



UNIVERSIDAD DE LOS ANDES  
FACULTAD DE INGENIERÍA  
ESCUELA DE INGENIERÍA ELÉCTRICA

**DISEÑO DE UN SISTEMA DE MEDICIÓN DIGITAL PARA  
DETERMINAR EL NIVEL EN UN TANQUE DE  
ALMACENAMIENTO DE LÍQUIDOS.**

Br. Daniel José Aponte Mahfoud

Mérida, noviembre, 2022



UNIVERSIDAD DE LOS ANDES  
FACULTAD DE INGENIERÍA  
ESCUELA DE INGENIERÍA ELÉCTRICA

**DISEÑO DE UN SISTEMA DE MEDICIÓN DIGITAL PARA  
DETERMINAR EL NIVEL EN UN TANQUE DE  
ALMACENAMIENTO DE LÍQUIDOS.**

Trabajo de Grado presentado como requisito parcial para optar al título de Ingeniero  
Electricista

Br. Daniel José Mahfoud Aponte  
Tutor: Prof. Arlex J. Cáliz A.  
Cotutor: Prof. Francisco J. Vilorio M.

Mérida, noviembre, 2022

UNIVERSIDAD DE LOS ANDES  
FACULTAD DE INGENIERÍA  
ESCUELA DE INGENIERÍA ELÉCTRICA

**DISEÑO DE UN SISTEMA DE MEDICIÓN DIGITAL PARA  
DETERMINAR EL NIVEL EN UN TANQUE DE  
ALMACENAMIENTO DE LÍQUIDOS.**

Br. Daniel José Mahfoud Aponte

Trabajo de Grado, presentado en cumplimiento parcial de los requisitos exigidos para optar al título de Ingeniero Electricista, aprobado en nombre de la Universidad de Los Andes por el siguiente Jurado.

www.bdigital.ula.ve

---

Francisco J. Araujo R.

---

Francisco J. Vilorio M.

---

Prof. Arlex J. Cáliz A.

## **DEDICATORIA**

Agradezco a Dios por haberme otorgado una familia maravillosa, especialmente por mis padres Miriam Aponte y Daniel Mahfoud, quienes han creído en mí siempre, dándome ejemplo de superación, humildad y sacrificio; enseñándome a valorar todo lo que tengo. A ellos dedico el presente trabajo, porque han fomentado en mí, el deseo de superación y de triunfo en la vida. Lo que ha contribuido a la consecución de este logro. Espero contar siempre con su valioso e incondicional apoyo.

[www.bdigital.ula.ve](http://www.bdigital.ula.ve)

## AGRADECIMIENTOS

Agradezco primeramente a **DIOS** por otorgarme el don de la vida y permitirme llegar hasta este punto, en el cual puedo disfrutar de todo el esfuerzo hecho a lo largo de mi ciclo académico.

A **MIS PADRES, MIRIAM APONTE y DANIEL MAHFOUD** por su sacrificio y esfuerzo incondicional lo cual me permitió cursar y terminar estudios universitarios, es un honor para mí ser su hijo. Sin ustedes hubiese sido imposible.

A **MI HERMANA, DANIELA MAHFOUD** que a pesar de la distancia siempre estuvo al pendiente de mi progreso, gracias a su ayuda y sus mensajes de ánimo pude retomar el trabajo en oportunidades donde el desánimo estuvo presente.

A **MIS COMPAÑEROS Y FUTUROS COLEGAS, JESÚS RUIZ y EBY ABREU,** excelentes amigos con los cuales pude contar siempre; además de tener la dicha de compartir y crecer como profesional.

A mis amigos **ÁNGEL RAMÍREZ y SHARON ROJAS,** que siempre me dieron mensajes de motivación, además de apoyarme y tener gestos tan bonitos conmigo. Existen acciones que trascienden las palabras por lo cual son parte de mi éxito.

A **MIS TUTORES, ARLEX CALIX y FRANCISCO VILORIA** por brindarme la oportunidad y sobre todo gracias a su guía pude ir construyendo este trabajo de grado paso a paso. Estaré eternamente agradecido por todo su conocimiento y consejos.

Por último, a la **ILUSTRE UNIVERSIDAD DE LOS ANDES,** por brindarme un lugar en donde crecer académica y personalmente. Además de permitirme vivir una de las experiencias más bonitas de mi vida.

**Br. Daniel José Mahfoud Aponte. Diseño de un sistema de medición digital para determinar el nivel en un tanque de almacenamiento de líquidos.** Universidad de Los Andes.

Tutor: Prof. Arlex J. Cáliz A. Cotutor: Prof. Francisco J. Viloría M. noviembre 2022.

## RESUMEN

El presente trabajo se enfocó en el diseño de un prototipo para sistema de medición y registro de nivel en tanques, haciendo énfasis en la facilidad de instalación y operación del sistema; proporcionando a su vez la capacidad de almacenar en una memoria tipo micro SD el registro detallado de las operaciones diarias del equipo y mostrando el estado del sistema a través de una pantalla LCD 16x2; además de los reportes de operaciones de manera inalámbrica por medio de conexión *Wi-Fi*, a través de una interfaz web, la cual muestra los reportes mediante diagramas de barras, presentando así las operaciones de entrada y salida del volumen contenido en el tanque.

El sistema de medición fue diseñado con módulos de bajo costo para garantizar la accesibilidad económica del sistema y codificado en base a herramientas de software libre como la interfaz de desarrollo de Arduino, la cual cuenta con la capacidad de programar microcontroladores como el ESP-32. Con esta herramienta de desarrollo también se codificó el servidor web el cual trabaja mediante el protocolo de comunicación WebSocket asíncrono para el intercambio de información, pudiendo trabajar con páginas web interactivas que atiendan a varios clientes en simultáneo. Finalmente, en cuanto a la interfaz web, fue desarrollada mediante el uso de lenguaje de marcado de hipertexto (HTML), hoja de estilos en cascada (CSS) y JavaScript para el manejo de toda la lógica de funcionamiento y comunicación logrando dar estructura a la página web para que fuese intuitiva y fácil de operar.

**Descriptores:** ESP-32, control, medición de nivel, servidor web, *WebSocket*, HTML, JavaScript.

# ÍNDICE GENERAL

DEDICATORIA .....	iv
AGRADECIMIENTOS.....	v
RESUMEN .....	vi
INTRODUCCION.....	1
CAPITULO 1 EL PROBLEMA DE INVESTIGACIÓN.....	3
1.1 PLANTEAMIENTO DEL PROBLEMA.....	3
1.2 OBJETIVOS DE LA INVESTIGACION .....	4
1.2.1 <i>Objetivo General</i> .....	4
1.2.2 <i>Objetivos Específicos</i> .....	4
1.3 JUSTIFICACIÓN.....	5
1.4 ALCANCE .....	6
1.5 LIMITACIONES.....	6
CAPITULO 2 MARCO TEORICO.....	7
2.1 ANTECEDENTES DE LA INVESTIGACION .....	7
2.2 DEFINICION DE MEDICIÓN DE NIVEL .....	9
2.3 MÉTODOS PARA LA MEDICIÓN DE NIVEL .....	9
2.3.1 <i>Métodos para medición directa de nivel</i> .....	9
2.3.2 <i>Métodos para medición indirecta de nivel</i> .....	10
2.4 LA CALIBRACIÓN O AFORO DE TANQUES .....	18
2.5 MÉTODOS PARA LA CALIBRACIÓN DE TANQUES .....	18
2.5.1 <i>Método de “strapping” (API MTSM) o medición manual de anillos</i> .....	19
2.5.2 <i>Método de calibración líquida (API STD 2555)</i> .....	20
2.6 ELABORACIÓN DE LA TABLA DE AFORO .....	22

2.7	EL MICROCONTROLADOR EN LOS SISTEMAS EMBEBIDOS .....	23
2.7.1	<i>El microcontrolador</i> .....	23
2.7.2	<i>¿Qué son los sistemas embebidos?</i> .....	23
2.7.3	<i>Bus de comunicación</i> .....	23
2.8	SOFTWARE.....	26
2.8.1	<i>Arduino Integrated Development Environment (IDE)</i> .....	26
2.8.2	<i>Servidor</i> .....	27
2.8.3	<i>Protocolos de comunicación</i> .....	27
2.9	HARDWARE.....	29
2.9.1	<i>ESP-WROOM-32 DEVKIT V1-DOIT (SoM)</i> .....	29
CAPITULO 3 MARCO METODOLÓGICO .....		31
3.1	TIPO DE INVESTIGACIÓN.....	31
3.2	DISEÑO DE LA INVESTIGACION.....	31
3.2.1	<i>Análisis del sistema</i> .....	32
3.2.2	<i>Diseño</i> .....	32
3.2.3	<i>Programación</i> .....	33
3.2.4	<i>Pruebas</i> .....	33
3.3	TÉCNICAS O INSTRUMENTOS PARA RECOLECCIÓN DE DATOS .....	33
3.4	METODOLOGÍA PARA EL DISEÑO DEL SISTEMA DE MEDICION .....	34
3.4.1	<i>Etapa 1: Análisis de los requisitos del sistema</i> .....	34
CAPITULO 4 DISEÑO DEL HARDWARE PARA EL SISTEMA DE MEDICION DE NIVEL .....		37
4.1	ETAPA 2: DISEÑO DEL HARDWARE .....	37
4.1.1	<i>Bloque 1: unidad central de procesamiento (CPU)</i> .....	39
4.1.2	<i>Bloque 2: acondicionador de señales</i> .....	40

4.1.3	<i>Bloque 3: medición de nivel.....</i>	41
4.1.4	<i>Bloque 4: registro de datos.....</i>	42
4.1.5	<i>Bloque 5: visualización y control .....</i>	43
4.1.6	<i>Bloque 6: comunicación inalámbrica.....</i>	44
4.1.7	<i>Especificaciones técnicas del hardware .....</i>	44
4.1.8	<i>Diagramas de conexión de las placas. ....</i>	45
CAPITULO 5 DISEÑO DEL SOFTWARE PARA EL SISTEMA DE MEDICION DE NIVEL .....		48
5.1	ETAPA 3: DISEÑO DEL SOFTWARE .....	48
5.2	Especificaciones de software.....	48
5.2.1	<i>Diseño del sistema de medición.....</i>	49
5.2.2	<i>Diseño del servidor e interfaz web para el sistema de medición.....</i>	59
5.2.3	<i>Comunicación entre Front End y Back End .....</i>	70
CAPITULO 6 PRUEBA DEL SISTEMA DE MEDICION .....		72
6.1	ETAPA 4: PRUEBAS DEL SISTEMA .....	72
6.1.1	<i>Obtención de la curva de calibración de forma experimental .....</i>	72
6.1.2	<i>Validación de los datos obtenidos la prueba de calibración.....</i>	76
6.1.3	<i>Funcionamiento del sistema de medición.....</i>	78
6.1.4	<i>Funcionamiento de la comunicación inalámbrica. ....</i>	83
6.1.5	<i>Factibilidad económica del prototipo.....</i>	85
CONCLUSIONES.....		88
RECOMENDACIONES .....		90
REFERENCIAS .....		91

## ÍNDICE DE FIGURAS

Figura.	pp.
<i>Figura 2.1: Medidores de nivel por principio de medición directa [6].</i> .....	9
<i>Figura 2.2: Medidor de presión hidrostática tipo Presión diferencial. [5]</i> .....	11
<i>Figura 2.3: Montaje para sistemas de medición y alarma por ultrasonido. [6]</i> .....	12
<i>Figura 2.4: Medidor de nivel por radar por espacio libre (a) y medidor de nivel por onda guiada (b). [1]</i> .....	14
<i>Figura 2.5: Sistema de almacenamiento con medidor de nivel por radar. [9]</i> .....	15
<i>Figura 2.6: Diagramas de bloque para sistemas de medición por radar de pulso (a) y radar FMCW (b). [9]</i> .....	16
<i>Figura 2.7: Determinación de la distancia mediante el uso de radar por FMCW. [10]</i> .....	17
<i>Figura 2.8: Posiciones para la medición de circunferencias en tanques soldados. [11]</i> .....	19
<i>Figura 2.9: Tabla de aforo para un tanque de la planta Puerto Salgar de la empresa ECOPETROL. [11]</i> .....	22
<i>Figura 2.10: Conexión del sensor DS18B20 de la empresa Dallas mediante bus 1-wire. [16]</i> .....	24
<i>Figura 2.11: Secuencia de bits para la identificación de cada sensor 1-wire. [16]</i> .....	24
<i>Figura 2.12: Esquema de conexión para múltiples periféricos con funcionamiento por I<sup>2</sup>C. [17]</i> .....	24
<i>Figura 2.13: Diagrama de conexión para protocolo de comunicación SPI. [18]</i> .....	26
<i>Figura 2.14: Diagrama de secuencia de una comunicación WebSocket</i> .....	28
<i>Figura 2.15: Esquema de pines para el módulo ESP-WROOM-32 DEVKIT V1-DOIT [25]</i> .....	30
<i>Figura 3.1: Modelo del ciclo de Vida incremental. [27]</i> .....	32
<i>Figura 3.2: Análisis de Hardware y software requerido para la implementación del prototipo de medición de nivel de tanques.</i> .....	35
<i>Figura 4.1: Diagrama de bloques del sistema de medición de tanque SINCONT ESP32</i> .....	38
<i>Figura 4.2: Conexiones de salida para el ESP-32</i> .....	40
<i>Figura 4.3: Interfaz de conexión entre los periféricos que usan 5 V y el ESP-32</i> .....	41
<i>Figura 4.4: Conexión del sensor de ultrasonido y temperatura</i> .....	42
<i>Figura 4.5: Contenido y formato de los datos para la creación del reporte diario</i> .....	42
<i>Figura 4.6: Conexión del módulo de memoria y reloj</i> .....	43
<i>Figura 4.7: Diagramas de conexión de los indicadores visuales y la LCD</i> .....	44
<i>Figura 4.8: Modulo medidor por ultrasonido JSN-SR04. [36]</i> .....	45
<i>Figura 4.9: Diagramas de conexión de la placa A</i> .....	46
<i>Figura 4.10: Diagrama de conexión de la placa B</i> .....	47

<i>Figura 5.1: Diagrama para la máquina de estados finitos implementada</i> .....	50
<i>Figura 5.2: Algoritmo de la función Leer_Sensor_Ultrasonido(), para determinar el nivel del tanque</i> .....	52
<i>Figura 5.3: Pantallas para los estados MIDIENDO y LLENANDO</i> .....	54
<i>Figura 5.4: Formato de almacenaje para el reporte de 1 operación de vaciado</i> .....	56
<i>Figura 5.5: Uso de la función ConnectWiFi_STA() para realizar la conexión del ESP-32 con el enrutador</i> .....	60
<i>Figura 5.6: Codificación del servidor web asíncrono</i> .....	61
<i>Figura 5.7: Función para la inicialización de la conexión WebSocket, InitWebSocket()</i> .....	62
<i>Figura 5.8: Función de eventos WebSocket</i> .....	62
<i>Figura 5.9: Función para el procesado de las solicitudes del cliente</i> .....	63
<i>Figura 5.10: Función para iniciar solicitud de reporte diario</i> .....	64
<i>Figura 5.11: Construcción de la ruta para acceder al archivo a consultar</i> .....	64
<i>Figura 5.12: Diagrama de flujo para la función de lectura de reporte diario</i> .....	65
<i>Figura 5.13: Continuación del diagrama de flujo de la función de lectura de reporte</i> .....	66
<i>Figura 5.14: Inicialización de la comunicación WebSocket y creación de manejador de archivos desde el cliente</i> .....	68
<i>Figura 5.15: Codificación de la función processData()</i> .....	69
<i>Figura 5.16: Diagrama de secuencia para la comunicación entre cliente y servidor del sistema de medición e inventario de tanque SINCONT ESP-32.</i> .....	71
<i>Figura 6.1: Tanque usado en el prototipo de sistema de medición</i> .....	73
<i>Figura 6.2: curva de calibración para el sistema de medición, obtenida a través de proceso experimental</i> .....	75
<i>Figura 6.3: Hardware del prototipo implementado para el sistema de medición de tanques SINCONT ESP-32</i> .....	78
<i>Figura 6.4: Circuito impreso correspondiente a la placa A</i> .....	79
<i>Figura 6.5: Vista 3D del circuito impreso correspondiente a la placa A</i> .....	79
<i>Figura 6.6: Circuito impreso correspondiente a la placa B</i> .....	80
<i>Figura 6.7: Vista 3D del circuito impreso correspondiente a la placa B</i> .....	80
<i>Figura 6.8: Función de autodiagnóstico del sistema a través de la pantalla LCD 16x2.</i> .....	81
<i>Figura 6.9: Visualización en la pantalla 16x2 de una operación de llenado.</i> .....	82
<i>Figura 6.10: Almacenaje de los registros de entrada- salida e inventario del tanque.</i> .....	83
<i>Figura 6.11: Conexión del esp-32 a la red local creada por el enrutador.</i> .....	83
<i>Figura 6.12: Vista inicial de la página web que sirve como interfaz gráfica inalámbrica para el sistema de medición.</i> .....	84
<i>Figura 6.13: Reporte de operaciones diario obtenido a través de la página web.</i> .....	85

## ÍNDICE DE TABLAS

Tabla.	pp.
<i>Tabla 2.1: Tipos de medidores de nivel.....</i>	<i>18</i>
<i>Tabla 2.2: Resumen métodos de calibración.....</i>	<i>21</i>
<i>Tabla 4.1: Hojas de datos de dispositivos principales del sistema de medición.....</i>	<i>44</i>
<i>Tabla 5.1: Secuencia de pulsos intermitentes para el LED de fallas según tipo de avería.....</i>	<i>56</i>
<i>Tabla 5.2: Descripción de las entradas a la máquina de estados.....</i>	<i>57</i>
<i>Tabla 5.3: Descripción de las salidas de la máquina de estados.....</i>	<i>57</i>
<i>Tabla 5.4: Variables del sistema.....</i>	<i>57</i>
<i>Tabla 5.5: Estados y fallas presentes en la máquina de estados.....</i>	<i>57</i>
<i>Tabla 5.6: Tabla resumen de las transiciones que componen al diagrama de estados.....</i>	<i>58</i>
<i>Tabla 6.1: recolección de datos experimentales para la curva de calibración del sistema de medición.....</i>	<i>74</i>
<i>Tabla 6.2: Comparación de los datos obtenidos por el método calibración por strapping y el método de calibración por volumen.....</i>	<i>77</i>
<i>Tabla 6.3: Presupuesto para la construcción del prototipo para medición de tanques.....</i>	<i>86</i>

# INTRODUCCION

Los instrumentos de medida han cobrado gran importancia en los últimos años, y el interés de los fabricantes por mejorar sus dispositivos en términos de fiabilidad, tipos de comunicación y tamaño de los equipos ha dado lugar a una gran variedad de instrumentos. En la actualidad la tendencia gira en torno a equipos que puedan establecer comunicación de manera inalámbrica, con lo cual se facilita los procesos de instalación y mantenimiento. En tal sentido, al contar con un sistema de medición moderno y adecuado para las necesidades de un proceso específico una empresa puede aumentar su eficiencia y a su vez facilitar las operaciones; pudiendo llevar el inventario de la materia prima que entra y sale de la empresa e incluso pudiendo visualizar los datos del proceso en tiempo real.

En el estado Mérida existen empresas que trabajan con tanques que almacenan materia prima de valor económico significativo, tal es el caso de LABORATORIOS CHEMYCAL'S SOMA, C.A. ubicada en el poblado de Lagunillas, municipio Sucre del estado Mérida, esta empresa posee 3 tanques para el almacenaje de combustible, el cual es usado para alimentar a los grupos electrógenos que proveen energía a la empresa en caso de fallas en la red eléctrica pública, además de surtir a la flota de camiones con los cuales cuenta la empresa. Sin embargo, estos tanques no poseen un sistema de medición para conocer el nivel en dichos contenedores, El proceso de medición lo realiza un operador mediante inspección visual del tanque para estimar el nivel y en función de esta estimación programar las descargas de más combustible a cada recipiente. Este ineficiente procedimiento provoca que la empresa no pueda tener control de cuánto combustible se debe comprar mensualmente y sobre todo cuanto combustible está consumiendo efectivamente en la operación de los generadores eléctricos y su flota de carga. De ahí que, esta compañía tiene la necesidad de un sistema para la medición de nivel de tanques, que sea fácil de operar, permita llevar y visualizar el registro de las operaciones diarias y sobre todo que sea factible desde un punto de vista económico.

El presente trabajo de grado se estructuró en base a 6 capítulos los cuales se describen a continuación:

En el capítulo 1 se encuentra el planteamiento del problema, los objetivos, la justificación, el alcance y las limitaciones de la investigación.

El Capítulo 2 contiene los fundamentos teóricos de la investigación, define los mecanismos clásicos y modernos para la medición del nivel, los métodos para el aforo o calibración de tanques, las características del microcontrolador usado y los demás periféricos con los cuales interactúa el sistema; también contiene una breve explicación de los protocolos de comunicación y las herramientas de programación usadas.

El capítulo 3 describe la metodología implementada para el desarrollo del prototipo, mostrando los pasos a seguir para el desarrollo de un *software* funcional y eficiente.

El capítulo 4 se centra en el diseño del *hardware* más concretamente en la descripción de los módulos usados y como están interconectados al microcontrolador, estructurando el sistema de medición en bloques según su funcionalidad, entre estos bloques se encuentra la unidad de procesamiento central de datos, la etapa de medición de nivel, el bloque de registro de operaciones, la sección de control, visualización de datos y alarmas, posteriormente se encuentra el bloque de acondicionamiento de señales y por último el bloque de comunicaciones inalámbricas y servidor web.

El capítulo 5 se enfoca en describir el diseño del *software* para el sistema de medición de nivel de tanques el cual se divide en 2 partes, la primera es la codificación del software para el servidor web y la segunda la codificación de la interfaz de usuario para el monitoreo y control inalámbrico del sistema de medición.

Por último, el capítulo 6 muestra los resultados obtenidos mediante la implementación de los capítulos anteriores, exponiendo el funcionamiento tanto del sistema de medición como de la interfaz web y mostrando una comparación económica del prototipo diseñado contra una opción comercial.

# CAPITULO 1

## EL PROBLEMA DE INVESTIGACIÓN

### 1.1 PLANTEAMIENTO DEL PROBLEMA

Es bien sabido que, para poder competir en la producción y comercialización de sus productos, una empresa necesita contar con abastecimiento oportuno de materia prima, es decir, tener la capacidad y el diseño adecuado a las necesidades del proceso. Dinámica que permite obtener productos refinados para abastecer el mercado. En consecuencia, la compañía debe tener el control sobre el inventario en los productos que requiere para su normal funcionamiento.

Si bien es cierto, en la industria es muy común usar tanques para el almacenaje de productos líquidos, por consiguiente, la medición de estos resulta de vital importancia para la empresa, puesto que resguardan materia prima con un valor económico y estratégico significativo. De acuerdo con Hägg y Sandberg [1], “la medición de tanques es la medición de líquidos en grandes tanques de almacenamiento con el fin de cuantificar el volumen y la masa del producto en los tanques”. Los sistemas modernos para la medición de tanques, digitalizan las mediciones de masa y volumen del líquido contenido en el tanque, esta información es enviada posteriormente al personal encargado de gestionar los datos de inventario.

Actualmente, en la empresa LABORATORIOS CHEMYCAL'S SOMA, C.A. se almacena combustible (gas-oil) en 3 tanques de emergencia que sirven para el abastecimiento de los grupos electrógenos, los cuales suministran energía al complejo al momento de ocurrir una falla en la red eléctrica pública, además, dicho combustible es usado para la movilización de su flota de vehículos de carga. No obstante, para que los operadores puedan realizar el llenado de los tanques de almacenamiento se realiza manualmente la apertura de válvulas, luego se

verifica de manera visual el nivel aproximado de líquido en cada tanque. En consecuencia, la empresa no puede llevar un control preciso de la compra y consumo mensual de combustible, de igual manera, ya que los tanques son supervisados de forma visual no existe un modo de conocer en tiempo real el inventario presente en el tanque.

Además, se corre el riesgo de que los trabajadores puedan sustraer de manera ilícita combustible de los depósitos sin dejar ningún tipo de registro, lo cual representa pérdida de dinero debido al elevado costo de esta materia prima, también puede darse el caso de presentarse un daño a la estructura del tanque que dé como resultado una fuga, misma que no sería detectada de manera rápida y al tratarse de combustible representara un riesgo significativo a la seguridad de los trabajadores y la empresa misma.

En atención a la problemática descrita anteriormente, se plantea la siguiente pregunta de investigación:

¿Será factible técnica y económicamente el diseño de un prototipo electrónico para la medición de nivel que permita llevar el registro y control de líquido en los tanques?

## **1.2 OBJETIVOS DE LA INVESTIGACION**

### **1.2.1 Objetivo General**

- Diseñar un prototipo para un sistema de medición digital que determine el nivel y registro entrada-salida de un tanque de almacenamiento de líquidos.

### **1.2.2 Objetivos Específicos**

- Realizar el estudio de sensores comerciales para medición de combustible.
- Diseñar un prototipo para la medición de nivel de un tanque de combustible.
- Diseñar un sistema electrónico para registrar el consumo y suministro de combustible.
- Diseñar un sistema para la visualización y descarga de los datos.

### 1.3 JUSTIFICACIÓN

La instrumentación forma parte muy importante en la automatización y control de los procesos industriales, mediante el uso de esta se puede diseñar equipos que se adapten a las necesidades que requiera cierto proceso de medición. Así mismo, la implementación de sistemas automatizados dota al proceso de mayor precisión y comodidad a la hora de realizar mediciones.

En la actualidad, existen muchas empresas comercializadoras de sistemas para la medición de tanques con gran precisión. Sin embargo, las opciones que estas empresas ofrecen pueden tener costos bastante elevados, además que, en la mayoría de los casos suelen requerir personal especializado por parte de la empresa que comercializa el sistema de medición para la instalación y mantenimiento de los equipos.

Ahora bien, ya que los sistemas digitales se han hecho más aplicables para el desarrollo de la tecnología, y al mismo tiempo, el uso de plataformas de código abierto (*open source*) han permitido tener acceso a una comunidad de desarrolladores muy extensa la cual aporta recursos tecnológicos que son de uso libre. Por consiguiente, los costos para el desarrollo e implementación de equipos para la medición se ven significativamente reducidos.

Esta investigación, analiza la importancia de desarrollar el diseño de un sistema de medición digital que determine el nivel en un tanque, además de llevar un registro sobre la entrada y salida del fluido albergado en el tanque de depósito.

Además de lo expuesto anteriormente, otro punto a considerar es la necesidad de diseñar una interfaz que permita procesar e interpretar de los datos capturados por el sistema de medición

Desde el punto de vista tecnológico, el diseño de un prototipo que cubra estas necesidades resultaría novedoso para la industria nacional, competitiva con la industria extranjera.

## **1.4 ALCANCE**

Este trabajo especial de grado titulado, diseño de un sistema de medición digital para determinar el nivel en un tanque de almacenamiento de líquidos, se enfoca en el análisis y diseño de un sistema prototipo que permita llevar el registro de entrada y salida del fluido a albergar en dicho tanque, mediante el uso del microcontrolador ESP-WROOM-32 en su placa de desarrollo ESP-WROOM -32 DEVKIT V1-DOIT, como elemento de control, el cual permitirá monitorear en tiempo real el inventario presente en el contenedor. Además, del desarrollo en un servidor web el cual permita realizar control remoto sobre las operaciones del sistema de medición y al mismo tiempo visualizar los datos de inventario procesados provenientes del sistema de medición.

## **1.5 LIMITACIONES**

En este estudio no se plantea el uso de internet para la conexión del servidor web, ya que todo el sistema funcionara de manera local, a través de los protocolos TCP/IP y WebSocket. No se considera el uso de otros protocolos de comunicación existentes actualmente

# **CAPITULO 2**

## **MARCO TEORICO**

El presente capítulo está orientado a exponer las investigaciones previas relacionadas con los sistemas de medición de nivel en tanques. De igual manera, define las bases teóricas y conceptuales requeridas para conocer los métodos y normas aplicadas a la medición y registro de nivel en tanques, además, describe los diferentes elementos que conforman al sistema de medición.

### **2.1 ANTECEDENTES DE LA INVESTIGACION**

Son los estudios previos a la investigación que tienen relación y relevancia, entre los cuales se menciona.

C. Valencia [2], realizó una investigación en la empresa EMCALI TELECOMUNICACIONES, dicha investigación se centró en diseñar un sistema para la medición y visualización de los niveles de combustible en los tanques de emergencia de la empresa, considerando los procesos de llenado, mantenimiento y distribución, en los diferentes puntos ubicados en la ciudad de Santiago de Cali, Colombia.

Desarrollo el proyecto documentándose a través de personal calificado en áreas como instrumentación, comunicación y desarrollo de software. Así mismo, realizo un estudio de costos para el desarrollo e implementación de dicha investigación Como resultado de la metodología implementada en el proyecto, pudo establecer los criterios más relevantes para realizar la selección del transductor para la medición de nivel, lo cual, llevo a establecer la tecnología más adecuada a usar para el sistema. También, realizo el diseño para la interfaz hombre maquina (IHM) necesaria para operar el equipo de medición

El trabajo especial de grado elaborado por M. Valencia [3], presenta el diseño y control automático de tanque de almacenamiento de crudo de petróleo para una refinería, este sistema fue diseñado con el propósito de llevar un seguimiento sobre el inventario presente en estos tanques, encargados de realizar la mezcla de diferentes tipos de crudo de petróleo con diferentes calidades, con el fin de conseguir mezclas más óptimas de este hidrocarburo, siguiendo la normativa establecida por la leyes peruanas y la norma API 650 (*American Petroleum Institute*).

La investigación abordó la deficiencia que tenía la empresa respecto a capacidad de almacenamiento de crudo y las condiciones operativas. Determinando así, la capacidad adecuada para el tanque en base a su capacidad de refinación, la cual permita satisfacer la demanda actual de la zona. Además, realizó un análisis en el que se pudo seleccionar y proponer el sistema de medición de nivel más adecuado. Llevando control de los niveles alto y bajo en el tanque, caudal y control de la mezcla. Finalmente presentó el estudio económico el cual mostraba los beneficios del sistema planteado, el incremento en la producción y consecuencias de pérdida económicas por multas a causa de desabastecimiento en su mercado local.

Sainz y Serna [4], a través de su investigación Diseño de entrenador de Arduino con aplicaciones industriales, mencionan que, la plataforma Arduino ha tomado gran relevancia en el mundo de los microcontroladores, ya que poseen una gran capacidad para el desarrollo de diferentes proyectos. A pesar que en el mayor número de casos estos proyectos no se realizan en ambientes industriales, existen muchas aplicaciones que permiten la mejora de los procesos o incluso la sustitución de sistemas antiguos por otros nuevos mediante el uso de una herramienta como Arduino.

El concepto de crear un entrenador para aplicaciones industriales, surgió al hacer la analogía con un equipo para el mismo propósito implementado mediante controladores lógicos programables (PLC). En estos últimos se pueden probar modificaciones al programa antes de ser implementados en áreas reales de trabajo, con lo cual se reduce costos debido a tiempos parada de los equipos

## 2.2 DEFINICION DE MEDICIÓN DE NIVEL

O. Cárdenas [5], define la medición de nivel como “la determinación de la posición de la interface entre dos medios”. Usualmente estos medios son dos fluidos como puede ser un líquido y un gas, un líquido y su vapor, dos líquidos o incluso un sólido o sólido diluido y un gas. Por su parte, A. Creus [6], establece en su libro Instrumentación Industrial que, “en la industria la medición de nivel es muy importante, tanto desde el punto de vista del funcionamiento correcto del proceso como de la consideración del balance adecuado de materias primas o de productos finales”.

