

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Facultad de Ingeniería en Electricidad y Computación

Sistema de monitoreo de consumo de energía eléctrica en tiempo real de
bajo costo basado en internet de las cosas

PROYECTO INTEGRADOR

Previo la obtención del Título de:

Ingeniero en Electricidad

Presentado por:

Kember Stalyn Buenaventura Jiménez

Wilson David Villamar Tandazo

GUAYAQUIL - ECUADOR

Año: 2022

DEDICATORIA

Dedico este trabajo la comunidad académica, emprendedores y al sector productivo empresarial del país para que amplíen sus horizontes a la comunidad estudiantil en busca de proyecto de alto impacto, tecnológicos, revolucionarios para que no se queden solo dentro del ámbito académico, sino que estos proyectos tengan cabida en el ámbito comercial, demostrando que el país tiene un potencial que debe ser aprovechado.

Kember Stalyn Buenaventura Jiménez

Dedico este trabajo a nuestra humanidad que en solo un par de generaciones desde la revolución industrial fue capaz de transformar para bien y para mal nuestro mundo.

Estamos viviendo recién el inicio del tiempo de nuestra especie con muchas generaciones de seres humanos por venir. Y son estos trabajos e innovaciones en ciencia y tecnología que permiten mejorar la vida de las personas que viven ahora y vivirán después.

Wilson David Villamar Tandazo

AGRADECIMIENTOS

Agradezco ante todo a mi familia que ha estado para mí en todo momento, en especial a mi mamá, mi abuelo y abuela de parte de madre, quienes me han apoyado en cada uno de mis metas desde los concursos de matemáticas hasta darme el ánimo de asumir el reto estudiar dos carreras universitarias, cada logro que he conseguido ha sido por y para ellos, y doy gracias a Dios por darme la dicha de tenerlos presente en este momento especial en mi vida.

Kember Stalyn Buenaventura Jiménez

El 99.9% de los agradecimientos en esta parte va a los padres, hermanos y abuelitos y está bien, sin su apoyo y consejo no hubiera llegado a este punto en mi vida, pero una gran parte del proyecto se debió gracias a mi falta de dedicación en mi carrera y a mi extrema dedicación de aprender muchas otras cosas y unir diferentes tecnologías para solucionar problemas.

Wilson David Villamar Tandazo

Un agradecimiento especial al PhD. Miguel Torres, PhD Síxifo Falcones y al Club de Desarrollo Sostenible de ESPOL (CDS) por la retroalimentación brindada en pro del desarrollo del proyecto.

DECLARACIÓN EXPRESA

"Los derechos de titularidad y explotación, nos corresponde conforme al reglamento de propiedad intelectual de la institución; Kember Stalyn Buenaventura Jiménez y Wilson David Villamar Tandazo damos nuestro consentimiento para que la ESPOL realice la comunicación pública de la obra por cualquier medio con el fin de promover la consulta, difusión y uso público de la producción intelectual"



Firmado electrónicamente por:
KEMBER STALYN
BUENAVENTURA
JIMENEZ

Kember Stalyn
Buenaventura Jiménez

A handwritten signature in blue ink, appearing to read 'Wilson David Villamar Tandazo', positioned above a horizontal line.

Wilson David Villamar
Tandazo

EVALUADORES

.....
MSc. Iván Endara

PROFESOR DE LA MATERIA

.....
PhD. Miguel Torres

PROFESOR TUTOR

.....
MSc. Ángel Recalde

PROFESOR COTUTOR

RESUMEN

Una gran parte de la energía utilizada por los consumidores residenciales, institucionales e industriales no es utilizada eficientemente debido a malos hábitos de consumo, desperfecto en los equipos o fugas eléctricas. Actualmente existen equipos de medición y monitoreo energético pero que no se pueden usar en diferentes ámbitos de aplicación debido a su elevado costo, dificultad de instalación y operación, limitación en la visualización de datos en las interfaces propietarias y tecnología cerrada que impide su reparación, reemplazo y actualización de los equipos.

En este proyecto desarrollamos un sistema de monitoreo del consumo eléctrico que consta de un equipo físico y una plataforma web. Se probó el equipo de monitoreo con la medición de variables eléctricas de voltaje y corriente a través de sensores disponibles de forma local, se calculó el valor económico del consumo energético y se mostró la información a tiempo real en una pantalla OLED y una página web a través del protocolo de internet de las cosas MQTT en la placa electrónica ESP32. Los datos se guardaron en una tarjeta microSD para el análisis.

El diseño modurable del equipo, así como la incorporación de sensores e impresión 3D para la carcasa permite el uso del equipo de monitoreo en diferentes entornos de medición, además de su fácil actualización y reemplazo de componentes. El uso de la plataforma de monitoreo basada en NODE-RED y Mosquitto Broker está documentada y es de fácil modificación por parte del usuario.

Palabras Clave: medidor, consumo, ESP32, MQTT.

ABSTRACT

A large part of the energy used by residential, institutional, and industrial consumers is not used efficiently due to poor consumption habits, equipment failure or electrical leaks. Currently there are energy metering and monitoring equipment, but they cannot be used in different areas of application due to their high cost, difficulty in installation and operation, limitation in data visualization in proprietary interfaces and closed technology that prevents their repair, replacement and upgrading of equipment.

In this project we developed an electrical consumption monitoring system consisting of physical equipment and a web platform. The monitoring equipment was tested with the measurement of electrical variables of voltage and current through locally available sensors, the economic value of energy consumption was calculated, and the information was displayed in real time on an OLED screen and a web page through the Internet of Things MQTT protocol on the ESP32 electronic board. Data was saved to a microSD card for analysis.

The modular design of the equipment, as well as the incorporation of sensors and 3D printing for the package, allows the use of the monitoring equipment in different measurement environments in addition to its easy updating and replacement of components. The use of the monitoring platform based on NODE-RED, and Mosquitto Broker is documented and easy to modify by the user.

Keywords: meter, consumption, ESP32, MQTT.

ÍNDICE GENERAL

EVALUADORES.....	I
RESUMEN.....	II
ABSTRACT.....	III
ÍNDICE GENERAL.....	IV
ABREVIATURAS.....	VI
SIMBOLOGÍA.....	VII
ÍNDICE DE FIGURAS.....	VIII
ÍNDICE DE TABLAS.....	X
CAPÍTULO 1.....	1
1. Introducción.....	1
1.1 Descripción del problema.....	1
1.2 Justificación del problema.....	1
1.3 Objetivos.....	3
1.3.1 Objetivo General.....	3
1.3.2 Objetivos Específicos.....	3
1.4 Marco teórico.....	4
1.4.1 Monitoreo de Energía Eléctrica.....	4
1.4.2 Transformador de Corriente (TC).....	9
1.4.3 Transformador de Voltaje o Potencial (TP).....	11
1.4.4 Placas de Desarrollo.....	14
CAPÍTULO 2.....	18
2. Metodología.....	18
2.1 Materiales.....	18
2.2 Criterios de diseño.....	18
2.3 Placa de Desarrollo ESP32.....	19

2.4	Conexiones físicas del Equipo de Monitoreo	21
2.5	Entorno de Programación Visual Studio Core	30
2.6	Comunicación por Protocolo MQTT	31
2.7	Visualización en Dashboard mediante NODE-RED	34
2.8	Diagrama de funcionamiento del Equipo de Monitoreo	36
2.9	Pruebas de funcionamiento del Equipo de Monitoreo	37
2.9.1	Prueba de funcionamiento del equipo de monitoreo eléctrico.....	37
2.9.2	Prueba de monitoreo Continuo	39
CAPÍTULO 3.....		40
3.	Resultados Y ANÁLISIS	40
3.1	Resultados de la prueba de funcionamiento.....	40
3.2	Resultado de la prueba de monitoreo.....	41
3.3	Análisis de Costos	47
3.4	Viabilidad Económica	48
CAPÍTULO 4.....		49
4.	Conclusiones Y Recomendaciones	49
4.1	Conclusiones.....	49
4.2	Recomendaciones.....	52
BIBLIOGRAFÍA.....		53
5.	Trabajos citados	53
APÉNDICES.....		55

ABREVIATURAS

ESPOL	Escuela Superior Politécnica del Litoral
ADC	Analog to Digital Converter - Convertidor analógico digital
TC	Transformador de corriente
TP	Transformador de Voltaje o Potencial
PCB	Printed circuit board - Placa de circuito impreso
MQTT	Message Queue Telemetry Transport
SoC	System on Chip
SoM	Module on Chip
PLA	Ácido Poli láctico
API	Application Programming Interface

SIMBOLOGÍA

A	Amperio
V	Voltaje
Ω	Resistencia eléctrica
CO ₂	Dióxido de carbono
V _{ADC}	Valor máximo de voltaje aceptado por los pines ADC del ESP32
V _{rms}	Valor eficaz de Voltaje
I _{rms}	Valor eficaz de Corriente
P _{inst}	Potencia Instantánea
VAC	Voltaje en Corriente Alterna
DC	Corriente Directa

ÍNDICE DE FIGURAS

Figura 1.1 Enfoque Holístico de eficiencia energética	4
Figura 1.2 Smapee Infinity dispositivos conectados	7
Figura 1.3 Eastron Europe SDM120M Series.....	7
Figura 1.4 Registrador de energía Fluke 1732 y 1734	8
Figura 1.5 Dispositivo Gen 2 Vue Energy Motor.....	8
Figura 1.6 Diagrama esquemático para un transformador de corriente	9
Figura 1.7 Transformador de Corriente ST 013-100.....	11
Figura 1.8 Sensor de Voltaje ZMPT101B	12
Figura 1.9 PINOUT del sensor de Voltaje ZMPT101B	12
Figura 1.10 Diagrama esquemático del módulo ZMPT101B	13
Figura 1.11 Placa Arduino Uno.....	15
Figura 1.12 Placa Raspberry Pi 4.....	16
Figura 1.13 Placa ESP32	16
Figura 2.1 Diagrama de bloques funcional de la placa ESP 32.....	19
Figura 2.2 Elementos que conforman la placa ESP32	20
Figura 2.3 Diagrama del PINOUT del ESP32 38 pines ESP WROOM 32.....	20
Figura 2.4 Señal acondicionada en los límites máximos y mínimos del pin ADC.....	24
Figura 2.5 Comportamiento no lineal de los pines ADC del ESP32	24
Figura 2.6 Pantalla OLED Modelo SSD1306.....	25
Figura 2.7 Módulo Micro SD	26
Figura 2.8 Conexiones físicas del equipo en el Protoboard.....	26
Figura 2.9 Modelamiento 3D del prototipo final	27
Figura 2.10 Modelamiento 3D de los componentes del dispositivo	27
Figura 2.11 Implementación física del dispositivo de monitoreo.....	28
Figura 2.12 Identificación de las partes físicas del dispositivo de monitoreo.....	28
Figura 2.13 Presentación final del dispositivo.....	29
Figura 2.14 Instalación y conexión del dispositivo físico.....	29
Figura 2.15 Instalación de la Plataforma IO en Visual Studio Code	30
Figura 2.16 Esquema de comunicación MQTT	33
Figura 2.17 Instalación del Bróker Mosquitto.....	33
Figura 2.18 Instalación de NODE-RED	34

Figura 2.19 Visualización del Dashboard en NODE-RED	35
Figura 2.20 Diagrama de flujo del funcionamiento del Equipo.....	36
Figura 2.21 Conexión del equipo durante la prueba de funcionamiento.....	37
Figura 2.22 Pinza amperimétrica EXTECH 380942	38
Figura 2.23 Multímetro marca Fluke 1587	38
Figura 2.24 Equipo de medición durante la prueba en el laboratorio	38
Figura 2.25 Impresora 3D utilizada como objeto de prueba	39
Figura 2.26 Equipo de monitoreo midiendo el consumo de la impresora 3D.....	39
Figura 3.1 Inicio de la prueba de monitoreo	41
Figura 3.2 Dashboard al finalizar la primera hora de monitoreo	42
Figura 3.3 Dashboard luego de tres horas de monitoreo.....	42
Figura 3.4 Dashboard al finalizar la prueba de monitoreo	43
Figura 3.5 Curva del Voltaje RMS durante el monitoreo.....	43
Figura 3.6 Curva del Corriente RMS durante el monitoreo	44
Figura 3.7 Curva de la Potencia Instantánea en Watts durante el monitoreo.....	45
Figura 3.8 Curva del Energía Acumulada en kWh durante el monitoreo	45
Figura 3.9 Valor en dólares del consumo energético durante el monitoreo	46
Figura 3.10 Logo del dispositivo de monitoreo	49
Figura 3.11 Ejes claves que distinguen a POWERTEC	49

ÍNDICE DE TABLAS

Tabla 2.1 Rango de entrada y salida para los distintos TC- SCT13.....	22
Tabla 2.2 Resistencia de burden en función de la corriente máxima	23
Tabla 3.1 Porcentaje de error del voltaje	40
Tabla 3.2 Porcentaje de error de la corriente	40
Tabla 3.3 Costo de fabricación del dispositivo de monitoreo.....	47
Tabla 3.4 Costo total de los TC colocados en el dispositivo de monitoreo	47

CAPÍTULO 1

1. INTRODUCCIÓN

Los equipos de monitoreo energético son dispositivos que miden voltaje y corriente dentro de casas y edificios con la finalidad de determinar el consumo tanto en potencia y energía de dichas construcciones. En los últimos años, estos dispositivos han cobrado especial importancia por el auge y concientización en temas de eficiencia energética y ahorro económico, ya que la caracterización del consumo permite identificar mal funcionamiento de equipos, desbalance de cargas, fallas y hurtos de electricidad, así como posibles mejoras en los hábitos de consumo, la infraestructura o los mismos equipos.

En la actualidad, la medición de la energía eléctrica se realiza mediante sensores especializados de voltaje y corriente que se conectan a placas electrónicas programables, las cuales, por medio de un código, realizan cálculos eléctricos que luego se muestran en pantallas o interfaces para la visualización del usuario y toma de decisiones.

1.1 Descripción del problema

Una gran parte de la energía eléctrica consumida por los clientes residenciales, comerciales e industriales no es utilizada eficientemente, lo que conlleva altos costos en la facturación de electricidad, baja competitividad y un mayor impacto ambiental debido a la imposibilidad de identificar fugas eléctricas, cortocircuitos y condiciones anormales en el funcionamiento de electrodomésticos y equipos que conllevan a una reducción en su vida útil. De allí, surge la necesidad de monitorear el consumo energético con el fin de minimizar el impacto ambiental, económico y social, promoviendo su uso óptimo de manera responsable y sostenible.

1.2 Justificación del problema

Actualmente existen equipos de medición y monitoreo energético que se utilizan para aplicaciones específicas, algunos de ellos se utilizan únicamente para el sector residencial con rangos de medición de corriente y voltaje estandarizados a este tipo de usuario, mientras que para usuarios comerciales existen medidores de mayor

capacidad, con interfaces gráficas que permiten la visualización de los datos eléctricos, y por último los medidores de energía industriales tienen características más avanzadas como la detección de transitorios, cálculos de armónicos, calidad de energía y eficiencia de determinados equipos industriales, estos medidores están pensados principalmente para el almacenamiento de los datos y su posterior análisis.

La segmentación de los medidores para diferentes aplicaciones además del conocimiento limitado del usuario de los procesos constructivos y códigos de programación sumado al soporte existente o no existente del fabricante impiden a los diferentes clientes instalar y operar estos dispositivos adecuadamente.

El software que vienen en los equipos de medición en muchos casos resulta incompatible con sistemas de gestión de edificios y sistemas de automatización industrial, esto conlleva al reemplazo de dispositivos o al cambio de protocolos o software ya utilizados. Las interfaces propietarias de los equipos de medición restringen la visualización a las variables o magnitudes definidas por el fabricante y no aquellas de sumo interés de cada usuario en particular, tampoco se permite su conexión con plataformas externas de notificaciones, análisis y almacenamiento de las mediciones.

La tecnología cerrada impide además la reparación, reemplazo y actualización de componentes de los equipos, ya que se ven limitados a la misma marca con sus rangos máximos de medición, de precisión y características que pueden no ser suficientes para entornos que requieren especificaciones determinadas.

Tecnologías emergentes como el Internet de las Cosas, que permiten la conexión de dispositivos e intercambio de datos a través de internet, el abaratamiento de sensores, las múltiples plataformas de procesamiento y la tendencia de generación distribuida, requieren que la próxima generación de medidores de energía tengan la capacidad de adaptarse de manera modular a las necesidades de los usuarios en los diferentes entornos de monitoreo.

1.3 Objetivos

1.3.1 Objetivo General

Construir un equipo de monitoreo de consumo de energía eléctrica mediante el módulo ESP32 y sensores compatibles para recolección y visualización de datos en una interfaz web.

1.3.2 Objetivos Específicos

- Identificar los rangos de operación de los transductores de corriente, sensores y el módulo ESP32 en función de las condiciones y el entorno de trabajo para el dimensionamiento del equipo de medición de energía eléctrica.
- Programar el procesamiento y envío de los datos recogidos por los sensores, así como el cálculo de variables eléctricas para la visualización de la información útil en eficiencia energética.
- Ajustar los parámetros de calibración de hardware y software necesarios en el equipo de monitoreo para la recolección correcta de datos de los sensores.
- Evaluar la recolección, procesamiento y visualización adecuada de las variables eléctricas mediante el uso de cargas de prueba.

1.4 Marco teórico

A continuación, se describirán los fundamentos teóricos de los elementos y conceptos necesarios para la diseño y construcción del equipo de medición de energía.

1.4.1 Monitoreo de Energía Eléctrica

La eficiencia energética ha cobrado mayor importancia en los últimos años debido al aumento de los precios de energía y una mayor consciencia del cambio climático. Incrementar la eficiencia energética de casas y edificios por medio de prácticas de sostenibilidad es importante para disminuir el impacto económico y ambiental del mal aprovechamiento de los recursos energéticos debido a que estos representan un porcentaje considerable del consumo eléctrico total y de emisiones de CO₂ [1].

