;
}
    if ( digitalRead(boton)==LOW ) { /// si esta vació el tinaco se va llenado
estado = APAGADO;

```

```
    }  
    break;  
case CISTERNA_VACIA:  
Serial.println("CISTERNA_VACIA");  
    digitalWrite(bomba,HIGH); // se activa  
    myData.bomba=0;  
    esp_now_send(broadcastAddress, (uint8_t *) &myData, sizeof(myData));  
    digitalWrite(valvula,LOW);  
    digitalWrite(zumbador,HIGH);  
        if ( digitalRead(boton)==LOW ) { /// si Esta vació el tinaco se va llenado  
estado = APAGADO;  
    }  
    break;  
} // fin switch  
}
```

### Código en lenguaje “c” utilizado para controlar la electrobomba.

```

#include <ESP8266WiFi.h>//Biblioteca para utilizar placa y wifi de ESP8266.
#include <espnw.h>// Biblioteca para utilizar protocolo Esp-Now.
int estadoBomba=0;// variable entera tipo entera inicializada en cero.
const int c2=16;// cisterna máx. pin0
const int c1=12;// cisterna medio pin6
const int led=2;//bomba de agua D5 led
const int zomm=14;//pin5
const int bomba=4;//como salida y on-off de la bomba.pin2
//Ejemplo de estructura para recibir datos
//debe coincidir con la estructura del remitente.
typedef struct struct_message { //Estructura para guardar los datos de estructura de mensaje emisor ESP32 válvula.
    int bomba;//variable de tipo entero bomba
} struct_message;// Datos guardados en estructura mensaje.
// creamos una estructura llamada struct_message myData.
struct_message myData;
// Función de devolución de llamada que se ejecutará cuando se reciban los datos.
void OnDataRecv(uint8_t * mac, uint8_t *incomingData, uint8_t len) { //Confirmación de envío y tipo de variable recibida.
    memcpy(&myData, incomingData, sizeof(myData)); //Copia del mensaje recibido.
    Serial.print("Recibido");
    Serial.println("Int: "); // impresión para bomba.
    Serial.println(myData.bomba); //Impresión del valor myData.bomba.
    Serial.println(); //solo salto de línea.
    estadoBomba=myData.bomba; //Creamos función myData llamada estadoBomba.
}
void setup() { //
    pinMode(bomba, OUTPUT); //declaramos la bomba como salida
    pinMode(led, OUTPUT); //declaramos la bomba como salida

```

```

// Inicializar monitor serie
Serial.begin(115200);
// Establecer el dispositivo como una estación Wi-Fi
WiFi.mode(WIFI_STA);
// Inicio ESP-NOW
if (esp_now_init() != 0) {
  Serial.println("Error initializing ESP-NOW");
  return;
}
// Una vez que ESP-NOW se inicie con éxito, nos registraremos para recv_cb para,
// obtener información del empaquetador de recepción
esp_now_set_self_role(ESP_NOW_ROLE_SLAVE);
esp_now_register_recv_cb(OnDataRecv);
}
void loop() {
  if(estadoBomba==1) {
    digitalWrite(led,LOW);
    //si cisterna está llena
    if (c1==HIGH && C2==HIGH)|| (c1==HIGH && C2==LOW)
    {
      //activa la bomba
      bomba=HIGH;
      //Enciende el zumbador
      zoom=HIGH;
    }
    //si Cisterna está vacía
    if (c1==LOW && C2==LOW)
    {
      //Enciende el zumbador
      zoom=LOW;
    }
  }
}

```