## 2.3 MÉTODOS PARA LA MEDICIÓN DE NIVEL

Los métodos más comunes para la medición de nivel en la industria se pueden clasificar en 2 grandes grupos, los de medición directa y los de medición indirecta: [5]

### 2.3.1 Métodos para medición directa de nivel

Como señala O. Cárdenas [5] “los métodos de medición directa utilizan para la medición de nivel, la altura del líquido sobre una línea de referencia”, estos métodos son los más anticuados y usualmente requieren de partes móviles que están sujetas a desgaste, por lo tanto, requieren de un mantenimiento más seguido y poseen una vida útil más corta. Debido a estas causas, no serán tomados en cuenta para el desarrollo de esta investigación.

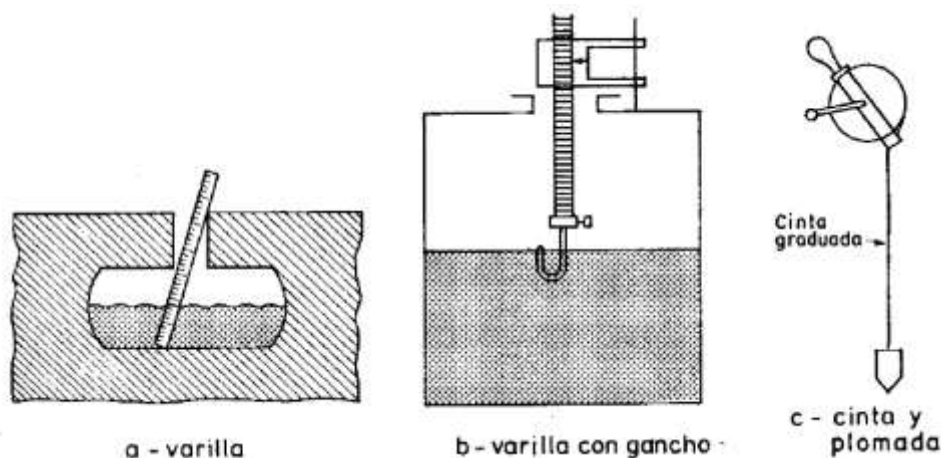


Figura 2.1: Medidores de nivel por principio de medición directa [6].

### 2.3.2 Métodos para medición indirecta de nivel

O. Cárdenas [5] indica que, “Los métodos de medición indirecta miden otros efectos que cambian con el nivel del líquido”. Entre los métodos más comunes para la medición indirecta de nivel de líquidos se encuentran los que aprovechan el empuje producido por el propio líquido, la presión hidrostática y los que aprovechan las características eléctricas del líquido. Sin embargo, para efectos de esta investigación solo se considerarán los que se basan en los 2 últimos principios.

#### 2.3.2.1 Medidores que aprovechan la presión hidrostática:

El nivel de líquido en un tanque abierto se puede medir usando la presión en un punto en el fondo del tanque porque es proporcional a la altura del líquido en ese punto y su gravedad específica [5]. Hay varios tipos de medidores de nivel que funcionan u operan según este principio, a continuación, se describe uno de los métodos más comunes usados en la industria:

##### Medidor de nivel por presión diferencial.

En un sistema de presión diferencial (Figura 2.2), se puede conectar una tubería entre la entrada de baja presión (L) y la parte superior del tanque para medir la diferencia de presión  $\Delta P$  entre la entrada superior e inferior del tanque, pudiéndose calibrar esta  $\Delta P$  en función de la altura del nivel en estudio. Los transductores de presión diferencial de diafragma se utilizan para medir niveles de líquido en recipientes a presión cerrados, aunque también pueden medir niveles de líquido en recipientes abiertos comparando la válvula de baja presión (L) con la presión atmosférica. Dado que el instrumento mide la presión diferencial, la presión estática del nivel del líquido no afecta la medición, por lo que un cambio en la presión estática no provocará errores de medición. Para fines de calibración, el nivel de líquido del eje cero debe seleccionarse a la altura del diafragma para que coincida con el 0% del manómetro. [5]

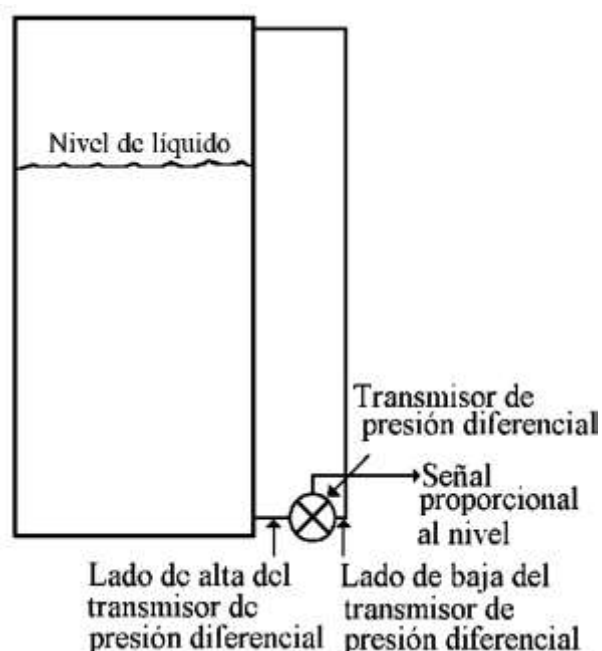


Figura 2.2: Medidor de presión hidrostática tipo Presión diferencial. [5]

### 2.3.2.2 Medidores que aprovechan las características eléctricas del líquido:

#### Medidores de nivel por ultrasonido.

A. Creus [6], indica que el sistema de medición de nivel ultrasónico se basa en transmitir pulsos ultrasónicos a una superficie reflectante y recibir sus ecos en un receptor. El retraso en recibir el eco depende del nivel en el tanque, ver ecuación (2.1). Estas ondas se propagan a través de un medio de gas o vapor con alguna atenuación o reflexión y se reflejan en la superficie de un sólido o líquido.

En la Figura 2.3, se muestra las diferentes disposiciones para la instalación de detectores para alarma o indicación continua de nivel. En el caso de una advertencia de nivel de líquido, el sensor vibra a una frecuencia resonante específica, que se reduce cuando se aplica líquido al sensor. En el segundo caso del indicador de nivel continuo, la fuente ultrasónica genera pulsos que son detectados por el receptor después de un tiempo correspondiente de ida y vuelta de la onda a la superficie del sólido o del líquido.

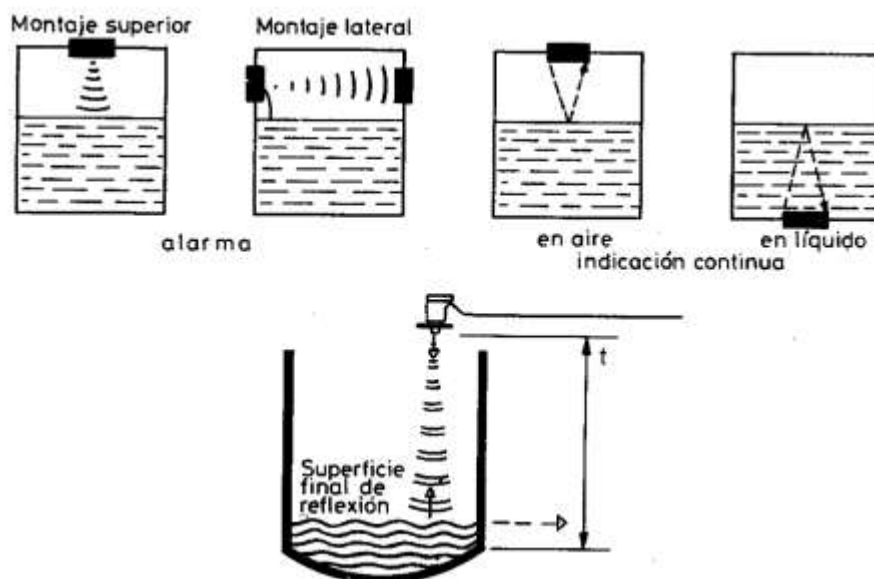


Figura 2.3: Montaje para sistemas de medición y alarma por ultrasonido. [6]

$$D = \frac{(v_s * t_v)}{2} \quad (2.1)$$

Donde:

D = distancia del emisor al líquido.

$v_s$  = Velocidad de propagación del sonido.

$t_v$  = tiempo de recorrido.

A. Creus [6] indica que, la precisión de estos instrumentos es de  $\pm (1 \text{ a } 3) \%$ . Además, son aptos para todo tipo de tanques y líquidos o lodos, pudiendo construirse a prueba de explosión. Su desventaja es que son sensibles a la densidad del líquido y dan señales falsas cuando la superficie del nivel no es transparente, como es el caso de los líquidos espumosos, debido a la creación de falsos ecos ultrasónicos.

El uso de un microprocesador permite que, a través, de la programación se registre el perfil ultrasónico del nivel, teniendo en cuenta las propiedades específicas de la superficie del líquido, como la espuma, lo que mejora la precisión de la medición. Las computadoras, por

otro lado, facilitan la conversión de los niveles de los tanques a volúmenes para su uso en el almacenamiento y brindan una función de autocomprobación del dispositivo. [6]

### La velocidad del sonido respecto a la temperatura

R. Serway y J. Jewett [7] Postulan que la velocidad de las ondas sonoras en un medio depende de la compresibilidad y densidad del medio; si es un líquido o un gas y tiene un módulo de volumen  $B$  y una densidad  $\rho$ , entonces la velocidad de las ondas sonoras en este medio es:

$$v_s = \sqrt{\frac{B}{\rho}} \quad (2.2)$$

De acuerdo con H. Young y R. Freedman [8], Para calcular la velocidad de las ondas sonoras en el aire, se debe recordar que el módulo de volumen de un gas depende de la presión del gas. La expresión para el módulo de volumen de gas utilizada en la ecuación (2.2) es:

$$B = \gamma * P_o \quad (2.3)$$

Donde  $P_o$  es la presión de equilibrio del gas. La cantidad  $\gamma$  (la letra griega gamma) se denomina la razón de capacidades caloríficas. Es un número adimensional que caracteriza las propiedades térmicas del gas. La densidad  $\rho$  de un gas también es dependiente de la presión que, a su vez, depende de la temperatura. Resulta que el cociente  $B/\rho$  para un tipo dado de gas ideal no depende de la presión, sólo de la temperatura. Esto significa que la velocidad del sonido en un gas es esencialmente una función de la temperatura  $T$ , ver ecuación (2.4)

$$v_s = \sqrt{\frac{\gamma * R * T}{M}} \quad (2.4)$$

Donde:

$v_s$  = rapidez del sonido en un gas. [ $\frac{m}{s}$ ]

$\gamma$  = razón de capacidad calorífica del aire [1.40] (valor adimensional)

$R$  = constante de los gases ideales [ $8.314 \frac{J}{mol * K}$ ]

$M$  = masa molar o masa por mol del gas [ $28.969 * 10^{-3} \frac{kg}{mol}$ ] (valor para la mezcla del aire)

$T$  = Temperatura absoluta del medio. [K]

Trabajando en la ecuación (2.4) para sustituir los valores anteriores y llevando la temperatura a grados Celsius [ $^{\circ}C$ ] tenemos que:

$$v_s = \left(331 \frac{m}{s}\right) * \sqrt{1 + \frac{T_c}{273^{\circ}C}} \quad (2.5)$$

### Medidores por radar.

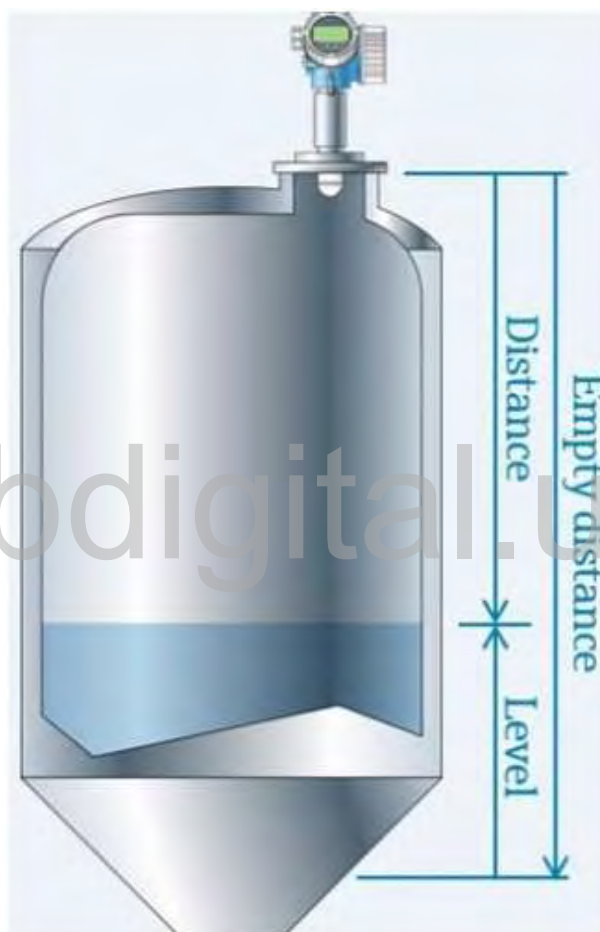
De acuerdo con A. Creus [6], Los sistemas de radar de microondas utilizan la propagación de ondas electromagnéticas que no se ven afectadas por los cambios de temperatura o densidad que pueden ocurrir en los líquidos. Por lo tanto, la espuma que es transparente para señales de radar ya no es un problema como es el caso de los medidores ultrasónicos



Figura 2.4: Medidor de nivel por radar por espacio libre (a) y medidor de nivel por onda guiada (b). [1]

El funcionamiento de la medición por radar es muy similar a como funciona el sistema de medición por ultrasonidos, A. Sánchez [9] describe que, existe un término conocido como ToF (Time of Flight, tiempo de vuelo), es el principio mediante el cual funcionan algunos

radares, radares guiados y sonares. Se refiere al tiempo en que tardar la onda en salir y regresa a la antena. Como se conocen las velocidades de las ondas electromagnéticas (velocidad de la luz) y de las ondas ultrasónicas (velocidad del sonido), es posible medir el tiempo que tardan las ondas en viajar de un lado a otro, de esta manera se puede calcular la distancia de la antena a la superficie del producto. Luego, en función de los datos ingresados sobre las dimensiones del tanque, el dispositivo calcula el nivel de líquido en el tanque, ver Figura 2.5



*Figura 2.5: Sistema de almacenamiento con medidor de nivel por radar. [9]*

La medición por radar se puede hacer de dos maneras. Por un lado, se puede hacer con un tren de pulsos (Figura 2.6a), por otro lado, se puede hacer con una onda continua con modulación de frecuencia, FMCW (onda continua con modulación de frecuencia), ver Figura 2.6b.

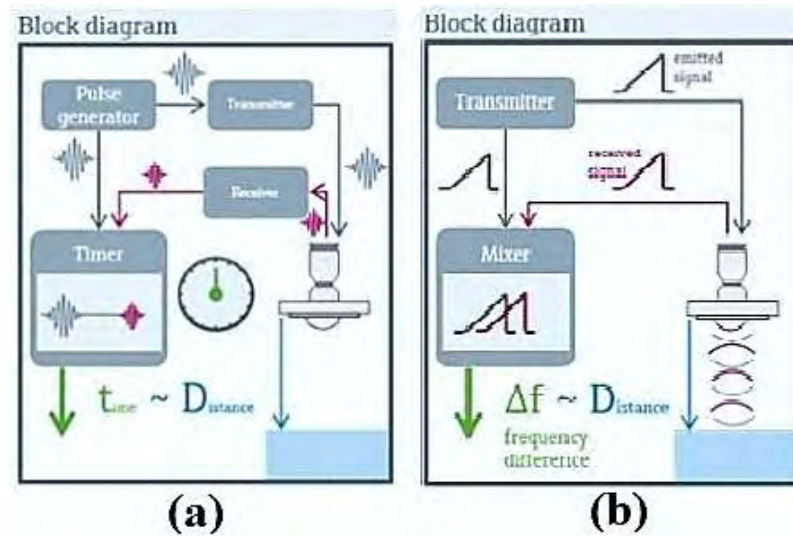


Figura 2.6: Diagramas de bloque para sistemas de medición por radar de pulso (a) y radar FMCW (b). [9]

En el primer método, llamado simplemente "de pulsos", funciona de manera similar al de ultrasonido, consiste en enviar un tren de pulsos y posteriormente obtener el tiempo que transcurre en ir y volver dicha señal hasta la antena, este método ha demostrado ser muy efectivo y se puede utilizar en el 90% de las aplicaciones, una de sus principales ventajas es su rápido tiempo de respuesta y componentes capaces de resistir altas temperaturas, por lo que hay radares TOF que pueden estar en tanques con procesos de hasta 450°C. [9]

De tal manera que si se adapta la ecuación (2.1) sabiendo que el radar propaga una onda electromagnética y no una onda mecánica, cuya rapidez es la velocidad de la luz, la cual es constante e independiente de la temperatura, ver ecuación (2.6), por lo tanto no es necesario hacer compensación en la medición por temperatura del medio de propagación.

$$D = \frac{(C * t_V)}{2} \quad (2.6)$$

Donde:

D = distancia del emisor al líquido[m].

C = Velocidad de la luz  $[\frac{m}{s}]$ .

$t_V$  = tiempo de recorrido[s].

Mientras que en el método de pulsos se envían, reciben y procesan una serie de pulsos, luego se envía otra serie de pulsos, en FMCW la onda electromagnética se envía todo el tiempo, no está modulada por pulsos. Por lo tanto, se requiere otro método para determinar la distancia de la antena desde la superficie del producto. Con este método, la frecuencia varía en una serie de rampas. La antena envía y recibe continuamente y compara la frecuencia de la onda recibida con la frecuencia de la onda transmitida. La diferencia de frecuencia la convierte en el tiempo que tarda la onda en viajar de un lado a otro, ver ecuación (2.7) lo que se traduce en la distancia de la antena al producto (Figura 2.7). [9]

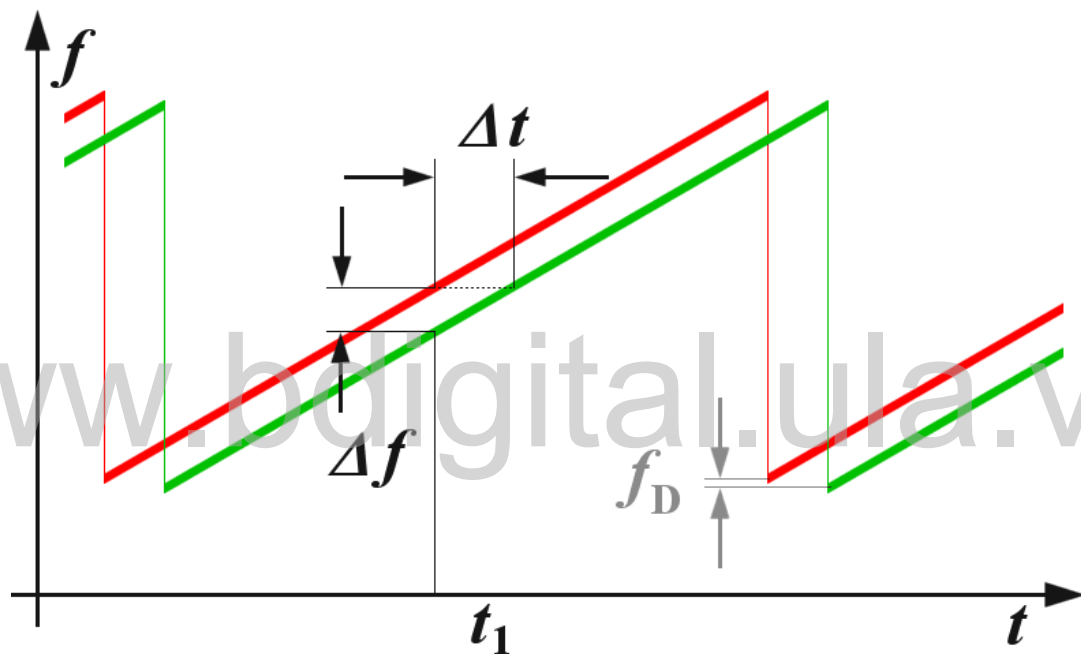


Figura 2.7: Determinación de la distancia mediante el uso de radar por FMCW. [10]

$$D = \frac{(C * |\Delta t|)}{2} \quad (2.7)$$

Donde:

D = distancia del emisor al líquido [m].

C = Velocidad de la luz [ $\frac{m}{s}$ ].

$\Delta t$  = tiempo de propagación [s].

Tabla 2.1: Tipos de medidores de nivel

Instrumento	Campo de medida	Precisión % escala	Presión máxima [bar]	Temperatura máxima del fluido [°C]	Desventajas	ventajas
<b>Presión diferencial</b>	(0-3) m	$\pm (0.15-0.5) \%$	150	200	-Posible agarrotamiento	-Interfaces líquido
<b>Ultrasonido</b>	(0-30)m	$\pm 1 \%$	400	200	-Sensible a densidad	-Todo tipo de tanques y líquidos -Sin contacto
<b>Radar</b>	(0-30)m	$\pm 2.5 \text{ mm}$	-	1500	-Sensible a constante dieléctrica	-Todo tipo de tanques y líquidos -Sin contacto

Nota: adaptada de A. Creus [6]

## 2.4 LA CALIBRACIÓN O AFORO DE TANQUES

M. Granadas y otros [11] definen al aforo de un tanque como, el proceso de determinar la capacidad total del tanque o las capacidades parciales en varios niveles bajo las condiciones operativas existentes. El elemento provisto en el procedimiento de medición es un certificado de calibración o tabla de aforo (ver Figura 2.9). La tabla de aforo muestra la cantidad de producto almacenado en el tanque a un nivel dado, medido desde un punto de referencia.

La calibración del tanque es necesaria cuando la integridad mecánica del tanque se ve afectada por una reparación o un cambio estructural, ya sea por un cambio en la inclinación, el diámetro, la altura de referencia o el espesor de la carcasa. [11]

## 2.5 MÉTODOS PARA LA CALIBRACIÓN DE TANQUES

“La selección del método de calibración dependerá del tipo y tamaño del tanque, la disponibilidad de tiempo, personal y equipo disponible” [11]. A continuación, se presenta una descripción corta de dos métodos aplicables a tanques cilíndricos verticales:

## 2.5.1 Método de “*strapping*” (API MTSM) o medición manual de anillos

El *strapping* de tanque es un término comúnmente usado para el proceso mediante el cual se determina las medidas de un contenedor, mismas que son necesarias para realizar la tabla de calibración o aforo de dicho tanque. Midiendo las circunferencias de cada anillo (Ver Figura 2.8) mediante el uso de una cinta métrica graduada y calibrada comparada con una cinta patrón. El número de mediciones por anillo depende de las uniones y arreglo de las láminas de los anillos del tanque. [11]

### 2.5.1.1 Prácticas generales

El *strapping* solo puede ser realizado luego de haber llenado al menos una vez el tanque al máximo de su capacidad en el lugar de uso con un líquido de igual o mayor densidad al que será albergado en el tanque. Es usual el uso de agua para dichas pruebas durante un lapso de 24 horas. [11]

- Tanques de 500 barriles o menos: se pueden calibrar en cualquier condición siempre y cuando se cumpla lo anterior
- Tanques de 500 barriles o más: Los tanques atornillados (*bolted joint*) pueden ser aforados estando llenos hasta con 2/3 y deben haber sido llenados por lo menos una vez antes del *strapping*

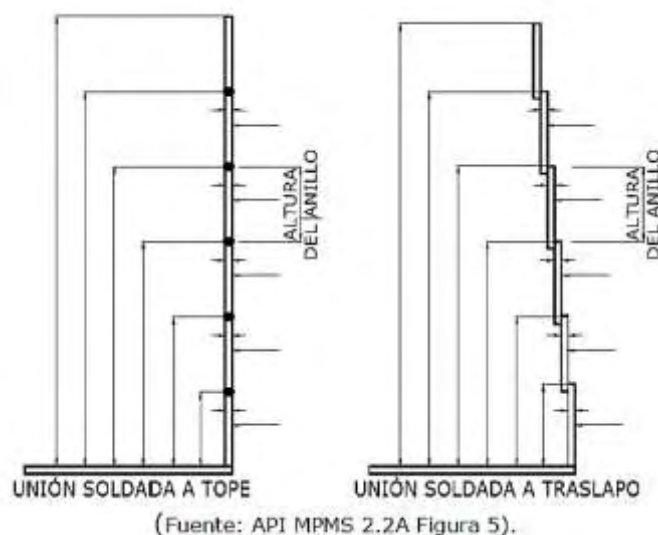


Figura 2.8: Posiciones para la medición de circunferencias en tanques soldados. [11]

## **2.5.2 Método de calibración líquida (API STD 2555)**

M. Granadas y otros [11] describen que, este método consiste en determinar la capacidad de un tanque mediante la transferencia de un volumen de líquido conocido desde o hacia otro tanque de menor tamaño, el cual haya sido previamente calibrado. Este método posee la ventaja de poder usarse en tanques de cualquier forma o diseño excepto en probadores de medidores. Ya que este método es lento, se aconseja para tanque de pequeño volumen, tanques horizontales, tanques subterráneos o aquellos que debido a su forma no puede aplicarse método geométrico

### **2.5.2.1 Procedimiento con un medidor de flujo**

Se debe instalar el medidor según las recomendaciones del fabricante, tan cerca como sea posible al tanque a aforar. El medidor de flujo debe ser probado previamente con el mismo líquido con el cual se hará la calibración. Cada vez que se realice una parada es necesario tomar nota del nivel del líquido, la temperatura del líquido en el tanque y temperatura promedio del líquido entregado por el medidor [11]

### **2.5.2.2 Procedimiento con tanque volumétrico**

Para este proceso se usa un tanque previamente aforado de menor tamaño al que se pretende calibrar, y debe estar lo más cerca posible al tanque más grande. Se recomienda usar agua para realizar dicha prueba, aunque se puede usar otros fluidos no volátiles. Además, se debe verificar las condiciones de las líneas de llenado y la hermeticidad de ambos depósitos, posteriormente se da inicio a la transferencia de líquido, se debe registrar la temperatura del líquido en tanque a ser calibrado como también la del tanque calibrador para poder realizar los ajustes si cambia la temperatura.

El trabajo debe ser realizado en la medida de lo posible sin interrupciones, ya que al haber interrupciones puede darse el caso de que cambié la temperatura, por consiguiente, introducir errores en la medición. Si la temperatura del líquido se aproxima a la temperatura ambiente se obtienen resultados más precisos. [11]

Tabla 2.2: Resumen métodos de calibración

Método calibración	Ventajas	Desventajas	Equipos necesarios
<b>Líquida</b>	<ul style="list-style-type: none"> <li>• Más exacto de todos</li> <li>• Se completa usando en el fondo el método lineal</li> <li>• Uso indicado en los tanques con formas irregulares</li> </ul>	<ul style="list-style-type: none"> <li>• se ejecuta en lapsos de tiempo muy largos, recomendado para tanques pequeños</li> </ul>	<ul style="list-style-type: none"> <li>• Medidor de flujo y demás indicados en la norma API STD 2555</li> </ul>
<b>Método Lineal</b>	<ul style="list-style-type: none"> <li>• Recomendado para tanques centrales , que no presentan variaciones de forma</li> </ul>	<ul style="list-style-type: none"> <li>• No indicado en tanques que presenten formas irregulares como los extremos laterales</li> </ul>	<ul style="list-style-type: none"> <li>• Cinta de medición para longitudes y alturas y demás indicados en la norma API MPMS Capitulo 2 Sección 2-A</li> </ul>

Nota: adaptada de M. Granadas y otros [11]

## 2.6 ELABORACIÓN DE LA TABLA DE AFORO

Los detalles para la elaboración de la tabla de aforo se especifican en el manual API MPMS 2.2ª numeral 2.2ª.19, *Tank Capacity Table Development: Calculation Procedure* [12]. A continuación la Figura 2.9 muestra un modelo de para una tabla de aforo:


NOMBRE Y LOGO DE EMPRESA AFORADORA			 <b>ECOPETROL S.A.</b> VICEPRESIDENCIA DE (TRANSPORTE) GERENCIA DE (POLIDUCTOS) PLANTA (PUERTO SALGAR) PUERTO SALGAR (CUNDINAMARCA)												TANQUE K-XXX *	
Dirección, teléfono, correo electrónico, página web, NIT, etc.															PRODUCTO: Nombre del producto	
FONDO			CILINDRO PRINCIPAL												FRACCIONES	
NIVEL	VOLUMEN	INCREMENTO	NIVEL	VOLUMEN	NIVEL	VOLUMEN	NIVEL	VOLUMEN	NIVEL	VOLUMEN	NIVEL	VOLUMEN	NIVEL	VOLUMEN	NIVEL	VOLUMEN
cm	Bls	Bls / mm	cm	Bls	cm	Bls	cm	Bls	cm	Bls	cm	Bls	cm	Bls	cm	Bls
0	1,38		5	50,00	240	2.737,32	480	5.493,62	720	8.250,90	960	11.007,33	1	11,48		
0,1	1,48	0,01	10	95,44	250	2.852,16	490	5.608,47	730	8.364,96	970	11.122,21	2	22,97		
0,9	3,70	0,22	20	210,78	260	2.967,01	500	5.723,32	740	8.479,83	980	11.237,07	3	34,46		
17	7,86	0,42	30	325,62	270	3.081,86	510	5.838,17	750	8.594,72	990	11.351,92	4	46,84		
2,7	16,44	0,78	40	440,47	280	3.196,70	520	5.953,03	760	8.709,61	1000	11.466,76	5	67,42		
4,6	34,29	1,89	50	555,31	290	3.311,55	530	6.067,88	770	8.824,49	1010	11.581,61	6	89,91		
6,0	60,00	1,67	60	670,15	300	3.426,39	540	6.182,73	780	8.939,38	1020	11.696,46	7	80,39		
			70	784,99	310	3.541,24	550	6.297,59	790	9.054,26	1030	11.811,31	8	91,88		
			80	899,84	320	3.656,09	560	6.412,44	800	9.169,15	1040	11.926,16	9	103,38		
			90	1.014,68	330	3.770,93	570	6.527,30	810	9.284,04	1050	12.041,00				
			100	1.129,52	340	3.885,78	580	6.642,15	820	9.398,92	1060	12.155,85				
			110	1.244,36	350	4.000,62	590	6.757,00	830	9.513,81	1070	12.270,69				
			120	1.359,21	360	4.115,47	600	6.871,86	840	9.628,69	1080	12.385,54				
			130	1.474,05	370	4.230,31	610	6.986,71	850	9.743,58	1090	12.500,39	1	1,16		
			140	1.588,89	380	4.345,16	620	7.101,56	860	9.858,47	1100	12.615,24	2	2,30		
			150	1.703,74	390	4.460,01	630	7.216,42	870	9.973,35	1110	12.730,08	3	3,44		
			160	1.818,58	400	4.574,85	640	7.331,27	880	10.088,24	1120	12.844,93	4	4,68		
			170	1.933,42	410	4.689,70	650	7.446,13	890	10.203,12	1130	12.959,78	5	6,74		
			180	2.048,26	420	4.804,54	660	7.560,98	900	10.318,01	1140	13.074,62	6	8,89		
			190	2.163,11	430	4.919,39	670	7.675,83	910	10.432,90	1150	13.189,47	7	8,04		
			200	2.277,95	440	5.034,24	680	7.790,69	920	10.547,78	1160	13.304,32	8	9,18		
			210	2.392,79	450	5.149,08	690	7.905,54	930	10.662,67	1170	13.419,17	9	10,33		
			220	2.507,63	460	5.263,93	700	8.020,39	940	10.777,55	1180	13.534,01				
			230	2.622,48	470	5.378,77	710	8.135,25	950	10.892,44	1190	13.648,86				
<b>TANQUE CILINDRICO VERTICAL SOLDADO</b>																
ALTURA DE REFERENCIA, m		DATO														
CAPACIDAD NOMINAL, bls		DATO														
DIAMETRO DEL CILINDRO, m		DATO														
ALTURA EFECTIVA, m		DATO														
ESFUERZO POR CABEZA DE		DATO														
LÍQUIDO, GRAVAPIA 60 °F		DATO														
TEMPERATURA BASE DE LÁMINA, °F		DATO														
TIPO DEL TECHO:		DATO														
TIPO DEL FONDO:		DATO														
<b>INCERTIDUMBRE ESTIMADA</b>																
FACTOR DE COBERTURA		2														
NIVEL DE CONFIANZA		95%														
INCERTIDUMBRE EXPANDIDA		0,00%														
INCERTIDUMBRE RELATIVA		0,00%														
La aplicación del nivel de la tabla es para medir a fondo. La calibración del tanque y los cálculos fueron realizados según API MPMS Capítulo 2, sección 2A "Measurement and Calibration of Upright Cylindrical Tanks by the Manual Tank Strapping Method" Primera Edición febrero de 1995, reconfirmada febrero 2.007 y Sección 2B, "Optical Reference Line Method" Primera Edición, marzo 1999, reconfirmada marzo de 2.007																
FECHA DE CALIBRACIÓN DD/MM/AAAA																
AFORADO POR:					REVISADO POR:					REP. MINISTERIO MINAS Y ENERGÍA:						
NOMBRE TÉCNICO/INGENIERO					Responsable verificación					NOMBRE/FIRMA INSPECTOR						
Matrícula Profesional					Matrícula Profesional											

Figura 2.9: Tabla de aforo para un tanque de la planta Puerto Salgar de la empresa ECOPETROL. [11]

## 2.7 EL MICROCONTROLADOR EN LOS SISTEMAS EMBEBIDOS

### 2.7.1 El microcontrolador

D. Apaza [13] Establece que el microcontrolador es un computador integrado en un chip de características limitadas, destinado a controlar una sola tarea. Dicho circuito integrado posee una memoria destinada a albergar un conjunto de instrucciones las cuales constituyen lo que se conoce como programa.