A continuación, se presenta el estado de la modernización energética, la cual junto con sus desafíos asociados y requisitos de rendimiento, se propone por medio de una estrategia holística de 4 principios para lograr la eficiencia energética en edificios [2]:

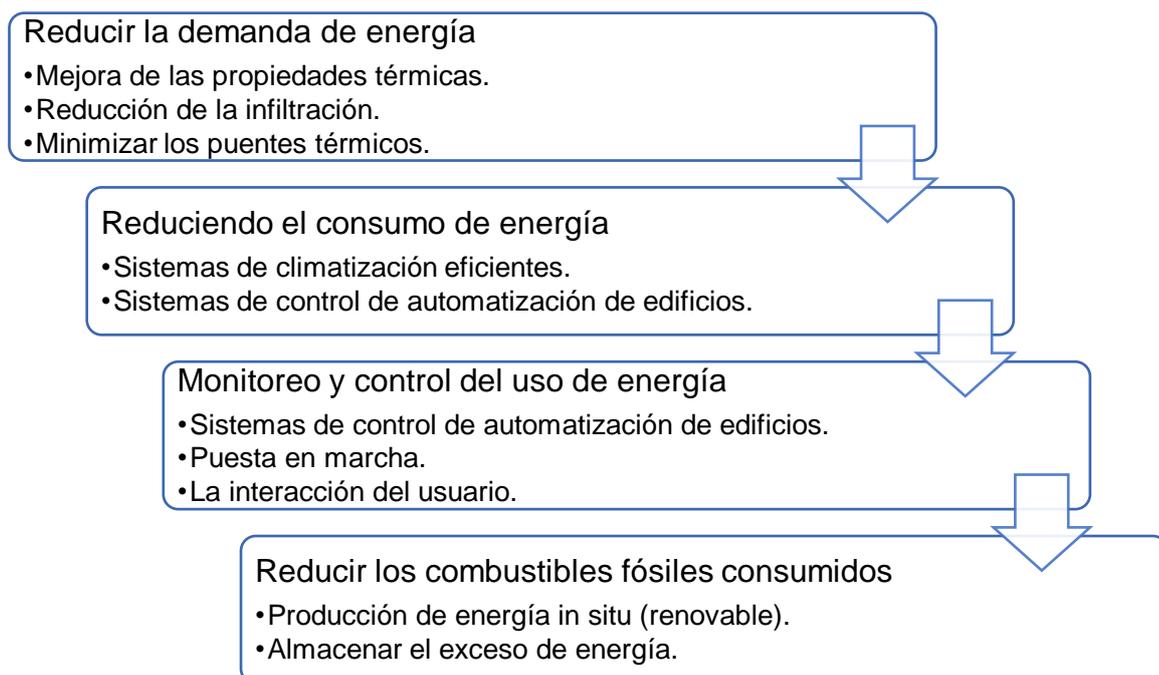


Figura 1.1 Enfoque Holístico de eficiencia energética

Dentro de esta estrategia se incluye la reducción del consumo de energía y el monitoreo del uso de energía; en el caso de la electricidad la reducción del consumo conlleva a la disminución del precio de la factura eléctrica y un menor uso de unidades generadoras menos eficientes y de alto costo basadas generalmente en combustibles fósiles.

Por su parte, el monitoreo es importante debido a que, en la actualidad la creciente demanda energética implica un mayor uso de los recursos renovables y no renovables [3], por lo que para alcanzar un desarrollo sustentable y sostenible no basta con el desarrollo de nuevas energías verdes o denominadas eco-amigables, sino que se debe crear una conciencia colectiva de uso eficiente y responsable de la energía eléctrica.

Esta demanda de electricidad ha ido en aumento con el pasar de los años debido al desarrollo de las ciudades, el aumento poblacional, el uso de más dispositivos electrónicos, entre otros, por lo que ante esta creciente demanda es necesario llevar un control del consumo energético, para monitorear en caso de mal uso, equipos electrodomésticos antiguos, alguna fuga no detectada, o cualquier otra razón que resulte en un consumo atípico, y con esto tomar los correctivos necesarios.

En el mercado existen diversos equipos de medición de energía los cuales pueden ser instalados generalmente en el panel de distribución si es que se desea monitorear el consumo energético de los circuitos principales de una instalación.

Los dispositivos de monitoreo de energía permiten obtener el consumo de energía en un punto determinado en el cual se esté realizando la medición. Dependiendo las aplicaciones y el alcance que este equipo se pueden obtener los valores de las distintas variables eléctricas como corriente, voltaje, potencia, energía, y para equipos más avanzados es posible la detección de armónicos.

Cada vez existe una mayor oferta de equipos de monitoreo de consumo eléctrico que en ciertos modelos resultan ser invasivos, ya que requieren la desconexión o modificación de los enchufes o paneles instalados, o por temas de costo omiten la medición de voltaje tomando un valor fijo, esto conlleva a errores de medición dado que el valor de voltaje al igual que la corriente fluctúa en función de la carga total y las unidades de generación conectadas al sistema.

Estos equipos de monitoreo muestran las variables eléctricas comunes: V_{rms} , I_{rms} , potencia instantánea, factor de potencia y energía, en las pantallas instaladas en el dispositivo o a través de una interfaz accesible mediante internet o mediante la conexión de un puerto de comunicación, en muchos casos las mediciones se pueden descargar para análisis posterior, así como toma de decisiones.

Algunos modelos más avanzados incluso permiten etiquetar las mediciones por dispositivo y ubicación, así como emitir alertas personalizadas cuando se alcanza un consumo máximo en kWh o se llega a un tiempo de encendido definido por el usuario.

Los equipos de medición miden los valores instantáneos, tanto de corriente y voltaje, por medio de los transformadores de corriente y de potencial, y luego a través de un microprocesador las convierte en valores eficaces.

Dado que la electricidad es suministrada por medio de corriente alterna, el significado de valor eficaz cobra importancia para el registro y monitoreo de energía. El valor eficaz de tensión es una medida de la eficacia que tiene la fuente al suministrar la potencia a una carga resistiva; por su parte para la corriente se definió que su valor eficaz resulta igual al valor de la corriente directa que fluye a través de una resistencia entregando la misma potencia promedio a esta resistencia como si esta fuera la corriente periódica [4]. Una vez definidos los valores eficaces tanto de corriente como de voltaje

es posible calcular la potencia promedio, y son estos valores los que arrojan los distintos equipos de medición que existen en la actualidad.

Entre los distintos dispositivos de monitoreo que se ofrecen en el mercado tenemos:

- Smapee Infinity es un dispositivo que ofrece un sistema de gestión energética que permite llevar un control de electricidad, que permite el manejo de información histórica y en tiempo real por medio de su interfaz de aplicación basada en la nube [5].



Figura 1.2 Smapee Infinity dispositivos conectados

- Eastron SDM120 es un medidor monofásico de corriente, voltaje y factor, además de medir energía y potencia de manera bidireccional de manera que se puede obtener los valores de energía importada, exportada y neto. Su capacidad de medición es de máximo 45 A AC [6].



Figura 1.3 Eastron Europe SDM120M Series

- Fluke 1732 y 1734 son registradores de energía eléctrica integrales, mide y registra los parámetros de tensión, corriente, y energía, incluso también permite obtener factor de potencia, cuenta con un software de aplicación Energy Analyze Plus, lo cual permite llevar el registro del consumo energético [7].



Figura 1.4 Registrador de energía Fluke 1732 y 1734

- Gen 2 Vue Energy Motor fabricado por la empresa Emporia Corp. es un medidor de energía que por medio de dos sensores de corriente miden la energía consumida en el panel de distribución de una casa y en caso de mediciones a circuitos individuales cuenta con un aproximado de hasta 16 sensores de 50 A para realizar la medición, el dispositivo se conecta a internet y envía los datos a una aplicación de Android, desde donde se puede visualizar los datos [8].



Figura 1.5 Dispositivo Gen 2 Vue Energy Motor

1.4.2 Transformador de Corriente (TC)

Para poder medir la corriente de diferentes equipos es necesario el uso de dispositivos de medición no invasivos para poder obtener tanto los datos de corriente y no tener que realizar conexiones directas en el circuito de interés. Para esto se emplean TC del tipo toroidal con la finalidad de que estos puedan ser fácilmente insertados en cada uno de los circuitos de la instalación y obtener las mediciones de la corriente de manera más fácil y práctica.

De manera general, el diagrama equivalente para un transformador tiene varias componentes entre los elementos que se describen en la Figura 1.6 [9] a continuación, está la impedancia de magnetización de excitación Z_e , la cual consta de dos partes Z'_e asociada con el flujo de fuga dentro del núcleo del transformador relacionada con la reactancia de dispersión X , y Z''_e asociada con el flujo que no alcanza a llegar al núcleo. Por otra parte, tenemos X_p que representa la reactancia de fuga del flujo que no circula por el núcleo del transformador y finalmente tenemos las resistencias R_p y R_s que representan tanto la resistencia del devanado primario y secundario respectivamente [9].

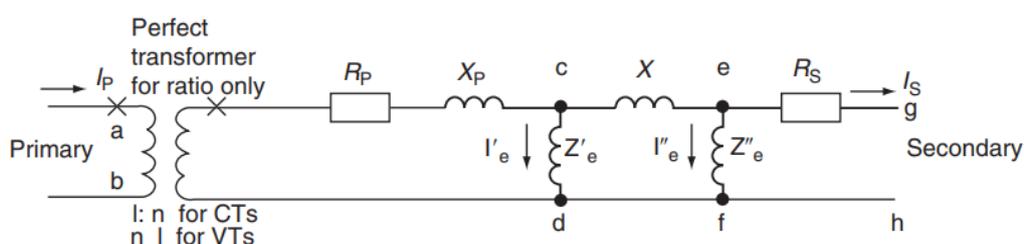


Figura 1.6 Diagrama esquemático para un transformador de corriente

Los transformadores de corriente se diferencian de los transformadores de voltaje a nivel constructivo: el núcleo se compone generalmente de laminaciones metálicas de silicio acero que soporta las densidades de flujo altas, esto es necesario para controlar la saturación del núcleo y en consecuencia la precisión y rangos de las mediciones de corriente. El instrumento de medición de la corriente del devanado secundario está en

serie, el devanado primario tiene un pequeño número de vueltas mientras que el devanado secundario tiene un gran número de vueltas y no puede estar en circuito abierto. Estas características hacen que el TC tengan una alta relación de transformación y una baja impedancia que permita el paso de corriente en el lado secundario.

Los transformadores de núcleo toroidal poseen algunas ventajas que lo vuelve más atractivo en comparación a otro tipo de TC, esto se debe principalmente a su bajo costo al aislamiento galvánico, la simplicidad de su aplicación y la confiabilidad que se entrega. No obstante, este equipo también presenta ciertas desventajas que se deben a la saturación del equipo y al flujo remanente.

Hay que tener en consideración que los TC se clasifican según su funcionamiento, el cual se subdivide en dos tipos, el primero es para uso de medición de corriente de operación, y el siguiente tipo es para medir corrientes de fallas con el fin de suministrar la señal de control a los relés de protección del sistema eléctrico de potencia para su posterior actuación para el despeje de la falla. De esta forma, dependiendo el enfoque y objetivo del proyecto se tiene que escoger el TC más adecuado [10].

Para el caso de este proyecto se escogió un TC de núcleo partido modelo SCT 013-100 [11] , este equipo tiene la relación de transformación de 100 amperios en el lado del primario a 50 miliamperios DC en el lado del secundario [12].

Se escogió este equipo debido a su bajo costo fácil instalación para la medición de corriente y el rango de mediciones que es posible tomar por medio de este equipo, lo cual lo vuelve viable para el equipo de medición.



Figura 1.7 Transformador de Corriente ST 013-100

1.4.3 Transformador de Voltaje o Potencial (TP)

Los transformadores de voltaje proveen aislamiento de los valores primarios de voltaje de gran magnitud y los reduce a valores secundarios que pueden ser leídos por microprocesadores utilizados por lo general en los relés de medición y protección que se utilizan en los sistemas de transmisión y distribución. En esencia el funcionamiento del equipo para ambos casos tiene la misma característica, lo que cambia es el fin con el cual se van a tratar las mediciones.

El diagrama equivalente de los transformadores de voltaje es similar al diagrama esquemático de los TC de la Figura 1.6, la diferencia radica en los aspectos constructivos: El núcleo está hecho de acero de alta calidad que funciona con bajas densidades de flujo magnético, se conecta en paralelo al instrumento, posee un mayor número de vueltas en el devanado primario y en el devanado secundario posee un menor número de vueltas que además se encuentra en circuito abierto, estas condiciones hace que el transformador de voltaje sea de baja relación de transformación y de alta impedancia para impedir así el paso de corriente entre los devanados.

El sensor que se utilizara en el equipo de medición es el módulo transformador de voltaje AC [13] de la empresa Qingxian Zeming Langxi Electronic, es un módulo popular y de precio accesible que mide hasta 250 VAC.

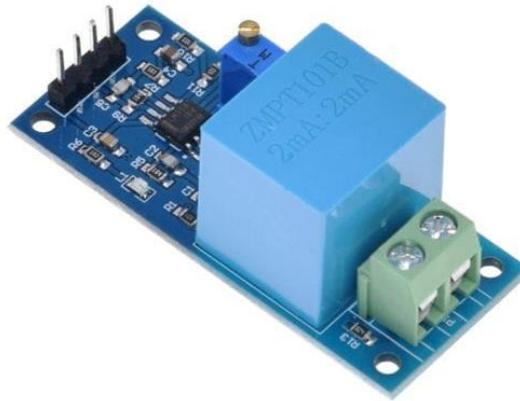


Figura 1.8 Sensor de Voltaje ZMPT101B

El módulo de la Figura 1.8 recibe un voltaje de entrada entre 0 a 250 VAC, obteniendo un voltaje de salida de 0 a 5 VAC, para su funcionamiento este debe ser alimentado con un voltaje DC de 5 V a 30 V.



Figura 1.9 PINOUT del sensor de Voltaje ZMPT101B

Adicionalmente, el error de fase de ángulo es menor igual a 20 minutos, el fabricante ofrece un error máximo entre -0.3% y 0.2% de la medida real de voltaje [13].

En la siguiente Figura 1.10 se puede observar el esquemático del módulo ZMPT101B conectado al transformador de voltaje con el fin de reducir el voltaje AC de entrada a un voltaje menor para poder ser leído por el microcontrolador. El módulo está compuesto por el TP cuya función es de aislamiento galvánico para una mayor seguridad de este módulo de manera que en el lado primario se conecta el voltaje de la red eléctrica de la instalación a la cual se quiera realizar la medición y en el lado secundario existe un divisor de tensión, un filtro de paso alto basado en el OPAMP LM358 y aumento de una señal offset de $VCC/2$ donde VCC es el valor de alimentación del módulo (5V o 3.3V). El objetivo de desplazar positivamente la onda es para que no tenga valores negativos y de esta forma la señal pueda ser leída promedio del convertidor analógico digital (ADC) del microcontrolador [15].

Este módulo fue escogido para el proyecto debido a su facilidad en el uso, además de ser viable para el monitoreo de energía eléctrica, ya que puede ser conectado a distintos microcontroladores y mandar los datos a la nube para su posterior análisis.

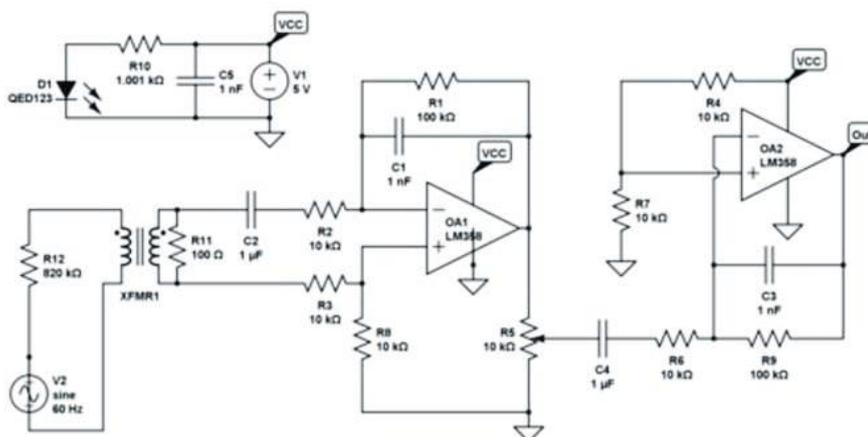


Figura 1.10 Diagrama esquemático del módulo ZMPT101B

1.4.4 Placas de Desarrollo

Las placas de desarrollo programables permiten procesar las señales de entrada provenientes de sensores y enviarlo a dispositivos de salida según instrucciones escritas en un lenguaje de programación en dichas placas.

Actualmente existen una multitud de placas electrónicas que se utilizan desde el aprendizaje, proyectos domésticos hasta productos industriales. Las placas de desarrollo poseen un microcontrolador programable, el cual maneja de uno de los múltiples lenguajes para sistemas electrónicos como C, C++, C#, Arduino, Micro Python, etc., además del uso de uno de muchos entornos de desarrollo de programación disponibles que permiten la escritura de código de programación en uno o varios lenguajes.

El microcontrolador está conectado a pines generales de entrada y salida, y a través del código de programación se define si dicho pin recibe señales de entrada de un sensor o en cambio envía señales de salidas a un actuador. Esta capacidad de modificación permite su uso con una gran cantidad de periféricos que van desde focos LED, motores, servomotores, relés, sensores temperatura, presión, humedad, cámaras, módulos Wifi, Bluetooth y más.

Existen además pines de entrada analógicos que permiten la lectura de señales analógicas de voltaje, que a su vez son equivalentes a señales analógicas reales de sensores; este tipo de pines tiene rangos de funcionamiento y una linealidad de conversión de señales analógicas. Las placas de desarrollo poseen además sistemas secundarios para la alimentación de energía y programación a través de puertos Jack o USB, reguladores de voltaje, indicadores LED, botones de encendido, cristal oscilador como señal de reloj estable para circuitos integrados, entre otros componentes.