```
    //Apaga la bomba
    bomba=LOW;
    }
}
Serial.println("Bomba prendida");
digitalWrite(bomba,HIGH);
}
else {
    Serial.println("Bomba apagada");
    digitalWrite(bomba,LOW);
    digitalWrite(led,HIGH);
}
}
```

## Referencias.

- [1] Electro nivel Control Inalámbrico De Bombeo P/tinaco 1100lt en venta en Tlaxcala Tlaxcala por sólo \$ 2,095.00 - OCompra.com México. (n.d.). <https://www.ocompra.com/mexico/item/electronivel-control-inalambrico-de-bombeo-p-tinaco-1100lt-556710719/>
- [2] Racom. (n.d.). Electronivel RACOM | Control de Bombeo. <https://racom.com.mx/content/9-control-de-bombeo>
- [3] Sensor De Nivel De Agua Para Tinaco Wifi. (n.d.). Meses Sin Intereses. [https://articulo.mercadolibre.com.mx/MLM-1495367930-sensor-de-nivel-de-agua-para-tinaco-wifi-\\_JM?matt\\_tool=17264456&matt\\_word=&matt\\_source=google&matt\\_campaign\\_id=15700894061&matt\\_ad\\_group\\_id=130350624206&matt\\_match\\_type=&matt\\_network=g&matt\\_device=c&matt\\_creative=571726024527&matt\\_keyword=&matt\\_ad\\_position=&matt\\_ad\\_type=pla&matt\\_merchant\\_id=633209388&matt\\_product\\_id=MLM1495367930&matt\\_product\\_partition\\_id=1801632139836&matt\\_target\\_id=pla-1801632139836&gclid=CjwKCAjw67ajBhAVEiW2Ag\\_jEM2hHmHpcY80moVCXPJpQdqUsp7FleWsA6trZ65p-v6QVaOnDIZ6mhoC\\_VkQAvD\\_BwE](https://articulo.mercadolibre.com.mx/MLM-1495367930-sensor-de-nivel-de-agua-para-tinaco-wifi-_JM?matt_tool=17264456&matt_word=&matt_source=google&matt_campaign_id=15700894061&matt_ad_group_id=130350624206&matt_match_type=&matt_network=g&matt_device=c&matt_creative=571726024527&matt_keyword=&matt_ad_position=&matt_ad_type=pla&matt_merchant_id=633209388&matt_product_id=MLM1495367930&matt_product_partition_id=1801632139836&matt_target_id=pla-1801632139836&gclid=CjwKCAjw67ajBhAVEiW2Ag_jEM2hHmHpcY80moVCXPJpQdqUsp7FleWsA6trZ65p-v6QVaOnDIZ6mhoC_VkQAvD_BwE)
- [4] Yonusa. (n.d.). <https://www.yonusa.com/>
- [5] Figura N°2 diagrama general del prototipo: editado en software CAD 2020.
- [6] Santos, S., & Santos, S. (2022). Getting Started with ESP-NOW (ESP32 with Arduino IDE) | Random Nerd Tutorials. Random Nerd Tutorials. <https://randomnerdtutorials.com/esp-now-esp32-arduino-ide/>
- [7] <https://www.sapalomera.cat/moodlecf/RS/1/course/module4/4.2.4.4/4.2.4.4.html#:~:text=IEEE%20802.11g%3A%20opera%20en,de%20banda%20de%20802.11a>.
- [8] INTESC. (2022, 10 marzo). Módulo Relevador de 2 canales - INTESC. <https://intesc.mx/productos/modulo-relevador-de-2-canales/>
- [9] ESP8266 Wi-Fi MCU i Espressif Systems. (n.d.). <https://www.espressif.com/en/products/socs/esp8266>
- [10] Válvula solenoide 12V DC 1 pulgada - aelectronics. (n.d.). Aelectronics. <https://aelectronics.com.mx/valvula/2261-valvula-solenoide-12v-dc-1-pulgada.html>
- [11] Nodemcu ESP8266 Base Shield. (n.d.). MercadoLibre. [https://articulo.mercadolibre.com.mx/MLM-824374881-nodemcu-esp8266-base-shield-\\_JM](https://articulo.mercadolibre.com.mx/MLM-824374881-nodemcu-esp8266-base-shield-_JM)
- [12] Base 38 Pines Shield ESP32 ESP32S Wifi Regulador 5V 2A. (n.d.). MercadoLibre. [https://articulo.mercadolibre.com.mx/MLM-959748542-base-38-pines-shield-esp32-esp32s-wifi-regulador-5v-2a-\\_JM](https://articulo.mercadolibre.com.mx/MLM-959748542-base-38-pines-shield-esp32-esp32s-wifi-regulador-5v-2a-_JM)