### 2.7.2 ¿Qué son los sistemas embebidos?

De acuerdo con F. Bordignon y A. Iglesias [14], “un sistema embebido es una combinación de hardware y software que trabaja junto con algún sistema mecánico o electrónico diseñado para cumplir una función específica. Por lo tanto, se usa para dotar de inteligencia a un artefacto”. Un sistema embebido está diseñado para desempeñar una tarea en específico ya que su *hardware* y *software* es implementado según la tarea a requeridas. El *hardware* de un sistema embebido esta principalmente compuesto por un microprocesador, transductores y actuadores.

### 2.7.3 Bus de comunicación

Según F. León [15], un bus de comunicación hace referencia al medio mediante el cual se transmiten los datos entre 2 o más dispositivos electrónicos.

#### 2.7.3.1 Bus 1-wire

Como plantea L. Llamas [16], el bus *1-wire* permite la conexión de múltiples periféricos a través de un solo cable, se basa en un complejo sistema de *timming* para enviar y recibir las señales. Como desventaja se tiene un código complejo para realizar la consulta de los sensores conectados a dicho bus. (Figura 2.10). Para poder usar varios periféricos conectado al mismo bus por defecto el fabricante asigna una dirección de 64 bits, en la cual los primeros 8 bits corresponden a la familia del periférico (0x28 para el DS18B20). Los 48 bits subsecuentes son el número de serie y los últimos 8 bits son un código CRC, Ver Figura 2.11

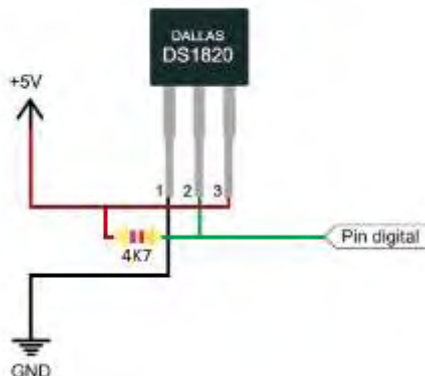


Figura 2.10: Conexión del sensor DS18B20 de la empresa Dallas mediante bus 1-wire. [16]

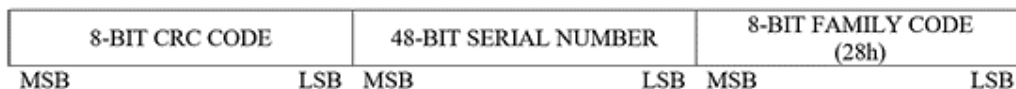


Figura 2.11: Secuencia de bits para la identificación de cada sensor 1-wire. [16]

### 2.7.3.2 Bus I<sup>2</sup>C

Con base en L. Llamas [17], el estándar I<sup>2</sup>C (*Inter-Integrated Circuit*) solo requiere de 2 cables para realizar su comunicación, uno para la señal de reloj (SCL) y otro para el envío de datos (SDA), ver Figura 2.12, lo cual representa una clara ventaja si se compara con el Bus SPI. Sin embargo, requiere de una electrónica más compleja para su implementación y posee una menor velocidad de comunicación, ya que los datos en ambos sentidos de la comunicación se transmiten por el mismo conductor.

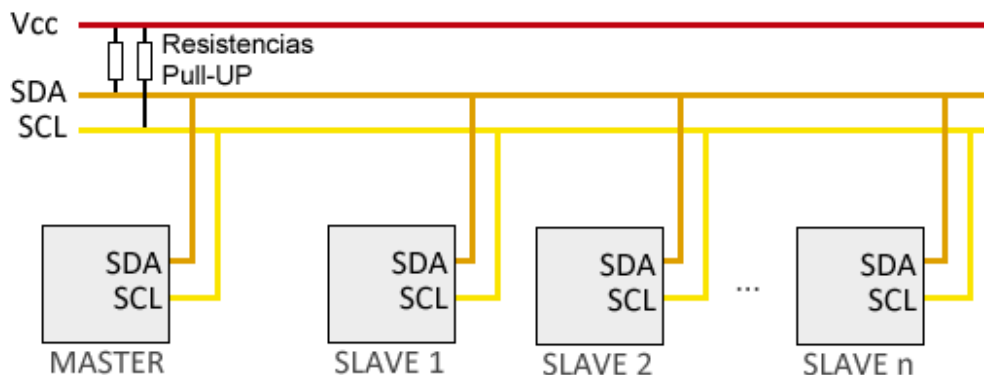


Figura 2.12: Esquema de conexión para múltiples periféricos con funcionamiento por I<sup>2</sup>C. [17]

L. Llamas [17] explica que, El bus I<sup>2</sup>C posee una arquitectura del tipo maestro-esclavo. Esto quiere decir que para iniciar la comunicación el dispositivo maestro (microcontrolador) debe preguntar al dispositivo esclavo (periféricos) si está disponible para establecer la comunicación, esto es posible debido que en este tipo de protocolo cada dispositivo esclavo contiene una dirección hexadecimal única (en teoría hasta 128 dispositivos) con la cual el microcontrolador puede acceder a la información de cada periférico de forma independiente.

### 2.7.3.3 Bus SPI

Como expresa L. Llamas [18], el bus SPI (*Serial Peripheral Interface*) tiene una arquitectura de tipo maestro-esclavo al igual que Bus I<sup>2</sup>C, de igual manera el dispositivo maestro (microcontrolador) es el que inicia la comunicación con uno o varios dispositivos esclavos (periféricos). Los periféricos no pueden iniciar la comunicación, ni intercambiar datos entre ellos de forma directa. En el bus SPI a diferencia del I<sup>2</sup>C la comunicación es full dúplex (ver Figura 2.13) lo que quiere decir que el envío de datos se realiza por un cable independiente al de recepción de datos, esto permite manejar velocidades de transmisión de datos más altas, ya que el maestro puede enviar y recibir datos a la misma vez.

El bus SPI también es síncrono, ya que cuenta con una señal de reloj que permite mantener todos sus dispositivos sincronizado. Esto reduce la complejidad del sistema frente a los sistemas asíncronos.

Por tanto, el bus SPI requiere un mínimo de 3 líneas.

- MOSI (*Master-out, Slave-in*) para la comunicación del maestro al esclavo.
- MISO (*Master-in, Slave-out*) para comunicación del esclavo al maestro.
- SCK (*Clock*) señal de reloj enviada por el maestro.

Como una desventaja frente a I<sup>2</sup>C este bus requiere una línea adicional SS (*Slave Select*) para cada dispositivo esclavo conectado, por el cual el maestro indica a cada periférico cuando se va a comunicar.

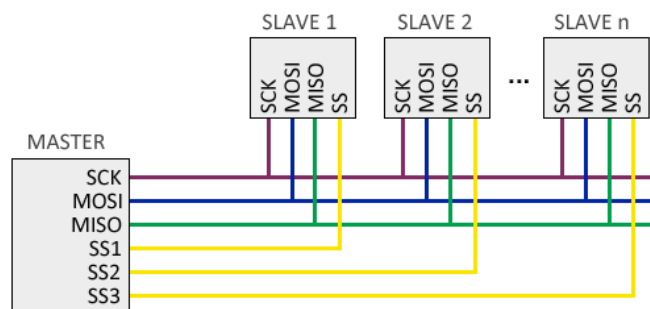


Figura 2.13: Diagrama de conexión para protocolo de comunicación SPI. [18]

## 2.8 SOFTWARE

Lázaro [19], define al software como “la parte inmaterial del ordenador que permite que los diferentes componentes de hardware funcionen. Se trata de un conjunto de instrucciones, datos o programas que ejecutan tareas concretas dentro del sistema informático”. En ciertas ocasiones se hace referencia al software como la parte que varía del ordenador o los estados que toma la máquina y las señales que fuerzan dichos estados.

Para la elaboración del prototipo para el sistema de medición de tanques se utilizó el entorno de desarrollo integrado (IDE) proporcionado por Arduino, el cual es de uso libre y gratuito, este puede ser descargado a través de la página web oficial de Arduino

### 2.8.1 Arduino *Integrated Development Environment* (IDE)

Según Crespo [20], es un programa informático el cual posee las herramientas necesarias para el desarrollo de otros software. Un IDE es un entorno de programación el cual posee un editor de código, un compilador, un depurador y un constructor de interfaz gráfica (GUI). Además, en el caso particular de Arduino incorpora las herramientas para cargar el programa ya compilado en la memoria flash del microcontrolador.

Como afirma [21], Arduino usa un lenguaje propio el cual se basa en el lenguaje de programación de alto nivel *Processing*, lo que significa que es similar a C++.

## 2.8.2 Servidor

García [22] define al servidor como una computadora u otro tipo de dispositivo informático que se encarga de proporcionar información a una amplia gama de clientes, ya sean humanos u otros dispositivos conectados a él. Los datos que se pueden transferir pueden ser provenientes de programas informáticos, bases de datos, archivos de texto, etc.

## 2.8.3 Protocolos de comunicación

Un protocolo es un método estándar que permite la comunicación entre procesos (capaces de ejecutarse en diferentes computadoras) y un conjunto de reglas y procedimientos que deben seguirse para enviar y recibir datos a través de una red. [23]

### 2.8.3.1 Protocolo TCP/IP

El Protocolo de control de línea es uno de los protocolos básicos en Internet. Es uno de los principales protocolos de capa de transporte del modelo TCP/IP. A nivel de aplicación, permite gestionar los datos entrantes o salientes al nivel más bajo del modelo (es decir, el protocolo IP). Dado que es un protocolo orientado a la conexión, permite que dos máquinas conectadas controlen el estado de transmisión. Las características principales de TCP son las siguientes: Admite la mayoría de las aplicaciones de Internet más populares, incluidas HTTP, SMTP, SSH y FTP. [23]

### 2.8.3.2 Protocolo HTTP

El protocolo de transferencia de hipertexto es el protocolo más utilizado en Internet. Es el protocolo usado en cada transacción de la Web. El propósito del protocolo HTTP es permitir la transferencia de archivos (a menudo en formato HTML) entre un navegador (cliente) y un servidor web. HTTP define la arquitectura y la semántica que utilizan los elementos de software en la arquitectura web (clientes, servidores y *proxies*) para comunicarse. Es un protocolo orientado a transacciones que sigue el modelo de petición- respuesta entre cliente-servidor. El cliente que realiza la solicitud (el navegador) se denomina "agente de usuario". La información que se va a transmitir se denomina recurso y se identifica mediante una

cadena de caracteres denominada URL. Los recursos pueden ser archivos, resultados de programas en ejecución, consultas a bases de datos, etc. [23]

### 2.8.3.3 Protocolo *WebSocket*

Un *WebSocket* es una conexión permanente entre el cliente (navegador) y el servidor (ESP32) que permite la comunicación bidireccional entre los dos a través de una conexión TCP. Esto significa que puede enviar datos de cliente a servidor y de servidor a al instante. El cliente establece una conexión *WebSocket* con el servidor a través de un proceso conocido como protocolo de enlace *WebSocket*. La conexión inicia una solicitud/respuesta HTTP, lo que permite que el servidor maneje conexiones HTTP y conexiones *WebSocket* en el mismo puerto. Una vez que se establece la conexión, tanto el cliente como el servidor pueden enviar datos en modo full dúplex. Gracias al protocolo *WebSocket*, el servidor puede enviar información a todos los clientes sin que se lo pidan. También permite enviar información al navegador web cuando se realizan cambios. Este cambio podría ser algo que sucedió en la página web (hizo clic en un botón) o algo que sucedió en el lado del ESP32, como presionar un botón físico en un circuito eléctrico. [24]

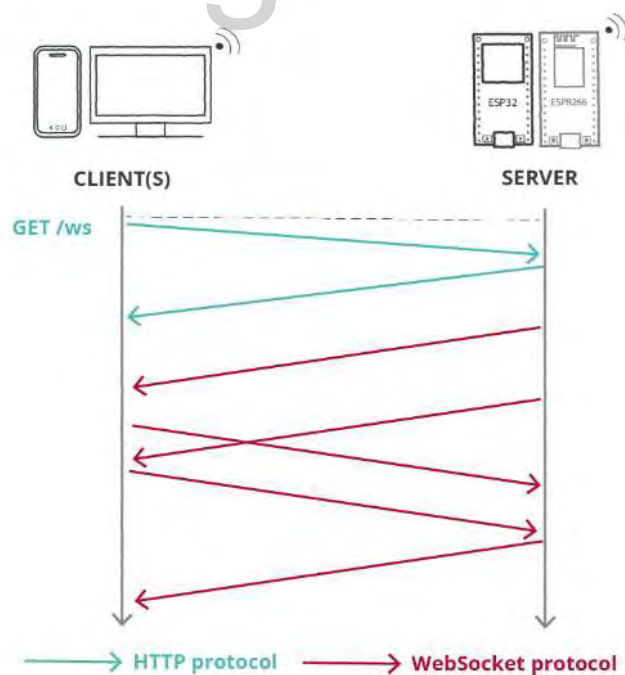


Figura 2.14: Diagrama de secuencia de una comunicación *WebSocket*

## 2.9 *HARDWARE*

Como expresa A. Lázaro [19], El *hardware* se refiere al conjunto de componentes físicos que interaccionan entre sí ya sea de manera analógica o digital, para dar forma al sistema embebido. El hardware es el soporte físico sobre el que se instala, opera y funciona cualquier software; esto quiere decir que, sin el *hardware*, no existe el ordenador.

### 2.9.1 **ESP-WROOM-32 DEVKIT V1-DOIT (SoM).**

Según [25], el ESP-WROOM -32 DEVKIT V1-DOIT es una placa de desarrollo muy parecida a Arduino en ciertas características, pero orientada especialmente al internet de las cosas (*IoT*). La placa de desarrollo cuenta con un núcleo SoM (*System On a Module*) ESP-WROOM-32 que a su vez está basado en el SoC (*System On a Chip*) ESP-32, así mismo trabaja con un convertidor USB-Serial TTL CP210x y conector micro-USB necesario para realizar la programación del microcontrolador. Además de poder establecer comunicación con el ordenador a través de este puerto, ESP-WROOM-32 está diseñado especialmente para trabajar de manera inalámbrica ya que cuenta con la capacidad de establecer conexiones mediante *wi-fi* y *Bluetooth*, su diseño es práctico para ser usado en el desarrollo en *protoboard* o soldado sobre una placa. Cuenta con regulador de voltaje de 3.3V, esto permite alimentar la placa directamente del puerto micro-USB o por los pines 5V y GND. Ya que los pines de entradas/salidas (*GPIO*) con los cuales se comunica esta placa trabajan a 3.3V se requieren de convertidores lógicos de nivel para periféricos que operen a 5V

El *SoM* ESP-WROOM-32 de *Espressif Systems*, integra el poderoso microcontrolador Tensilica Xtensa LX6 con doble núcleo de 32 bits, el cual puede trabajar con velocidades de hasta 240 MHz. Además, incorpora una memoria flash externa de 4 Mb (pudiendo ser de hasta 16 Mb), cristal oscilador y antena wifi en el PCB (ver Figura 2.15). [25]

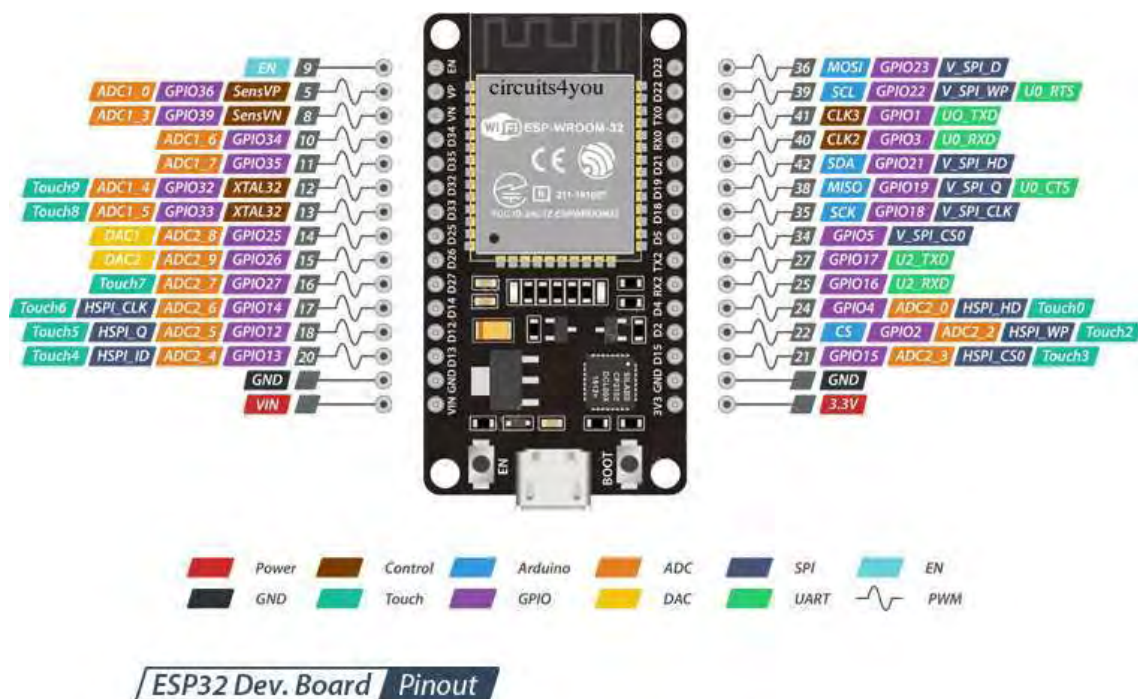


Figura 2.15: Esquema de pines para el módulo ESP-WROOM-32 DEVKIT V1-DOIT [25]

www.bdigital.ula.ve

# **CAPITULO 3**

## **MARCO METODOLÓGICO**

### **3.1 TIPO DE INVESTIGACIÓN**

Por las características mostradas, esta investigación está enmarcada en las investigaciones proyectivas [26], porque consiste en la elaboración de una propuesta abordando problemas prácticos de índole tecnológicos y centrándose en una aplicación concreta que dé respuesta a un problema específico. Este tipo de investigación engloba “inventos, programas, diseños o creaciones dirigida a cubrir una determinada necesidad” [26]

### **3.2 DISEÑO DE LA INVESTIGACION**

Para lograr los objetivos planteados, esta investigación se diseñó en base al modelo incremental de la metodología de desarrollo de software. Maida y Pacienza [27], expresan que “este modelo permite construir el proyecto en etapas incrementales en donde cada etapa agrega funcionalidad. Estas etapas, consisten en requerimientos, diseño, codificación, pruebas y entrega” (ver Figura 3.1).

Pressman [28].especifica que, cuando se utiliza un modelo incremental, es común que el primer incremento este orientado a cumplir las características fundamentales del producto. En decir, este incremento obvia funcionalidades complementarias que de momento puede que sean conocidas o incluso puede darse el caso que surjan nuevas funcionalidades en las iteraciones posteriores. Por consiguiente, el cliente debe poner a prueba el producto fundamental permitiendo crear la evaluación necesaria para trazar el siguiente incremento. Este proceso debe ser repetido después de la entrega de cada nuevo incremento, hasta lograr

cumplir con las características del producto final. El modelo incremental se centra en hacer entrega al final de cada incremento de un producto funcional



Figura 3.1: Modelo del ciclo de Vida incremental. [27]

A continuación, se describen las etapas que constituyen el proceso de iteración o incremento de la siguiente manera:

### 3.2.1 Análisis del sistema

Teniendo en cuenta a [27], el proceso para la recolección de requisitos se enfoca esencialmente en el software. Es trabajo del analista a cargo del proyecto familiarizarse con el proceso al que se debe aplicar el programa a diseñar, haciendo especial énfasis en la funcionalidad, rendimiento y las interfaces requeridas. Como señala [29], en esta etapa se puede entender la esencia del problema al que se está buscando solución, determinando las acciones específicas necesarias para conseguir la solución. Las actividades a realizar durante esta fase son: (a) precisar las necesidades del sistema, (b) considerar los recursos de hardware necesarios para implementar la solución, por ejemplo: equipos, módulos, sensores, etc., y (c) puntualizar los requerimientos de software necesarios.

### 3.2.2 Diseño

La etapa de diseño del programa hace enfoque en cuatro atributos diferentes del software: (a) la estructura de los datos, (b) la arquitectura del software, (c) el detalle procedimental y (d)

la caracterización de la interfaz (plantear prototipos de pantalla). El proceso de diseño permite convertir los requerimientos en una representación del software como paso previo a la codificación. [27]

### **3.2.3 Programación**

Según [27], consiste en el proceso de escribir las instrucciones que deberá seguir el equipo en un lenguaje que la maquina pueda interpretar. Si la etapa de diseño se realiza de manera adecuada entonces la codificación se puede realizar de modo más rápido y sencillo. Por otro lado [29], indica que para programar es necesario primero conocer la sintaxis del lenguaje a usar, parte de las obligaciones de esta etapa son: (a) Escribir el código fuente y (b) realizar pruebas unitarias y de funcionalidad.

### **3.2.4 Pruebas**

Con base en [27], ya concluido el proceso de escritura del código fuente es necesario empezar la etapa de pruebas, en este proceso se evalúan aspectos como la lógica interna y externa, efectuando pruebas que garanticen que las entradas definidas produzcan los consecuencias previstas. Como hace notar [29], las actividades pertinentes a esta etapa son: (a) realizar la inspección del prototipo para garantizar la conexión correcta de todo el hardware asociado en el lugar de pruebas y (b) efectuar las pruebas garantizar que el equipo cumple con lo previsto.

## **3.3 TÉCNICAS O INSTRUMENTOS PARA RECOLECCIÓN DE DATOS**

En todo trabajo de investigación se requiere de técnicas o herramientas que permitan al investigador estar informado a cerca de tema a tratar, de acuerdo con , Hernández y otros [30] “recolectar los datos implica elaborar un plan detallado de procedimientos que nos conduzcan a reunir datos con un propósito específico”, en tal sentido, Arias [31], señala que un instrumento de recolección de información es “Cualquier recurso, dispositivo o Formato (en papel o digital), que se utiliza para obtener, registrar o almacenar información” siguiendo el orden de ideas de este último autor para realizar la recolección de información requerida

en esta investigación se usó el método de análisis documental, más específicamente usando a la computadora como principal método para la consulta de fuentes bibliográficas como libros, manuales y normas, además de otras fuentes digitales como páginas web y tesis [31].

### **3.4 METODOLOGÍA PARA EL DISEÑO DEL SISTEMA DE MEDICION**

Basado en lo explicado anteriormente en el DISEÑO DE LA INVESTIGACION, en relación al modelo de metodología incremental, a continuación se expone la primera etapa del proceso de iteración para el diseño del sistema de medición:

#### **3.4.1 Etapa 1: Análisis de los requisitos del sistema**

##### **3.4.1.1 Requerimientos generales del sistema**

Cuando nos referimos a software para instrumentos de medida, se hace referencia a programas especialmente desarrollados para la medición de variables de proceso mediante herramientas y la gestión y/o configuración de los parámetros correspondientes [32]. En tal sentido, los requerimientos planteados en el análisis para la creación del prototipo de sistema de medición de tanques que debe cumplir el software son:

- a) El sistema debe ser de fácil uso, con una interfaz intuitiva la cual permita operar el equipo de medición sin muchas dificultades.
- b) Debe ser de fácil instalación, esto es tanto para la instalación como para el mantenimiento periódico del equipo.
- c) Bajo costo de producción, para garantizar que el equipo sea accesible y factible para la empresa.
- d) El sistema debe llevar un registro diario del inventario actual, fecha y hora en la cual se modifica el nivel del líquido en el tanque, además de los litros que ingresan o salen en total en cada operación.
- e) El registro de la información debe ser realizado en una memoria tipo Micro SD, para que posteriormente pueda ser consultado a través de una interfaz web local.

- f) Para poder realizar una comprobación en tiempo real del inventario, el sistema debe contar con una pantalla para mostrar datos como el nivel actual del tanque.

### 3.4.1.2 Análisis de la plataforma *hardware/software*

Para la elaboración del software que usará el sistema de medición y registro de nivel en tanques se empleará la herramienta de código abierto Arduino IDE, en conjunto con HTML5, JavaScript y CSS3 para implementar la página web que servirá de interfaz hombre máquina permitiendo realizar control en los procesos del sistema y a su vez visualizar el estado de la máquina y los reportes almacenados en la memoria micro SD. En cuanto al hardware se maneja el microcontrolador ESP-32, más concretamente trabajando con la placa de desarrollo ESP-WROOM-32 DEVKIT V1-DOIT (SoM). Entre los módulos y sensores usados (ver Figura 3.2) se encuentran el DS18B20 (sensor digital de temperatura), DS3231 (RTC), pantalla LCD 16x2 con módulo para comunicación I<sup>2</sup>C incorporado, módulo para el manejo de la memoria Micro SD, convertidor de nivel lógico de 8 canales y como elemento principal de medición el sensor ultrasónico HC-SR04 (ver Figura 3.2). Para establecer la comunicación entre el microprocesador y los módulos se usarán protocolos de comunicación 1-wire, I<sup>2</sup>C y SPI.

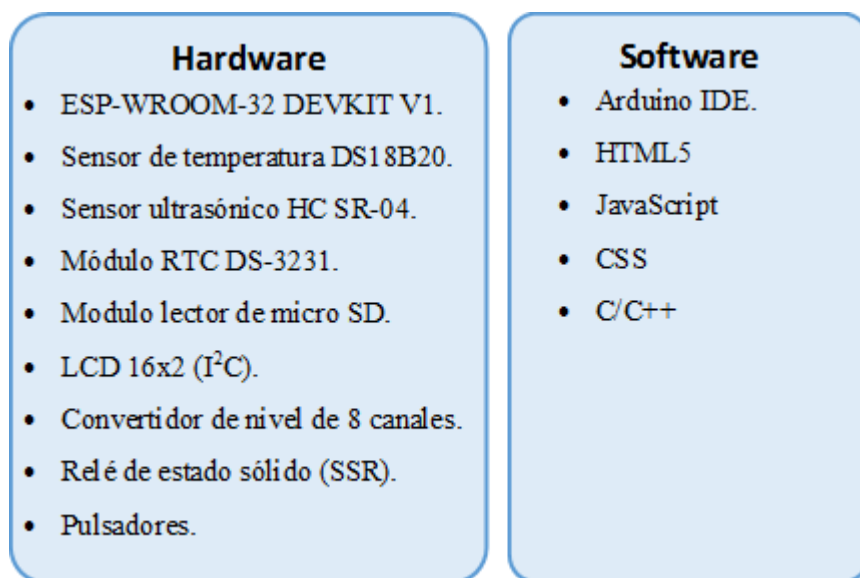


Figura 3.2: Análisis de Hardware y software requerido para la implementación del prototipo de medición de nivel de tanques.

### 3.4.1.3 Normativas legales

En la etapa de análisis también es necesario contemplar los criterios legales que deben orientar el estudio. De acuerdo con Hägg y Sandberg [1], existen estándares internacionales para la medición de tanques que son de gran relevancia. Estos cumplen el objetivo de servir como guía para los fabricantes de equipos de medición de tanques. En tal sentido ISO (Organización Internacional de Estandarización) y API (instituto americano de petróleo) son los responsables con mayor relevancia en cuanto a medición de tanques se refiere, sin embargo, existen otras organizaciones, las cuales tienen un aporte significativo en esta área de la metrología, a continuación, se nombran algunas de estas organizaciones:

- **Instituto Americano de Petróleo (API):** la función principal de esta organización es la de ser el portavoz de la industria de los hidrocarburos con el fin de tener influencia en la política pública en apoyo de la industria. Entre sus funciones también destacan la investigación y la certificación de los estándares de la industria [1].
- **Organización Internacional de Normalización (ISO):** cumple con la función de establecer y publicar normas que puedan crear un estándar mundial para empresas y organizaciones que trabajan en la elaboración de productos y equipos. Anteriormente ISO y API poseían diferencias en sus estándares para la medición de tanque, pero con el pasar de los años ocurrió una homologación entre ambas organizaciones, provocando así que ISO ya no emita nuevas normas en el área de medición de tanques, por el contrario, ahora colabora junto a API en la revisión y creación de nuevos estándares para la medición de tanques [1].
- **Organización Internacional de Metrología Legal (OIML):** Es una organización intergubernamental. Promueve y facilita la normalización internacional de los procedimientos metrológicos legales que sustentan el comercio internacional. La certificación metrológica reglamentaria garantiza que la certificación de equipos de medición en un país sea compatible con la certificación de otro país. La OIML ha desarrollado pautas para ayudar a sus miembros a desarrollar legislación relevante y pautas de certificación para metrología.

# **CAPITULO 4**

## **DISEÑO DEL HARDWARE PARA EL SISTEMA DE MEDICION DE NIVEL**

En el presente capítulo se explicará a detalle la conexión del hardware y la funcionalidad que proporcionará cada módulo al sistema de medición e inventario.