Otras de las ventajas de las placas de desarrollo, además de la gran cantidad de periféricos a los que se pueden conectar, radica en su pequeño tamaño, ideal para uso de electrónica en movimiento. Además la variedad de placas electrónicas en el mercado permite el desarrollo de diversos proyectos de automatización, robótica o en este caso equipos de medición.

Algunas de las placas de desarrollo populares son Arduino Uno, Raspberry Pi 4, ESP32.

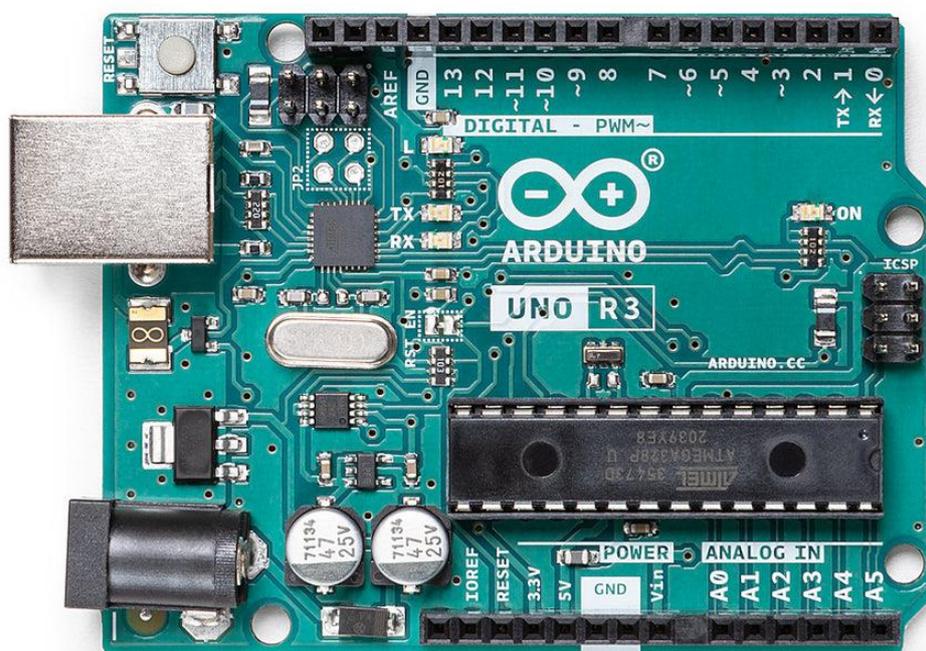


Figura 1.11 Placa Arduino Uno

La placa de Arduino UNO es una opción popular para iniciar en el mundo de la programación y electrónica, tiene 14 pines generales de entrada y salida GPIO, 6 entradas analógicas, conexión USB, ICSP y botón de reinicio. Posee el microprocesador ATmega328P, es ideal es para aplicaciones pequeñas que no requieren una gran capacidad de procesamiento, no posee conexiones Wifi ni Bluetooth por lo que para la comunicación se requiere módulos externos, el tamaño es grande en comparación a otras placas de desarrollo [16]

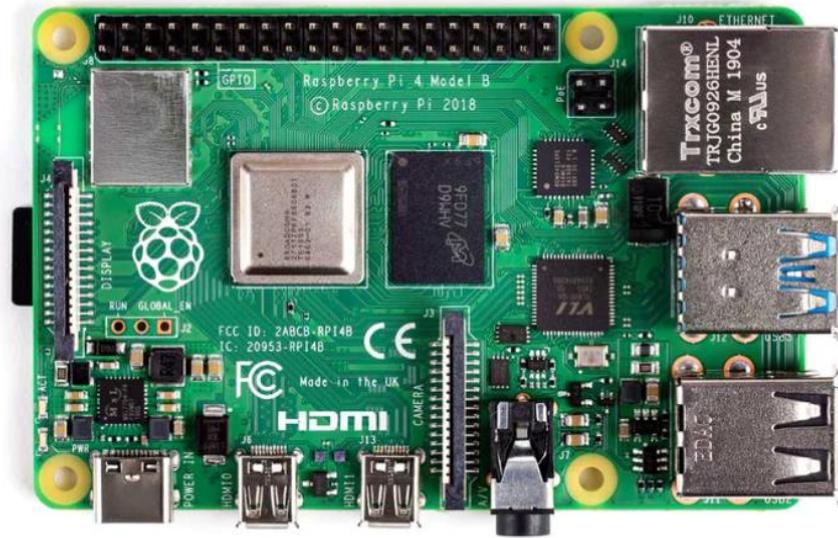


Figura 1.12 Placa Raspberry Pi 4

La placa de desarrollo Raspberry Pi 4 se acerca mucho más a una computadora de escritorio de bajo costo, posee comunicación Ethernet, Bluetooth 5.0, Wifi, opciones de RAM de 1GB, 2GB, 4GB, 8 GB. Posee un sistema operativo completo basado en Linux y su capacidad le permite aplicaciones para el control de robots, procesamiento de inteligencia artificial, centro de control de hogares y mucho más [17].

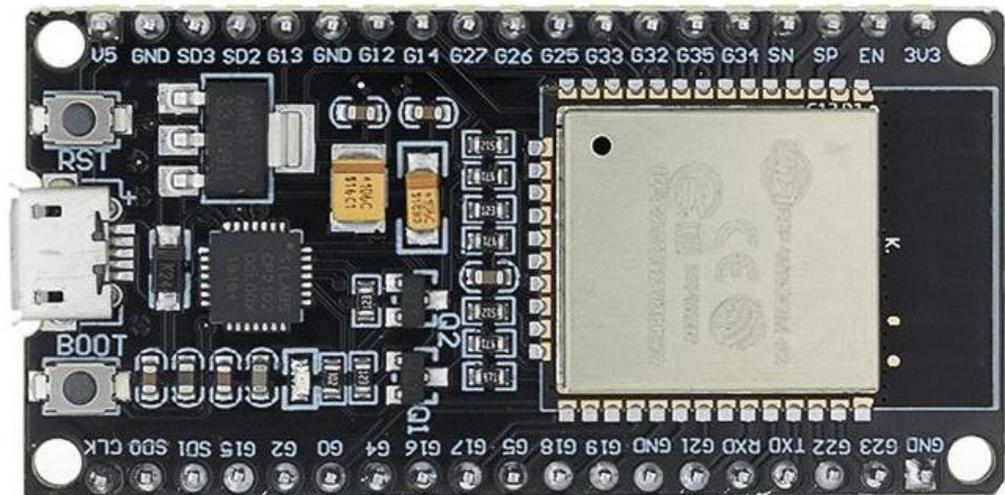


Figura 1.13 Placa ESP32

La placa de desarrollo ESP32 es de bajo costo y de tamaño muy pequeño ideal para aplicaciones de electrónica en movimiento, posee en sus distintas versiones hasta 24 pines generales de entrada y salida, 18 pines analógicos, además de conexión inalámbrica Wifi y Bluetooth. El microprocesador es de doble núcleo permitiendo así hacer ejecutar comandos en modo de ultra bajo consumo. La programación de la placa electrónica se puede realizar en Arduino, Micro Python, LUA, Esp-idf, Python, RTO's, JavaScript y Basic [18].

CAPÍTULO 2

2. METODOLOGÍA

En este capítulo se describe los elementos necesarios para la construcción del equipo de monitoreo, así como sus conexiones físicas y protocolo de comunicación.

2.1 Materiales

A continuación, se da una lista de los materiales que se utilizaron para la construcción del equipo de monitoreo:

- Placa ESP32
- Transformador de corriente SCT 013-100
- Sensor de Voltaje ZMPT101B
- Pantalla OLED modelo SSD1306
- Módulo de tarjeta Micro SD
- Potenciómetro trimmer
- Resistencias de 10k
- Capacitor
- Puertos Jack
- Insumos: cables

2.2 Criterios de diseño

Para el diseño del equipo se tomó en cuenta criterios, tales como:

- Modulable, permitiendo el intercambio de piezas para una mayor durabilidad o para mejorar las capacidades según el ámbito de aplicación.
- Escalable, que el equipo sea capaz de medir en instalaciones de diversos tipos en diferentes ámbitos, ya sea un hogar, una fábrica, un sistema de paneles solares, entre otros.
- Económico, el costo del equipo debe ser menor al de otros equipos que ofrezcan servicios similares en el mercado

2.3 Placa de Desarrollo ESP32

La placa de desarrollo seleccionada para la lectura de los sensores, el procesamiento y la visualización de estos es la placa electrónica NODEMCU ESP32 debido a su integración, robustez y precio para aplicaciones de Internet de las Cosas.

Cuenta con el Sistema en Chip Soc. ESP32 de la empresa Espressif Systems, este posee un procesador Tensilica Xtensa de 32-bit LX6, 2 núcleos de procesamiento, frecuencia del reloj hasta 240 MHz, interfaces SPI, I2C, UART, entre otras para conexión con periféricos como pantallas OLED y LCD, además de conectividad Wifi y Bluetooth [19].

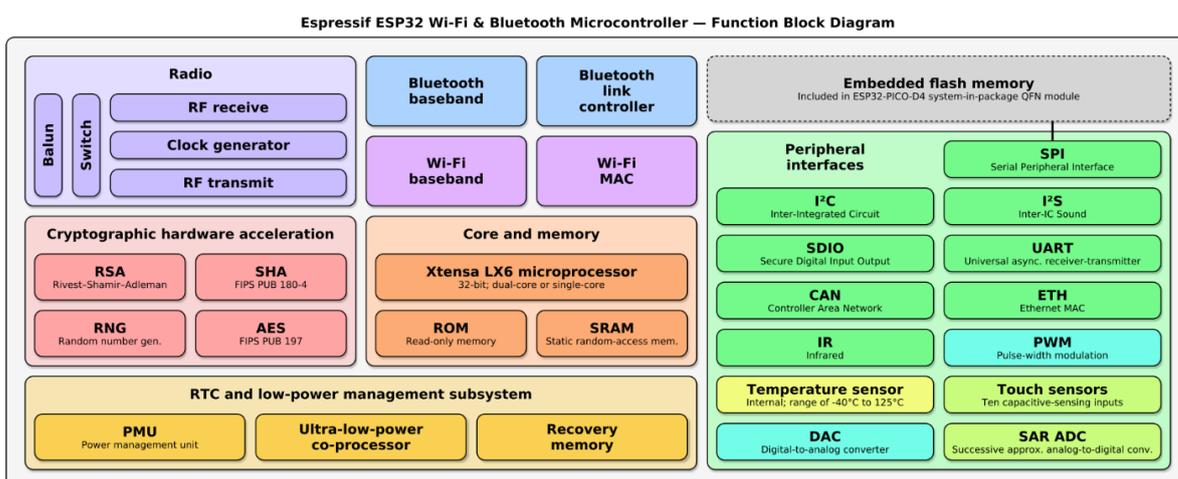


Figura 2.1 Diagrama de bloques funcional de la placa ESP 32

El Soc. viene incluido en el Módulo en Chip SoM ESP-WROOM-32 que integra una memoria flash de 4 MB, cristal oscilador, pines digitales, pines conversores analógicos a digital, ADC (Analog to Digital Converter) y una antena Wifi.

La placa de desarrollo NODEMCU incluye al SoM ESP-WROOM-32 y añade otras características como energización a través de micro-USB, regulador de voltaje, conversor USB-serial CP2102 para programación por micro-USB, leds indicativos y puertos para montaje en protoboard y uso con cables Dupont.



Figura 2.2 Elementos que conforman la placa ESP32

Los puertos de la placa NODEMCU-ESP32 tienen diferentes funciones asignadas:

- Los pines 1, 14, 19, 32 y 38 permiten la energización de periféricos.
- Los pines ADC miden voltajes analógicos provenientes de sensores.
- Los pines 33 y 36 permite comunicar dispositivos como pantallas OLED.

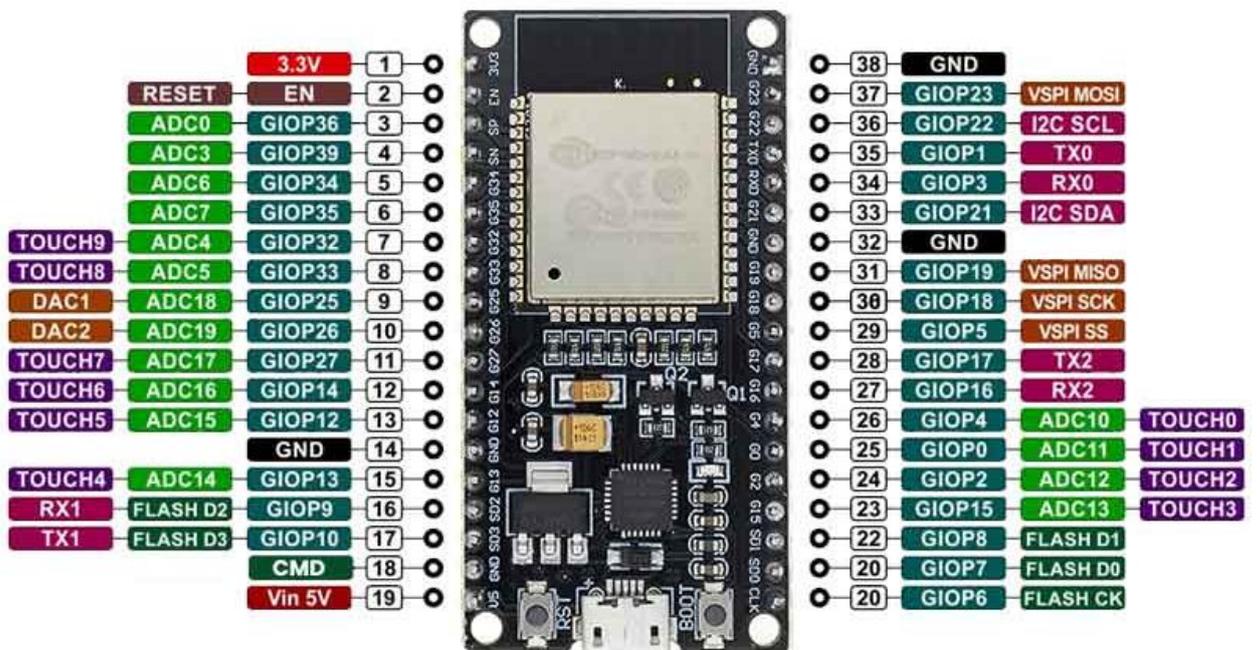


Figura 2.3 Diagrama del PINOUT del ESP32 38 pines ESP WROOM 32

El diseño robusto está pensado para usos industriales entre -40°C y 125°C , el ultra bajo consumo apoyado por el coprocesador de ultra baja energía permite su aplicación dentro de dispositivos móviles y electrónica portátil.

Al ser una placa de desarrollo popular existe suficiente documentación y oferta de manera local para el reemplazo o actualización como unidad de lectura de datos, procesamiento y comunicación. La razón de usar la placa de 38 pines y no las placas de 30 o 36 pines es porque permite una mayor cantidad de pines ADC necesarios para medir los valores de voltaje y corriente, además que esta versión es compatible con las protoboard comunes en el proceso de prototipado, las otras versiones no se pueden conectar a una sola protoboard, ya que no son de las dimensiones para calzar en una protoboard.

2.4 Conexiones físicas del Equipo de Monitoreo

Para las conexiones se requiere energizar todos los tipos de dispositivos con los pines 3.3V y GND, a excepción del TC, este último puede requerir un potenciómetro de escalamiento, además de un offset DC compuesto de un arreglo de resistencias y un capacitor.

Existen dos tipos de TC que se pueden utilizar para la medición de la corriente, el primero es aquellos cuya salida de esta dada por una señal de corriente que va de 0 a 50 mA y otro en el cual la salida viene dada por una señal de 0 a 1 V a diferentes rangos de entrada de corriente. El detalle de los diferentes modelos de TC se describen en la Tabla 2.1, en donde los valores de corriente son descritos en términos de su valor eficaz

Como la corriente puede ir en ambos sentidos, los rangos de la señal de salida son -50 mA a $+50\text{ mA}$ y -1V a 1V para los dos tipos de TC. Dado a que los pines ADC de la placa solo aceptan valores positivos de tensión, para ambos casos se debe acondicionar la señal para que puedan ser conectados a los pines sin que se vaya a afectar los datos a medir ni afectar a la placa de desarrollo, el offset DC permite acondicionar la señal de entrada del TC a valores positivos dentro del rango analógico de voltaje que aceptan las entradas analógicas de la placa de desarrollo.

Tabla 2.1 Rango de entrada y salida para los distintos TC- SCT13

Modelo	SCT013-100 A	SCT013-005	SCT013-010	SCT013-015	SCT013-020
Rango	0A – 100 A	0A – 5 ^a	0A – 10A	0A – 15A	0A – 20A
Salida	Corriente	Voltaje	Voltaje	Voltaje	Voltaje
	0 mA – 50 mA	0V – 1V	0V – 1V	0V – 1V	0V – 1V
Modelo	SCT013-025	SCT013-030	SCT013-050	SCT013-060	SCT013-100V
Rango	0A – 25 A	0A – 30A	0A – 50A	0A – 60A	0A – 100A
Salida	Voltaje	Voltaje	Voltaje	Voltaje	Voltaje
	0V – 1V	0V – 1V	0V – 1V	0V – 1V	0V – 1V

Para el caso de los TC, cuya señal de salida sea en función de la corriente es necesario que esta sea conectada a una resistencia de burden con la finalidad de que ahora la señal de salida esté en función del voltaje, no obstante al ser un elemento puramente resistivo la señal mantiene la onda sinusoidal tomando valores instantáneos tanto positivos como negativos, por esto igual sigue siendo necesario emplear el circuito adicional de offset DC a la señal de manera que los valores se mantengan dentro del rango de 0 a 3.3 V aceptados por los pines ADC.