- [13] Bomba sumergible 24V10l minuto ELEGANT COMET para caravanas y autocaravanas. (n.d.). Mecatech. [https://www.mecatech.com/es-ES/bomba-sumergible-24v10l-minuto-elegant-comet-para-caravanas-y-autocaravanas\\_CW10891.htm](https://www.mecatech.com/es-ES/bomba-sumergible-24v10l-minuto-elegant-comet-para-caravanas-y-autocaravanas_CW10891.htm)
- [14] <https://tresdprinttech.com/mx/sensores/620-sensor-nivel-de-agua-sin-contacto-xkc-y25-v-7503040289518.html>

[15] <https://tameson.es/pages/interruptor-de-flotador-como-funciona#:~:text=Un%20interruptor%20de%20flotador%20detecta,dela%20C3%ADquido%20sube%20o%20baja.>

[16] Contactor de CA doméstico 2P 2NO 40A 220V/230V Contactor modular, Montaje en riel DIN de 35 mm Contactor de automatización 50/60HZ, Frecuencia máxima de encendido/apagado 30 veces/h : Amazon.com.mx: Industria, Empresas y Ciencia. (n.d.-b). [https://www.amazon.com.mx/Contactor-dom%C3%A9stico-automatizaci%C3%B3n-Frecuencia-encendido/dp/B08GCCRTMC/ref=d\\_pb\\_allspark\\_dp\\_sims\\_pao\\_desktop\\_session\\_based\\_scl\\_4\\_4/134-8550359-3945643?pd\\_rd\\_w=wPX1c&content-id=amzn1.sym.149331ea-1901-4cd1-b5db-ec38af41db26&pf\\_rd\\_p=149331ea-1901-4cd1-b5db-ec38af41db26&pf\\_rd\\_r=5SA15JXQ1KSFC7R2NCMC&pd\\_rd\\_wg=GBm9I&pd\\_rd\\_r=e2b3cd6a-dd83-47ceb79-4ad7788c859e&pd\\_rd\\_i=B08GCCRTMC&psc=1](https://www.amazon.com.mx/Contactor-dom%C3%A9stico-automatizaci%C3%B3n-Frecuencia-encendido/dp/B08GCCRTMC/ref=d_pb_allspark_dp_sims_pao_desktop_session_based_scl_4_4/134-8550359-3945643?pd_rd_w=wPX1c&content-id=amzn1.sym.149331ea-1901-4cd1-b5db-ec38af41db26&pf_rd_p=149331ea-1901-4cd1-b5db-ec38af41db26&pf_rd_r=5SA15JXQ1KSFC7R2NCMC&pd_rd_wg=GBm9I&pd_rd_r=e2b3cd6a-dd83-47ceb79-4ad7788c859e&pd_rd_i=B08GCCRTMC&psc=1)

[17] [https://es.aliexpress.com/item/1005004263444094.html?srcSns=com.whatsapp.w4b&spreadType=socialShare&bizType=ProductDetail&social\\_params=21019446061&aff\\_fcid=2cf035cd7f424c08828f6d630f914ada-1689966363013-01106-\\_mLNVsfW&tt=MG&aff\\_fsk=\\_mLNVsfW&aff\\_platform=default&sk=\\_mLNVsfW&aff\\_trace\\_key=2cf035cd7f424c08828f6d630f914ada-1689966363013-01106-\\_mLNVsfW&shareId=21019446061&businessType=ProductDetail&platform=AE&terminal\\_id=a0fb02c7ccb24e58a43df84c591d85ac&afSmartRedirect=y](https://es.aliexpress.com/item/1005004263444094.html?srcSns=com.whatsapp.w4b&spreadType=socialShare&bizType=ProductDetail&social_params=21019446061&aff_fcid=2cf035cd7f424c08828f6d630f914ada-1689966363013-01106-_mLNVsfW&tt=MG&aff_fsk=_mLNVsfW&aff_platform=default&sk=_mLNVsfW&aff_trace_key=2cf035cd7f424c08828f6d630f914ada-1689966363013-01106-_mLNVsfW&shareId=21019446061&businessType=ProductDetail&platform=AE&terminal_id=a0fb02c7ccb24e58a43df84c591d85ac&afSmartRedirect=y) tarjeta leds azules ELECTRO NIVELES..

[18] Aurelio. (2020, March 20). NodeMCU: 04\_ESPNOW (Comunicación maestro-esclavo). <http://agrportfolioeducativo.blogspot.com/2020/03/nodemcu-04espnow-comunicacion.html>

[19],[20],[21], Santos, S., & Santos, S. (2022). Getting Started with ESP-NOW (ESP32 with Arduino IDE) | Random Nerd Tutorials. Random Nerd Tutorials. <https://randomnerdtutorials.com/esp-now-esp32-arduino-ide/>

[22] <https://electronicabasica.site/live-wire-y-pcb-wizard/>

[23] <https://forum.arduino.cc/t/calculo-de-panel-solar-para-proyectos-iot/609655>

[24] [https://media.digikey.com/pdf/Data%20Sheets/DFRobot%20PDFs/SEN0208\\_Web.pdf](https://media.digikey.com/pdf/Data%20Sheets/DFRobot%20PDFs/SEN0208_Web.pdf)

[25] [https://www.youtube.com/watch?v=g4U0cro2GHE&ab\\_channel=LeonardoP%C3%A9rez](https://www.youtube.com/watch?v=g4U0cro2GHE&ab_channel=LeonardoP%C3%A9rez)