### **4.1 ETAPA 2: DISEÑO DEL *HARDWARE***

En el diseño del sistema de medición de nivel, se aborda el estudio e implementación de los diferentes módulos, sensores, actuadores y pulsadores, con los cuales se añade las funcionalidades necesarias para dar solución a lo planteado en la sección 3.4.1, considerando este análisis, se diseñó un diagrama de bloques que puede ser observado en la Figura 4.1, con la finalidad de organizar a los periféricos según su utilidad en el sistema de medición y la interconexión entre el microcontrolador y los periféricos. Cada bloque fue creado producto de la interacción que existe entre los periféricos, ya que algunos trabajan en conjunto para poder dar solución a etapas del proceso.

Para la implementación de este equipo de medición fue necesario el uso de los siguientes módulos y equipos: fuente de alimentación SUNPOWER DC 5V, módulo regulador de tensión AMS1117 3,3 V, microcontrolador ESP32, sensor ultrasónico HC-SR04, sensor digital de temperatura DS18B20, modulo para memoria micro SD, módulo RTC DS3231, convertidor de nivel de 8 canales, LCD 16x2 con interfaz I<sup>2</sup>C, LED's indicadores, pulsadores para acciones de control, y relés de estado sólido (SSR) SHARP S216S02, los cuales funcionan articuladamente para dar solución al problema planteado.

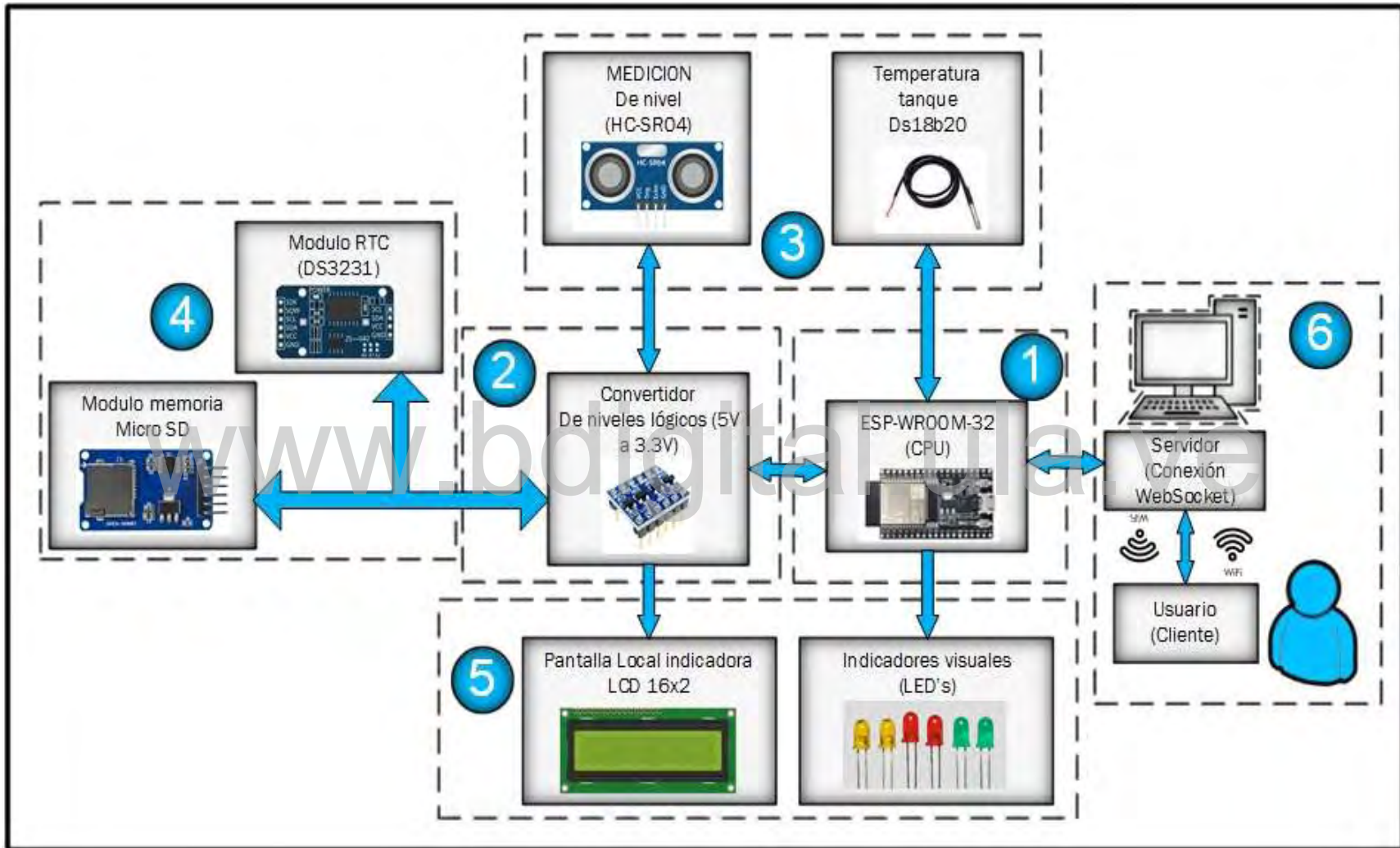


Figura 4.1: Diagrama de bloques del sistema de medición de tanque SINCONT ESP32

A continuación, se hará una descripción detallada de cada uno de los bloques y módulos que componen al esquema de la Figura 4.1, en cuanto a especificaciones técnicas, acondicionamiento, diagrama eléctrico, interconexión y funcionamiento,

#### 4.1.1 Bloque 1: unidad central de procesamiento (CPU)

Es la etapa más importante del sistema y está compuesta por el microcontrolador ESP-WROOM-32, este fue descrito en la sección 2.9.1 de este trabajo, como se mencionó anteriormente esta placa de desarrollo trabaja a 3,3 V aunque posee también una entrada de alimentación de 5V la cual se encuentra identificada como pin Vin (ver Figura 2.15), ya que internamente posee un regulador de tensión que adecua esa tensión de entrada a la requerida por el microcontrolador para trabajar apropiadamente. Este chip posee la capacidad de comunicarse mediante protocolo I<sup>2</sup>C a través de sus pines GPIO 21 (SDA) y GPIO 22(SCL). Mediante este bus controlara a periféricos como la pantalla LCD 16x2 la cual posee un módulo convertidor paralelo a serial y al reloj en tiempo real DS3231, aunque estos periféricos trabajan a 5V por lo que es necesario crear un acondicionamiento de las señales del bus I<sup>2</sup>C para que pueda haber compatibilidad entre los módulos y el ESP-32, este acondicionamiento será descrito más adelante.

Además, a través de los pines GPIO5 (CS), GPIO18 (SCK), GPIO19 (*MISO*) y GPIO23 (*MOSI*) el microcontrolador se conecta a la tarjeta SD para crear y leer registros albergados en la tarjeta SD. Para ingresar las señales de control del tanque para llenar y vaciar se incorporaron pulsadores mecánicos a los GPIO 26 y 27, se trabajó con las resistencias de *Pull-Down* internas, las cuales son programables por *software*. Cabe destacar que según el modulo para tarjeta micro SD que se use puede requerirse o no un condicionamiento de señal, a que hay módulos que trabajan a 5V y otros que trabajan a 3,3 V directamente. Para este caso en particular se empleó uno diseñado para 5V compatible con los Arduino por lo que en los esquemáticos se verá reflejada la conexión con la etapa de acondicionamiento.

Para el caso del sensor usado para hacer la medición de nivel, el HC-SR04 el cual se conectó a los GPIO16 y 17, en este caso se incluyó acondicionamiento de señal ya que este también funciona a 5 V. ahora bien para periféricos como el sensor de temperatura conectado al GPIO15, los relés de estado sólido conectados a los pines GPIO25 y GPIO33 para el manejo

de las cargas de potencia y las salidas usadas para la visualización de alarmas se realizó una conexión directa al microcontrolador, ya que funcionan perfectamente a 3,3 V. A continuación, se muestra el diagrama de conexión para el ESP-32 con todos los periféricos.

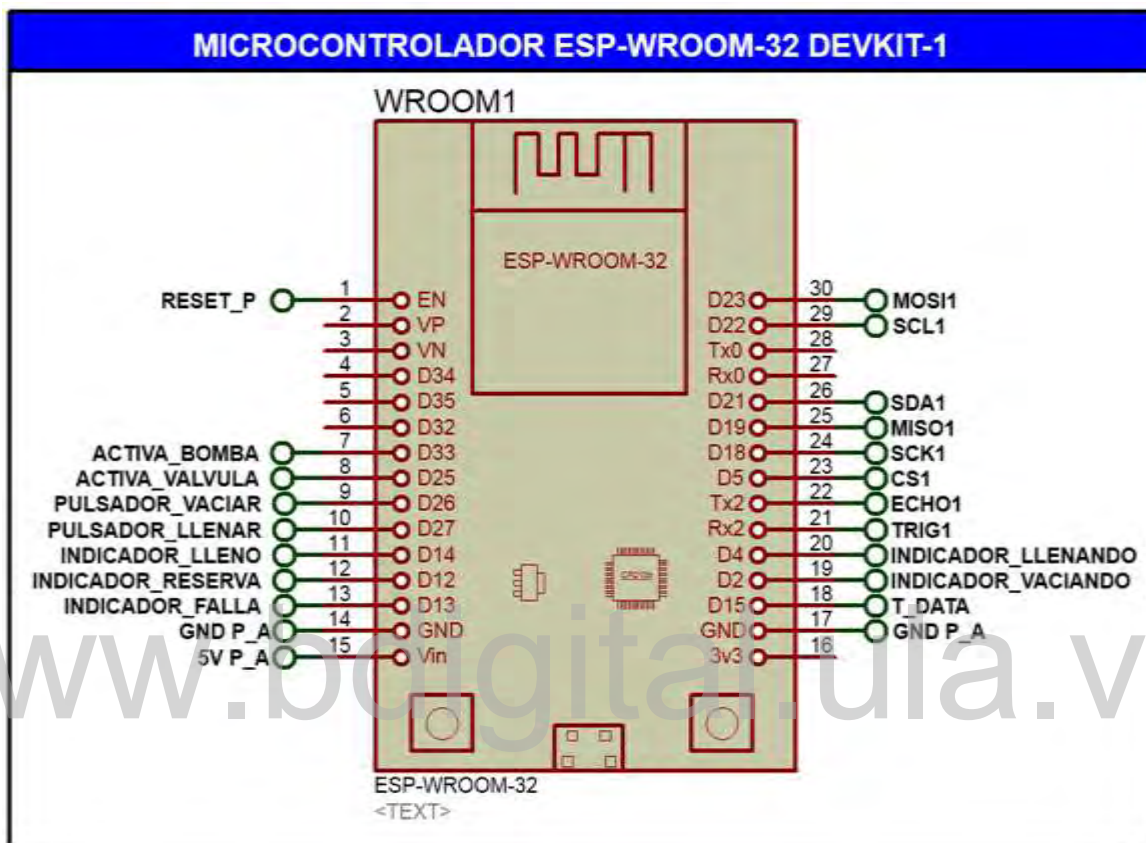


Figura 4.2: Conexiones de salida para el ESP-32

#### 4.1.2 Bloque 2: acondicionador de señales

Debido a que el microcontrolador trabaja a nivel de tensión CMOS (3.3 V) es necesario crear una interfaz que permita comunicar los periféricos que funcionan a 5 V con el microcontrolador, sin causar ningún daño. Por esta razón se requieren de un convertidor bidireccional de señales que cuente con al menos 8 canales para poder conectar el sensor ultrasónico HC-SR04, la pantalla LCD16x2, el RTC DS3231 y el módulo para micro SD (ver Figura 4.3).

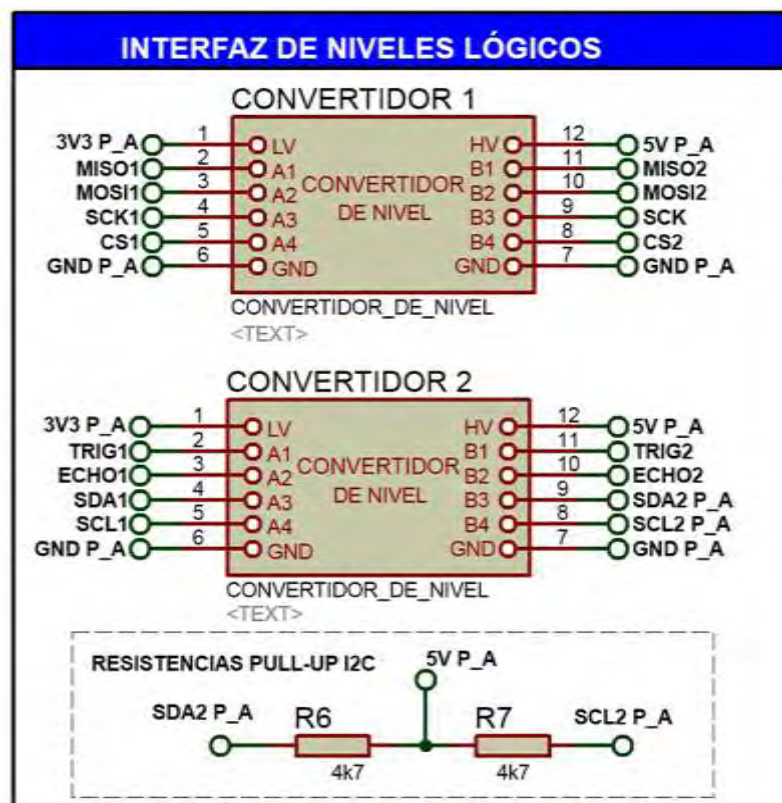


Figura 4.3: Interfaz de conexión entre los periféricos que usan 5 V y el ESP-32

En la figura anterior se puede apreciar que se usó 2 convertidores de nivel bidireccionales de 4 canales cada uno, en el convertidor 1 se conectó el módulo para tarjeta micro SD por el lado de tensión de 5 V (HV) y por el lado opuesto de 3,3 V (LV) las conexiones hacia el microcontrolador. Por otro lado, en el convertidor 2 tenemos al bus I2C con sus resistencias de *Pull-Up* y los pines *Trigger* y *Echo* del sensor ultrasónico por el lado HV y del lado LV al igual que en el caso anterior van a conexión con el esp-32.

### 4.1.3 Bloque 3: medición de nivel

La etapa de medición consta de un sensor ultrasónico HC-SR04 para medir el nivel en el tanque y un sensor de temperatura digital DS18B20 para la compensación por cambios de temperatura de la velocidad del sonido, ya que la velocidad de propagación del sonido cambia en función de la temperatura, lo cual genera que la precisión de la medición se vea afectada durante el día. Tal como muestras la Figura 4.4 se puede apreciar que el sensor de ultrasonido debe pasar por el convertidor de nivel lógico numero 2 por el lado HV primero para luego

seguir la conexión desde el lado LV hasta el microcontrolador ESP-32, esto no pasa con el sensor de temperatura que fácilmente puede trabajar a 3,3 V y puede establecer conexión directa con el esp32.

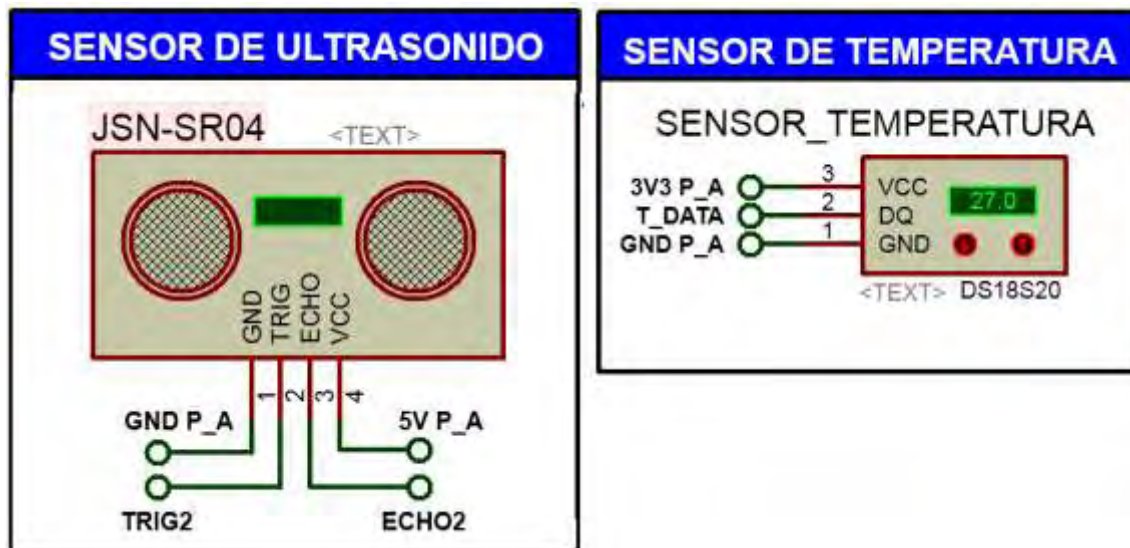


Figura 4.4: Conexión del sensor de ultrasonido y temperatura

#### 4.1.4 Bloque 4: registro de datos

Esta unidad es la encargada de construir en la memoria micro SD los archivos en formato .txt del registro diario de información correspondiente a las operaciones de entrada y salida de materia prima albergada en el tanque ver. Por su parte el RTC DS3231 es usado para llevar control de la fecha y hora del sistema, y con estos datos construir los nombres de los archivos a almacenar, de igual manera proporciona estos mismos datos de fecha y hora para el registro detallado de cada operación (Figura 4.5).

Día de la semana	Fecha	Hora de la operación	Cantidad de entrada	Cantidad de salida	Inventario actual
------------------	-------	----------------------	---------------------	--------------------	-------------------

Figura 4.5: Contenido y formato de los datos para la creación del reporte diario

A continuación, la Figura 4.6 muestra el diagrama de conexión eléctrica de este bloque al resto del sistema, como es de notar la tarjeta de memoria se encuentra conectada por el lado HV del convertidor de nivel lógico número 1 y el reloj está conectado al bus I<sup>2</sup>C por el lado

HV del convertidor 2, posteriormente del lado LV de ambos módulos convertidores de nivel salen las respectivas conexiones del reloj y la memoria micro SD hasta la placa de desarrollo ESP-WROOM-32.

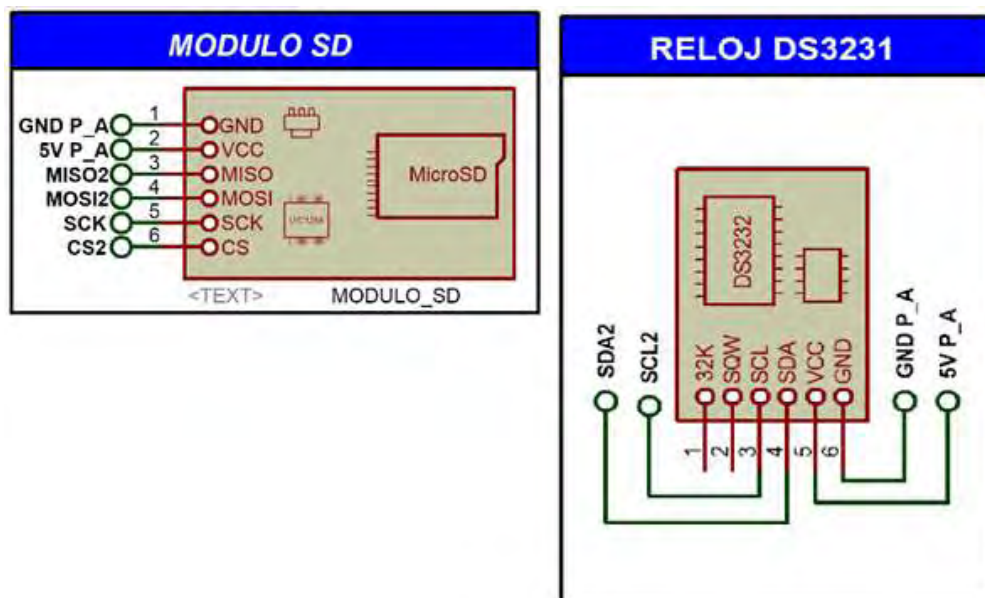


Figura 4.6: Conexión del módulo de memoria y reloj

#### 4.1.5 Bloque 5: visualización y control

El sistema requiere de indicadores visuales como la pantalla LCD 16x2 y los LED's conectados a los GPIO 2, 4, 12,13 y 14 que permitan aportar información de importancia a los operadores sobre el estado del equipo, con la finalidad de detectar averías lo más rápido posible y a su vez poder conocer directamente el nivel del tanque en su lugar de instalación. Por otro lado, en este bloque también se encuentran los pulsadores conectados a los GPIO 26 y 27 como ya se había mencionado, estos permitirán activar procesos de llenado y vaciado del tanque de manera local, además del pulsador para reinicio en caso de que el equipo lo requiera. En el diagrama circuital de la Figura 4.7 se puede observar la conexión de los elementos mencionados en este bloque, es de notar que para el caso de los pulsadores para activar los procesos de llenado o vaciado no requiere resistencias de *Pull-Down* ya que son asignadas internamente por *software* y en el caso del botón de reset ya la placa trae de fabrica la resistencia necesaria. Además, en la figura se puede observar la conexión de la pantalla con el bus I2C el cual se encuentra del lado HV del convertidor de nivel lógico numero 2

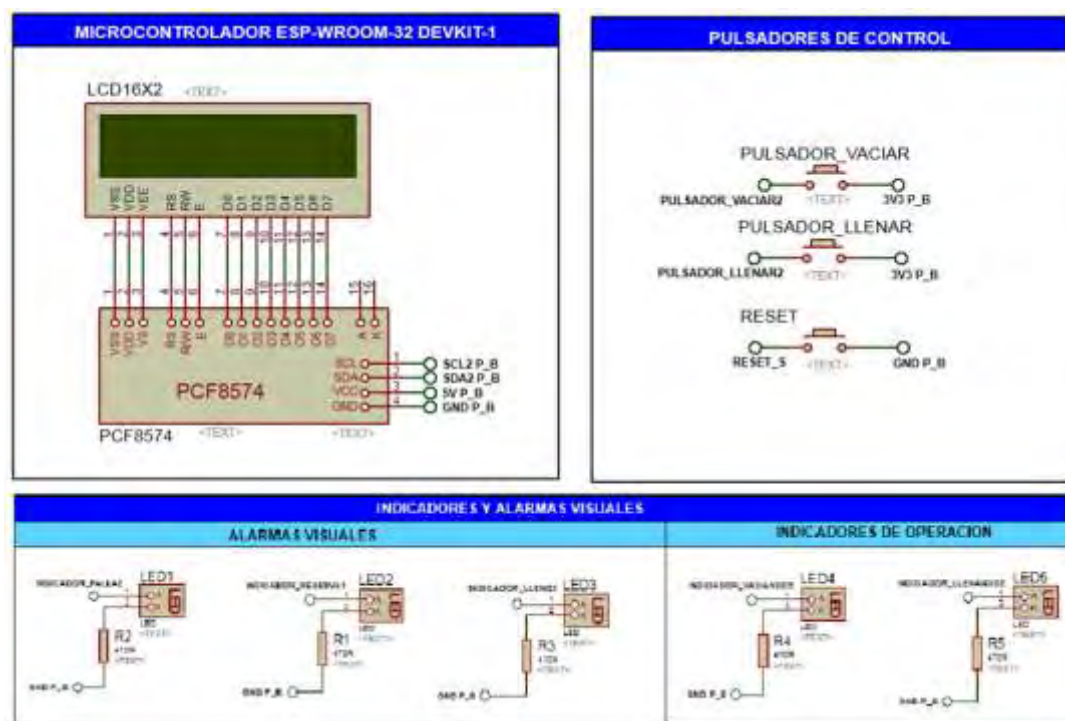


Figura 4.7: Diagramas de conexión de los indicadores visuales y la LCD

#### 4.1.6 Bloque 6: comunicación inalámbrica

El microcontrolador cuenta con la capacidad de establecer comunicaciones inalámbricas de 2 clases (*Bluetooth* y *Wi-Fi*), en este caso se usó la tecnología *Wi-Fi* mediante el protocolo *WebSocket* para establecer una comunicación entre el equipo de medición y el usuario (cliente), en este caso el microcontrolador hará el papel de servidor, ya que alojará la página web que se usará de interfaz de usuario y proveerá los datos de inventario que podrá solicitar los operadores del sistema. Por otro lado, el operador establecerá conexión a través de una computadora la cual debe tener conexión física o inalámbrica al enrutador con el cual se conecta el equipo de medición.

#### 4.1.7 Especificaciones técnicas del hardware

Tabla 4.1: Hojas de datos de dispositivos principales del sistema de medición

Hardware	Descripción
ESP-WROOM-32	Módulo ESP-32 [33]
HC-SR04	Sensor de ultrasonido [34]
DS18B20	Sensor de temperatura [35]

#### 4.1.8 Diagramas de conexión de las placas.

Debido a propósitos prácticos el equipo se diseñó en base a 2 placas, las cuales compartirán información entre sí, esto se hizo con la finalidad de pensar en un producto con una terminación más atractiva y comercial. En primer lugar hay que describir la conexión de la placa A con todos sus respectivos partes. Esta tarjeta contiene la gran mayoría de componentes a usar, está constituida por el ESP32, el módulo micro SD, el módulo DS3231, convertidores de nivel lógico, los relés de estado sólido (SSR) para el manejo de cargas de potencia, un regulador de 3,3 V, la conexión para el sensor de temperatura DS18B20 y por último cuenta con diferentes borneras que permiten tanto conectar la alimentación principal de toda la electrónica como también las borneras para conectar entre sí las dos placas.

Para efectos de una implementación real se contempla el uso de sensor JSN-SR04 el cual posee las mismas características eléctricas y modo de operación del sensor HC-SR04, con la diferencia de que este sensor se encuentra adaptado para trabajar en ambientes con mucha humedad, posee un único transmisor /receptor el cual viene separado de la placa para aumentar la seguridad del módulo (ver Figura 4.8: Módulo medidor por ultrasonido JSN-SR04.) ya que de esta manera en el módulo siempre permanece seco y libre de humedad. A continuación, se muestra el esquemático de la placa A, (Figura 4.9). Por otra parte, la placa “B” está destinada netamente a visualización por lo cual va a conectar la pantalla LCD16x2, los LED’s indicadores de alarma y de operaciones activas en el tanque, además, se incorpora un tercer pulsador para reiniciar el ESP32 cuando lo requiera el operador (Figura 4.10)



Figura 4.8: Módulo medidor por ultrasonido JSN-SR04. [36]

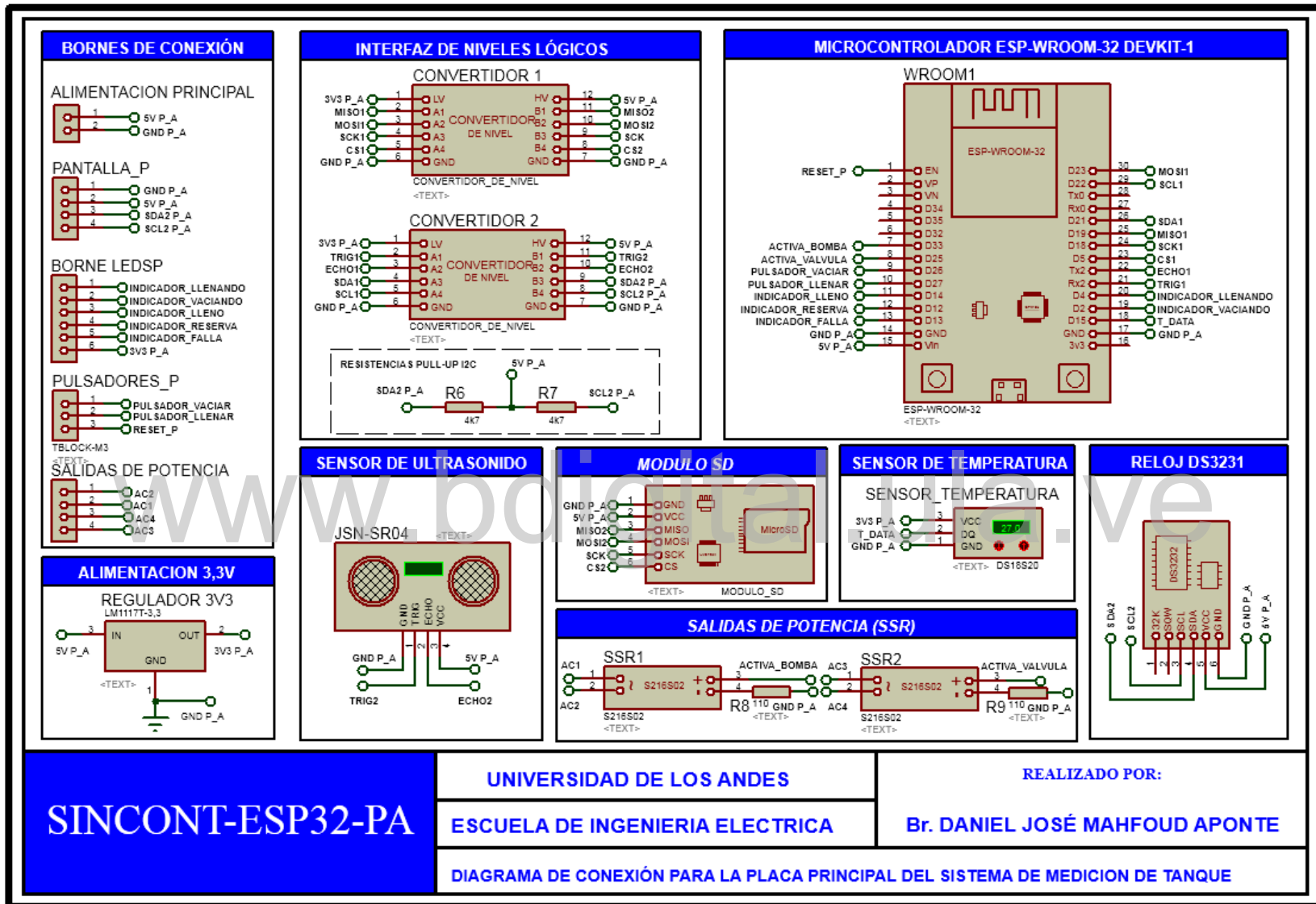


Figura 4.9: Diagramas de conexión de la placa A

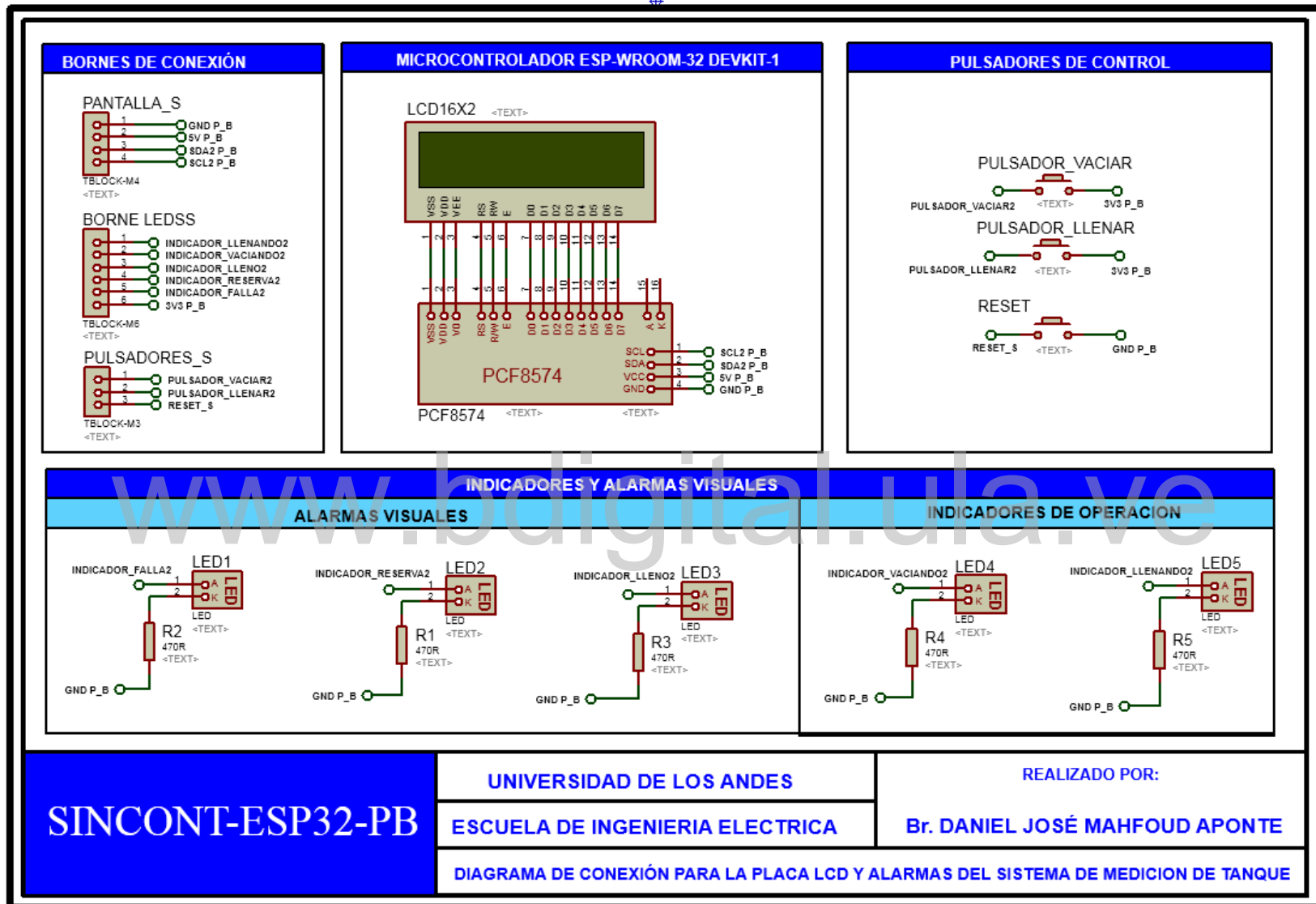


Figura 4.10: Diagrama de conexión de la placa B

SINCONT-ESP32-PB

UNIVERSIDAD DE LOS ANDES

ESCUELA DE INGENIERIA ELECTRICA

REALIZADO POR:

Br. DANIEL JOSÉ MAHFOUD APONTE

DIAGRAMA DE CONEXIÓN PARA LA PLACA LCD Y ALARMAS DEL SISTEMA DE MEDICION DE TANQUE

# CAPITULO 5

## DISEÑO DEL SOFTWARE PARA EL SISTEMA DE MEDICION DE NIVEL

### 5.1 ETAPA 3: DISEÑO DEL SOFTWARE

En el siguiente capítulo, se procederá a explicar todo lo relacionado a la codificación tanto del sistema de medición como de la página web, indicando la manera en la que se modelo el sistema y como se debe operar el equipo de medición.