La resistencia de burden se dimensiona para los valores máximos de corriente Irms que se desea medir y está dada por las siguientes fórmulas:

$$I_{sec_{RMS}} = I_{prim_{RMS}} \cdot N \quad (2.1)$$

$$I_{sec_{pico}} = I_{sec_{RMS}} \cdot \sqrt{2} \quad (2.2)$$

$$V_{ref} = \frac{V_{ADC}}{2} \quad (2.3)$$

$$R_{burden} \leq \frac{V_{ref}}{I_{sec_{pico}}} \quad (2.4)$$

Donde:

$$N = \frac{1}{2000} \text{ vueltas}$$

$$V_{ADC} = 3.3 \text{ V}$$

Como los valores de resistencias se rigen con magnitudes estándar, esta debe ser menor o igual al valor teórico calculado. N es el número de vueltas que es un valor fijo para el sensor de corriente SCT013-100A y V_{ADC} es el valor máximo de voltaje aceptado por los pines ADC de la placa de desarrollo ESP32.

A continuación, en la Tabla 2.2 se muestran los valores de la resistencia de burden calculados en base al valor máximo de entrada para cada tipo de TC descritos previamente.

Tabla 2.2 Resistencia de burden en función de la corriente máxima

Valor máximo de corriente a medir [A]	Resistencia de Burden [Ω]
100	22
90	22
80	27
70	30
60	30
50	39
40	51
30	68
20	100
10	200

Para el caso de 100 A de corriente en el lado primario, la resistencia de burden es 22 ohmios, los límites máximos y mínimos del pin ADC, así como el rango de la señal acondicionada del TC se muestran en la siguiente imagen.

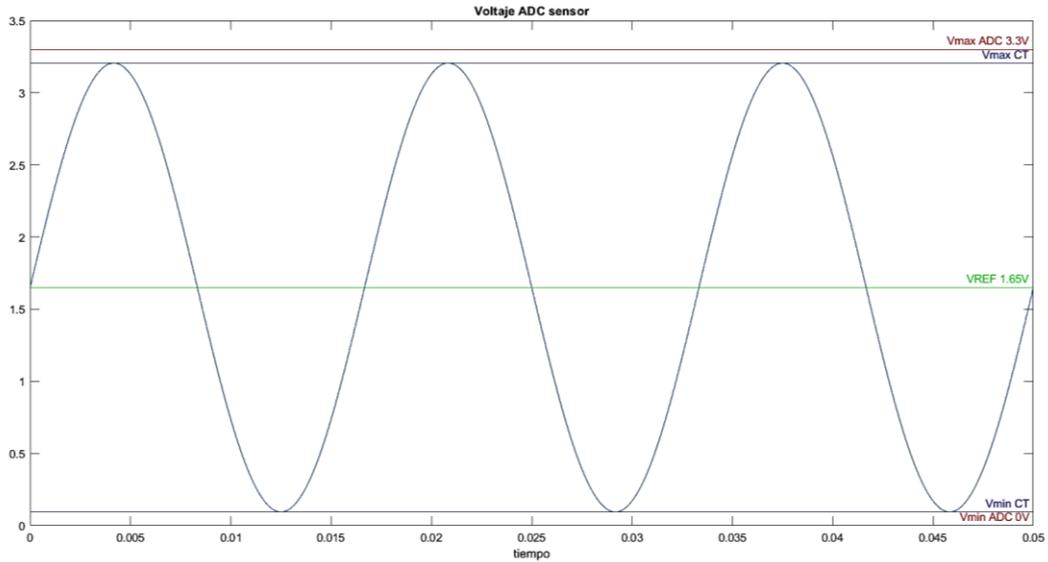


Figura 2.4 Señal acondicionada en los límites máximos y mínimos del pin ADC

Para minimizar el error en la medición de los datos de entrada se acondiciona la señal de tal manera que los valores se encuentran dentro del rango de linealidad de los pines ADC del ESP32. Para esto se toma referencia la curva en la Figura 2.5, en donde es posible establecer que entre 3.2 a 3.3 V y 0 a 0.1 V, la lectura analógica de los pines pierde la linealidad, ya que en estos rangos la curva se satura por lo que permanece en un valor constante por esto para una correcta medición es necesario que la señal que ingresa por los pines ADC este acondicionada de forma que se encuentren en la zona de linealidad.

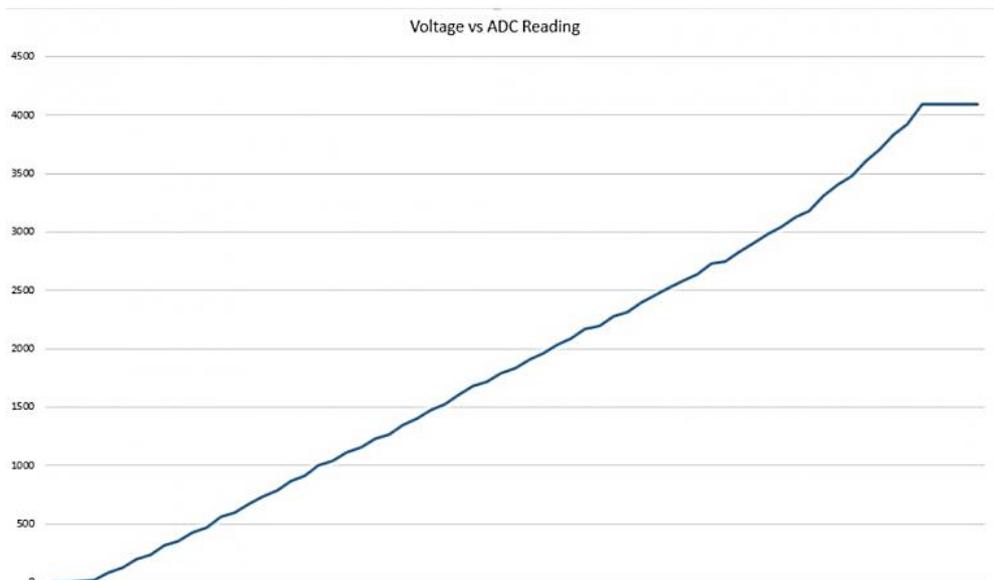


Figura 2.5 Comportamiento no lineal de los pines ADC del ESP32

El transformador de voltaje se conecta a un pin ADC disponible de acuerdo con el diagrama esquemático del sensor ZMPT101B, solo están habilitados 12 pines de los 18 en total debido a que los otros pines son necesarios para otras funcionalidades de la placa de manera que su conexión con los TC o TP podría interferir al momento de cargar el programa en la placa o cuando esté funcionando el Wifi.

Para la visualización de datos se utilizó una pantalla OLED de 128 pixeles por 64 pixeles de 0.96 pulgadas modelo SSD1306. Al ser el único elemento de su tipo se conecta a los pines I2C SDA y SCL en 34 y 36 respectivamente. Se escogió esta pantalla debido a gran contraste y uso mínimo de energía, al ser OLED no necesita retroiluminación como los LCD y se puede controlar cada píxel individualmente.



Figura 2.6 Pantalla OLED Modelo SSD1306

Para guardar los datos de manera local se utilizó un módulo de tarjeta Micro SD, que permite almacenar y acceder a información contenida dentro de la tarjeta. Se comunica a través del protocolo SPI, mediante los pines en la placa de desarrollo ESP32 dedicados exclusivamente para periféricos SPI.

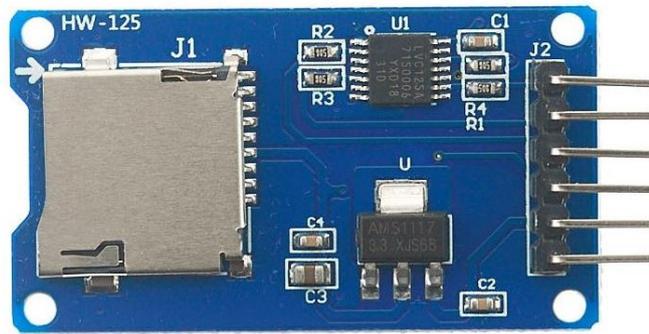


Figura 2.7 Módulo Micro SD

Las conexiones en protoboard con un TC de salida de corriente conectado es:

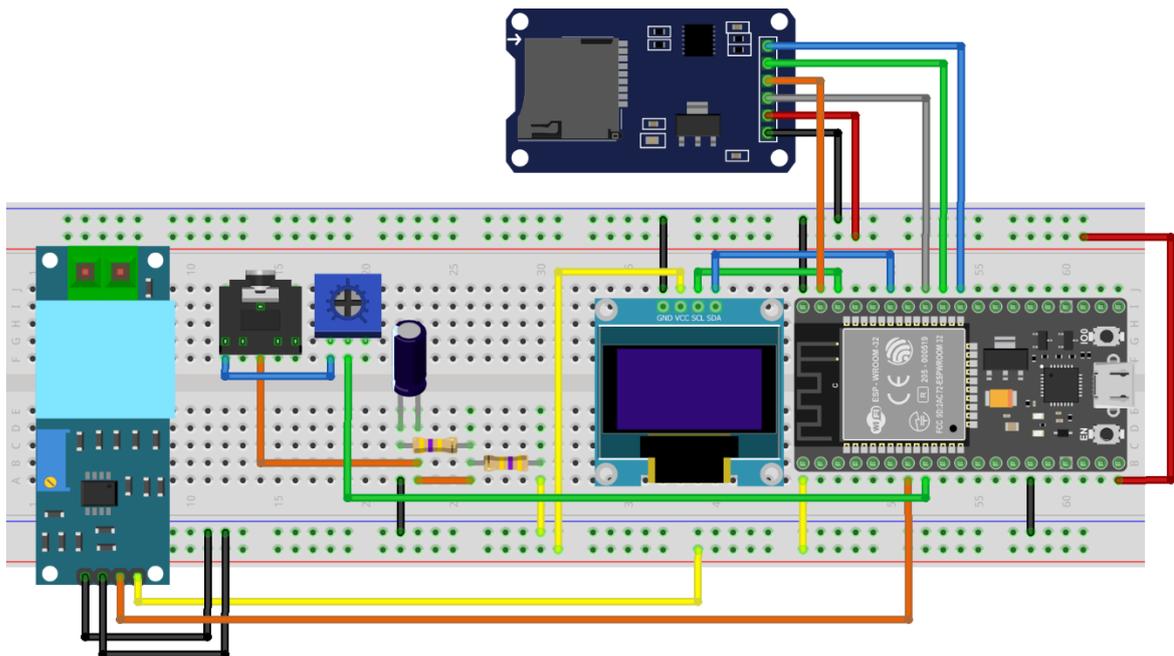


Figura 2.8 Conexiones físicas del equipo en el Protoboard

Se elaboró e imprimió diseños en PCB únicos e independientes para los TC de salida de voltaje, los TC de salida de corriente y la parte de procesamiento que incluye el ESP32, la pantalla OLED y el lector de tarjetas Micro SD. De esta forma se posibilita el reemplazo de componentes internos en caso de actualización o reemplazo.

Para la carcasa se elaboró un modelo 3D en TinkerCAD, este se exportó y se imprimió en una impresora 3D en material de PLA.

Los detalles de las partes del equipo de monitoreo se muestran en la siguiente imagen.

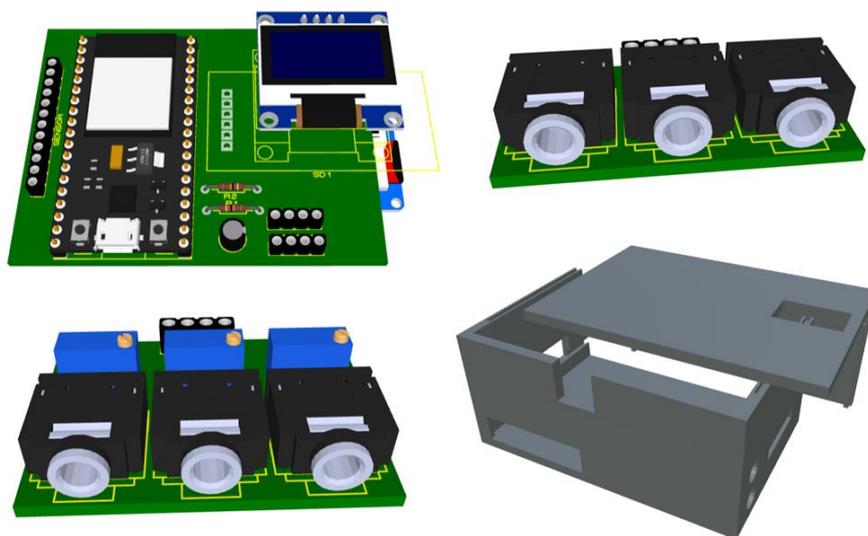


Figura 2.10 Modelamiento 3D de los componentes del dispositivo

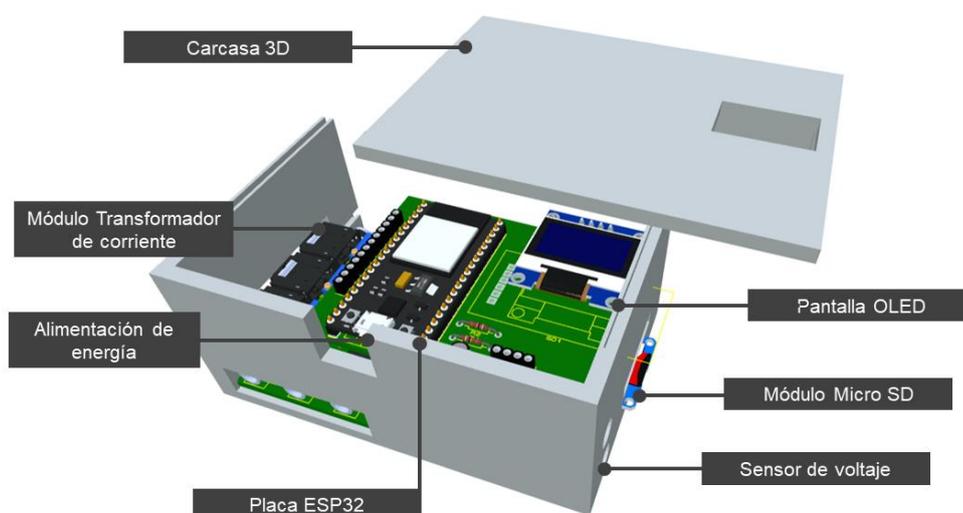


Figura 2.9 Modelamiento 3D del prototipo final

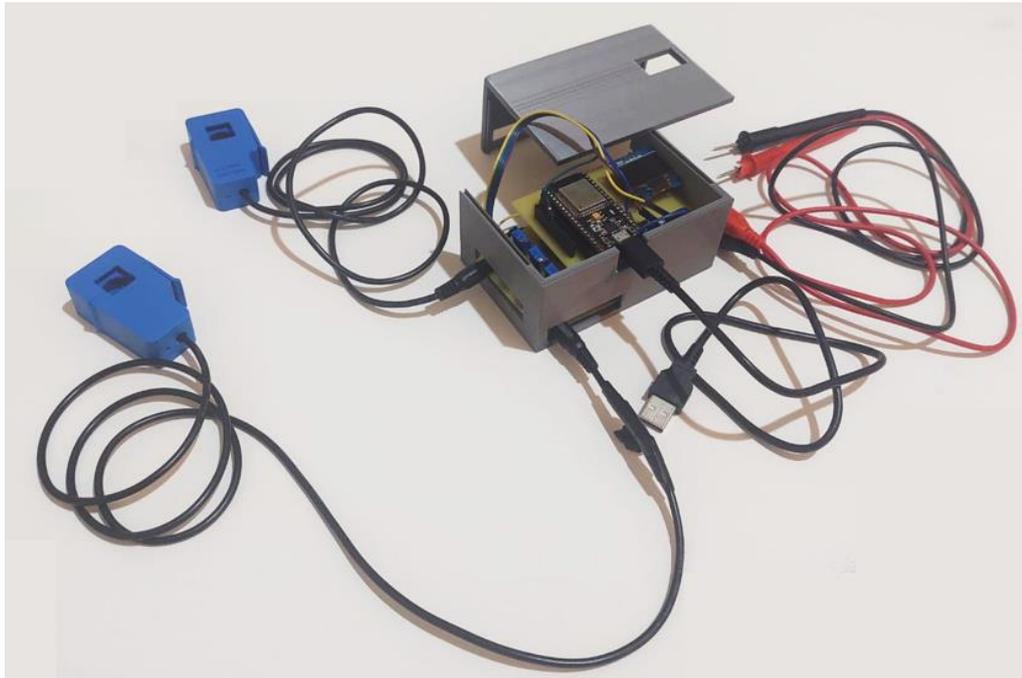


Figura 2.11 Implementación física del dispositivo de monitoreo

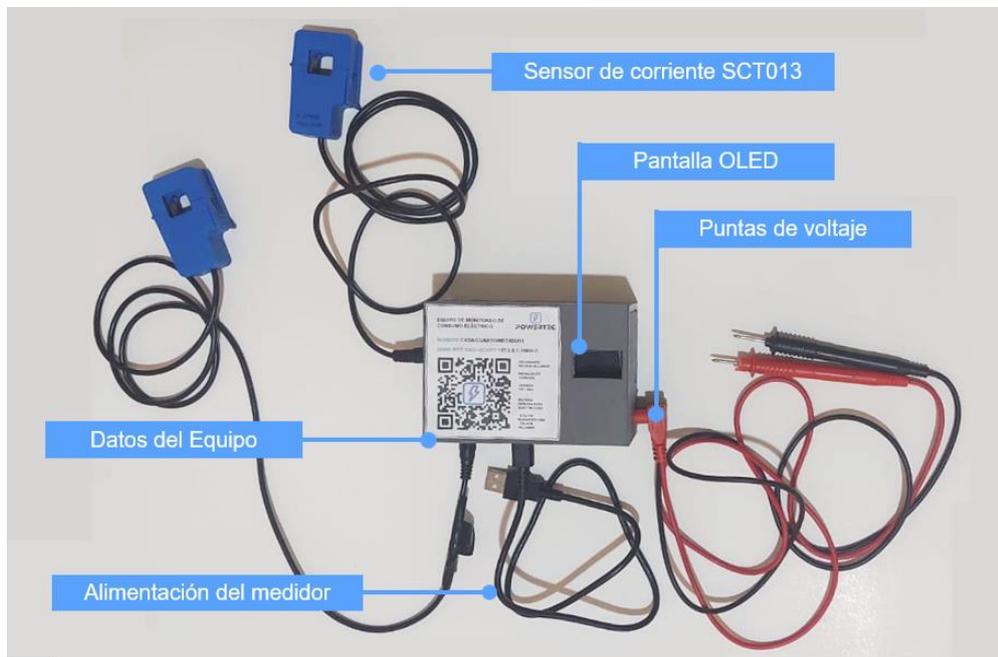


Figura 2.12 Identificación de las partes físicas del dispositivo de monitoreo



Figura 2.13 Presentación final del dispositivo



Figura 2.14 Instalación y conexión del dispositivo físico

2.5 Entorno de Programación Visual Studio Core

Para la programación de la placa electrónica se utilizó Visual Studio Code como IDE Entorno Integrado de Desarrollo, la instalación requiere descargar e instalar tanto Visual Studio Code desde <https://code.visualstudio.com/download> y Python desde <https://www.python.org/downloads/>.

Se ingresó a Visual Studio Code, desde las extensiones se instaló Platform IO que es una plataforma Open Source que permite características más avanzadas como finalización automática, la comprobación de errores y gestión de múltiples archivos y bibliotecas, funciones que no están disponibles en IDE como Arduino.

La generación de código multiplataforma sin dependencias externas a un software de sistema permite más de 1000 placas de desarrollo, más de 40 plataformas de desarrollo y más de 20 Frameworks, entre los cuales están el ESP32, y las bibliotecas necesarias para la visualización en la pantalla OLED, elaboración de cálculos eléctricos, conexión Wifi y comunicación mediante MQTT.

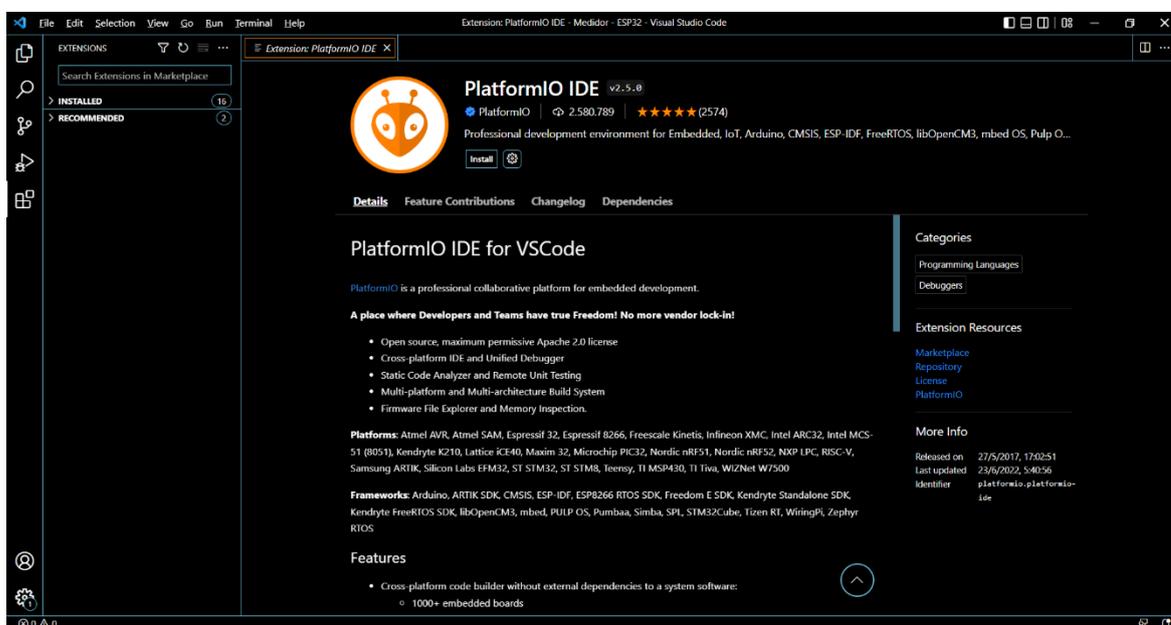


Figura 2.15 Instalación de la Plataforma IO en Visual Studio Code

Para el procesamiento de las magnitudes de voltaje y corriente instantáneas recogidas se utilizó la biblioteca Open Energy Monitor, esta implementa algoritmos

para la calibración de las señales de entrada y el cálculo de valores RMS de voltaje, corriente, potencia y energía aparente. Se prefirió usar la biblioteca y no crear los códigos desde cero debido a que la biblioteca tiene varios años de desarrollo y está optimizada para dicha función. Se utilizan específicamente estas líneas de código:

```
#include "EmonLib.h"
```

Incluye la biblioteca Open Energy Monitor y sus funciones al código actual

```
energyMonitor.voltage(pin, calibración, diferencia de fase)
```

Define el pin donde está conectado el sensor de voltaje y su calibración

```
energyMonitor.current(pin, calibración);
```

Define el pin donde está conectado el sensor de corriente y su calibración

```
energyMonitor.calcVI(periodo, muestreo);
```

Define que el periodo de 1/60Hz y cuánto tiempo toma datos para obtener el valor RMS.

```
energyMonitor.apparentPower*(millis()-tiempo)/3600000000;
```

Se obtiene la energía en kWh como la multiplicación de la potencia aparente y la diferencia de tiempo en horas entre cada medición.

2.6 Comunicación por Protocolo MQTT

MQTT es un protocolo ligero de mensajería de bajo ancho de banda, es decir, una cantidad muy limitada de información por segundo diseñado para intercambiar datos, por lo general de sensores, entre múltiples dispositivos de Internet de las Cosas. Esto permite el envío de información extremadamente rápida y un alto nivel de calidad de servicio sobre protocolos como HTTP.

El método de funcionamiento es mediante el sistema Publicación y Suscripción, un dispositivo publica mensajes sobre un tema y otro dispositivo se suscribe al mismo, de este modo recibe los mensajes asociados a ese tema.

Los mensajes pueden ser comandos de control o datos provenientes de sensores, mientras que los temas especifican las direcciones a los cuales se envían o de los cuales se reciben los mensajes. Estos se representan por texto separado por barras "/". Ejemplo, en el caso de una casa en la cual se realice el monitoreo de varios dispositivos eléctricos, los temas se pueden representar de la siguiente forma:

casa/cocina/refrigeradora
casa/cuarto/AC
casa/lavandería/secadora
casa/lavandería/lavadora

Las lecturas de los diferentes dispositivos se publican en dichos temas y solo se pueden acceder suscribiéndose a estos. Los temas tienen niveles, por lo cual para el ejemplo, si un dispositivo se suscribe al tema casa/lavandería/ recibirá las mediciones de todos los dispositivos debajo de ese nivel de jerarquía, en este caso los temas secadora y lavadora. Los temas además son sensibles a mayúsculas y minúsculas, de tal modo que los temas casa/CUARto/AC y casa/cuarto/AC no son los mismos [20].

Para la recepción de mensajes, clasificación de los temas y envío de los mensajes a todos los dispositivos suscritos se utiliza el concepto de Bróker. El bróker es un software que se ejecuta en un dispositivo local o remoto como una Raspberry Pi, una computadora o un servidor. Existen varios tipos, el más popular es Mosquitto Broker, y es el responsable de recibir las mediciones eléctricas del ESP32 de los distintos temas y enviarlo a la plataforma para la visualización y almacenamiento de las lecturas; la plataforma escogida para la visualización es NODE-RED. Un esquema del envío de información es el siguiente:

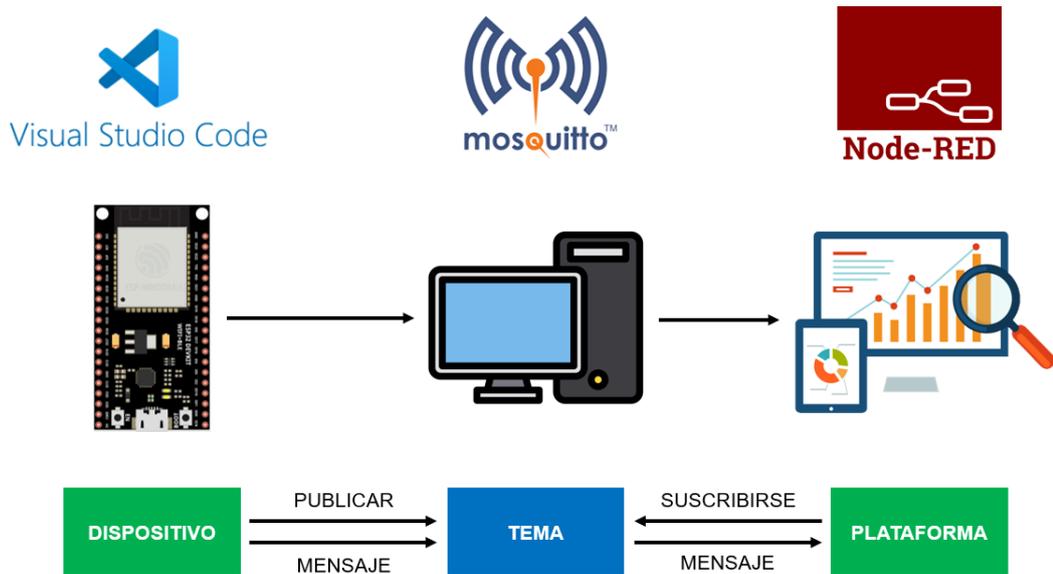


Figura 2.16 Esquema de comunicación MQTT

Se instaló el Bróker Mosquitto en una computadora Windows 11 de 64bits, se descarga desde <https://mosquitto.org/download/>, se siguió el proceso de instalación normal, luego se accedió a la consola de comandos de Windows, se usó el código Mosquitto -v esto permite iniciar el bróker en el puerto 1883 por defecto, es decir, en ese puerto se pueden publicar y suscribir a temas en la red de Wifi local.

```

C:\Windows\System32\cmd.exe - mosquitto -v
Microsoft Windows [Versión 10.0.22000.795]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Program Files\mosquitto>mosquitto -v
1658974092: mosquitto version 2.0.14 starting
1658974092: Using default config.
1658974092: Starting in local only mode. Connections will only be possible from clients running on this machine.
1658974092: Create a configuration file which defines a listener to allow remote access.
1658974092: For more details see https://mosquitto.org/documentation/authentication-methods/
1658974092: Opening ipv4 listen socket on port 1883.
1658974092: Opening ipv6 listen socket on port 1883.
1658974092: mosquitto version 2.0.14 running
  
```

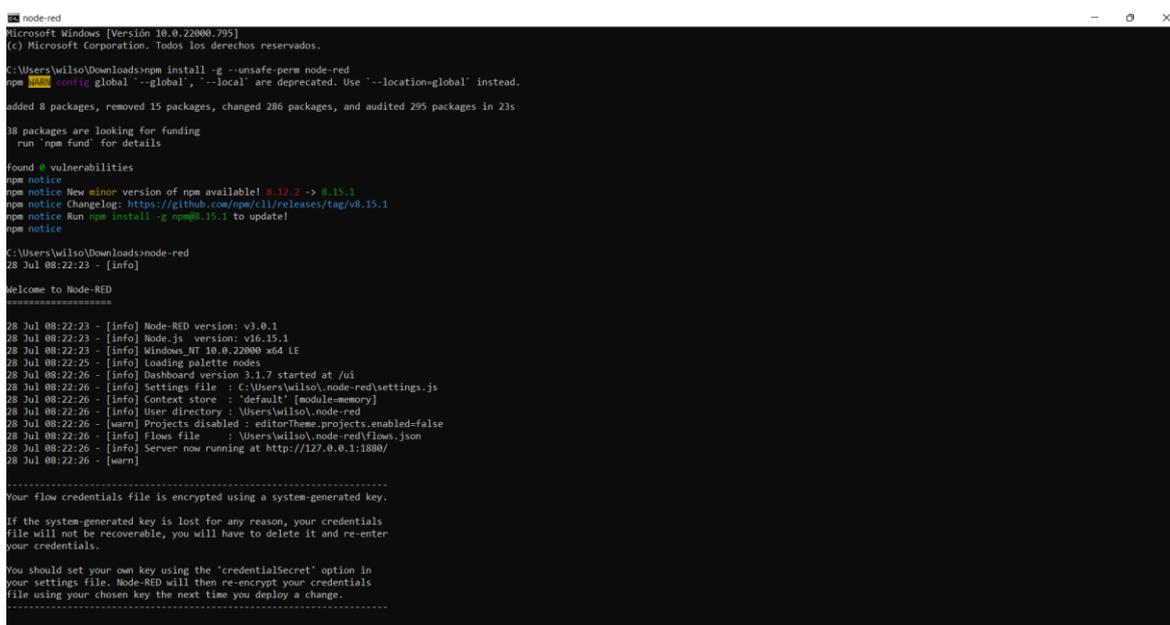
Figura 2.17 Instalación del Bróker Mosquitto

2.7 Visualización en Dashboard mediante NODE-RED

NODE-RED es una herramienta para conectar dispositivos de hardware, API y servicios en línea enfocado a internet de las cosas.

La conexión y programación de las conexiones entre dispositivos y servicios se realiza mediante la creación de flujos de nodos basados en un navegador web. Está construido en Node.js, lo que le permite una ejecución en hardware de bajo costo como computadoras, Raspberry Pi y en nube.

Para la ejecución de NODE-RED se instaló Node.js desde <https://nodejs.org/en/download/>, una vez instalado se abrió el CMD consola de comandos en Windows y se escribió el siguiente código `npm install -g --unsafe-perm node-red`.



```
node-red
Microsoft Windows [Versión 10.0.22000.795]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\wilso\Downloads>npm install -g --unsafe-perm node-red
npm WARN config global '--global', '--local' are deprecated. Use '--location=global' instead.
added 8 packages, removed 15 packages, changed 286 packages, and audited 295 packages in 23s

38 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities

npm notice New minor version of npm available! 8.12.2 -> 8.35.1
npm notice Changelog: https://github.com/npm/cli/releases/tag/v8.15.1
npm notice Run `npm install -g npm@8.15.1` to update!
npm notice

C:\Users\wilso\Downloads>node-red
28 Jul 08:22:23 - [info]
-----
Welcome to Node-RED
-----

28 Jul 08:22:23 - [info] Node-RED version: v3.0.1
28 Jul 08:22:23 - [info] Node.js version: v16.15.1
28 Jul 08:22:23 - [info] Windows_NT 10.0.22000 amd64 LE
28 Jul 08:22:25 - [info] Loading palette nodes
28 Jul 08:22:26 - [info] Dashboard version 3.1.7 started at /ui
28 Jul 08:22:26 - [info] Settings file : C:\Users\wilso\node-red\settings.js
28 Jul 08:22:26 - [info] Context store : 'default' [module=memory]
28 Jul 08:22:26 - [info] User directory : \Users\wilso\node-red
28 Jul 08:22:26 - [warn] Projects disabled: editorTheme.projects.enabled=false
28 Jul 08:22:26 - [info] Flows file : \Users\wilso\node-red\flows.json
28 Jul 08:22:26 - [info] Server now running at http://127.0.0.1:1880/
28 Jul 08:22:26 - [warn]

-----
Your flow credentials file is encrypted using a system-generated key.

If the system-generated key is lost for any reason, your credentials
file will not be recoverable, you will have to delete it and re-enter
your credentials.

You should set your own key using the 'credentialSecret' option in
your settings file. Node-RED will then re-encrypt your credentials
file using your chosen key the next time you deploy a change.
-----
```

Figura 2.18 Instalación de NODE-RED

Luego de instalar NODE-RED, se desarrolla la programación en la dirección <http://127.0.0.1:1880/>

Para la comunicación de NODE-RED con Mosquitto se activó el bróker mediante la consola de comandos en la carpeta raíz de Mosquitto con la configuración en el archivo mosquitto.conf.

Para el envío y recibo de información se corre los códigos de publicación y suscripción Mosquitto_pub y Mosquitto_sub respectivamente.

```
Mosquitto_sub -h localhost -t tema -m "mensaje"
```

```
Mosquitto_pub -h localhost -t tema -m "mensaje"
```

Donde el prefijo -t define el tema y -m define el mensaje para ambos casos.

La creación y visualización de información y gráficas se realizó en NODE-RED una vez conectado al bróker MQTT. La interfaz es completamente configurable, se decidió un diseño sencillo que muestre todos los datos y gráficas en una sola pantalla tipo Dashboard.