### 5.2 Especificaciones de software

Para crear la lógica de operación del sistema de medición de nivel, el software debe dividirse en dos partes. La primera parte consta concretamente de la función de medición, en esta etapa, se utilizó el IDE de Arduino ya que gracias a su vasta comunidad de desarrolladores cuenta con soporte para programar de forma fácil y rápida los microcontroladores de la familia ESP-32.

Por otro lado, la segunda parte del software trata sobre el servidor web (*Back End*) y la funcionalidad de la página web o interfaz de usuario para el control del tanque y visualización de reportes (*Front End*), para codificar y dar funcionalidad al servidor del lado del microcontrolador se usó Arduino mediante el uso de las librerías para el manejo de servidor web y comunicación *WebSocket*, ahora bien, para la interfaz de usuario se usó Lenguaje de Marcado de Hipertexto (HTML) para dar estructura a la página web, agregando estilo usando Cascade Style Sheets (CSS) para diseñar la página descrita. Sin embargo, el uso de HTML y CSS solo está relacionado con la visualización en pantalla que tiene el cliente (*Front End*), pero no brinda ningún tipo de funcionalidad. Por ello, es necesario recurrir a un lenguaje de

programación como JavaScript, que permite asignar la lógica de funcionamiento de la página web y configurar el protocolo que se utilizará para enviar información del cliente al servidor y viceversa.

### 5.2.1 Diseño del sistema de medición

El sistema de medición se modeló como una máquina de estados finitos, ya que son intuitivas y fáciles de entender, este tipo de diagramas puede modelar en forma gráfica sistemas complejos mediante la definición estados o etapas por los cuales ira pasando el instrumento de medición según las señales de entrada que vaya recibiendo. Por otro lado, este método por ser grafico facilita la codificación del proceso, pudiendo identificar en forma clara las estructuras de decisión a requerir para dar sentido a las transiciones necesarias para cambiar de un estado a otro. Uno de los objetivos en el modelo de funcionamiento de la máquina de estados, es conocer en todo momento que operación se encuentra ejecutando, debido a posibles fallos de energía, la maquina debe continuar la operación que estaba realizando al momento de fallar el servicio eléctrico, de esta forma se garantiza la confiabilidad del sistema, sin la intervención de un operador, esto presenta gran utilidad, ya que así se puede certificar que la maquina no perderá la información concerniente a inventario entrada/salida de un proceso y al mismo tiempo presentar funcionabilidad de autodiagnóstico si algún de los módulos llega a fallar, de esta forma el operador puede rápidamente restituir su funcionamiento.

Cabe destacar que, para sintetizar un sistema de esta complejidad y utilidades a través de otras herramientas de programación, ameritaría un esfuerzo un poco mayor a nivel de organizar y estructurar el funcionamiento, para posteriormente construir el código fuente que permita aportar la funcionalidad requerida. Pudiendo generarse casos como el de no haber tomado en cuenta alguno de los escenarios posibles para un buen funcionamiento. Tomando en cuenta estas características, se establecieron 6 estados los cuales conforman al diagrama de la máquina, estos son: CONFIGURACIÓN, MIDIENDO, LLENANDO, VACIANDO, ALMACENAR y por ultimo FALLA. Estos interactúan entre sí a través de 22 transiciones las cuales serán los indicadores de cambio entre estados. A continuación, en la Figura 5.1 se mostrará el diagrama de estados finitos implementado en el sistema de medición.



### 5.2.1.1 Descripción de los estados del diagrama

**CONFIGURACIÓN:** es el estado con el cual arranca la maquina por defecto, y el encargado de inicializar todo el sistema. Como primer acto da inicio a la comunicación con la memoria *EEPROM* del módulo ESP-32, esto con el objetivo de evaluar cuál era el último estado en el que se encontraba la maquina al momento de ser apagada, con la finalidad de dar continuidad a la operación de almacenaje de la información en caso de pérdida de energía a mitad de una operación de carga o descarga de inventario. Posteriormente establece todas las comunicaciones, entre ellas está la inicialización de la LCD 16x2 y RTC ubicados en el bus I<sup>2</sup>C, establece comunicación con la memoria micro SD para comprobar la existencia del archivo de reporte correspondiente al día actual y para consultar el ultimo valor de inventario almacenado, evalúa la operatividad de los sensores de ultrasonido y temperatura y en función de todas estas operaciones se desprenden 5 posibles transiciones de salida a los diferentes estados del diagrama , estas transiciones serán explicadas en forma breve en la Tabla 5.6.

**MIDIENDO:** este estado está constituido con el fin de censar el nivel del tanque periódicamente cada 60 segundos, aunque puede ser variable y ajustado mediante código, dependiendo de las características que amerite el proceso en particular, esta medición constante se hace con el objetivo de descartar fugas en el contenedor y a su vez poder estar atento a las señales de control para las operaciones de llenado y vaciado, proporcionadas por los pulsadores manuales o los digitales los cuales se encuentran en la interfaz web. Ya que al momento de realizar una medición el microcontrolador entra en un bucle con duración de 1 segundo durante el cual no puede recibir señales de control. Además, estando en este estado el microcontrolador monitorea en forma constante la existencia del archivo actual en el cual se debe realizar el guardado del reporte de operación, ya que al cambiar de día es importante que se cree el nuevo archivo correspondiente a la fecha para poder mantener organizado los reportes de inventario diario. Además, imprime en la pantalla LCD los datos de fecha, hora, identificación del tanque y por último el nivel en tiempo real de dicho contenedor (Ver Figura 5.3)

En base al diagrama de estado mostrado en la Figura 5.1, existen 4 transiciones de salida para la etapa de medición. Cabe destacar que deben considerarse factores de seguridad para este

estado ya que el equipo debe responder con alarmas visuales en caso de que ocurra una fuga, o en casos donde no resulte conveniente realizar la operación que indique el usuario, como por ejemplo mandar a llenar el tanque cuando ya se encuentra lleno o a vaciar cuando ya se encuentra en un nivel de reserva. En este punto es importante explicar el algoritmo usado para realizar una medición que determine el nivel del tanque. En la Figura 5.2 se muestra el algoritmo requerido para realizar una medición de nivel.

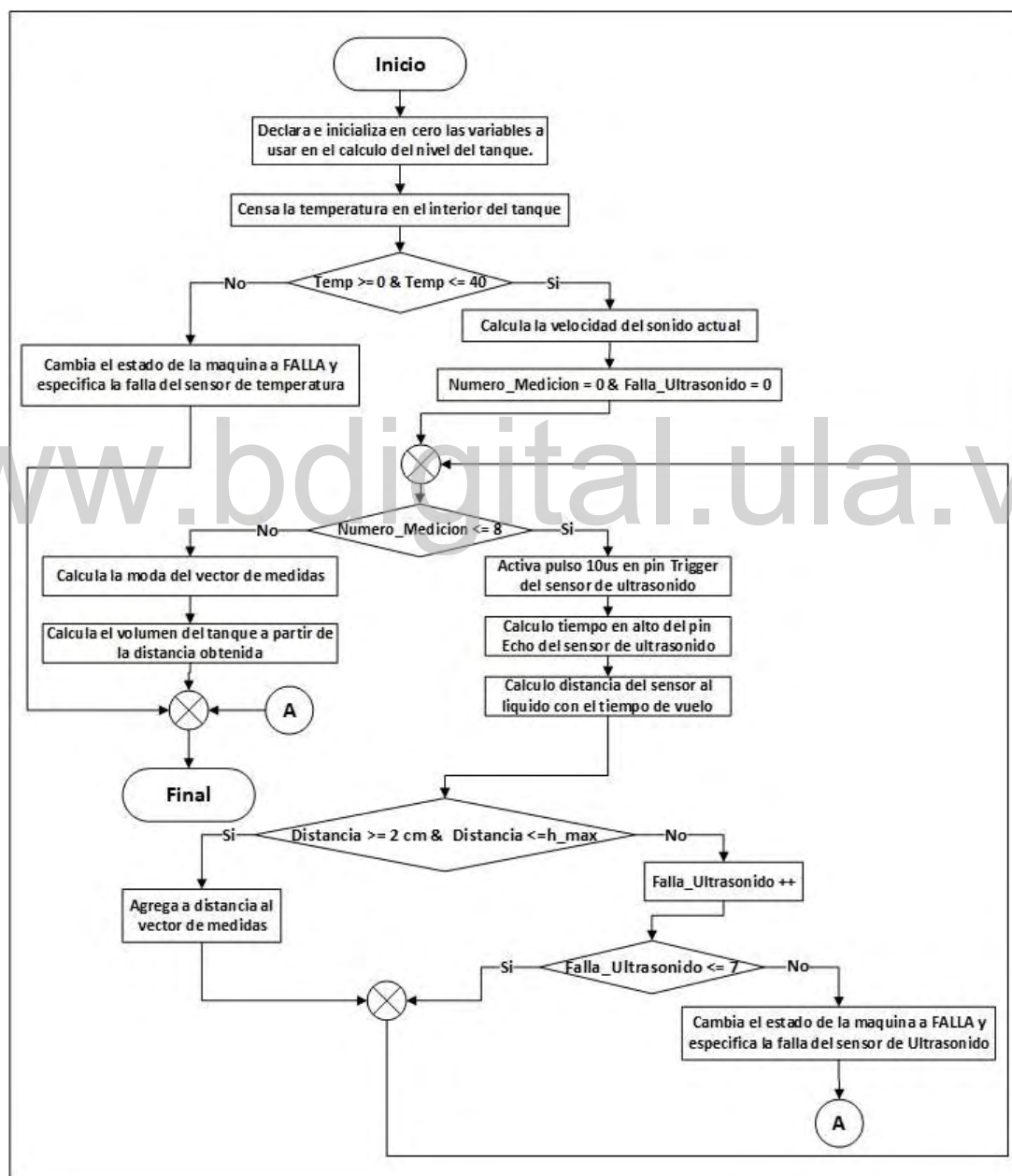


Figura 5.2: Algoritmo de la función Leer\_Sensor\_Ultrasonido(), para determinar el nivel del tanque

Siguiendo al diagrama de flujo mostrado, a continuación, se hará una breve descripción del proceso. Para realizar una medición el software llama a la función `Leer_Sensor_Ultrasonido()`, esta inicia declarando las variables necesarias para el cálculo de nivel como temperatura del tanque, distancia del sensor al líquido y la altura del líquido; posteriormente hace una lectura de temperatura en el interior del tanque a través del sensor DS18B20; en este punto se hace la primera comprobación de seguridad ya que la temperatura debe estar en un rango dentro de (0 a 40) °C, considerando que el tanque debe estar a temperatura ambiente. Si la temperatura se encuentra por fuera de este rango entonces la maquina pasara al estado de falla y se especificara en el tipo de falla, cual es el modulo que se encuentra averiado.

Ahora bien, si la temperatura se encuentra dentro del rango antes mencionado el sistema entrará en un bucle, el cual se repetirá 8 veces; este bucle consiste en poner en nivel lógico alto el pin *Trigger* por un periodo de 10 $\mu$ s y posteriormente a valor lógico bajo; en este instante el módulo HC-SR04 manda a través del aire una serie de ondas ultrasónicas y el pin *Echo* pasa a nivel alto el tiempo en que tardan las ondas ultrasónicas en ir hasta el líquido y volver. Sabiendo esto, el sistema obtiene el tiempo en que el pin *Echo* permanece en alto y con este valor calcula la distancia del sensor al líquido, a través de la ecuación (2.1)

En este punto se encuentra la siguiente valoración de seguridad ya que la distancia obtenida por el sensor debe estar en el rango entre 20 mm y la altura máxima del tanque a usar como contenedor, ya que el sensor posee una zona muerta de medición de (0 a 20) mm. En este sentido si la medida se encuentra por fuera del rango (20 a 320) mm la medida se descarta y se aumenta un contador de medida fuera de rango el cual puede aumentar hasta un valor de 7 fallos en la medición antes de pasar la maquina al estado falla por sensor de ultrasonido, con esto se garantiza que al menos 1 medida del bucle sea válida. Ahora bien, si la medida entra entre el rango correcto de medición esta se añade a un vector que servirá de contenedor momentáneo; después de cumplir el ciclo de medición las 8 veces el microcontrolador cuenta con un vector de medidas validas que reflejan la distancia del sensor al líquido, pero estas medidas pueden no ser la misma debido a ondas en el tanque, espuma generada por el líquido o a las propias características del sensor por lo cual es necesario calcular un parámetro estadístico a los valores almacenados en el vector mediciones, en tal sentido al aplicarle la

moda a dicho conjunto de datos se garantiza que se obtiene el valor de medición más exacto. A partir de este valor se obtiene la altura del líquido calculando la diferencia entre la altura del tanque y la distancia del líquido al sensor de ultrasonido el cual se encuentra en la parte superior del contenedor. Conocido el nivel, es cuestión de calcular el volumen de líquido a partir de la forma que contenga el tanque bien sea por una ecuación si el contenedor posee una forma que lo permita o en caso de tanques irregulares a través de su tabla de calibración.

**LLENANDO:** a este estado se puede acceder a través de estado de medición, pulsado el botón de llenar siempre y cuando el tanque este en la capacidad de recibir más volumen de líquido o si al momento de reiniciar el esp-32 detecta que se apagó antes de finalizar un proceso de llenado anterior de forma completa. Si se intenta entrar a este estado y el tanque se encuentra al 100% de su capacidad entonces se retornará en forma automática al estado de medición. Pero, en el caso de que el tanque tenga espacio de almacenamiento disponible efectivamente pasara a LLENANDO; en este momento el tiempo entre mediciones se ve reducido a 2,5 segundos esto con el fin de poder detectar las variaciones en el nivel del tanque en forma más rápida y a su vez monitorear que el tanque no sobrepase su nivel máximo de almacenamiento. Durante este estado, la pantalla LCD cambiara a otra distinta a la del estado MIDIENDO en la que se refleja el proceso de llenado y el nivel en el que se encuentra el tanque, Figura 5.3: Pantallas para los estados MIDIENDO y LLENANDO.



*Figura 5.3: Pantallas para los estados MIDIENDO y LLENANDO*

Ahora bien, para concluir con el proceso de llenado el operador del sistema debe pulsar el botón físico llenar, ubicado en el panel de control del equipo o pulsar el botón digital llenar ubicado en la interfaz web. Al detectar que el usuario dio la orden de terminar el llenado, el equipo espera a que la medición se estabilice, estableciendo un rango de tolerancia en la coincidencia de medición de  $\pm 3$  mm del nivel, esto debido a la característica de precisión del sensor de ultrasonido usado.

Una vez que la medición se encuentra estabilizada el equipo compara el nivel actual del tanque con el último valor de inventario que se almacena en la memoria micro SD y en función de esto calcula la diferencia, si esta diferencia de nivel es significativa, es decir, está por arriba del error de medición del sensor entonces el equipo pasará a ALMACENAR. En el caso de que la diferencia en el nivel que se calculó sea menor al error de medición del sensor, el sistema no debe crear un registro, ya que este caso indicaría que se activó el proceso de llenado, pero no ingreso líquido al contenedor o que la cantidad de material que ingreso no es significativa como para ser detectada por el sensor, en este caso la máquina de estados debe regresar al estado de MEDICION sin pasar por el almacenaje de la información.

**VACIANDO:** similar a LLENANDO, este estado a nivel de lógica funciona de la misma forma, haciendo mediciones en el tanque de manera periódica, con la diferencia de que en vez calcular el volumen de entrada al tanque, se busca obtener el volumen que abandono dicho contenedor, por su parte también este estado monitorea que el nivel no baje más allá del 5% del volumen total del sistema, ya que en una aplicación real se trabaja con bombas sumergibles para extraer producto del contenedor y estos tipos de bombas no pueden quedar completamente secas, ya que esto degrada su vida útil o incluso pueden dañarse por completo.

**ALMACENAR:** es el estado más sencillo del sistema a nivel de operaciones realizadas; en esta etapa del proceso el microcontrolador llama a las funciones que solicitan el formato de hora y fecha, con el fin de consultar si existe el archivo correspondiente para ese instante de tiempo y en caso de que no existan esos documentos, los crea, organizando los reportes diarios en carpetas mensuales. Después de realizar la comprobación del archivo el microcontrolador llama a la función encargada de hacer el guardado del reporte y es en este

momento cuando el sistema crea una cadena de texto en la que asigna el formato con el cual debe ser guardado el reporte; posteriormente esa cadena de caracteres se llena a partir de los datos que indica la Figura 5.4: Formato de almacenaje para el reporte de 1 operación de vaciado, la cual muestra el formato con el cual se crea el registro de una operación de llenado o vaciado.

Día de la semana	fecha	Hora de la operación	Entró	Salió	Inventario
Dom	15-12-1994	12:25:21 p.m.	0.00	6.65	1.23

*Figura 5.4: Formato de almacenaje para el reporte de 1 operación de vaciado*

Después de concluido el proceso de almacenamiento, el ESP-32 actualiza en la memoria EEPROM en una celda específica para esta variable que se completó de forma satisfactoria el guardado, en caso contrario, si no se logra hacer el guardado por fallas con la memoria, el estado cambiara a falla y especificara el tipo de falla en una variable, o en caso de no completarse el proceso por pérdida de energía al alimentar de nuevo la placa de desarrollo, esta podrá detectar mediante la consulta de esta celda de la EEPROM el estado en el cual se encontraba el proceso y lo concluirá. Posteriormente la máquina de estados volverá a su estado de reposo MIDIENDO.

**FALLA:** es el encargado de hacer la función de autodiagnóstico ya que utiliza una salida digital del ESP-32 para dar una secuencia de encendido a un LED para indicar el fallo de un módulo en específico, esto permite al operador detectar mediante esta señal luminosa que sector de la placa debe revisar para encontrar el desperfecto. A continuación, se muestra en una tabla la secuencia de parpadeos del LED para cada tipo de falla.

*Tabla 5.1: Secuencia de pulsos intermitentes para el LED de fallas según tipo de avería*

Tipo de falla	Secuencia de encendido
Falla micro SD	2 pulsos intermitentes
Falla RTC DS3231	3 pulsos intermitentes
Fuga del tanque	4 pulsos intermitentes
Falla DS18B20	5 pulsos intermitentes
Falla HC-SR04	6 pulsos intermitentes

### 5.2.1.2 Descripción de las transiciones del diagrama

Al haber definido los estados del sistema y las funciones que cumple cada uno, es tiempo de describir las transiciones asociadas al cambio entre dichos estados. Debido a la complejidad del diagrama de estados, las transiciones serán explicadas a través de las tablas que se presenta a continuación:

*Tabla 5.2: Descripción de las entradas a la máquina de estados*

Símbolo	Descripción
P_1	Llenar el tanque
P_2	Vaciar el tanque

*Tabla 5.3: Descripción de las salidas de la máquina de estados*

Símbolo	Descripción
RP1	Relé de potencia válvula
RP2	Relé de potencia bomba
FP	Alarma de falla

*Tabla 5.4: Variables del sistema*

Símbolo	Descripción
G_C	Guardado correcto del reporte
E_Act	Estado actual de la maquina
E_Ant	Estado anterior de la maquina
T_Falla	Tipo de falla del sistema
Entró	Cantidad de litros entrando al tanque
Salió	Cantidad de litros saliendo del tanque
E_M	Resolución de medida del sensor

*Tabla 5.5: Estados y fallas presentes en la máquina de estados*

Estado	Nº	Falla	Nº
CONFIGURACIÓN	0	Falla_SD	0
MIDIENDO	1	Falla_RTC	1
LLENANDO	2	Fuga_tanque	2
VACIANDO	3	Falla_DS18B20	3
ALMACENAR	4	Falla_HC_SR04	4
FALLA	5	No_Hay_Falla	5

Tabla 5.6: Tabla resumen de las transiciones que componen al diagrama de estados

Transición N°	Condición lógica	Modifica a la variable
1	$E\_Act = 1 \ \& \ E\_Ant = 4 \ \& \ T\_Falla = 5$	RP1=0, RP2=0, FP=0
2	$T\_Falla \neq 5$	RP1=0, RP2=0, FP=1
3	$(E\_Act = 3 \ \& \ E\_Ant = 1 \ \& \ T\_Falla = 5) \    \ (E\_Act = 4 \ \& \ E\_Ant = 3 \ \& \ G\_C = 0)$	RP1=0, RP2=0
4	$(E\_Act = 2 \ \& \ E\_Ant = 1 \ \& \ T\_Falla = 5) \    \ (E\_Act = 4 \ \& \ E\_Ant = 2 \ \& \ G\_C = 0)$	RP1=0, RP2=0
5	$(E\_Act = 4 \ \& \ G\_C = 1 \ \& \ T\_Falla = 5)$	RP1=0, RP2=0
6	$E\_Act = 1 \ \& \ P\_1 = 0 \ \& \ P\_2 = 0$	RP1=0, RP2=0
7	$T\_Falla \neq 5$	RP1=0, RP2=0, FP=1
8	$P\_1 = 1 \ \& \ P\_2 = 0$	RP1=1, RP2=0
9	$P\_1 = 0 \ \& \ P\_2 = 1$	RP1=0, RP2=1
10	$P\_1 = 1 \ \& \ P\_2 = 0$	RP1=1, RP2=0
11	$P\_1 = 0 \ \& \ P\_2 = 0 \ \& \ Entró > E\_M$	RP1=0, RP2=0
12	$P\_1 = 0 \ \& \ P\_2 = 0 \ \& \ Entró < E\_M$	RP1=0, RP2=0
13	$T\_Falla \neq 5$	RP1=0, RP2=0, FP=1
14	$P\_1 = 0 \ \& \ P\_2 = 1$	RP1=0, RP2=1
15	$P\_1 = 0 \ \& \ P\_2 = 0 \ \& \ Salió < E\_M$	RP1=0, RP2=0
16	$P\_1 = 0 \ \& \ P\_2 = 0 \ \& \ Salió > E\_M$	RP1=0, RP2=0
17	$T\_Falla \neq 5$	RP1=0, RP2=0, FP=1
18	$G\_C = 0$	RP1=0, RP2=0
19	$G\_C = 1$	RP1=0, RP2=0, G_C = 0
20	$T\_Falla \neq 5$	RP1=0, RP2=0, FP=1
21	$T\_Falla \neq 5$	RP1=0, RP2=0, FP=1
22	$T\_Falla = 5$	RP1=0, RP2=0, FP=0

## 5.2.2 Diseño del servidor e interfaz web para el sistema de medición

El sistema de control y medición de tanques requiere de una interfaz inalámbrica a través de *Wi-Fi*, que permita tener acceso a los procesos de control, además de brindar acceso a los registros de operaciones a través de gráficos de barras. Por lo cual, es necesario construir un servidor web en el microcontrolador ESP-32 que pueda aportar estas funcionalidades al operador del sistema (cliente).

Este servidor está compuesto por 2 partes, la primera está asociada a la lógica de funcionamiento del lado del sistema de medición (*Back End*), y la otra parte está vinculada al funcionamiento, estructura y estética de la página web (*Front End*) en la cual funcionara la interfaz de usuario para el sistema de medición. Ambas partes tienen tareas diferentes, pero intercambian entre sí información acerca del estado del sistema y las acciones de control a través de distintas peticiones o solicitudes por parte del cliente.

### 5.2.2.1 Funcionamiento del servidor (*Back End*).

La placa de desarrollo ESP-WROOM-32 es la encargada de manejar la funcionalidad del servidor, en conjunto con la funcionalidad del diagrama de estados antes descrito, ambos códigos coexisten en la memoria de programa del microcontrolador y trabajan en forma simultánea, ya que este chip cuenta con la capacidad de ser multitareas. En este apartado será descrito el funcionamiento del servidor, visto desde el microcontrolador, centrando la explicación en cómo se establece la conexión entre el microcontrolador y el cliente, como es la comunicación entre ambos y cuál es el contenido de dicha comunicación

#### **Inicio de la conexión inalámbrica:**

Siguiendo el orden de ideas, se puede decir que para iniciar el servidor web lo primero que debe hacer el microcontrolador es establecer un enlace con el enrutador que maneja la conexión de red local mediante la función `WiFi.begin(ssid, password)`, a través de su modo *Station*, en el cual se le pasa como parámetros el nombre de la red con la cual hará conexión (SSID) y la contraseña de dicha red inalámbrica (PASSWORD) ver Figura 5.5. Posteriormente el sistema informa por la pantalla LCD 16x2 que se está haciendo conexión con la red antes mencionada y entra en un bucle que se repetirá hasta que se establezca la

conexión entre el ESP-32 y el enrutador. Por último, después de establecida la comunicación con la red local el sistema imprime en la LCD la dirección IP que fue asignada por el enrutador al sistema de medición.

```

1 void ConnectWiFi_STA() {
2 //void ConnectWiFi_STA(bool useStaticIP = false) {
3   Serial.println("");
4   WiFi.mode(WIFI_STA);
5   WiFi.begin(ssid, password);

```

Figura 5.5: Uso de la función `ConnectWiFi_STA()` para realizar la conexión del ESP-32 con el enrutador

Es importante mencionar que el microcontrolador también posee la capacidad de crear su propia red local (modo *Access Point*), en casos donde no se cuente con un enrutador, en este caso se le puede asignar nombre y contraseña a esta red, pero esta red posee un alcance más bajo que el de la red creada por un enrutador debido a la potencia de la antena del microcontrolador. Por lo cual en espacios donde ya se cuente con una red inalámbrica local es mejor usar el modo de conexión *Station* para garantizar una mejor calidad de conexión.

### Creación del servidor web:

Una vez conectado el ESP-32 al enrutador es momento de iniciar el servidor web asíncrono, esta etapa se encarga de llamar los archivos desde el *Serial Peripheral Interface File System* (SPIFFS), este es un sistema de ficheros que forma parte de la memoria flash externa perteneciente al módulo ESP-WROOM-32; en este caso se usa para almacenar y gestionar el archivo HTML que contiene la estructura de la página web, además del archivo de estilo CSS para dar la estética al contenido de la página y el archivo de JavaScript con las instrucciones que permiten dar el funcionamiento a la página.

Ya que el servidor es asíncrono, permite implementar páginas web dinámicas, en las cuales se puede servir contenido en forma espontánea, sin la necesidad de que el cliente interactúe con la página o refresque el navegador. Ahora bien, para llamar al archivo que contiene la página web, el sistema aprovecha a la instancia *AsyncWebServer* usando el puerto 80 ya que es el estandarizado para navegación web HTTP, posteriormente llama a la función

InitServer(), esta función se encarga de buscar en el sistema de ficheros del microcontrolador el archivo con nombre index.html y asigna la página web con ese nombre a la URL raíz del servidor (ver Figura 5.6). Además el servidor describe una respuesta para el caso en que el cliente solicite una URL que no esté definida y por ultimo inicializa el servidor con la función server.begin().

```

1 AsyncWebServer server(80);
2 void InitServer(){
3     server.serveStatic("/", SPIFFS, "/").setDefaultFile("index.html");
4     server.onNotFound([](AsyncWebServerRequest *request) {
5         request->send(400, "text/plain", "Not found");
6     });
7     server.begin();
8     Serial.println("HTTP server started");
9 }

```

Figura 5.6: Codificación del servidor web asíncrono

#### **Configuración del protocolo de comunicación AsyncWebSocket:**

El protocolo *websocket* asíncrono es un protocolo *full-duplex* de comunicación el cual permite conectar a diferentes usuarios a un servidor a través de un mismo canal de comunicación, el cual permanece abierto una vez que se inicie la conexión, esto resulta muy eficiente, ya que por dicho canal ambas partes pueden intercambiar información entre sí, pudiendo mantener a todos los clientes actualizados sin la necesidad de la interacción con la interfaz, esto resulta ideal para la implementación del sistema, ya que con esta capacidad se puede mantener actualizado los diferentes clientes en forma instantánea y tener reportes en tiempo real, esto garantiza la detección de averías de manera rápida y eficiente.