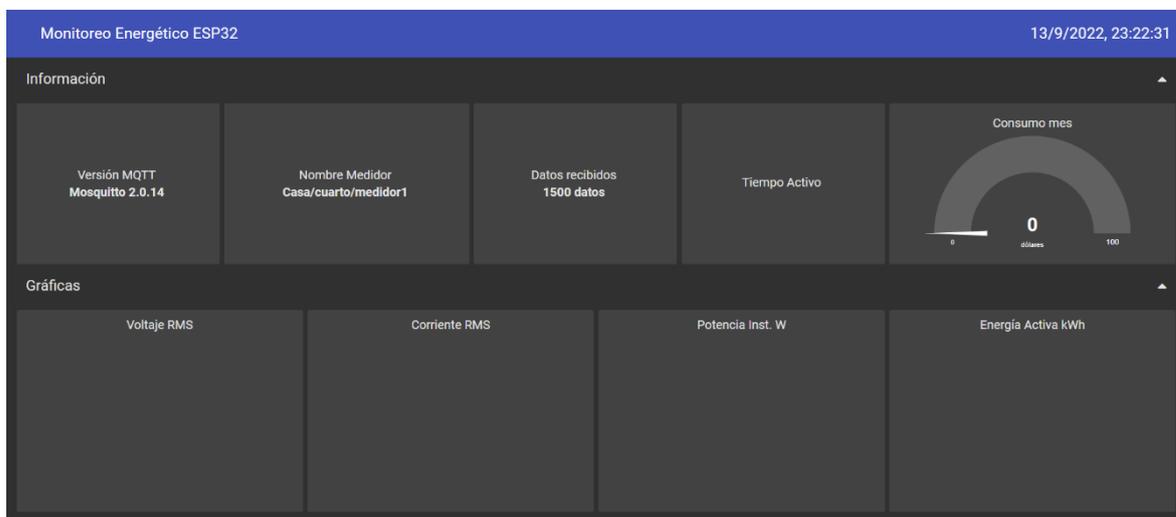


Figura 2.19 Visualización del Dashboard en NODE-RED

2.8 Diagrama de funcionamiento del Equipo de Monitoreo

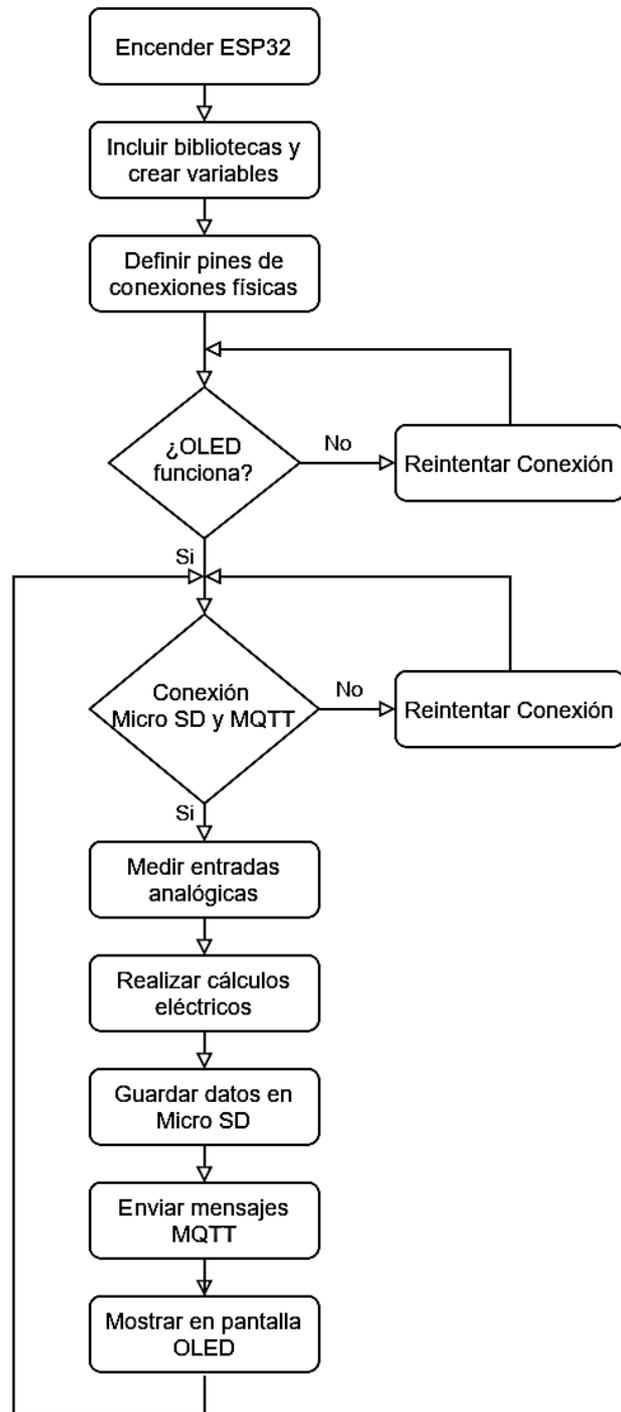


Figura 2.20 Diagrama de flujo del funcionamiento del Equipo

2.9 Pruebas de funcionamiento del Equipo de Monitoreo

Para comprobar el correcto funcionamiento del equipo se realizaron distintas pruebas, la primera se enfocó en verificar que el equipo mida correctamente las variables fundamentales, corriente y voltaje. Luego, se realizó otra prueba con énfasis en el monitoreo, es decir, se enfocó en que el equipo sea capaz de medir en distintos periodos de tiempo, verificando que todos los datos sean visualizados correctamente en el Dashboard y los datos sean almacenadas en la tarjeta de memoria.

2.9.1 Prueba de funcionamiento del equipo de monitoreo eléctrico

Para comprobar el correcto funcionamiento del equipo se realizó la medición en un tablero de distribución del laboratorio de maquinaria eléctrica de la FIEC. Para esto los TC fueron conectados tal como se muestra en la Figura 2.21,

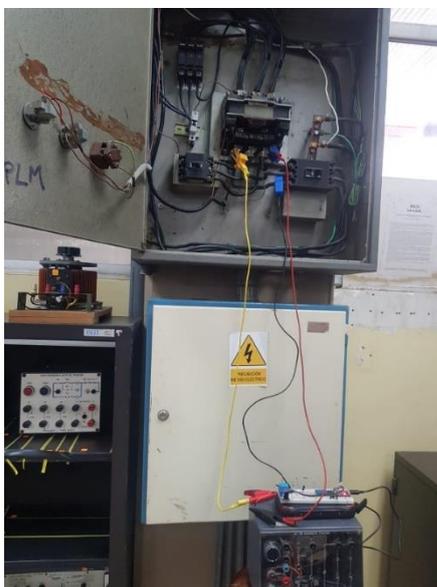


Figura 2.21 Conexión del equipo durante la prueba de funcionamiento

Para comparar los valores medidos por nuestro equipo tanto para el caso de la corriente como para el voltaje se hicieron uso una pinza amperimétrica de la marca EXTECH 380942 con capacidad de medir con resolución de

0.1 mA para corrientes bajas de hasta 30 A y un multímetro marca Fluke 1587.



**Figura 2.23 Multímetro
marca Fluke 1587**



**Figura 2.22 Pinza amperimétrica
EXTECH 380942**

Para el cálculo del error se consideró el valor que aparece en la pinza amperimétrica y el multímetro como el valor teórico y el valor de nuestra la pantalla del equipo como valor experimental, los resultados de esta prueba son mostrados en el siguiente capítulo.

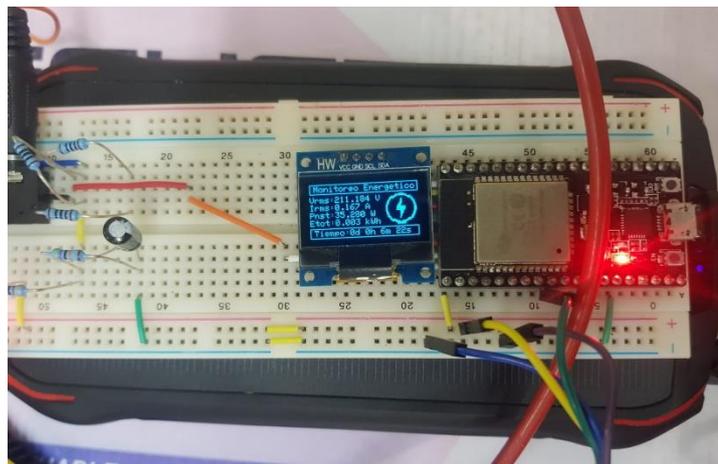


Figura 2.24 Equipo de medición durante la prueba en el laboratorio

2.9.2 Prueba de monitoreo Continuo

Se realizó la medición para un lapso de 8 horas de manera ininterrumpida, esto con la finalidad de comprobar el funcionamiento de Dashboard en NODE-RED y el almacenamiento de los datos en la tarjeta microSD.

El objeto de prueba fue la impresora 3D que se utilizó para la impresión de la caja exterior del prototipo final.

Los datos almacenados en la memoria microSD fueron almacenados en archivo y con estos datos se replicó las gráficas con la finalidad de comparar si se obtuvieron los mismos que se observaron en el Dashboard.

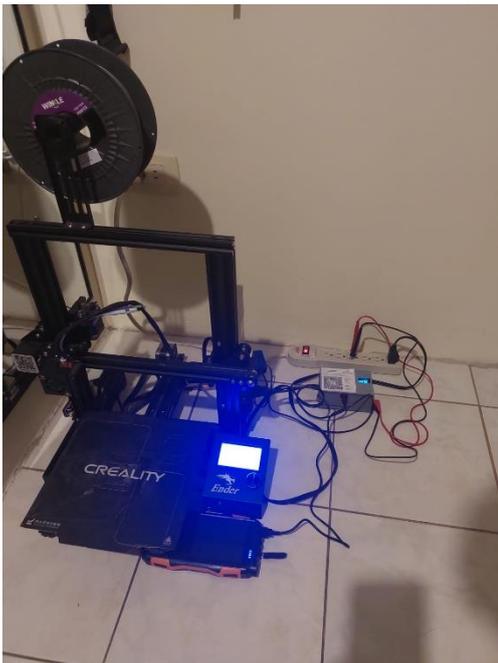


Figura 2.25 Impresora 3D utilizada como objeto de prueba



Figura 2.26 Equipo de monitoreo midiendo el consumo de la impresora 3D

CAPÍTULO 3

3. RESULTADOS Y ANÁLISIS

En esta sección se analizará los resultados obtenidos en las pruebas de funcionamiento del equipo de monitoreo eléctrico acorde con lo establecido en el capítulo anterior.

3.1 Resultados de la prueba de funcionamiento

Se realizaron varias mediciones para obtener error ante cargas diferentes, la cual fue modificada por medio del encendido o apagado de los aires condicionados en las instalaciones, obteniendo los siguientes porcentajes de error:

Tabla 3.1 Porcentaje de error del voltaje

Voltaje	Valores		Error		
	Teórico	Experimental	Absoluto	Relativo	%
Sin carga	211.5	211.184	0.316	0.00149409	0.14940898
Con carga	210.9	210.62	0.28	0.00132764	0.13276434

Para el caso del voltaje se realizó la comparación con carga y sin carga, en ambos casos el error obtenido es bajo, menor al 0.15%, lo cual es un buen indicador de que el valor de voltaje medido por el equipo es correcto y es bien calibrado.

Tabla 3.2 Porcentaje de error de la corriente

Corriente	Valores		Error		
	Teórico	Experimental	Absoluto	Relativo	%
Prueba 1	14.670	14.685	0.015	0.00102249	0.10224949
Prueba 2	34.59	35.151	0.561	0.01621856	1.62185603

Para el caso de la corriente se realizó la medición para dos carga distintas, en el primer caso el error es bajo de aproximadamente el 0.10%, lo cual da a certeza que el sensor de corriente está midiendo correctamente. Luego, al aumentar la corriente, se observó que el error aumento a 1.62%, este aumento se debe principalmente a que la pinza amperimétrica que se está usando tiene una mayor precisión para corriente menor a 30 A, al sobrepasar este valor se introduce este

error debido a la limitación del equipo de laboratorio. No obstante este aumento, el error sigue siendo lo suficientemente bajo, por lo que se determinó que los sensores de corrientes funcionaban correctamente.

Los resultados obtenidos durante esta prueba de funcionamiento demuestran la efectividad al momento de medir las variables eléctricas de voltaje y corriente que son la base para el resto de los cálculos que se pueden realizar con el equipo como la potencia y energía, o el valor a facturar por el consumo de energía.

3.2 Resultado de la prueba de monitoreo

Para este caso se muestra los datos en tiempo real, por medio del Dashboard diseñado para el usuario final.

Al inicio de la prueba:

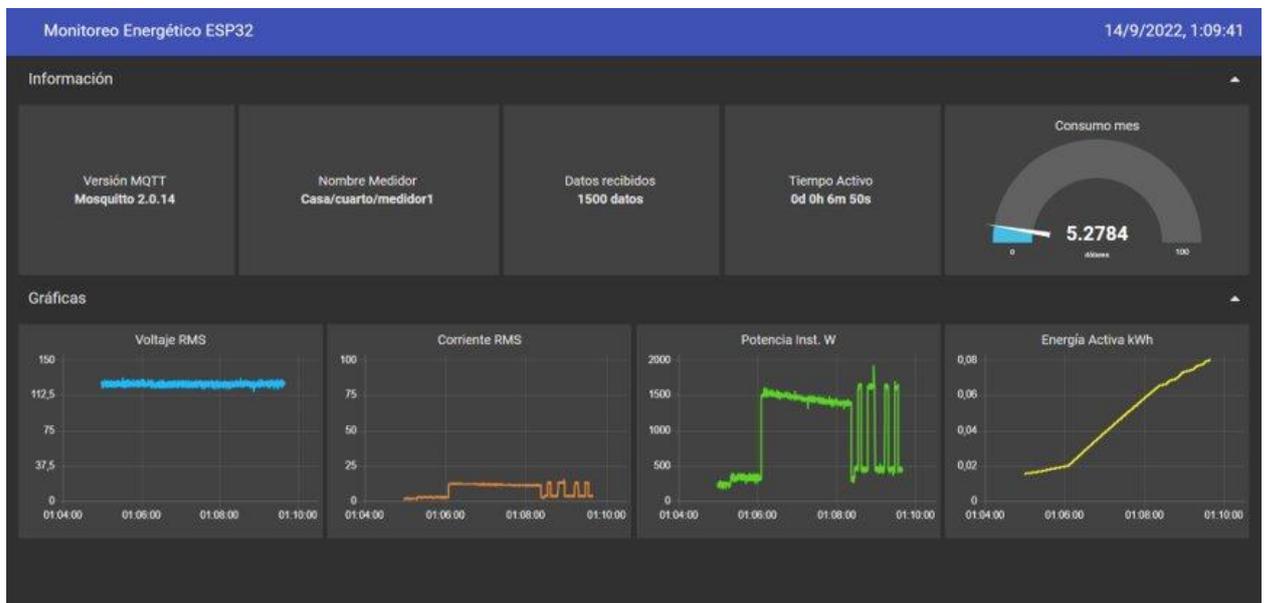


Figura 3.1 Inicio de la prueba de monitoreo

Como se muestra en la imagen anterior, el usuario es capaz de visualizar la ubicación donde está colocado el medidor, el tiempo que ha permanecido activo, la gráfica del voltaje y corriente RMS en tiempo real, así como a potencia en Watts y la energía activa que lleva consumiendo en kWh.

Al finalizar la primera hora los datos mostraban la siguiente información:

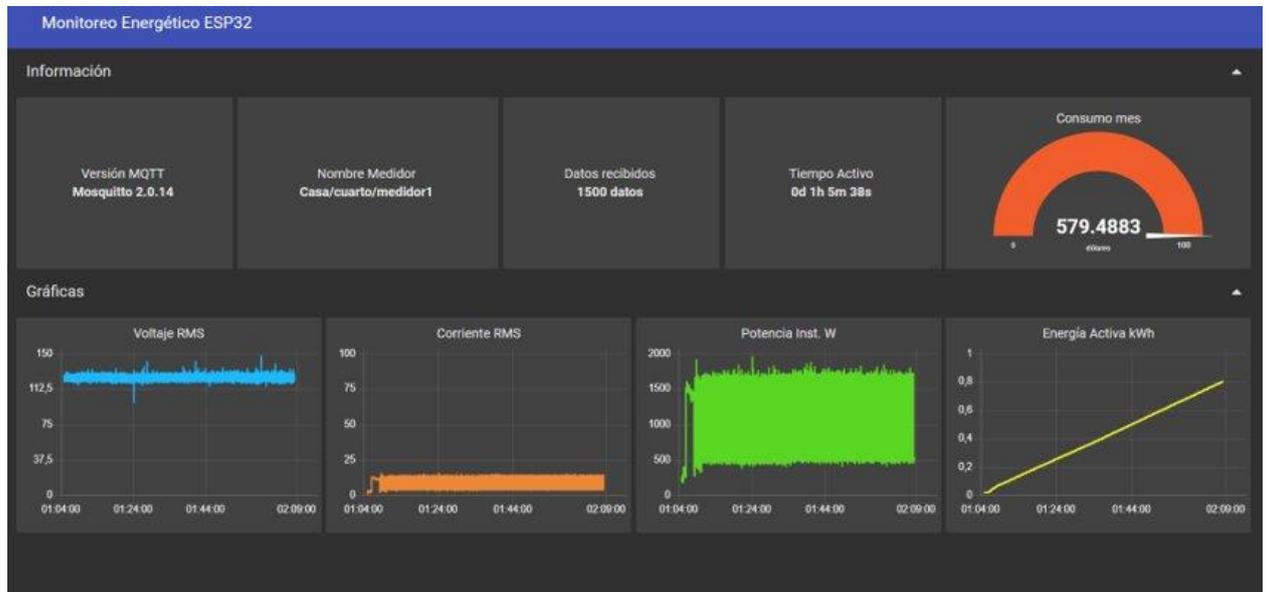


Figura 3.2 Dashboard al finalizar la primera hora de monitoreo

A las 3 horas de haber iniciado la medición:

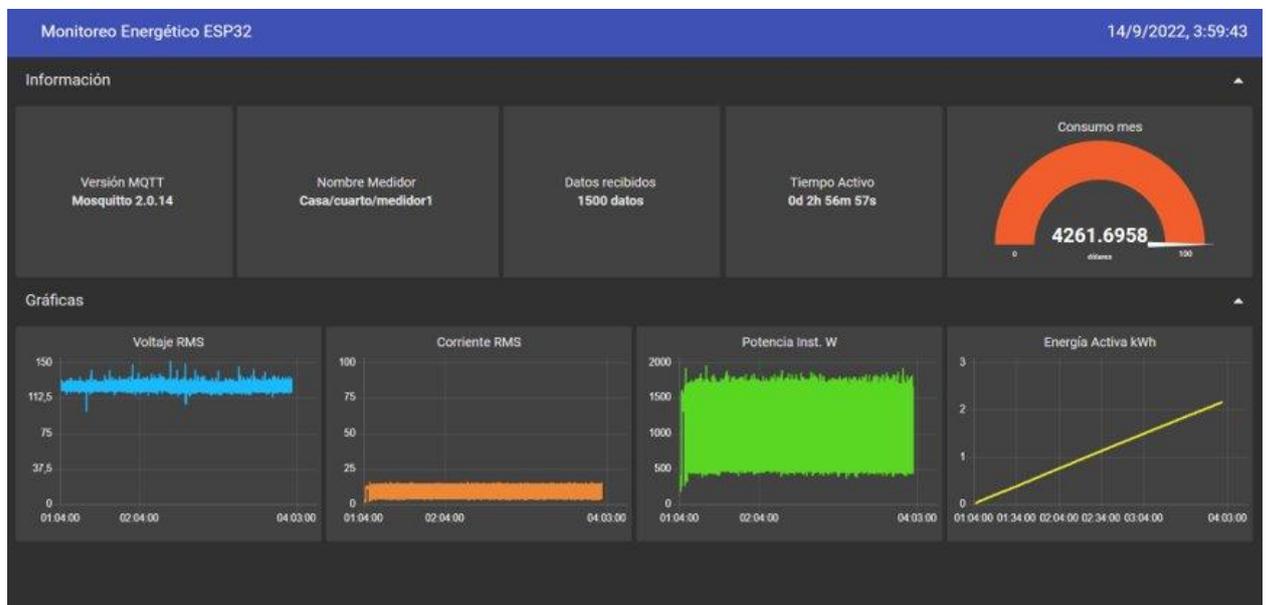


Figura 3.3 Dashboard luego de tres horas de monitoreo

Finalmente, una vez cumplida las 8 horas, la información fue visualizada de forma completa, mostrando la información en tiempo real, lo que demostró que la comunicación ente el equipo y la interfaz fue configurada de manera exitosa.

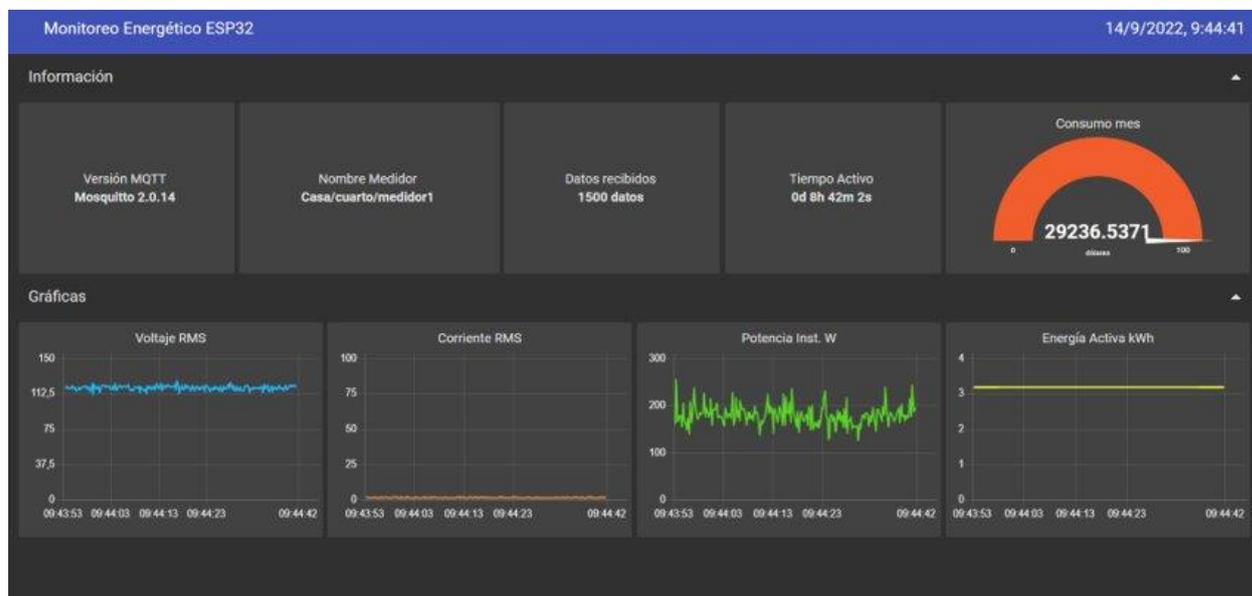


Figura 3.4 Dashboard al finalizar la prueba de monitoreo

Mientras la prueba se estaba ejecutando, los datos fueron almacenados a su vez en la tarjeta microSD, de manera que los datos almacenados fueron tabulados y se realizaron las gráficas correspondientes.

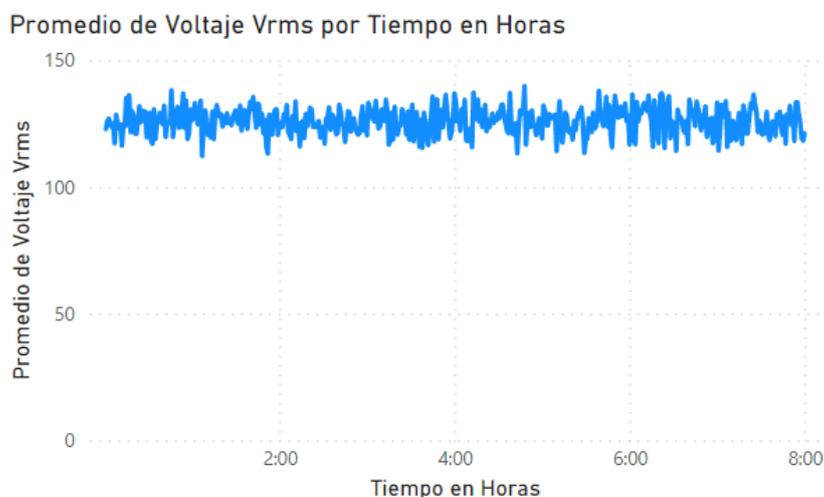


Figura 3.5 Curva del Voltaje RMS durante el monitoreo

La corriente muestra una variación a lo largo de la medición debido a que dependiendo de los movimientos que realice la impresora 3D, una vez que termina la impresión se observa que la corriente disminuye considerablemente; dado a que sigue encendida existe una pequeña corriente que mide el dispositivo.

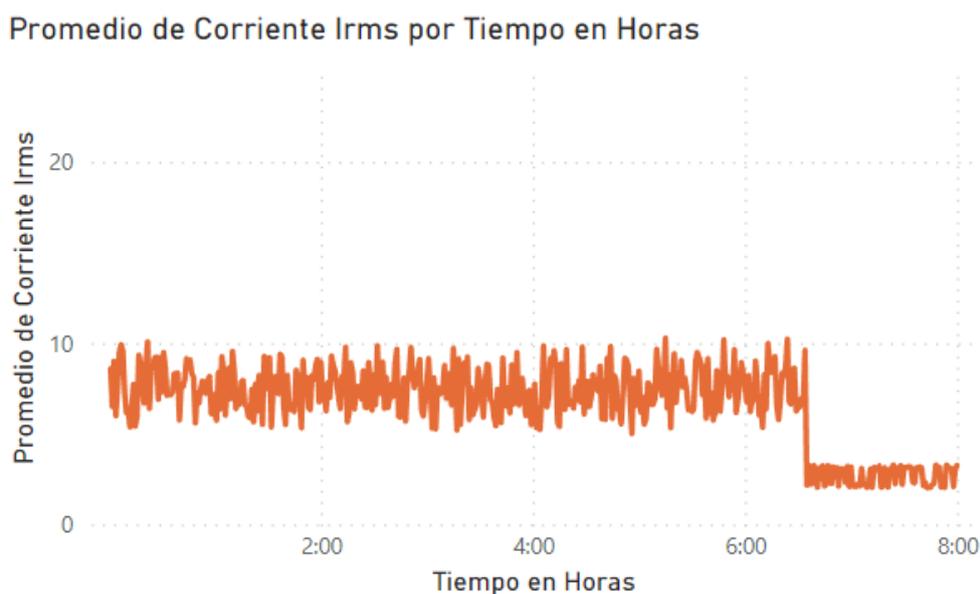


Figura 3.6 Curva del Corriente RMS durante el monitoreo

Este comportamiento de la corriente influye directamente en los valores de potencia instantánea y la energía acumulada. Para el caso de la energía se ve que crece de manera casi lineal en cada intervalo de tiempo, la pendiente de dicha curva dependerá del aumento o disminución de cargas, como se observa en la última hora de monitoreo dado a que la impresión termino. La tasa de crecimiento es mucho menor, aunque no del todo nula debido a que la impresora sigue encendida, existiendo un pequeño consumo residual de energía.

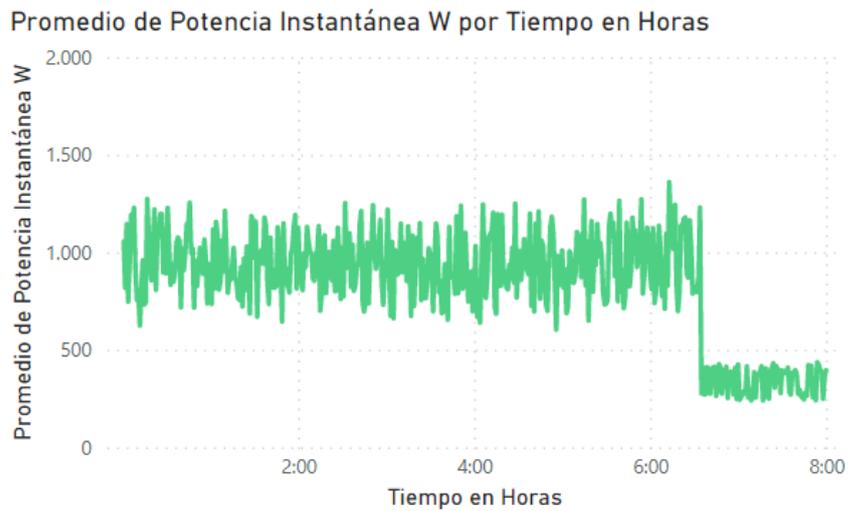


Figura 3.7 Curva de la Potencia Instantánea en Watts durante el monitoreo

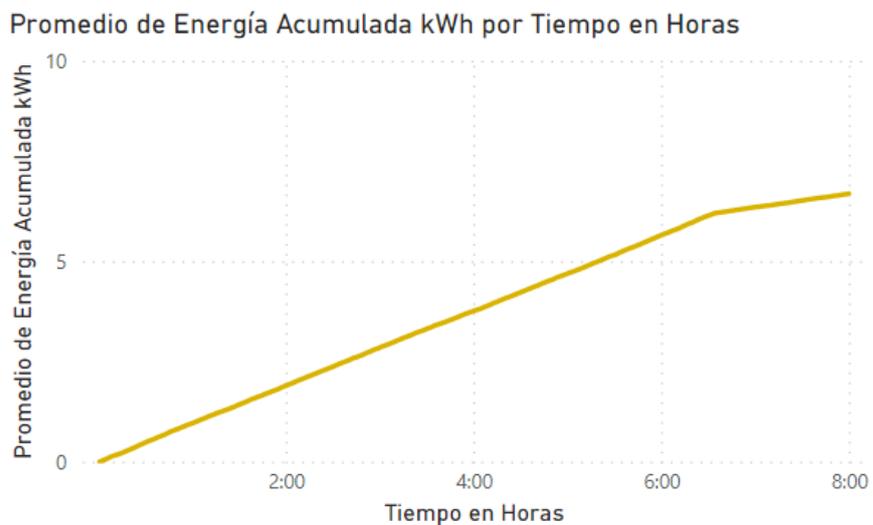


Figura 3.8 Curva del Energía Acumulada en kWh durante el monitoreo

Por otra parte, uno de los datos de importancia para el usuario es la repercusión económica que tiene el consumo energético, de allí que por medio de los datos almacenados se pudo establecer la curva del consumo en dólares, mostrando el ritmo a que aumenta en distintos periodos de tiempo. Para la obtención del consumo de energía se estableció un valor fijo de 9 ctvs. el kWh, un valor aproximado para consumidores residenciales. No obstante este valor es posible modificarlo en la programación para que se adapte al tipo de cliente.

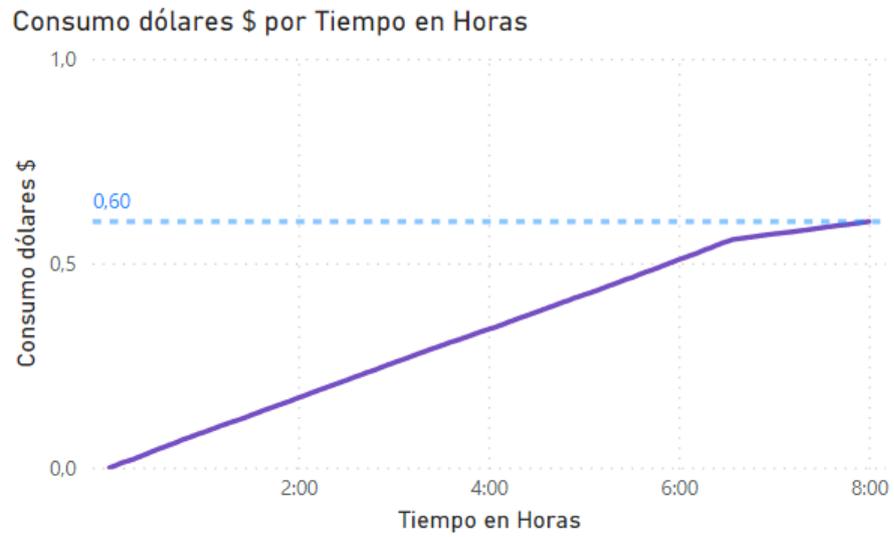


Figura 3.9 Valor en dólares del consumo energético durante el monitoreo

Las gráficas anteriores demuestran que la información es almacenada correctamente, y estos datos pueden ser utilizados por el usuario para un análisis posterior.

3.3 Análisis de Costos

Para la construcción del equipo de monitoreo se detallan los siguientes valores, considerando precio de venta al por menor en las tiendas electrónicas en Ecuador.

Tabla 3.3 Costo de fabricación del dispositivo de monitoreo

Ítem	Descripción	Costo Unitario	Cantidad	Costo total
1	Placa ESP32	\$ 10.58	1	\$ 10.58
2	Cable micro USB	\$ 1.07	1	\$ 1.07
3	Sensor de Voltaje ZMPT101B	\$ 6.50	1	\$ 6.50
4	Pantalla OLED modelo SSD1306	\$ 8.00	1	\$ 8.00
5	Módulo de tarjeta Micro SD	\$ 2.50	1	\$ 2.50
6	Potenciómetro trimmer	\$ 0.25	4	\$ 1.00
7	Resistencias de 10k	\$ 0.15	2	\$ 0.30
8	Capacitor	\$ 0.25	1	\$ 0.25
9	Puertos Jack	\$ 0.25	9	\$ 2.25
10	Insumos generales	\$ 1.00	1	\$ 1.00
11	Mano de obra-PCB	\$ 10.00	1	\$ 10.00
12	Mano de obra-Caja en impresión 3D	\$ 5.00	1	\$ 5.00
TOTAL				\$ 48.45

Adicional al costo descrito en la Tabla 2.1Tabla 3.3, se deben considerar un costo variable que depende de la cantidad de sensores de corriente que el cliente necesite, para el caso base se considera solo 2 TC.

Tabla 3.4 Costo total de los TC colocados en el dispositivo de monitoreo

Ítem	Descripción	Costo Unitario	Cantidad	Costo total
1	Sensores de corriente	\$ 10.50	2	\$ 21.00
TOTAL				\$ 21.00

Tomando en cuenta estos dos valores, nos da un costo final para el equipo de \$69.5. Entre los equipos de monitoreo existentes en el mercado con el menor costo y con características similares, se encontró el GEN 2 Emporia Vue, destinado al

uso residencial, y que posee la misma cantidad de TC, cuyo valor en el extranjero sumado a los costos de envío al país da un valor de \$111.43.

De manera que, comparando nuestro equipo versus uno ya existente en el mercado es un 38% más barato, sobre el cual se puede añadir un margen de ganancia acorde al PVP final que se escoja para la venta del equipo. Cabe destacar que, si bien se comparó con el GEN 2 Emporia Vue, nuestro equipo presenta diferencia que le dan un valor agregado, como lo es su característica escalable al no estar limitado al uso residencial sino para otros ámbitos, lo que le da una ventaja competitiva.

3.4 Viabilidad Económica

Los precios mostrados en la sección anterior son el PVP de las tiendas electrónicas al público general. No obstante, el objetivo del proyecto es la producción de varias unidades del equipo, por lo que al comprar los materiales base a un mayor volumen estos costos se reducen, incluso si se importa estos materiales se pueden conseguir precios aún menores y más competitivos, de manera que podamos reducir los costos de producción de cada unidad.

De esta forma aumentar la diferencia de costos frente a los equipos similares en el mercado.

CAPÍTULO 4

4. CONCLUSIONES Y RECOMENDACIONES

A continuación, se presentan las conclusiones a las que se han llegado luego del diseño, elaboración y prueba del equipo y la plataforma de monitoreo de consumo eléctrico.

4.1 Conclusiones

Se construyó el equipo de monitoreo de consumo eléctrico mediante la conexión de la placa de desarrollo ESP32 con sensores de voltaje y corriente disponibles de manera local. El equipo puede realizar mediciones con el código de programación, pero se debe configurar manualmente la precisión antes de las mediciones. Las mediciones tomadas por los sensores se envían hacia la placa a través de cables comunes jumpers o a través del circuito ampliamente disponibles para proyectos electrónicos; las mediciones se enviaron por MQTT al Dashboard de NODE-RED a tiempo real. También es necesario que la conexión Wifi se mantengan durante la medición para el envío de datos, las credenciales de la red se colocan dentro de la programación antes de las sesiones de monitoreo de consumo eléctrico.

Los transductores de corriente y voltaje utilizados acorde a sus características de fábrica permiten la lectura de hasta 250 voltios de voltaje y hasta 100 amperios de corriente que es más que suficiente para instalaciones residenciales. No obstante, para las pruebas se comprobó su funcionamiento para corrientes de hasta 30 A, debido a las limitantes de las pinzas amperimétricas usadas en el laboratorio, al hacer pruebas para corrientes mayores se notó un ligero aumento en el error puesto a que el instrumento utilizado como referencia perdía precisión a medida que se excedía su capacidad máxima de medición introduciendo error en las mediciones. Para el caso del voltaje no existió inconvenientes puesto a que el máximo voltaje de salida del sensor de voltaje para este caso es de 250 V, y el multímetro utilizado tiene una capacidad máxima de medición mayor, por lo que se pudo verificar que el sensor estaba midiendo correctamente puesto a que se obtuvieron error porcentuales menores al 0.2%.