Ya que la comunicación *websocket* es bidireccional es necesario definirla en ambos extremos del canal de comunicación, por el momento se indicará como se realizó configuración del protocolo del lado del sistema de medición y posteriormente se hará la explicación del lado de la interfaz web. Para crear una conexión web es necesario crear una instancia a través del comando AsyncWebSocket ws("/ws") con esta instancia se hará referencia a esta conexión y se podrá acceder a las funcionalidades de este protocolo

A continuación, se muestra un fragmento del código asociado a la inicialización del *websocket*.

```

64 void InitWebSockets() {
65     ws.onEvent(onWsEvent);
66     server.addHandler(&ws);
67     Serial.println("WebSocket server started");
68 }

```

Figura 5.7: Función para la inicialización de la conexión WebSocket, *InitWebSocket()*

La función mostrada en la figura anterior tiene como finalidad indicarle al servidor que hacer al recibir un evento, indicándole a través del parámetro *onWsEvent* que función debe ejecutar. A continuación, se procede a mostrar dicha función de eventos.

```

void onWsEvent(AsyncWebSocket * server, AsyncWebSocketClient * client,
               AwsEventType type, void * arg, uint8_t *data, size_t len){

    if(type == WS_EVT_CONNECT){
        Actualizar_Pulsadores();
        Actualizar_Estado();
        Actualizar_Nivel();

        client->ping();
    } else if(type == WS_EVT_DISCONNECT){
    } else if(type == WS_EVT_ERROR){

    } else if(type == WS_EVT_PONG){

    } else if(type == WS_EVT_DATA){
        AwsFrameInfo * info = (AwsFrameInfo*)arg;
        String msg = "";
        if(info->final && info->index == 0 && info->len == len){
        }
    }
}

```

Figura 5.8: Función de eventos WebSocket

La Figura 5.8 muestra las estructuras de decisión que canalizan el flujo del programa dependiendo de los distintos eventos que reciba la conexión *WebSocket*, de estos eventos vale la pena destacar 2 de ellos, el primero sería *WS\_EVT\_CONNECT*, este evento se recibe cuando un nuevo cliente establece conexión con el servidor ingresando la dirección IP asignada por el enrutador al ESP-32, según describe la figura cada vez que un nuevo cliente establece conexión, el servidor llama a 3 funciones importantes para la interfaz web, las cuales son *Actualizar\_Pulsadores()*, *Actualizar\_Estado()*, *Actualizar\_Nivel()*.

Estas funciones permiten que al cargar la página web a través de la petición HTTP://dirección\_IP/ esta se actualice con los valores en tiempo real para el estado de los pulsadores, el estado en el cual se encuentra el sistema y el nivel del tanque. Ahora bien, el evento WS\_EVT\_DATA es el más importante de todos, ya que este es el encargado de recibir y procesar los datos cuando un cliente hace una petición, en tal sentido obtiene el mensaje en notación de objeto de JavaScript comúnmente conocido por su acrónimo en inglés (JSON) el cual es la manera estandarizada de compartir información en páginas web. Al final de este evento se encuentra una función llamada ProcessRequest(client, msg), a esta última función se le pasan 2 parámetros, el primero sería el cliente el cual hizo la solicitud y el segundo sería el mensaje en formato JSON enviado por el cliente al servidor, con el contenido se la solicitud.

```

118 void ProcessRequest (AsyncWebSocketClient * client, String request) {
119     Serial.println(request);
120     StaticJsonDocument<200> doc;
121     DeserializationError error = deserializeJson(doc, request);
122     if (error) { return; }
123     String command = doc["command"];
124     if (command == "Set") {
125         setData(client, request);
126     }
127     if (command == "Get") {
128         getData(client, request);
129     }
130 }

```

*Figura 5.9: Función para el procesamiento de las solicitudes del cliente*

Tal como muestra la Figura 5.9 la función ProcessRequest recibe un mensaje en formato JSON el cual es deserializado para obtener la primera clave “command” con el fin de saber si el cliente quiere hacer un cambio en las entradas de control para llenar o vaciar el tanque a través de la función setData o si quiere realizar una solicitud de reporte diario por medio de la función getData.

Al entrar a la función setData el servidor obtiene mediante deserialización cuál de los 2 botones pulso el cliente en la interfaz web y en función de eso alterna el valor lógico de la variable correspondiente al estado del pulsador por consiguiente activa de forma remota la operación de llenado o vaciado el botón que allá pulsado el cliente.

Ahora bien, la función `getData` es una de las funciones con gran relevancia dentro del código ya que esta permite empezar con la solicitud de un reporte de operación diario, a continuación, se presenta el código que compone dicha función:

```

79 void getData(AsyncWebSocketClient * client, String request){
80     StaticJsonDocument<300> doc;
81     DeserializationError error = deserializeJson(doc, request);
82     if (error) {
83         return;
84     }
85     String Tipo_Reporte = doc["id"];
86
87     if(Tipo_Reporte == "Reporte_Mensual"){
88         String Mes_A_Reportar = doc["Mes"];
89         Serial.print("se solicito un reporte mensual: ");
90         Serial.print(Mes_A_Reportar);
91     }
92     else if(Tipo_Reporte == "Reporte_Diario"){
93         String Mes_A_Reportar = doc["Mes"];
94         String Dia_A_Reportar = doc["Dia"];
95         Serial.print("se solicito un reporte diario: ");
96         Serial.println(Mes_A_Reportar);
97         Serial.println(Dia_A_Reportar);
98         Elaborar_Reporte_Diario(Mes_A_Reportar, Dia_A_Reportar);
99     }
100 }

```

*Figura 5.10: Función para iniciar solicitud de reporte diario*

Al entrar a esta función el ESP-32 deserializa la petición para canalizar que tipo de reporte desea el cliente ya que inicialmente se dimensionó para trabajar con reportes diarios y, aunque posteriormente se le podrá añadir la funcionalidad de reportes mensuales, por lo que el camino a seguir es el de guardar en variables que día y de qué mes desea el reporte y pasar esas variables por referencia a la función `Elaborar_Reporte_Diario`. Esta función se encarga de construir la ruta del archivo a consultar

```

96 void Elaborar_Reporte_Diario(String *Mes_A_Reportar, String *Dia_A_Reportar ){
97     String Ruta_De_Carpeta = "/" + *Mes_A_Reportar;
98     String Ruta_De_Reporte_Diario = Ruta_De_Carpeta + "/" + *Dia_A_Reportar;
99     readfileDailyReport(SD, Ruta_De_Reporte_Diario.c_str());
100     Serial.println(" ");
101 }

```

*Figura 5.11: Construcción de la ruta para acceder al archivo a consultar*

A continuación, se muestra el diagrama de flujo de la función `readFileDailyReport ()` encargada de dar lectura a los archivos de reporte solicitados por el cliente.

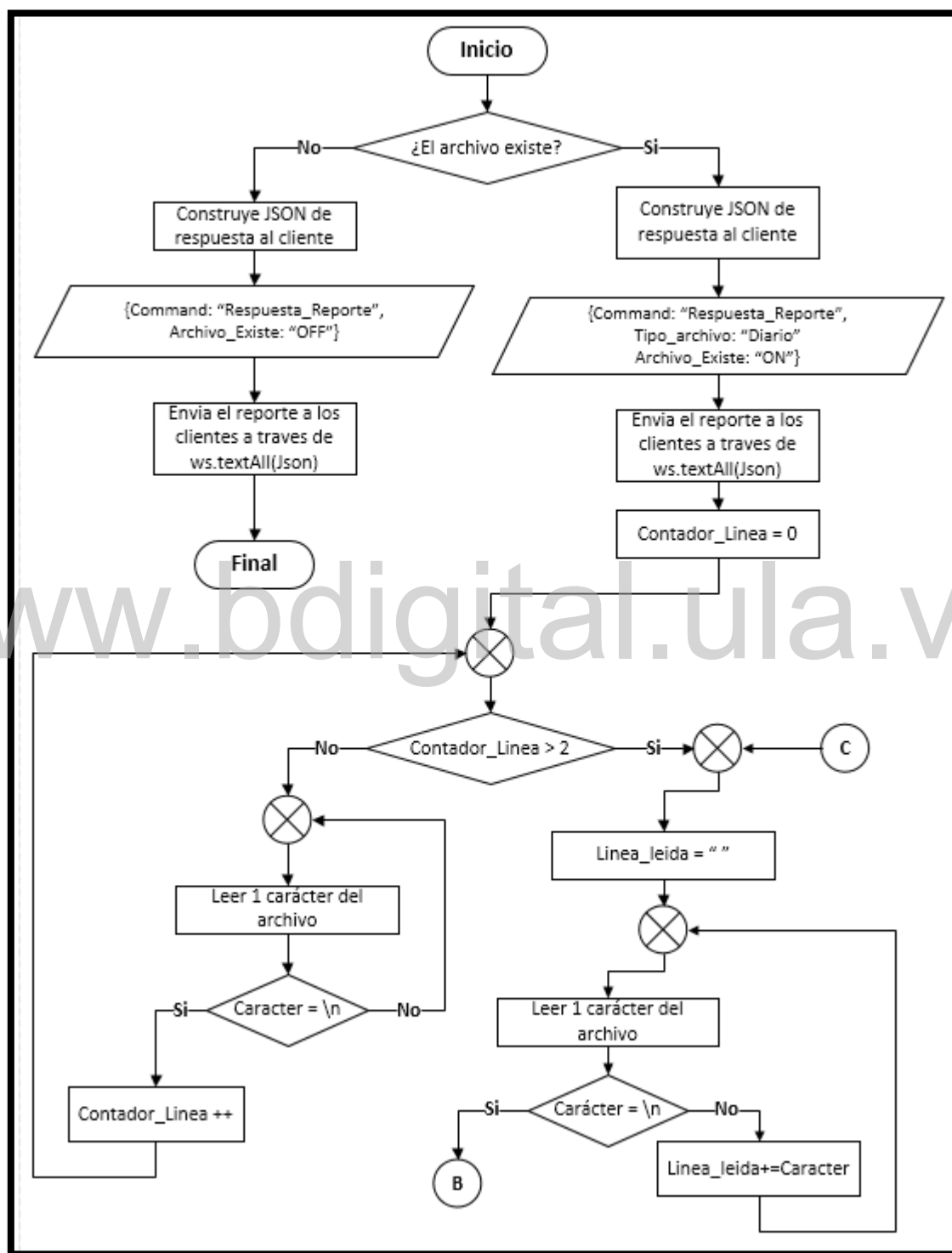


Figura 5.12: Diagrama de flujo para la función de lectura de reporte diario

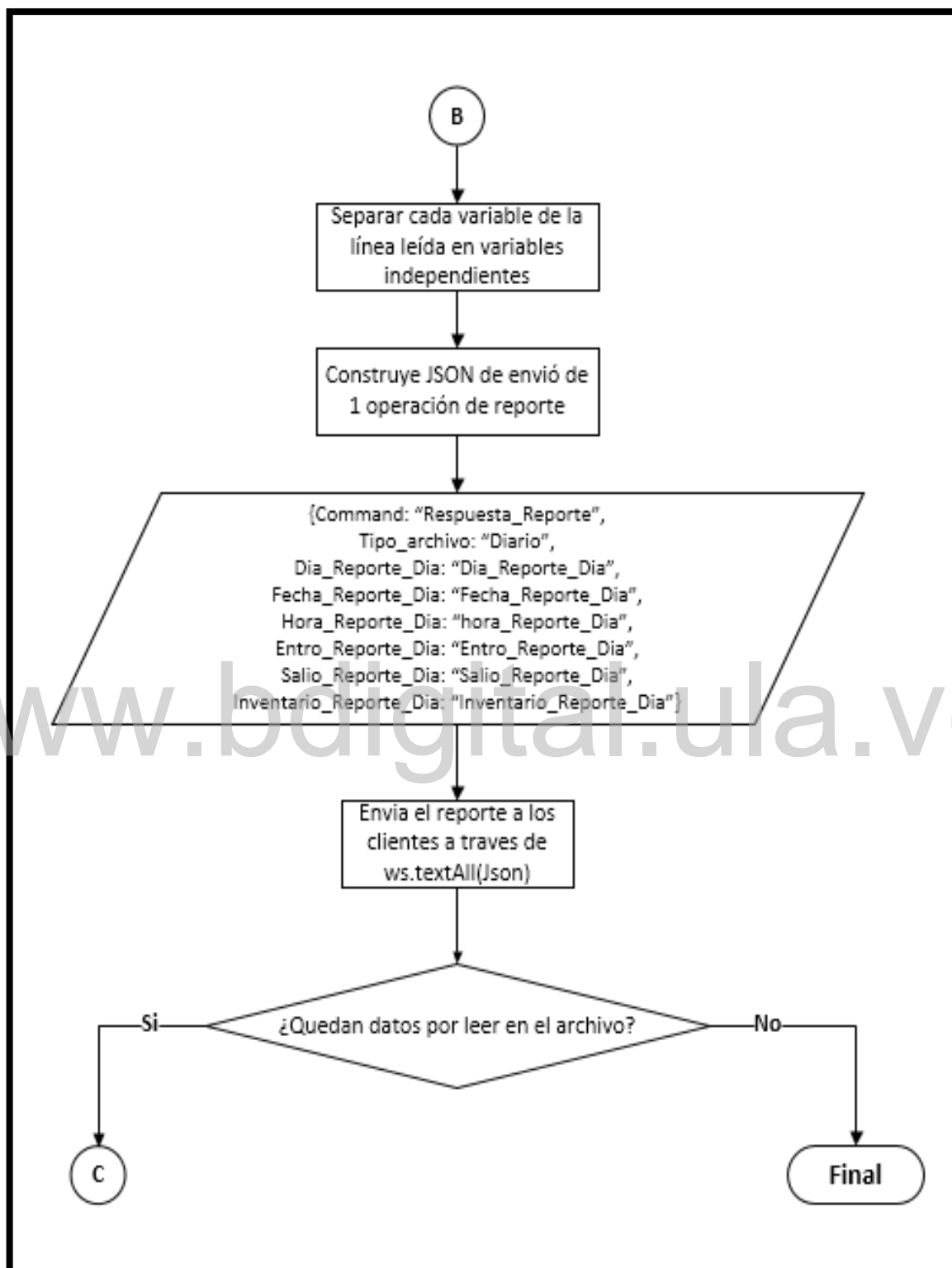


Figura 5.13: Continuación del diagrama de flujo de la función de lectura de reporte

La función `readFileDailyReport ()` descrita en el diagrama de flujo de las últimas dos figuras explica el proceso mediante el cual se lee un archivo de reporte diario y se envía estos datos a través del canal *WebSocket* asíncrono, como aspecto importante de este diagrama se puede apreciar cómo se estructura los mensajes a través del formato JSON y posteriormente como son enviados al cliente, línea por línea hasta concluir con todo el archivo. Con esto se concluye con la explicación de funcionamiento del servidor desde el punto de vista del ESP-32

### 5.2.2.2 Funcionamiento de la interfaz de usuario (*Front End*)

La página web a implementar como interfaz de control debe cumplir con ciertas características, una de estas es poder visualizar contenido de forma dinámica y automática, sin la necesidad de que el operador este constantemente interactuando con la interfaz, ya que esta página es un sistema que sirve para monitorear alarmas y estados del proceso, en base a esta premisa esta interfaz debe poseer la capacidad de visualizar el estado de las variables del proceso en tiempo real, además de poder enviar parámetros de control del proceso; en este caso particular se hace mención de los botones con los cuales se inicia el llenado o vaciado del tanque. Así mismo, se debe incluir un tercer botón para solicitar reportes de operaciones. Por ultimo debe contar con una sección en donde se puedan visualizar los gráficos de las operaciones diarias

En base a estas necesidades se estructuro una página web mediante el uso de HTML y CSS, esta página se encuentra dividida en tres secciones, la primera de estas es el *Header*, es una etiqueta bastante sencilla y se usó para contener el título de la página además del logo de la compañía *Espressif Systems* y se asignó color y otros estilos CSS para hacerla estéticamente más atractiva. Posteriormente, en el siguiente bloque de código llamado *Section* se encuentra lo más importante de la página, ya que alberga una sección que contiene los 2 botones para control del tanque y el botón para solicitud de reporte. Después de esta sección de botones en página se encuentra una división hecha por 2 contenedores, el primero a su vez contiene una sección destinada a para monitoreo en tiempo real de las variables y el estado del proceso. Además, este contenedor alberga un espacio destinado a la sección de solicitud de reporte, pero esta unidad es dinámica y solo aparece cuando se pulsa el botón de solicitud de reporte.

El segundo contenedor está destinado a cargar el espacio para realizar los gráficos de reporte diario de entrada y salida de materia prima, esta grafica posee un eje x en el cual se ubica la hora en la cual se realizó la operación de guardado y un eje y que indica el volumen en litros que entro o salió del tanque, siendo más específico, en el diagrama de barras se representa en color azul las operaciones de llenado y en color rojo las operaciones de vaciado del tanque.

En el último bloque de la página *Footer* se añade 2 enlaces, el primero para regresar al inicio de la página y el segundo para contactar al desarrollador web, por último, se añade el año de creación y el nombre del desarrollador de la página. Ahora bien, para que toda esta estructura tenga una funcionalidad es necesario añadir lógica a través del uso de JavaScript, llamando desde el código HTML un archivo de esta clase el cual se encarga de inicializar la comunicación WebSocket y al igual que cuando se explicó la conexión *WebSocket* del lado del servidor se debe definir un manejador de eventos que permita canalizar las acciones a tomar en base a las peticiones que hace el servidor. A continuación, se muestra el fragmento de código de este archivo encargado de inicializar la comunicación.

```

1 //-----conexion websocket-----//
2 var connection = new WebSocket('ws://' + location.hostname + '/ws', ['arduino']);
3 function connect() {
4     console.log('entro a connect()')
5     connection = new WebSocket('ws://' + location.hostname + '/ws', ['arduino']);
6     connection.onopen = function () {
7         connection.send('Received from Client');
8         console.log('Connected');
9     };
10    connection.onerror = function (error) {
11        console.log('WebSocket Error ', error);
12        connection.close();
13    };
14    connection.onmessage = function (e) {
15        console.log('Server: ', e.data);
16        processData(e.data);
17    };
18    connection.onclose = function (e) {
19        console.log('El Websocket está cerrado. Se intentará volver a conectar en 10 segundo.', e.reason);
20        setTimeout(function () {
21            connect();
22        }, 10000);
23        /*location.reload();*/
24    };
25 }
26 connect();
27 //-----//

```

Figura 5.14: Inicialización de la comunicación WebSocket y creación de manejador de archivos desde el cliente

De la figura anterior es importante destacar el evento `connection.onmessage` el cual ocurre cuando el servidor envía un mensaje al cliente, este evento llama a la función `processData(e.data)`; y le pasa como parámetro el mensaje recibido para que posteriormente dicha función se encargue de deserializar el mensaje y evaluar que acción desea hacer el servidor en la página web. Otro evento importante a considerar es `connection.onclose` el cual ocurre cuando por alguna razón se pierde o se cierra el canal de comunicación WebSocket, este evento activa la función `setTimeout` la cual espera un tiempo de 10 segundos y posteriormente llama a la función `connect()` para intentar recuperar la conexión. A continuación, se muestra la función `processData(data)`.

```
378 function processData(data) {
379     let json = JSON.parse(data);
380     console.log(json);
381
382     if (json.command == 'Actualizar_Pulsadores') {
383         ActualizarPulsadores(json.Boton_Llenar, json.Boton_Vaciar);
384     }
385     else if (json.command == 'Actualizar_Estado') {
386         ActualizarEstado(json.Estado_De_Maquina, json.Tipo_De_Falla);
387     }
388     else if (json.command == 'Actualizar_Medicion') {
389         ActualizarMedicion(json.Nivel_Tanque)
390     }
391     else if (json.command == 'Respuesta_Reporte') {
392         if (json.Archivo_Existente == 'OFF') {
393             window.alert("¡El archivo solicitado NO existe!");
394         }
395         else {
396             if (json.Tipo_archivo == 'Diario') {
397                 Grafica_Diaria(json);
398             }
399             else if (json.Tipo_archivo == 'Mensual') {
400                 Grafica_Mensual(json);
401             }
402         }
403     }
404 }
```

Figura 5.15: Codificación de la función `processData()`.

Esta función direcciona las acciones de la página web, según la información que reciba desde el servidor. Es usada al momento de recibir una actualización de los pulsadores, el nivel del tanque y el estado en el cual se encuentra el sistema de medición, también cuando se va a graficar un reporte diario. Para solicitar un reporte diario la página web cuenta con una etiqueta `<Input type: "date">` para ingresar la fecha a solicitar, al seleccionar una fecha en esta entrada, el archivo JavaScript extrae los datos y los envía al servidor para que valide si existe el documento y posteriormente recibe los datos del reporte enviado por el servidor, en caso de no existir el documento la página web despliega una ventana emergente que indica la ausencia del archivo en los registros del ESP-32.

Para realizar los gráficos se recurrió a la librería de *JavaScript Highcharts*, esta cuenta con la funcionalidad necesaria para crear diferentes tipos de gráficas. Como un punto a favor de esta librería es que resulta bastante intuitiva y entrega gráficos muy llamativos e interactivos. Además, puede representar tanto gráficos estáticos como también en tiempo real. Para hacer uso de esta librería es necesario crear una instancia `var chartT = new Highcharts.Chart` y esta requiere de ciertos parámetros, los cuales son ingresados en formato JSON, algunos de estos parámetros son `renderTo`, el cual es usado para indicar a través de un Id en que parte del HTML se desea elaborar el gráfico. En las series se le indica todo lo necesario para cada gráfico que se quiera dibujar, en tal sentido se asocia un nombre a cada gráfico, un tipo de gráfica, pudiendo trabajar con líneas, columnas e incluso hasta con diagramas circulares. Posteriormente se establece el formato y tipo de datos que será ingresado tanto en el eje X como en el Y. con estos datos ingresados ya se puede realizar un gráfico por medio de la función de Highcharts `chartT.series[i].addPoint([x, y], true, true, true)`, la cual añade los puntos (X,Y) conforme los va recibiendo la página web.

### 5.2.3 Comunicación entre *Front End* y *Back End*

Habiendo ya descrito ambas partes del servidor web es tiempo de dar un ejemplo gráfico de cómo se crea el canal de conexión WebSocket y posteriormente como intercambian entre si la información el servidor y los diferentes clientes. A continuación, la figura

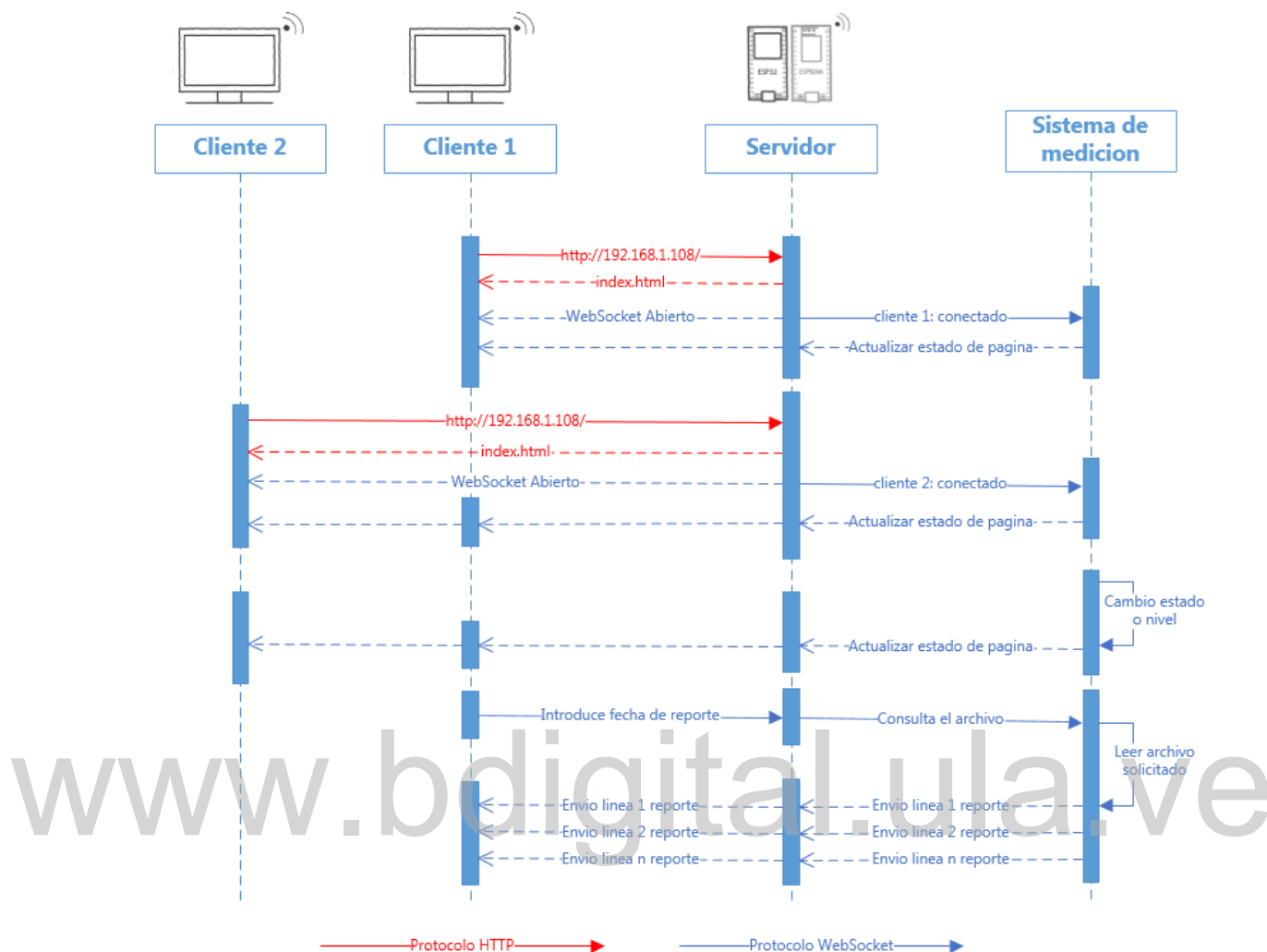


Figura 5.16: Diagrama de secuencia para la comunicación entre cliente y servidor del sistema de medición e inventario de tanque SINCONT ESP-32.

Tal como muestra la figura anterior, para iniciar la comunicación un cliente debe realizar una solicitud `HTTP://Direccion_IP/` al servidor y este responde de vuelta con la página web asignada por defecto, en este momento el servidor también envía un mensaje al sistema de medición para informar que un nuevo cliente se conectó y el sistema responde a todos los clientes con la actualización del estado actual del sistema. Además, en caso de que una variable del sistema de medición cambie de manera local, también envía una actualización de estado a todos los clientes. Por último, se puede apreciar el intercambio de datos al momento en que un cliente solicite un registro de reporte de operaciones diario

# **CAPITULO 6**

## **PRUEBA DEL SISTEMA DE MEDICION**

En los capítulos anteriores, se explicó todo el proceso necesario para el diseño del sistema de medición e inventario de tanques. Ahora bien, en este capítulo se expondrá las pruebas realizadas con el sistema de medición, la página web y se propondrá un diseño de placas para una posible implementación del sistema de medición.

### **6.1 ETAPA 4: PRUEBAS DEL SISTEMA**

En el diseño de todo equipo de medición, es necesario realizar pruebas de funcionamiento con la finalidad de garantizar la calidad de las mediciones hechas con dicho equipo. Sin embargo, para validar estas pruebas es necesario usar otro equipo de medición, que sirva como patrón, este elemento debe poseer una mejor resolución y precisión de medida que el instrumento a calibrar. En tal sentido para dar validación a las pruebas realizadas se implementó como instrumento patrón la regla graduada, esta herramienta de medición resulta ideal para contrastar el nivel real del líquido en el tanque con el indicado por el sistema de medición y registro. La Figura 6.1 muestra el registro fotográfico del tanque usado en el prototipo para sistema de medición, este tanque fue elaborado en vidrio, cuenta con una base en forma cuadrada con un área interior de  $(19.6 \text{ cm})^2$  y posee en su exterior una regla graduada, que permite comprobar las mediciones hechas por el sensor de ultrasonido.

#### **6.1.1 Obtención de la curva de calibración de forma experimental**

Como una primera prueba de medición se realizó una curva de calibración del instrumento mediante Método de calibración líquida (API STD 2555) descrito en la sección 2.5.2 utilizando el Procedimiento con tanque volumétrico, estas pruebas fueron realizadas a una temperatura de

28 °C y considerando la totalidad del rango de medida para el tanque de la Figura 6.1, dicho rango de nivel va de (0 a 260) mm lo que se traduce en una capacidad total de 9.98 litros



*Figura 6.1: Tanque usado en el prototipo de sistema de medición*

Debido que el procedimiento para la calibración mediante tanque volumétrico exige el uso de un tanque considerablemente menor se tomó entonces un recipiente previamente medido, cuyo volumen es de 0,5 litros; representando un valor bastante cercano al 5 % del volumen nominal del tanque a aforar. En base a esta premisa surgen las 20 medidas tomadas para la elaboración de la tabla de calibración del tanque, estas medidas fueron registradas en la Tabla 6.1. Posteriormente, mediante los datos tabulados se pudo construir la curva de calibración, esta permitió obtener la máxima desviación de la curva característica del sistema con respecto a una línea de referencia. A continuación, se muestra tanto la tabla de registro como la curva de calibración obtenida:

Tabla 6.1: recolección de datos experimentales para la curva de calibración del sistema de medición.

Ascendente					Descendente				
Nivel Real (cm)	Nivel Software (cm)	Volumen (%)	Error Absoluto (cm)	Error Relativo (%)	Nivel Real (cm)	Nivel Software (cm)	Volumen (%)	Error Absoluto (cm)	Error Relativo (%)
0,0	0,0	0,00%	0,0	0,00%	0,0	0,0	0,00%	0,0	0,00%
1,4	1,3	4,99%	0,1	7,14%	1,1	1,1	4,23%	0,0	0,00%
2,8	2,6	9,99%	0,2	7,14%	2,4	2,3	8,84%	0,1	4,17%
4,0	3,9	14,98%	0,1	2,50%	3,6	3,6	13,83%	0,0	0,00%
5,3	5,2	19,98%	0,1	1,89%	4,9	4,9	18,82%	0,0	0,00%
6,6	6,5	24,97%	0,1	1,52%	6,3	6,3	24,20%	0,0	0,00%
7,9	7,8	29,96%	0,1	1,27%	7,6	7,6	29,20%	0,0	0,00%
9,2	9,1	34,96%	0,1	1,09%	8,9	8,9	34,19%	0,0	0,00%
10,5	10,4	39,95%	0,1	0,95%	10,4	10,3	39,57%	0,1	0,96%
11,8	11,7	44,95%	0,1	0,85%	11,8	11,6	44,56%	0,2	1,69%
13,2	13,1	50,32%	0,1	0,76%	13,0	12,9	49,56%	0,1	0,77%
14,5	14,3	54,93%	0,2	1,38%	14,4	14,3	54,93%	0,1	0,69%
15,8	15,7	60,31%	0,1	0,63%	15,8	15,5	59,54%	0,3	1,90%
17,2	17,0	65,31%	0,2	1,16%	17,1	16,9	64,92%	0,2	1,17%
18,5	18,3	70,30%	0,2	1,08%	18,4	18,5	71,07%	0,1	0,54%
19,7	19,5	74,91%	0,2	1,02%	19,7	19,5	74,91%	0,2	1,02%
21,0	20,7	79,52%	0,3	1,43%	21,0	20,8	79,91%	0,2	0,95%
22,3	21,9	84,13%	0,4	1,79%	22,3	22,1	84,90%	0,2	0,90%
23,7	23,3	89,51%	0,4	1,69%	23,6	23,2	89,13%	0,4	1,69%
25,0	24,7	94,89%	0,3	1,20%	25,0	24,7	94,89%	0,3	1,20%
26,3	26,1	100,27%	0,2	0,76%	26,3	26,1	100,27%	0,2	0,76%
Error Max			0,4		Error Max			0,4	
Error Promedio			0,2		Error promedio			0,1	



Figura 6.2: curva de calibración para el sistema de medición, obtenida a través de proceso experimental

La Figura 6.2: curva de calibración para el sistema de medición, obtenida a través de proceso experimental describe la curva característica del sistema de medición, según la Tabla 6.1 la mayor desviación del valor real ocurre al 85 % del volumen total del tanque, dicha desviación fue de 4 mm, representando un error relativo de 1.79 %. Sin embargo, en la mayoría de las mediciones el error absoluto es mucho más bajo, teniendo un promedio de  $\pm 2$  mm, lo que resulta en un error aceptable dentro de los parámetros.

### 6.1.2 Validación de los datos obtenidos la prueba de calibración

Para dar validación a los datos de la prueba de calibración se recurrió al Método de “*strapping*” (API MTSM) o medición manual de anillos descrito en la sección 2.5.1 al aplicar este método se calculó el volumen del tanque en función de sus medidas, ya que la forma del contenedor permite obtener de forma sencilla una ecuación para representar su volumen en función de la altura.

$$V = \frac{b^2 * h}{1000} \quad (6.1)$$

Donde:

V = volumen del tanque[L].

b = lado de cuadrado de la base [cm].

h = nivel del líquido [cm].

En función de la ecuación (6.1) se elaboró la Tabla 6.2, la cual permitió comprobar los datos obtenidos en la prueba de calibración de tanque. A continuación de muestra una comparación entre los datos teóricos de la calibración por *strapping* contra los datos proporcionados por la prueba de calibración.

Tabla 6.2: Comparación de los datos obtenidos por el método calibración por strapping y el método de calibración por volumen

PRUEBA ASCENDENTE DE LA CURVA DE CALIBRACION					
Nivel real [cm]	Nivel <i>strapping</i> [cm]	Volumen <i>strapping</i> [%]	Nivel sistema medición [cm]	Volumen sistema medición [%]	Error absoluto [cm]
0,0	0,0	0,00%	0,0	0,00%	0,0
1,4	1,4	5,38%	1,3	4,99%	0,1
2,8	2,8	10,76%	2,6	9,99%	0,2
4,0	4,0	15,37%	3,9	14,98%	0,1
5,3	5,3	20,36%	5,2	19,98%	0,1
6,6	6,6	25,36%	6,5	24,97%	0,1
7,9	7,9	30,35%	7,8	29,96%	0,1
9,2	9,2	35,34%	9,1	34,96%	0,1
10,5	10,5	40,34%	10,4	39,95%	0,1
11,8	11,8	45,33%	11,7	44,95%	0,1
13,2	13,2	50,71%	13,1	50,32%	0,1
14,5	14,5	55,70%	14,3	54,93%	0,2
15,8	15,8	60,70%	15,7	60,31%	0,1
17,2	17,2	66,08%	17,0	65,31%	0,2
18,5	18,5	71,07%	18,3	70,30%	0,2
19,7	19,7	75,68%	19,5	73,76%	0,2
21,0	21,0	77,20%	20,7	78,37%	0,3
22,3	22,3	85,67%	21,9	83,36%	0,4
23,7	23,7	91,05%	23,3	88,36%	0,4
25,0	25,0	96,04%	24,7	94,89%	0,3
				Error <sub>promedio</sub>	0,17

### 6.1.3 Funcionamiento del sistema de medición

#### 6.1.3.1 El prototipo para sistema de medición de tanques

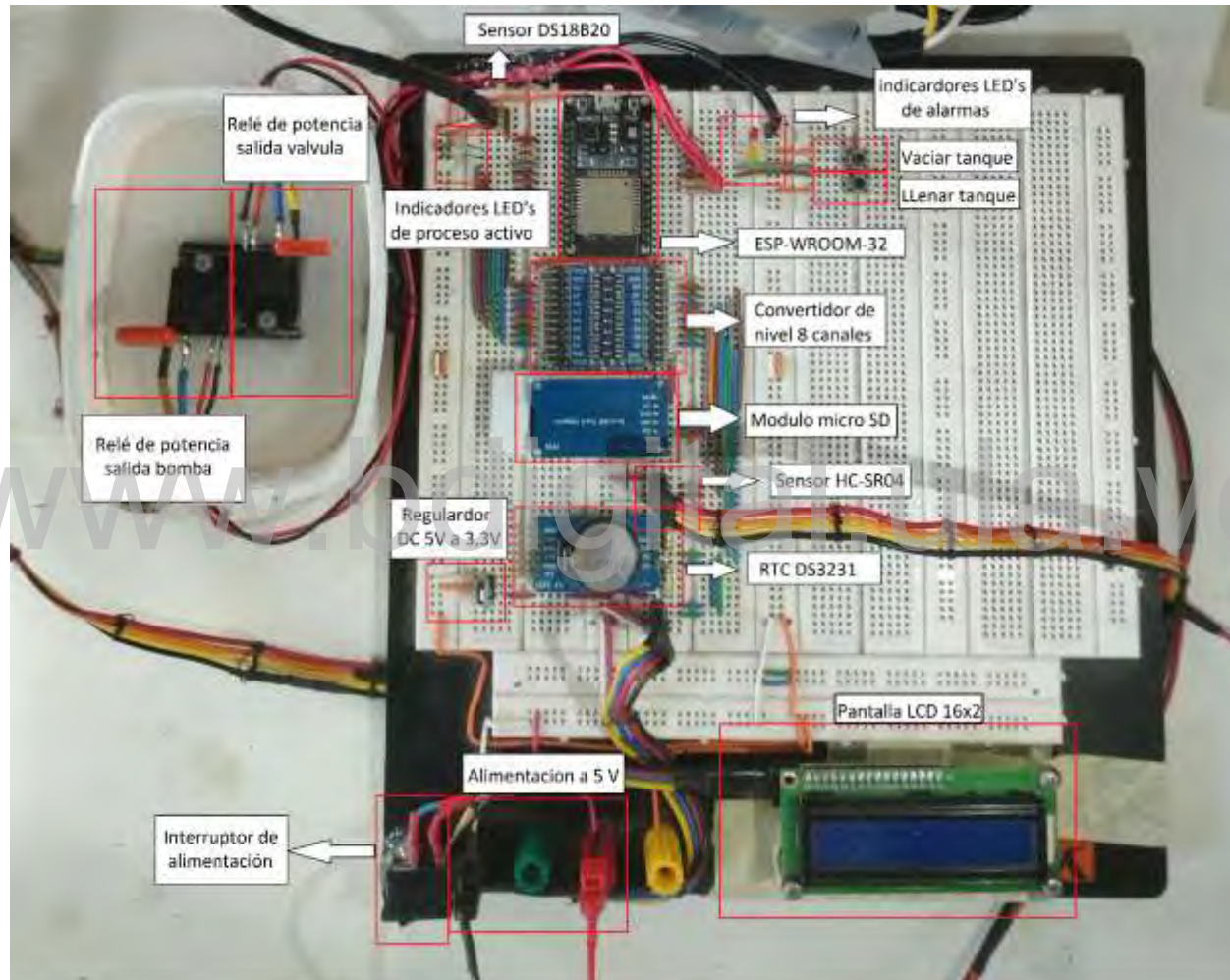


Figura 6.3: Hardware del prototipo implementado para el sistema de medición de tanques SINCONT ESP-32

La imagen anterior muestra el prototipo final usado en el sistema de medición de tanques, en esta imagen se puede apreciar todo el *hardware* implementado para el desarrollo del prototipo.

### 6.1.3.2 Diseño de los circuitos impresos para futura implementación real.

Tal como se mencionó en la sección 4.1.8 los circuitos fueron divididos en 2 placas físicas, a continuación se presenta el diseño del circuito impreso y una vista 3D de dichas placas:

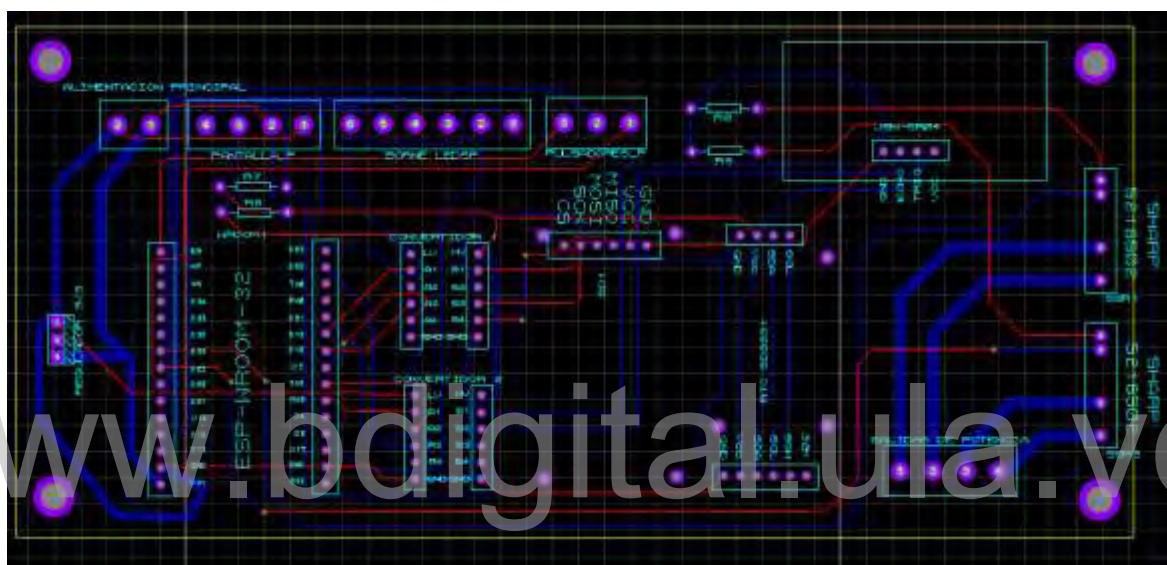


Figura 6.4: Circuito impreso correspondiente a la placa A

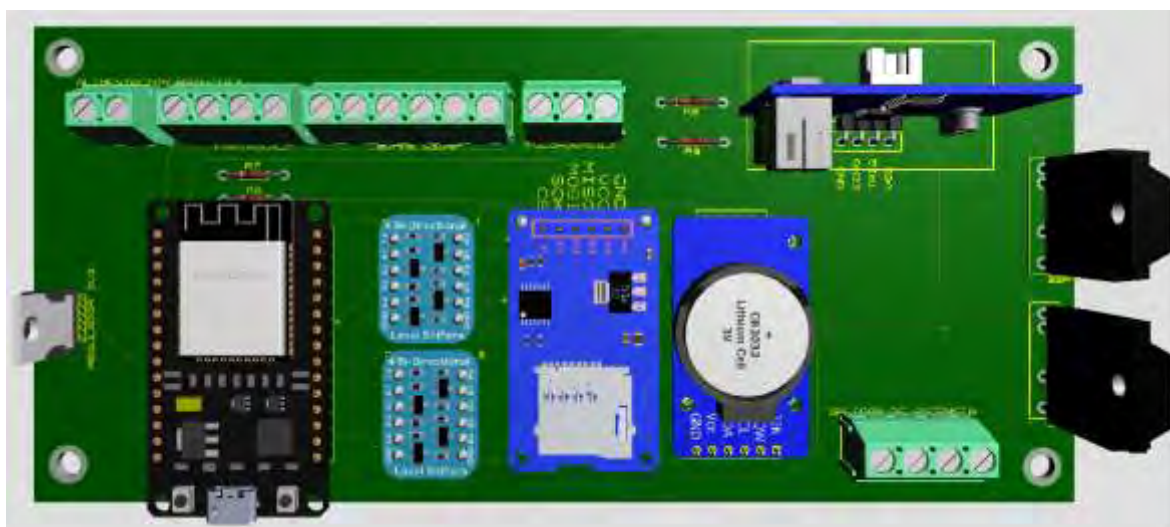


Figura 6.5: Vista 3D del circuito impreso correspondiente a la placa A

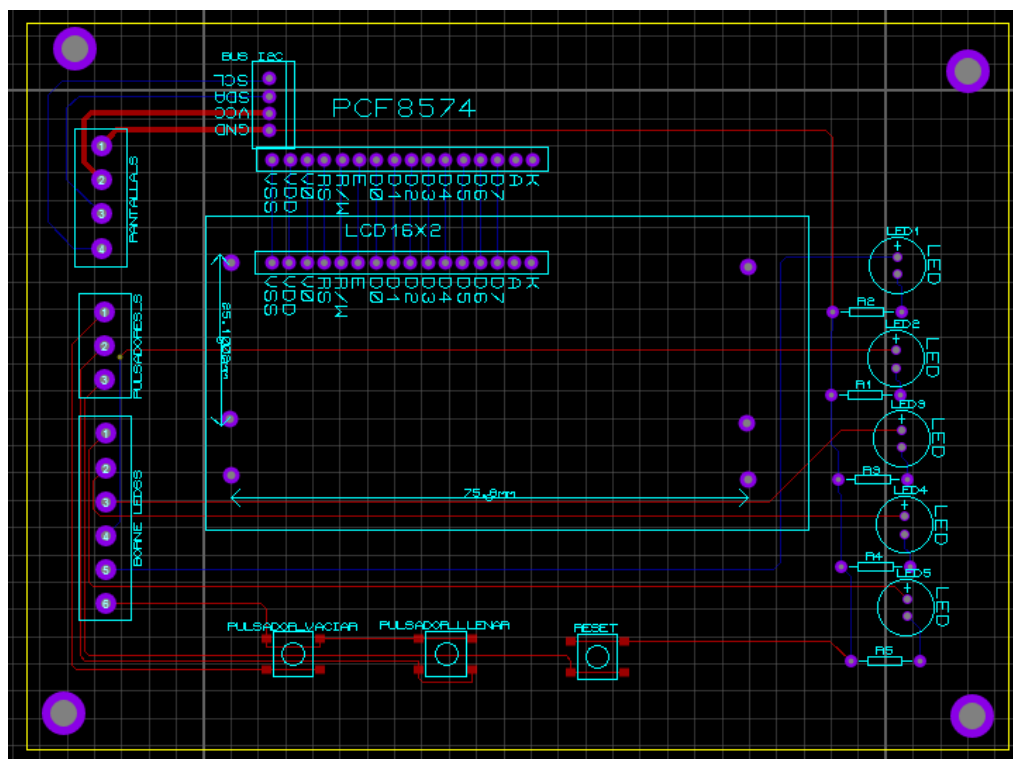


Figura 6.6: Circuito impreso correspondiente a la placa B

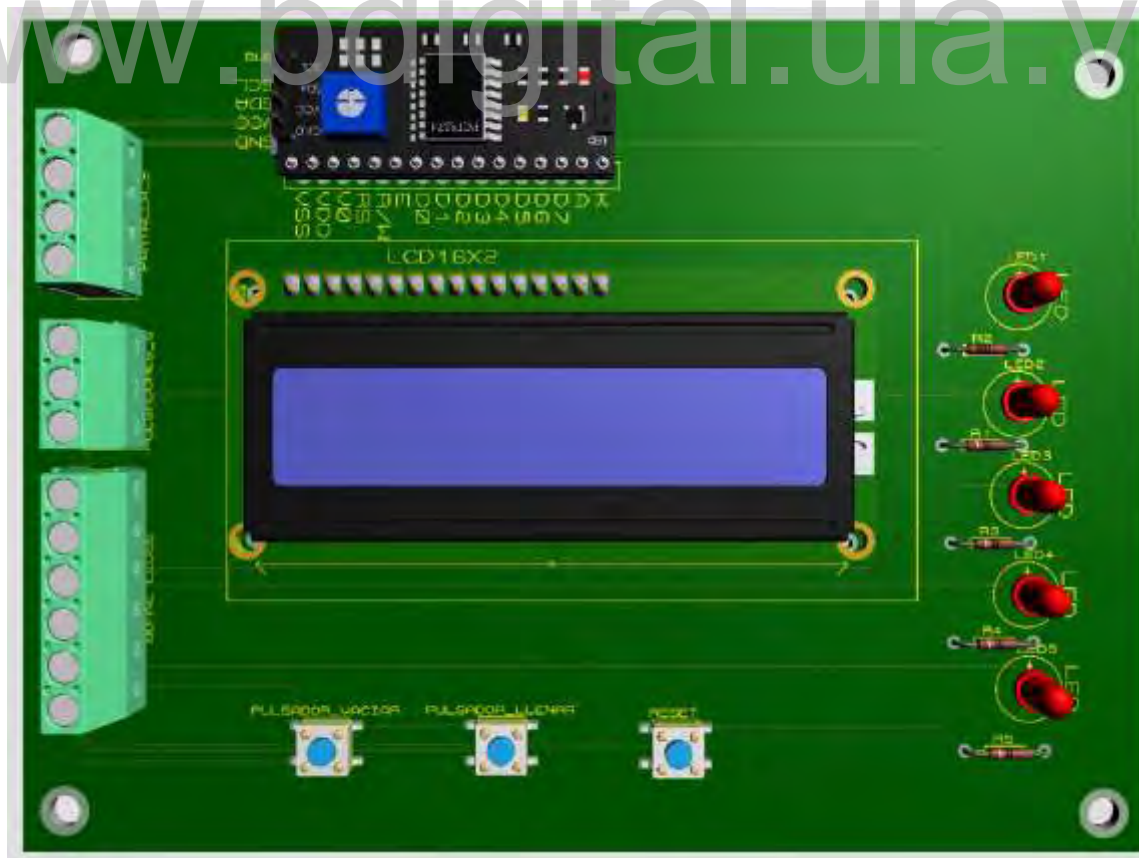


Figura 6.7: Vista 3D del circuito impreso correspondiente a la placa B

### 6.1.3.3 Visualizaciones de pantalla del sistema de medición funcionando

A continuación la Figura 6.8 muestra los mensajes que muestra la pantalla LCD 16x2 al momento de realizar el estado de configuración hasta llegar al estado de medición.

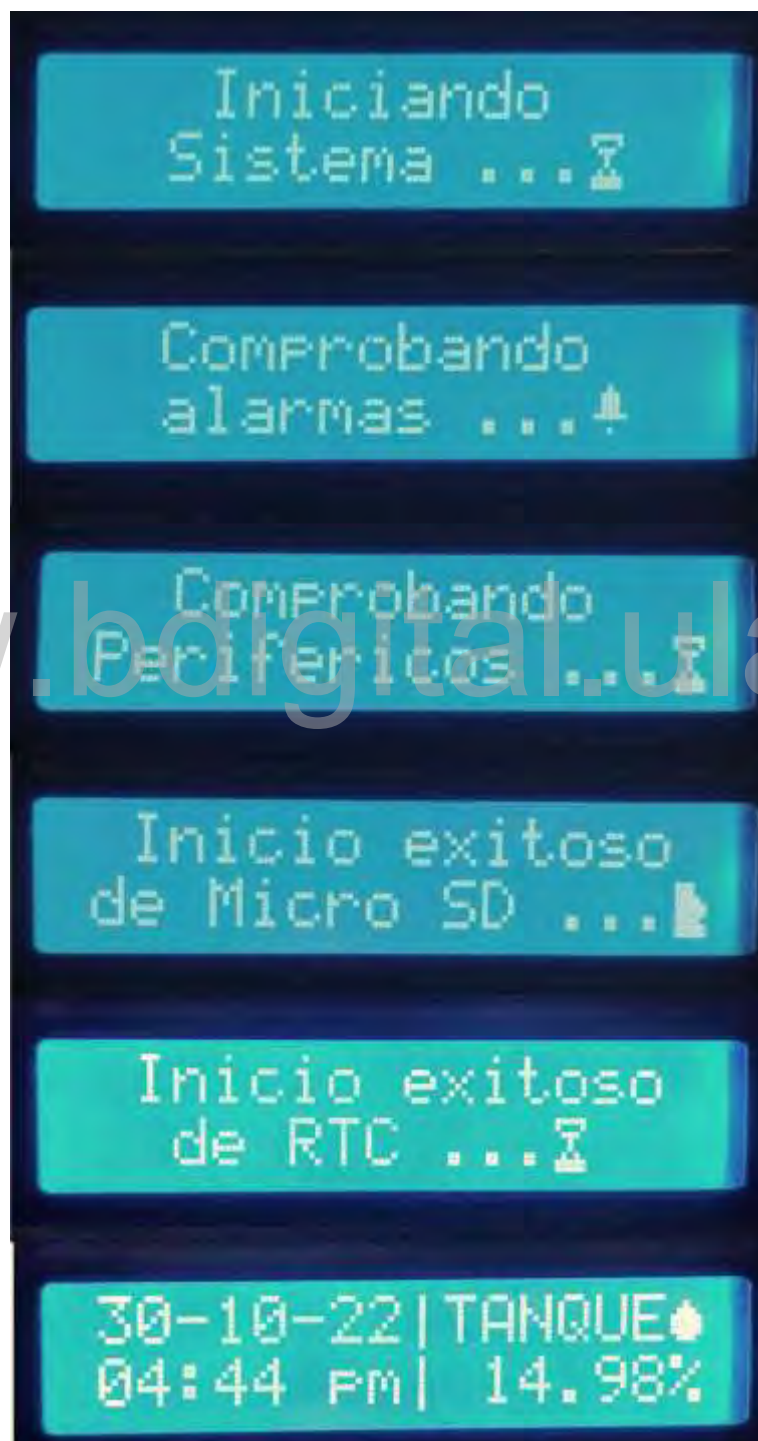


Figura 6.8: Función de autodiagnóstico del sistema a través de la pantalla LCD 16x2.

La Figura 6.9 muestra la secuencia de visualización mostrada por la pantalla LCD en una operación de llenado del tanque.



*Figura 6.9: Visualización en la pantalla 16x2 de una operación de llenado.*

Después de haber mostrado la visualización de la configuración y la operación de llenado, es tiempo de mostrar como son almacenados los reportes de inventario en la memoria micro SD. La figura x muestra un archivo de tipo .txt con un reporte de operaciones diarias, en este se puede apreciar los datos albergados por cada operación de entrada o salida del tanque tal como fue mostrado en la Figura 5.4

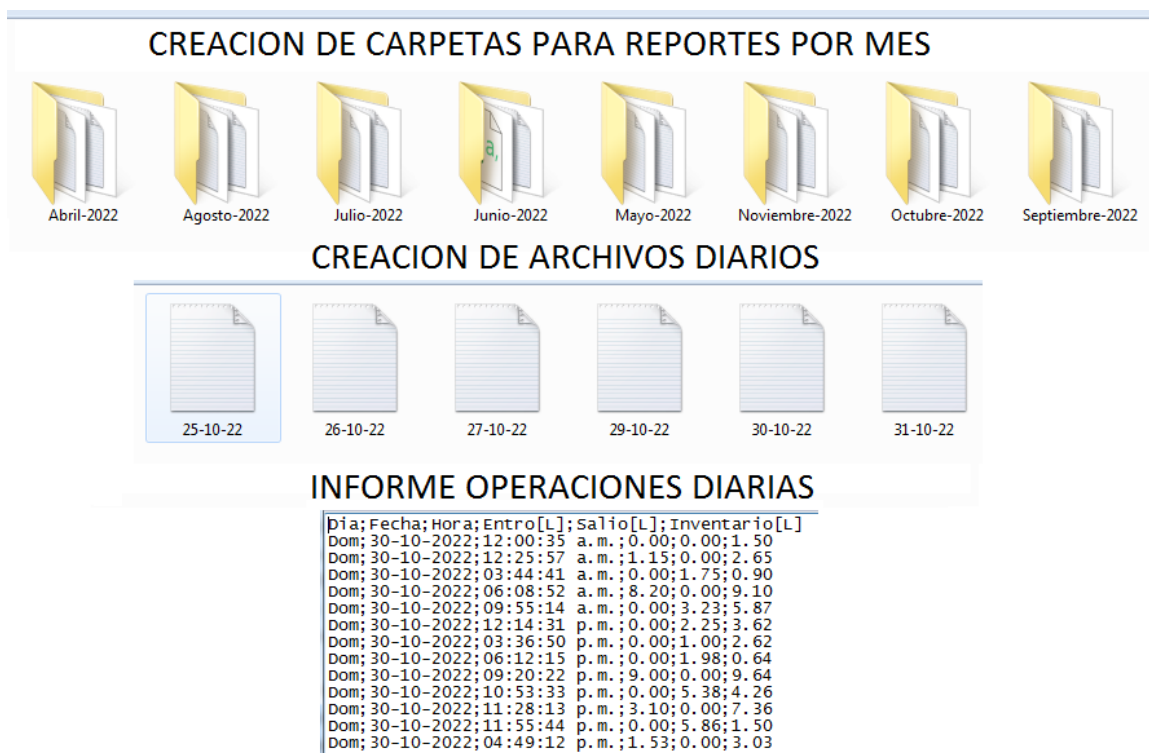


Figura 6.10: Almacenaje de los registros de entrada- salida e inventario del tanque.

## 6.1.4 Funcionamiento de la comunicación inalámbrica.

### 6.1.4.1 Conexión del servidor al enrutador

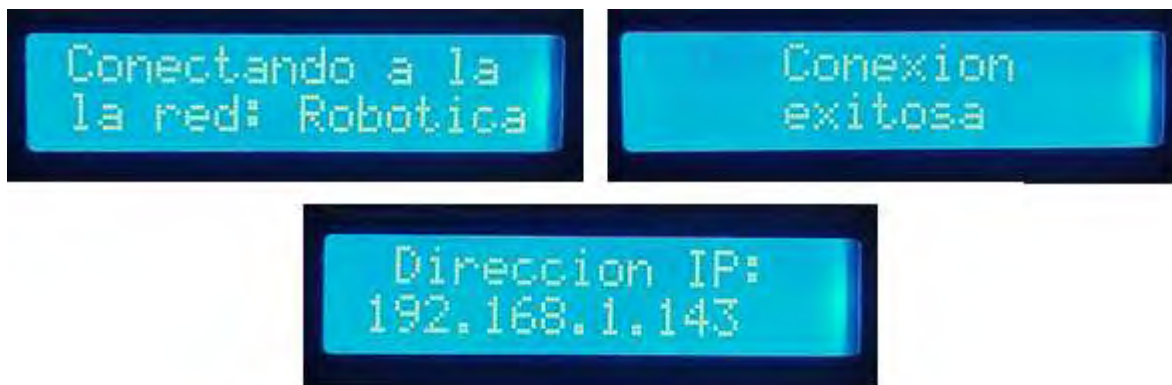


Figura 6.11: Conexión del esp-32 a la red local creada por el enrutador.

La Figura 6.11 muestra la secuencia de visualización que muestra el microcontrolador ESP-32 al establecer comunicación con un enrutador a través de su modo *Station*, una vez que se logra comprobar las credenciales de la red se asigna una dirección IP al microcontrolador con la cual se podrá establecer comunicación entre el servidor del sistema de medición y los dispositivos conectados a dicha red.

### 6.1.4.2 Funcionamiento de la página web



Figura 6.12: Vista inicial de la página web que sirve como interfaz gráfica inalámbrica para el sistema de medición.

La Figura 6.12 muestra la página web que se carga por defecto al momento de ingresar la dirección IP asignada al servidor alojado en el microcontrolador ESP-32, esta página cuenta con una sección para los botones de control y un tercer botón para solicitar los reportes de operación, además se puede visualizar el área de monitoreo del sistema de medición, a su vez incorpora la sección en la que se ingresa la fecha de reporte a solicitar y por último se muestra el área en la cual se muestran los reportes diarios mediante diagramas de barras.

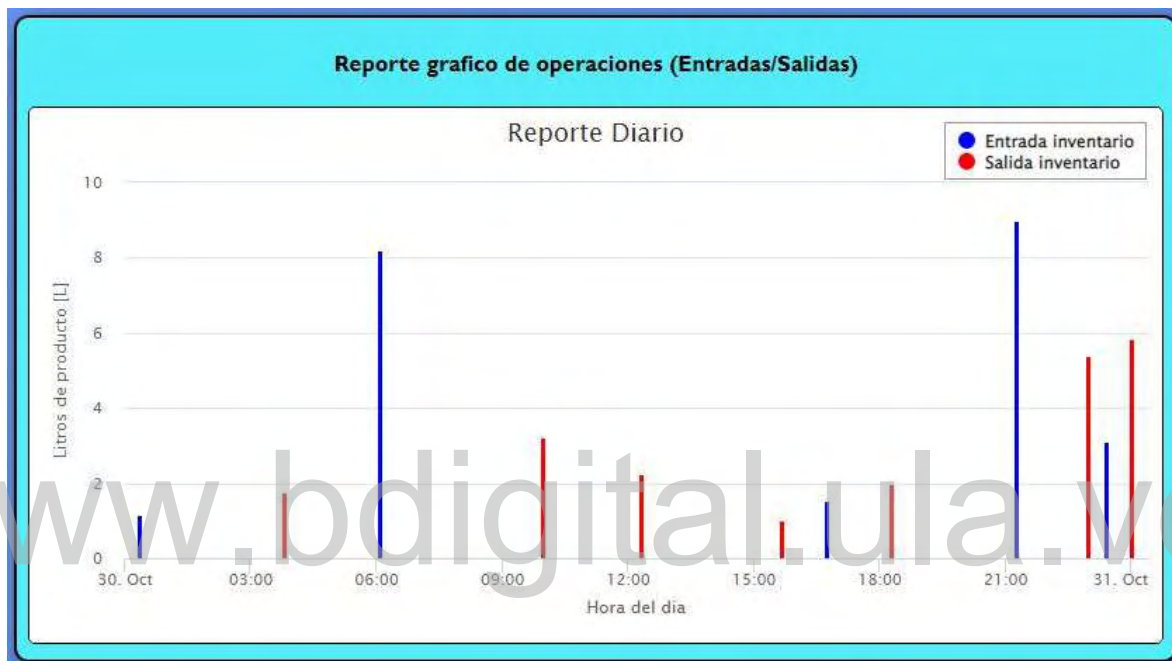


Figura 6.13: Reporte de operaciones diario obtenido a través de la página web.

### 6.1.5 Factibilidad económica del prototipo

Al comienzo de esta investigación se planteó la siguiente pregunta, ¿Será factible técnica y económicamente el diseño de un prototipo electrónico para la medición de nivel que permita llevar el registro y control de líquido en los tanques? Hasta el momento se ha podido probar desde un punto de vista técnico que si es posible llevar a cabo este proyecto. Sin embargo, es necesario probar la factibilidad del proyecto desde un punto de vista económico; para realizar esta actividad se consultó a uno de los proveedores de componentes electrónicos más grandes y accesibles del mundo, pensando en que el mejor balance beneficio/costo se obtendrá al adquirir los componentes en grandes cantidades, con lo cual se reduce los gastos de envío. En este sentido, el proveedor con el cual se elaboró el siguiente presupuesto es AliExpress. A continuación, la Tabla 6.3 muestra se muestra los precios estimados de los componentes:

Tabla 6.3: Presupuesto para la construcción del prototipo para medición de tanques

Cantidad	Componente	Costo unitario (usd)	Costo total (usd)
1	ESP-32 DEVKIT V1 [37]	3.70	3.70
2	PCB [38]	2.50	5.00
1	Pantalla LCD 16X2 I2C [39]	0.91	0.91
1	Sensor JSN-SR04 [40]	5.44	5.44
1	Módulo RTC DS3231 [41]	1.90	1.90
1	Módulo Micro SD [42]	1.06	1.06
1	Tarjeta de memoria 2 Gb [43]	2.54	2.54
2	Convertor de niveles lógicos (5v-3.3v) [44]	1.58	3.16
1	Fuente de Alimentación DC 5V@2A [45]	3.51	3.51
1	Regulador DC 5V-3.3V [46]	0.60	0.60
1	Interruptor de energía [47]	0.60	0.60
3	Pulsadores [48]	0.07	0.21
5	LED's varios colores [49]	0.03	0.15
9	Resistencias [50]	0.02	0.18
2	Relé de estado sólido S216S02 [51]	4.74	9.47
1	Envío E.E.U.U. – Venezuela [52]	4.89	4.89
<b>TOTAL(USD):</b>			<b>43.32</b>

Para la elaboración de este presupuesto no se tomó en cuenta los costos de desarrollo del software ni los costos de armado de la placa. Sin embargo, este presupuesto toma en cuenta los gastos de envío desde china hasta E.E.U.U y posteriormente se añade el costo de envío desde E.E.U.U hasta Venezuela, esto permite disminuir los costos por gastos de envío. Debido que el sistema de medición de tanque diseñado compone un sistema completo en el cual se abarca medición, control y comunicaciones no se encontró en la investigación un equipo que abarque todo esto en un solo dispositivo embebido, por lo cual se decidió realizar la comparación con un PLC, ya que es uno de los controladores más usados en la industria.

En tal sentido, se seleccionó el PLC de la marca Siemens modelo LOGO! 6ED1052-1CC08-0BA0 el cual posee la capacidad de albergar un servidor web, además de contar con espacio para una memoria micro SD para registro de datos y comunicación a través de su conectividad *Ethernet* integrada, este equipo posee un costo de 115,07 \$ [53], sin incluir los gastos de envío desde E.E.U.U hasta Venezuela. Además, debe añadirse el costo de la fuente

para el PLC, el transductor con el cual se hará la medición de tanque y otros periféricos. Como es evidente, tan solo el equipo PLC supera el costo del presupuesto planteado para el desarrollo del sistema de medición de tanque, por lo cual hace que el proyecto cumpla con la característica de ser económicamente viable.

[www.bdigital.ula.ve](http://www.bdigital.ula.ve)

## CONCLUSIONES

La instrumentación industrial y la automatización de procesos en la actualidad se han vuelto de vital importancia para la productividad y competitividad de una empresa. En este sentido, para poder contar con los equipos que presten la mayor confiabilidad, comodidad y eficiencia se requiere mantener actualizada la tecnología a usar en los procesos industriales.

Con el avance de este trabajo de grado, se logró cumplir en su totalidad los objetivos planteados al inicio de esta investigación. A través del estudio de los diferentes métodos existentes para la medición de nivel y los nuevos protocolos de comunicación inalámbrica permitidos por las placas de desarrollo ESP-32. Logrando así, diseñar e implementar un prototipo funcional para la medición y registro del nivel de tanques, el cual es adaptable a una variedad de procesos diferentes. En este sentido, resulta bastante atractivo ya que el diseño no se particularizó para un tipo de tanque en específico, sino que puede ser adaptado según las necesidades del proceso.

Además, el prototipo cuenta con la interfaz web de comunicación inalámbrica desarrollada mediante el uso de HTML, CSS y JavaScript, con la cual se facilita considerablemente el proceso. Ya que, al poder tener acceso inalámbrico tanto a las opciones de control y monitoreo en tiempo real del contenedor, así como a los datos censados en el proceso a través del servidor web asíncrono por medio del protocolo de comunicación WebSocket, prácticamente se puede prescindir de la presencia de un operador para que este constantemente monitoreando el tanque. Facilitando así, las operaciones de la empresa, puesto que al contar con un sistema inteligente que monitoree constantemente el proceso se pueden reducir fallas y aumentar la productividad. Por otro lado, el equipo también es escalable a futuro para añadir las funcionalidades de monitoreo inalámbrico a través de internet con lo cual el supervisor del proceso podrá evaluar los reportes desde cualquier lugar con una conectividad a internet.

El producto final permite realizar las siguientes acciones:

- ✓ Medir el nivel de líquido en un tanque previamente calibrado.
- ✓ Calcular el volumen de líquido contenido en un tanque a través del nivel medido.
- ✓ Registrar la entrada y salida del producto contenido en el recipiente.
- ✓ Monitorear en tiempo real el estado del sistema de manera local y remota.
- ✓ Controlar el estado del sistema a través de los pulsadores físicos y digitales de la página web.
- ✓ Establecer conectividad inalámbrica para solicitar y graficar reportes de consumo del líquido a albergar.

Por otro lado, se pudo probar la ventaja económica que representa este diseño, ya que desde un punto de vista técnico y económico resultó ser factible y cómodo de usar con un costo de significativamente menor al de un sistema de medición comercial. Además, se logró conseguir una tasa de error promedio de  $\pm 2$  mm, lo cual resulta bastante aceptables y dentro de los parámetros de medición.

[www.bdigital.ula.ve](http://www.bdigital.ula.ve)

## RECOMENDACIONES

- ✓ Añadir al servidor una página web de configuración inicial del sistema en la que se pueda configurar las credenciales desde cero o incluso cambiar las credenciales ya establecidas para iniciar la conexión *Wi-Fi* con la red local a conectarse.
- ✓ Aumentar la seguridad de la página web a través de un sistema de usuarios y contraseñas, para restringir el acceso al control del proceso y la información de reportes únicamente al personal autorizado.
- ✓ Completar la sección de reportes agregando la opción de solicitar reportes de operación mensual.
- ✓ Estudiar a profundidad el uso del doble núcleo en el microcontrolador ESP-32 para trabajar en cada uno las tareas de medición y servidor web por separado.
- ✓ Incluir un respaldo de energía a través de baterías de litio y un sistema de detección, que monitoree la entrada de energía de la red de tal manera que el equipo de medición pueda detectar la ausencia de energía, culminar la tarea en curso y posteriormente entre en modo de sueño profundo.

## REFERENCIAS

- [1] L. Hägg y J. Sandberg, La guía del ingeniero para la medición de tanques, 2021 ed., Emerson, 2021.
- [2] C. A. Valencia Aguilar, «Diseño de un sistema de monitoreo de nivel de los tanques de emergencia de EMCALI Telecomunicaciones,» Santiago de Cali, 2013.
- [3] M. E. Valencia Valencia, «Diseño y control automático de tanque de almacenamiento de crudo de petróleo para una refinería,» Piura, Perú, 2019.
- [4] R. Sainz y M. Serna, «Diseño de un entrenador de Arduino con aplicaciones industriales,» 2017.
- [5] O. Cárdenas, Transductores para procesos industriales, 1 ed., Mérida: Talleres gráficos universitarios, 2010.
- [6] A. Creus Solé, Instrumentación industrial, 6 ed., Barcelona: 1998 ALFAOMEGA GRUPO EDITOR, S.A, 1997.
- [7] R. Serway y J. Jewett, Física para ciencias e ingeniería, 7 ed., vol. I, S. R. C. González, Ed., Santa Fe: Cengage Learning Editores, 2008.
- [8] H. D. Young y R. A. Freedman, Física Universitaria, 12 ed., vol. 1, R. F. Rivera, Ed., D.F: Pearson Educación, 2009, p. 760.
- [9] A. Sánchez, «Diferencias entre los métodos para la medición de nivel con radar: Pulsos vs FMCW,» 2019. [En línea]. Available: <https://www.isamex.org/intechmx/index.php/2019/11/05/endresshauser-diferencias-entre-metodos-para-la-medicion-de-nivel-con-radares-pulsos-vs-fmcw/>. [Último acceso: 2 Febrero 2022].

- [10] c. wolff, «Radar de onda continua modulado en frecuencia,» 2011. [En línea]. Available:  
[https://www.radartutorial.eu/02.basics/rp08.es.html#:~:text=El%20radar%20de%20onda%20continua,continua%20simple%20\(radar%20CW\)..](https://www.radartutorial.eu/02.basics/rp08.es.html#:~:text=El%20radar%20de%20onda%20continua,continua%20simple%20(radar%20CW)..) [Último acceso: 2 Febrero 2022].
- [11] M. A. Granada Cañas, D. Buitrago Patiño, E. M. Claro Martínez, D. A. Silva Rincón y A. P. Alvarez, Manual de medicion de hidrocarburos y biocombustibles, Capitulo 2 - Medicion de tanques, Bogotá: ecopetrol, 2013.
- [12] American Petroleum Institute, «Manual of Petroleum Management Standard - Chapter 2: Tank Calibration,» de *Manual of Petroleum Management Standard*, washington D.C, 1995.
- [13] D. Apaza Condori, Microcontroladores PIC fundamentos y aplicaciones un enfoque didactico, Arequipa: Universidad autónoma San Francisco, 2017.
- [14] F. Bordignon y A. Iglesias, Diseño y construcción de objetos interactivos digitales : experimentos con la plataforma Arduino, Gonnet: UNIPE, 2015.
- [15] F. Leon, «Buses de datos en electrónica Paralelo, UART, SPI, I2C,» Dynamo Electronics, [En línea]. Available: <https://dynamoelectronics.com/buses-de-datos-en-electronica-paralelo-uart-spi-i2c/>. [Último acceso: 22 Febrero 2022].
- [16] L. LLamas, «Medir temperatura de líquidos y gases con Arduino y DS18B20,» 27 Junio 2016. [En línea]. Available: <https://www.luisllamas.es/temperatura-liquididos-arduino-ds18b20/>. [Último acceso: 5 Febrero 2022].
- [17] L. LLamas, «El bus I2C en Arduino,» 18 Mayo 2016. [En línea]. Available: <https://www.luisllamas.es/arduino-i2c/>. [Último acceso: 5 Febrero 2022].

- [18] L. LLamas, «El bus SPI en Arduino,» 14 Mayo 2016. [En línea]. Available: <https://www.luisllamas.es/arduino-spi/>. [Último acceso: 5 Febrero 2022].
- [19] A. Lázaro, «Hardware y software: definiciones y conceptos,» Profesional review, 10 Noviembre 2019. [En línea]. Available: <https://www.profesionalreview.com/2019/11/10/hardware-software-definiciones/#:~:text=hardware%20y%20software-,Definici%C3%B3n%20de%20hardware%2C%20principales%20tipos%20y%20componentes,H%2FW%20o%20h%2Fw..> [Último acceso: 15 Enero 2022].
- [20] J. E. Crespo, «IDE Arduino,» 11 Diciembre 2016. [En línea]. Available: <https://aprendiendoarduino.wordpress.com/2016/12/11/ide-arduino/>. [Último acceso: 15 Enero 2022].
- [21] «QUÉ ES LA PROGRAMACIÓN CON ARDUINO Y PARA QUÉ SIRVE,» Bejob, 14 Febrero 2017. [En línea]. Available: <https://www.bejob.com/que-es-la-programacion-con-arduino-y-para-que-sirve/#:~:text=Lenguaje%20de%20la%20programaci%C3%B3n%20con%20Arduino%3A%2C%2B%2B&text=La%20plataforma%20Arduino%20se%20programa,es%20similar%20a%2C%2B%2B..> [Último acceso: 15 Enero 2022].
- [22] D. Garcia, «Definicion de servidor,» 5 Febrero 2018. [En línea]. Available: <https://www.economiasimple.net/glosario/servidor#:~:text=El%20servidor%20es%20la%20m%C3%A1quina,t%C3%A9cnicos%20que%20con%20un%20hosting..> [Último acceso: 23 Agosto 2022].
- [23] A. Caballero Oropeza, «Protocolos utilizados en internet,» 29 Octubre 2014. [En línea]. Available: <https://sites.google.com/site/instalacionderedesdecomputo/home/protocolos-utilizados-en-internet>. [Último acceso: 23 Agosto 2022].

- [24] R. Santos, «ESP32 WebSocket Server: Control Outputs (Arduino IDE),» [En línea]. Available: <https://randomnerdtutorials.com/esp32-websocket-server-arduino/>. [Último acceso: 23 Agosto 2022].
- [25] A. R. Bruno Saravia, «ESP-32 NODE MCU,» 2019. [En línea]. Available: [www.microelectronicash.com](http://www.microelectronicash.com). [Último acceso: 13 Octubre 2021].
- [26] J. Hurtado de Barrera, Metodología de la investigación: Guía para la comprensión holística de la ciencia, 4 ed., Caracas: Quiron ediciones, 2010.
- [27] E. G. Maida y J. Pacienza, «Metodologías de desarrollo de software,» Buenos Aires, 2015.
- [28] R. Pressman, Ingeniería del software: un enfoque práctico, 7 ed., P. R. Vazquez, Ed., Ciudad de México, D.F: McGraw-Hill, 2010.
- [29] J. L. Enriquez, E. Farias, E. Flores, C. Honores, R. Llanos, W. López, V. Medina, C. Olivos, C. Torres, G. Velázquez y A. Zuñiga, Metodología de desarrollo de software, 001 ed., Chimbote: División de sistemas, 2017.
- [30] R. Hernández Sampieri, C. Fernández Collado y P. Baptista Lucio, Metodología de la investigación, 6ta ed., Mexico D.F, Santa Fe: McGraw-Hill, 2014.
- [31] F. G. Arias, El proyecto de investigación: introducción a la metodología científica, 6ta ed., Caracas: Editorial EPISTEME C.A., 2012.
- [32] Y. M. Peña Puerto y C. A. Almeida Ortega, «Una visión de la aplicación del software en metrología,» *Met&Flu (Metrología & Fluidos): Ciencia, Tecnología e Innovación*, n° 3, pp. 6-13, 2010.
- [33] Espressif systems, «ESP32-WROOM-32 Datasheet,» 2022. [En línea]. Available: [https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32\\_datasheet\\_en.pdf](https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32_datasheet_en.pdf). [Último acceso: 1 Noviembre 2022].

- [34] Leantec robotics & Electronics, «datasheet HC-SR04,» [En línea]. Available: <https://leantec.es/wp-content/uploads/2019/06/Leantec.ES-HC-SR04.pdf>. [Último acceso: 30 Octubre 2022].
- [35] Maxim Integrated, «DS18B20,» 2019. [En línea]. Available: <https://datasheets.maximintegrated.com/en/ds/DS18B20.pdf>. [Último acceso: 2022 Noviembre 2022].
- [36] W. Ewald, «Sensores de distancia HC-SR04 y JSN-SR04T-2.0,» 7 Noviembre 2020. [En línea]. Available: <https://wolles-elektronikkiste.de/en/hc-sr04-and-jsn-sr04t-2-0-distance-sensors>. [Último acceso: 28 Octubre 2022].
- [37] aliExpress, «Placa central de ESP32-DevKitC, placa de desarrollo ESP32 V4, ESP32-WROOM-32D/U,» [En línea]. Available: [https://www.aliexpress.us/item/3256801434697430.html?spm=a2g0o.productlist.0.0.6ba121e5Waty64&algo\\_pvid=8ede857e-d332-4e42-b74b-64881a0f16aa&algo\\_exp\\_id=8ede857e-d332-4e42-b74b-64881a0f16aa-47&pdp\\_ext\\_f=%7B%22sku\\_id%22%3A%2212000028808965005%22%7D&pdp\\_npi](https://www.aliexpress.us/item/3256801434697430.html?spm=a2g0o.productlist.0.0.6ba121e5Waty64&algo_pvid=8ede857e-d332-4e42-b74b-64881a0f16aa&algo_exp_id=8ede857e-d332-4e42-b74b-64881a0f16aa-47&pdp_ext_f=%7B%22sku_id%22%3A%2212000028808965005%22%7D&pdp_npi). [Último acceso: 5 Noviembre 2022].
- [38] JLCPCB, «cart.jlpcb.com,» [En línea]. Available: <https://cart.jlpcb.com/quote?orderType=1&stencilLayer=2&stencilWidth=150&stencilLength=150&stencilCounts=5>. [Último acceso: 5 Noviembre 2022].
- [39] AliExpress, «Módulo LCD1602 + I2C, pantalla verde azul/amarilla, pantalla LCD de 16x2 caracteres PCF8574T PCF8574 IIC I2C, interfaz 5V para arduino,» [En línea]. Available: [https://www.aliexpress.us/item/2251832251811574.html?spm=a2g0o.productlist.0.0.50a05f86rMjiQK&algo\\_pvid=3f9d66ce-1016-439e-a9df-2e4587c709b5&algo\\_exp\\_id=3f9d66ce-1016-439e-a9df-2e4587c709b5-1&pdp\\_ext\\_f=%7B%22sku\\_id%22%3A%2212000026802941987%22%7D&pdp\\_npi](https://www.aliexpress.us/item/2251832251811574.html?spm=a2g0o.productlist.0.0.50a05f86rMjiQK&algo_pvid=3f9d66ce-1016-439e-a9df-2e4587c709b5&algo_exp_id=3f9d66ce-1016-439e-a9df-2e4587c709b5-1&pdp_ext_f=%7B%22sku_id%22%3A%2212000026802941987%22%7D&pdp_npi). [Último acceso: 5 Noviembre 2022].

- [40] AliExpress, «Módulo de Sensor ultrasónico a prueba de agua, JSN-SR04T, JSN SR04T, transductor de medición de distancia para Kit electrónico Arduino,» [En línea]. Available: [https://www.aliexpress.us/item/3256801407713294.html?spm=a2g0o.productlist.0.0.7c0c51490Oa1ep&algo\\_pvid=b846c2ce-3f5d-4cf0-81ee-6e266ae910bb&algo\\_exp\\_id=b846c2ce-3f5d-4cf0-81ee-6e266ae910bb-0&pdp\\_ext\\_f=%7B%22sku\\_id%22%3A%2212000016716811843%22%7D&pdp\\_npi=](https://www.aliexpress.us/item/3256801407713294.html?spm=a2g0o.productlist.0.0.7c0c51490Oa1ep&algo_pvid=b846c2ce-3f5d-4cf0-81ee-6e266ae910bb&algo_exp_id=b846c2ce-3f5d-4cf0-81ee-6e266ae910bb-0&pdp_ext_f=%7B%22sku_id%22%3A%2212000016716811843%22%7D&pdp_npi=). [Último acceso: 5 Noviembre 2022].
- [41] AliExpress, «Módulo DS3231 AT24C32 IIC DS1302 Módulo de reloj de precisión DS1307 módulo de memoria mini módulo en tiempo Real 3,3 V/5V para Raspberry Pi,» [En línea]. Available: [https://www.aliexpress.us/item/2251832646821825.html?spm=a2g0o.productlist.0.0.216d672bug9ghP&algo\\_pvid=f5b73908-e832-4abe-91ba-527831973a00&algo\\_exp\\_id=f5b73908-e832-4abe-91ba-527831973a00-0&pdp\\_ext\\_f=%7B%22sku\\_id%22%3A%2210000011431877724%22%7D&pdp\\_npi=](https://www.aliexpress.us/item/2251832646821825.html?spm=a2g0o.productlist.0.0.216d672bug9ghP&algo_pvid=f5b73908-e832-4abe-91ba-527831973a00&algo_exp_id=f5b73908-e832-4abe-91ba-527831973a00-0&pdp_ext_f=%7B%22sku_id%22%3A%2210000011431877724%22%7D&pdp_npi=). [Último acceso: 5 Niviembre 2022].
- [42] AliExpress, «Tarjeta Micro SD TF, módulo de protección de memoria SPI, tarjeta de expansión de almacenamiento Micro SD para arduino, Kit Diy,» [En línea]. Available: [https://www.aliexpress.us/item/2251832691016649.html?spm=a2g0o.productlist.0.0.7848https://www.aliexpress.us/item/3256802343782025.html?spm=a2g0o.productlist.0.0.38df690bzmyWgb&algo\\_pvid=9659d775-f688-43c6-b39e-1544dfcc0aff&algo\\_exp\\_id=9659d775-f688-43c6-](https://www.aliexpress.us/item/2251832691016649.html?spm=a2g0o.productlist.0.0.7848https://www.aliexpress.us/item/3256802343782025.html?spm=a2g0o.productlist.0.0.38df690bzmyWgb&algo_pvid=9659d775-f688-43c6-b39e-1544dfcc0aff&algo_exp_id=9659d775-f688-43c6-). [Último acceso: 5 Noviembre 2022].
- [43] AliExpress, «Tarjeta de memoria Micro TF de 128MB, 256MB, 512MB, 1GB, 2GB, 4GB, 8GB, 16GB, 32GB, tarjeta SD de clase 4 para teléfono inteligente de alta capacidad para adaptador de juegos,» [En línea]. Available: [https://www.aliexpress.us/item/3256803470893755.html?spm=a2g0o.productlist.0.0.137614c4iwWiBD&algo\\_pvid=8a3dd603-1c0c-4301-ae1e-f2a62ed35d52&algo\\_exp\\_id=8a3dd603-1c0c-4301-ae1e-f2a62ed35d52-](https://www.aliexpress.us/item/3256803470893755.html?spm=a2g0o.productlist.0.0.137614c4iwWiBD&algo_pvid=8a3dd603-1c0c-4301-ae1e-f2a62ed35d52&algo_exp_id=8a3dd603-1c0c-4301-ae1e-f2a62ed35d52-)

13&pdp\_ext\_f=%7B%22sku\_id%22%3A%2212000026678013845%22%7D&pdp\_npi. [Último acceso: 5 Noviembre 2022].

[44] AliExpress, «Convertidor de nivel lógico IIC I2C de 5 piezas y 4 canales, módulo bidireccional de 5V a 3,3 V,» [En línea]. Available: [https://www.aliexpress.us/item/2251832585558278.html?spm=a2g0o.productlist.0.0.735e47e27RJL6m&algo\\_pvid=6b17a162-78b3-45a7-bcdb-60458849aafb&algo\\_exp\\_id=6b17a162-78b3-45a7-bcdb-60458849aafb-1&pdp\\_ext\\_f=%7B%22sku\\_id%22%3A%2212000031118412998%22%7D&pdp\\_npi=](https://www.aliexpress.us/item/2251832585558278.html?spm=a2g0o.productlist.0.0.735e47e27RJL6m&algo_pvid=6b17a162-78b3-45a7-bcdb-60458849aafb&algo_exp_id=6b17a162-78b3-45a7-bcdb-60458849aafb-1&pdp_ext_f=%7B%22sku_id%22%3A%2212000031118412998%22%7D&pdp_npi=). [Último acceso: 5 Noviembre 2022].

[45] AliExpress, «Adaptador USB de viaje de 5V, 1A, 2A, 3A, cargador de teléfono, fuente de alimentación, cargador de pared de escritorio, enchufe de la UE/EE. UU./AU, negro y blanco,» [En línea]. Available: [https://www.aliexpress.us/item/3256801578303078.html?spm=a2g0o.productlist.0.0.6f3a4167y6EVWf&algo\\_pvid=b74e5390-0cfc-4c63-bc32-2f0bcbb80935&algo\\_exp\\_id=b74e5390-0cfc-4c63-bc32-2f0bcbb80935-52&pdp\\_ext\\_f=%7B%22sku\\_id%22%3A%2212000017474987896%22%7D&pdp\\_npi=](https://www.aliexpress.us/item/3256801578303078.html?spm=a2g0o.productlist.0.0.6f3a4167y6EVWf&algo_pvid=b74e5390-0cfc-4c63-bc32-2f0bcbb80935&algo_exp_id=b74e5390-0cfc-4c63-bc32-2f0bcbb80935-52&pdp_ext_f=%7B%22sku_id%22%3A%2212000017474987896%22%7D&pdp_npi=). [Último acceso: 5 Noviembre 2022].

[46] AliExpress, «Módulo de fuente de alimentación de DC-DC, AMS1117, 3,3 V/5V, reductor de potencia, AMS1117-5.0V, LDO, 800MA, AMS1117-3.3V,» [En línea]. Available: [https://www.aliexpress.us/item/3256802894390895.html?spm=a2g0o.productlist.0.0.33c13dce3hlLdU&algo\\_pvid=b773e6f7-03f8-4e8a-a815-7a34be8d3a10&algo\\_exp\\_id=b773e6f7-03f8-4e8a-a815-7a34be8d3a10-14&pdp\\_ext\\_f=%7B%22sku\\_id%22%3A%2212000023958402816%22%7D&pdp\\_npi=](https://www.aliexpress.us/item/3256802894390895.html?spm=a2g0o.productlist.0.0.33c13dce3hlLdU&algo_pvid=b773e6f7-03f8-4e8a-a815-7a34be8d3a10&algo_exp_id=b773e6f7-03f8-4e8a-a815-7a34be8d3a10-14&pdp_ext_f=%7B%22sku_id%22%3A%2212000023958402816%22%7D&pdp_npi=). [Último acceso: 5 Noviembre 2022].

[47] AliExpress, «Interruptor basculante de enclavamiento, interruptor de encendido e/O de 4/6 pines con luz 16A 250VAC 20A 125VAC KCD4, 1 ud.,» [En línea]. Available: <https://www.aliexpress.us/item/2251832644495685.html?spm=a2g0o.productlist.0.0>.

3e504324Sj7Poe&algo\_pvid=d196ee3c-edc4-4ef0-814a-d232dc81a777&algo\_exp\_id=d196ee3c-edc4-4ef0-814a-d232dc81a777-6&pdp\_ext\_f=%7B%22sku\_id%22%3A%2212000020812107111%22%7D&pdp\_npi=. [Último acceso: 5 Noviembre 2022].

- [48] AliExpress, «Interruptor de botón táctil momentáneo PCB de plástico, 6x6mm, 12x12mm, 4 pines, 4,3 2 pines de altura/5/6/7/10/12/15mm, 50 Uds.» [En línea]. Available:

[https://www.aliexpress.us/item/2251832760126793.html?spm=a2g0o.productlist.0.0.1d2f5ea6dLyoyY&algo\\_pvid=0ba2c717-8d46-4d2a-9a4b-c9cd02f1f53b&algo\\_exp\\_id=0ba2c717-8d46-4d2a-9a4b-c9cd02f1f53b-26&pdp\\_ext\\_f=%7B%22sku\\_id%22%3A%2266309820530%22%7D&pdp\\_npi=2%40d](https://www.aliexpress.us/item/2251832760126793.html?spm=a2g0o.productlist.0.0.1d2f5ea6dLyoyY&algo_pvid=0ba2c717-8d46-4d2a-9a4b-c9cd02f1f53b&algo_exp_id=0ba2c717-8d46-4d2a-9a4b-c9cd02f1f53b-26&pdp_ext_f=%7B%22sku_id%22%3A%2266309820530%22%7D&pdp_npi=2%40d). [Último acceso: 5 Noviembre 2022].

- [49] AliExpress, «Juego de surtido de diodos emisores de luces LED de 3mm y 5mm para Arduino, blanco brillante, rojo, azul, verde, amarillo, F3, F5,» [En línea]. Available:

[https://www.aliexpress.us/item/3256803478351473.html?spm=a2g0o.productlist.0.0.319fdfa3Rx6AeD&algo\\_pvid=f35fe837-51ca-413e-ad7b-8fb1f48c7228&algo\\_exp\\_id=f35fe837-51ca-413e-ad7b-8fb1f48c7228-3&pdp\\_ext\\_f=%7B%22sku\\_id%22%3A%2212000026708831704%22%7D&pdp\\_npi=.](https://www.aliexpress.us/item/3256803478351473.html?spm=a2g0o.productlist.0.0.319fdfa3Rx6AeD&algo_pvid=f35fe837-51ca-413e-ad7b-8fb1f48c7228&algo_exp_id=f35fe837-51ca-413e-ad7b-8fb1f48c7228-3&pdp_ext_f=%7B%22sku_id%22%3A%2212000026708831704%22%7D&pdp_npi=.) [Último acceso: 5 Noviembre 2022].

- [50] AliExpress, «Resistencia de película de carbono, 1/2w, 100 W, 0,5, 1R ~ 1M, 2.2R, 10R, 22R, 47R, 51R, 100R, 150R, 470R, 1K, 5% K, 10K, 47K, 1, 4,7, 10, 22, 47, 51, 2,2, 100, 150 unidades,» [En línea]. Available:

[https://www.aliexpress.us/item/3256801494097920.html?spm=a2g0o.productlist.0.0.579377d9z5ZRCo&algo\\_pvid=6f9faa60-814e-465c-8e7d-195a82db1a49&algo\\_exp\\_id=6f9faa60-814e-465c-8e7d-195a82db1a49-5&pdp\\_ext\\_f=%7B%22sku\\_id%22%3A%2212000017108863176%22%7D&pdp\\_npi=.](https://www.aliexpress.us/item/3256801494097920.html?spm=a2g0o.productlist.0.0.579377d9z5ZRCo&algo_pvid=6f9faa60-814e-465c-8e7d-195a82db1a49&algo_exp_id=6f9faa60-814e-465c-8e7d-195a82db1a49-5&pdp_ext_f=%7B%22sku_id%22%3A%2212000017108863176%22%7D&pdp_npi=.) [Último acceso: 5 Noviembre 2022].

- [51] AliExpress, «S202S02 S202SE2 S216S02 S216SE2 S216S01 S202S01 SIP-4 TO3P-4 en Stock,» [En línea]. Available: [https://www.aliexpress.us/item/3256802317020074.html?spm=a2g0o.productlist.0.0.4028b81cJt2JL0&algo\\_pvid=da58bd97-3739-4a01-8935-cb5282885e16&algo\\_exp\\_id=da58bd97-3739-4a01-8935-cb5282885e16-0&pdp\\_ext\\_f=%7B%22sku\\_id%22%3A%2212000020906639428%22%7D&pdp\\_npi=](https://www.aliexpress.us/item/3256802317020074.html?spm=a2g0o.productlist.0.0.4028b81cJt2JL0&algo_pvid=da58bd97-3739-4a01-8935-cb5282885e16&algo_exp_id=da58bd97-3739-4a01-8935-cb5282885e16-0&pdp_ext_f=%7B%22sku_id%22%3A%2212000020906639428%22%7D&pdp_npi=). [Último acceso: 5 Noviembre 2022].
- [52] Liberty Express, «tarifas de envío,» [En línea]. Available: <https://www.libertyexpress.com/es-ve/tarifas/>. [Último acceso: 5 Noviembre 2022].
- [53] AliExpress, «LOGO de SIEMENS 12/24RCE 6ED1052-1MD08-0BA1 reemplazo de 6ED1052-1MD08-0BA0 6ED1052-1FB08-0BA1,» [En línea]. Available: [https://www.aliexpress.us/item/2255799930410920.html?spm=a2g0o.productlist.0.0.676559f51tcGeF&algo\\_pvid=b592436e-5edb-4da6-a5d7-40d2bf9944b1&algo\\_exp\\_id=b592436e-5edb-4da6-a5d7-40d2bf9944b1-5&pdp\\_ext\\_f=%7B%22sku\\_id%22%3A%2212000028070550210%22%7D&pdp\\_npi=](https://www.aliexpress.us/item/2255799930410920.html?spm=a2g0o.productlist.0.0.676559f51tcGeF&algo_pvid=b592436e-5edb-4da6-a5d7-40d2bf9944b1&algo_exp_id=b592436e-5edb-4da6-a5d7-40d2bf9944b1-5&pdp_ext_f=%7B%22sku_id%22%3A%2212000028070550210%22%7D&pdp_npi=). [Último acceso: 5 Noviembre 2022].