Para instalaciones de instituciones como universidad, edificios del gobierno, condominios, así como instalaciones industriales, resulta más fácil la instalación de varios de los medidores para la obtención del consumo estratificado; el consumo total entonces es la suma de todos los consumos de los medidores instalados. La carcasa del equipo lo protege del polvo, humedad y condiciones que puedan afectar el funcionamiento del dispositivo de medición, en caso de daño es fácilmente reemplazable a través de la impresión de otra carcasa en PLA y el montaje de los electrónicos y los sensores en la nueva caja.

El uso de una biblioteca existente en la programación de la placa de desarrollo ESP32 permite obtener algoritmos ya refinados durante años de actualizaciones sin necesidad de hacer todo desde cero. Las partes de código que corresponden a la escritura de datos en la tarjeta MicroSD y en la pantalla OLED son modificables para almacenar y mostrar únicamente los valores importantes según los requerimientos del cliente, esto es especialmente importante debido a que en ciertos escenarios de medición solo se requieren medir corrientes, voltajes, potencia instantánea o facturación y no todas las variables. El aumento o disminución de la escritura o muestra en los datos no tiene implicaciones en la velocidad de procesamiento de la placa electrónica ESP32 debido a que el uso de potencia computacional en la escritura de datos es mínimo.

El entorno de trabajo basados en flujos de la interfaz de NODE-RED resulta muy visual para personas no técnicas en la parte eléctrica y de programación, sumado a la modificación de las gráficas basada en arrastrar, escribir y soltar se pueden tener múltiples versiones de las visualizaciones de las mediciones eléctricas.

La calibración del sensor de corriente depende de los valores máximos de medición que se ajustan con el potenciómetro según la tabla calculada. Al colocar un rango de medición y cambiar drásticamente la medición según las pruebas de linealidad se observó un aumento en el error de alrededor de 1%, este valor es aceptado dado que está por debajo de las especificaciones técnicas de los fabricantes de los sensores.

Las cargas utilizadas para la prueba del equipo y la plataforma de monitoreo se midieron en periodos cortos y largos de tiempo, se observó que la tasa de actualización en la plataforma fue instantáneo debido a la forma de funcionamiento del protocolo MQTT y los datos accedidos desde la tarjeta Micro SD coinciden con los mostrados en el Dashboard. No hubo cambios abruptos de las mediciones a excepción de la disminución de carga o desconexión de los sensores durante las pruebas.

El diseño del equipo mediante diversos componentes separados permiten el reemplazo de los mismos cuando estos presenten un mal funcionamiento. Para el caso de actualización, por ejemplo de la pantalla OLED unicolor a una pantalla más grande de múltiples colores, basta solo con imprimir una nueva carcasa que considere el cambio, lo mismo si se tratase de sensores o la adición de elementos como antenas para reforzar la conexión Wifi. De este modo se aporta al desarrollo sostenible, en especial a la economía circular, alargando la vida útil del equipo y que no se tenga que desechar. Si bien la programación es sencilla para una persona no técnica, no necesariamente el cliente va a realizar las configuraciones iniciales, el empleo del equipo de medición puede promover el empleo de personas especialistas en mediciones eléctricas, computación y eficiencia eléctrica.

El precio de fabricación del producto es 38% más barato considerando el precio del producto más económico entre los existentes en la competencia, ya que existen otros dispositivos con valores más elevados. Además, este porcentaje de ahorro económico puede incrementarse aplicando un modelo de economía de escala, es decir por medio de la producción de varias unidades, de manera que se pueden obtener los componentes más baratos comprando al por mayor directamente a los proveedores en el extranjero, y el dispositivo serían ensamblado en el Ecuador,

De esta forma, nuestro proyecto busca promover un auge en el uso de estos dispositivos, en consecuencia un aumento en los trabajos de mantenimiento, energías renovables e ingeniería para la mejora de la eficiencia de las instalaciones eléctricas una vez determinado el consumo real de las mismas.

4.2 Recomendaciones

Se recomienda la prueba del equipo de monitoreo con cargas de corriente directa, ya que este es un factor que no depende de los transductores sino de la programación y permite el uso del equipo de monitoreo en instalaciones de paneles solares, vehículos eléctricos y otras cargas de corriente directa. En dicho caso no se requiere usar la librería Open Energy Monitor sino la lectura analógica de los pines ADC.

Para verificar el funcionamiento para corrientes mayores a 30 A es necesario equipos de laboratorio con mayor capacidad de medición para que sirvan de referencia, y poder realizar una comparación mejor comparación entre datos teóricos y experimentales.

La carcasa 3D puede diseñarse bajo técnicas de modelado profesional para que sea más versátil, con la implementación de un soporte para rieles industriales, forma adecuada para el panel de interruptores e incluso la colocación en serie de diversos módulos de medición en una sola caja.

Para aumentar la robustez y velocidad del equipo se puede programar los comandos de funcionamiento en un lenguaje compilado mucho más simplificado, en este caso se entra a trabajar en temas como uso de la memoria de la placa electrónica ESP32, encendido y apagado de los núcleos de procesamiento, modos de sueño profundo, velocidad de procesamiento, entre otras cosas.

BIBLIOGRAFÍA

5. TRABAJOS CITADOS

- [1] J. J. Lamb y P. Bruno G, *Energy-Smart Buildings: Design, construction and monitoring of buildings for improved energy efficiency*, IOP Publishing, 2020.
- [2] L. C. Felius, F. Dessen y B. D. Hrynyszyn, «Retrofitting towards energy-efficient homes in European cold climates: a review,» *Energy Efficiency*, vol. 13, nº 1, pp. 101-125, 2020.
- [3] A. J. Wood, B. F. Wollenberg y G. B. Sheblé , *Power Generation, Operation and Control*, New Jersey: IEEE Wiley, 2014.
- [4] W. H. Hayt, J. E. Kemmerly y S. M. Durbin, *Análisis de circuitos en ingeniería*, Mexico: McGraw Hill, 2012.
- [5] Smappee, «Smappee Infinity- Sistema de gestión energética todo en uno.,» 2022. [En línea]. Available: Smappee Infinity. [Último acceso: 10 Julio 2022].
- [6] Eastron Europe, «SDM120M Series,» 2022. [En línea]. Available: <https://www.eastroneurope.com/products/view/sdm120modbus>. [Último acceso: 2022 julio 10].
- [7] Fluke Corporation, «Registradores de energía eléctrica trifásicos 1732 y 1734 de Fluke,» 2022. [En línea]. Available: <https://www.fluke.com/es-es/producto/comprobacion-electrica/calidad-electrica/1732-1734>. [Último acceso: 2022 Julio 15].
- [8] Emporia Corp, «How the Gen 2 Vue Energy Monitor Works.,» 2022. [En línea]. Available: <https://www.emporiaenergy.com/how-the-vue-energy-monitor-works>. [Último acceso: 2022 Julio 20].
- [9] J. L. Blackburn y T. J. Domin, *Protective Relaying Principles and Applications*, CRC press, 2014.
- [10] R. Thiara, *DESIGN AND IMPLEMENTATION OF IOT BASED REMOTE MONITORING SYSTEM AT ELECTRICAL ENGINEERING LABORATORY IN PRESIDENT UNIVERSITY*, Tesis Doctoral: President University, 2019.

- [11] Dechang Electronic Co., Ltd, «Split core current transformer, Model : SCT013-100,» 2022. [En línea]. Available: <https://en.yhdc.com/product/SCT013-401.html>. [Último acceso: 10 Junio 2022].
- [12] C. L. Cartagena Lobos y R. E. Quintanilla Padilla, «Plataforma IoT para el control y monitoreo de variables físicas con tecnología Open Hardware,» *Revista Tecnológica*, nº 12, 2019.
- [13] Qingxian Zeming Langxi Electronic Devices Co., Ltd. , «Current-type Voltage Transformer, ZMPT101B,» 2022. [En línea]. Available: <https://www.micro-transformer.com/2ma-2ma-voltage-transformer-ZMPT101B.html>. [Último acceso: 8 Junio 2022].
- [14] UNIT Electronics, «ZMPT101B Sensor De Voltaje AC 2mA - UNIT Electronics,» 2022. [En línea]. Available: <https://uelectronics.com/producto/zmpt101b-sensor-de-voltaje-ac-2ma/>. [Último acceso: 8 julio 2022].
- [15] I. Abubakar, M. W. Mustafa, S. Abd Khalid y H. Shareef, «Calibration of ZMPT101B voltage sensor module using polynomial regression for accurate load monitoring,» *Journal of Engineering and Applied Sciences*, vol. 12, nº 4, pp. 1077-1079, 2017.
- [16] Arduino Online Shop, «Arduino Uno Rev3,» 2021. [En línea]. Available: <https://store-usa.arduino.cc/products/arduino-uno-rev3>. [Último acceso: 2022 Julio 10].
- [17] R. P. (Trading) Ltd, «Buy a Raspberry Pi 4 Model B,» Raspberry Pi, 2022. [En línea]. Available: <https://www.raspberrypi.com/products/raspberry-pi-4-model-b/>. [Último acceso: 2022 Julio 10].
- [18] V. O. Oner, *Developing IoT Projects with ESP32: Automate your home or business with inexpensive Wi-Fi devices*, Packt Publishing, 2021.
- [19] A. S. Salunkhe , Y. K. Kanse y S. S. Patil, «Internet of Things based Smart Energy Meter with ESP 32 Real Time Data Monitoring,» *2022 International Conference on Electronics and Renewable Systems (ICEARS)*, pp. 446-451, 2022.
- [20] D. Llorca Llinares, *Desarrollo de aplicación IoT para la monitorización de consumos eléctrico en una vivienda*, Valencia: UNIVERSITAT POLITÈCNICA DE VALÈNCIA, 2020, p. 40.

APÉNDICES

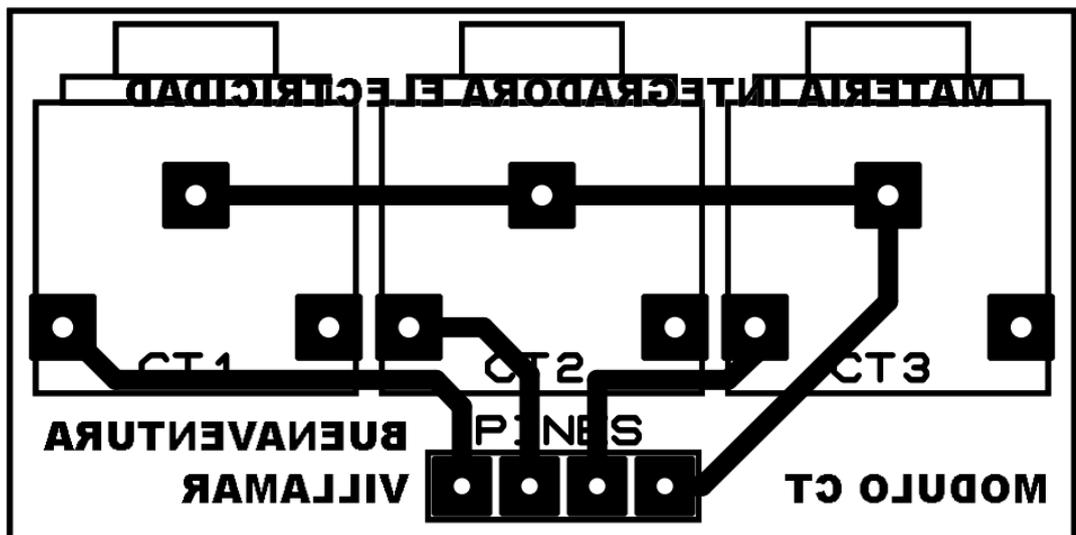
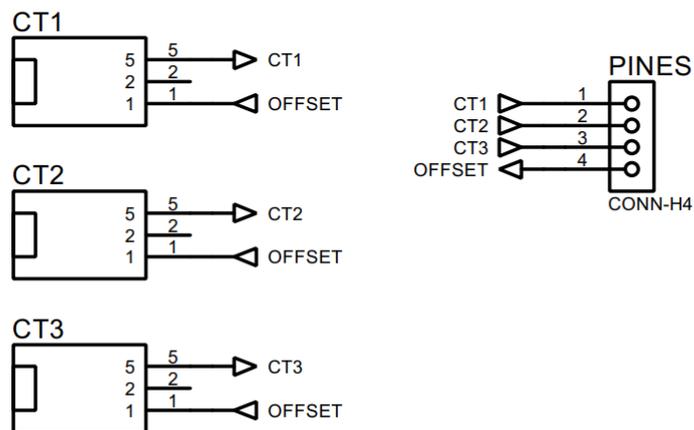
APÉNDICE A

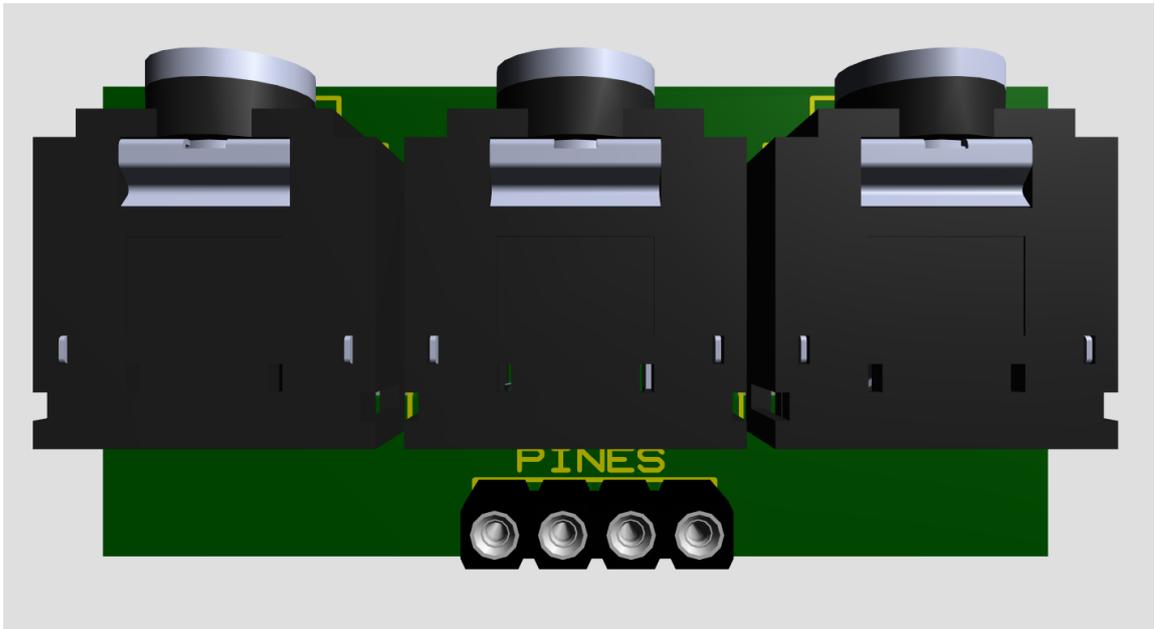
Módulo Transformador de corriente con salida de voltaje

EQUIPO DE MONITOREO DE CONSUMO ELÉCTRICO

KEMBER BUENAVENTURA, WILSON VILLAMAR

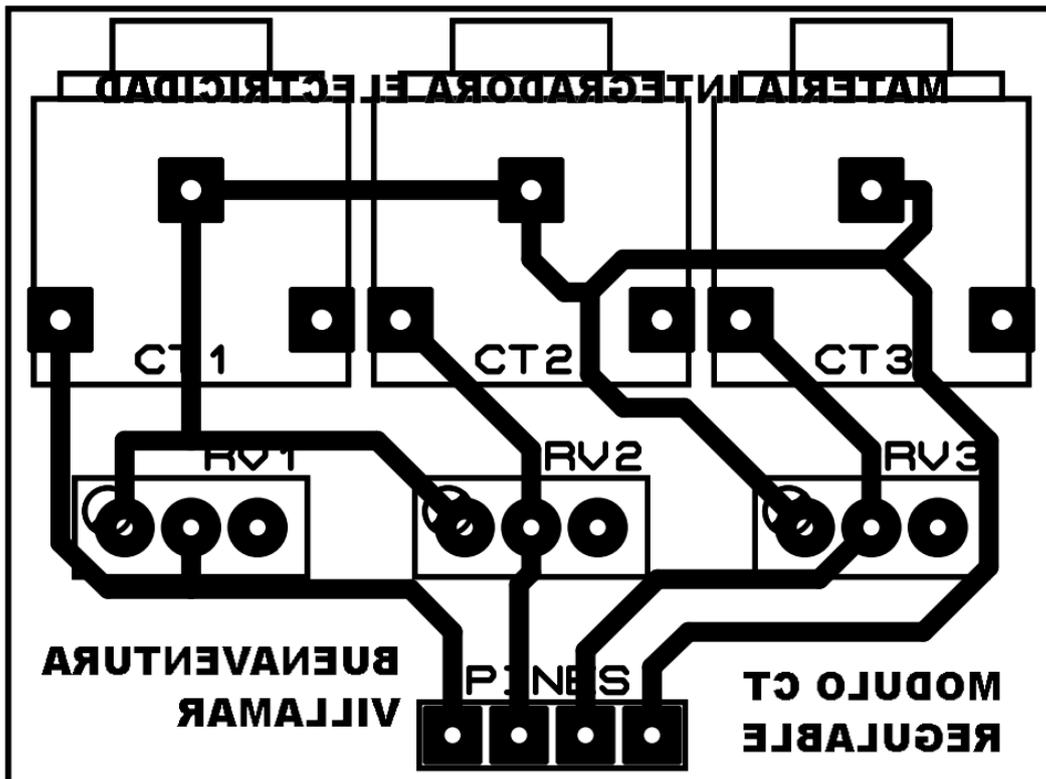
Módulo Transformador de corriente de voltaje

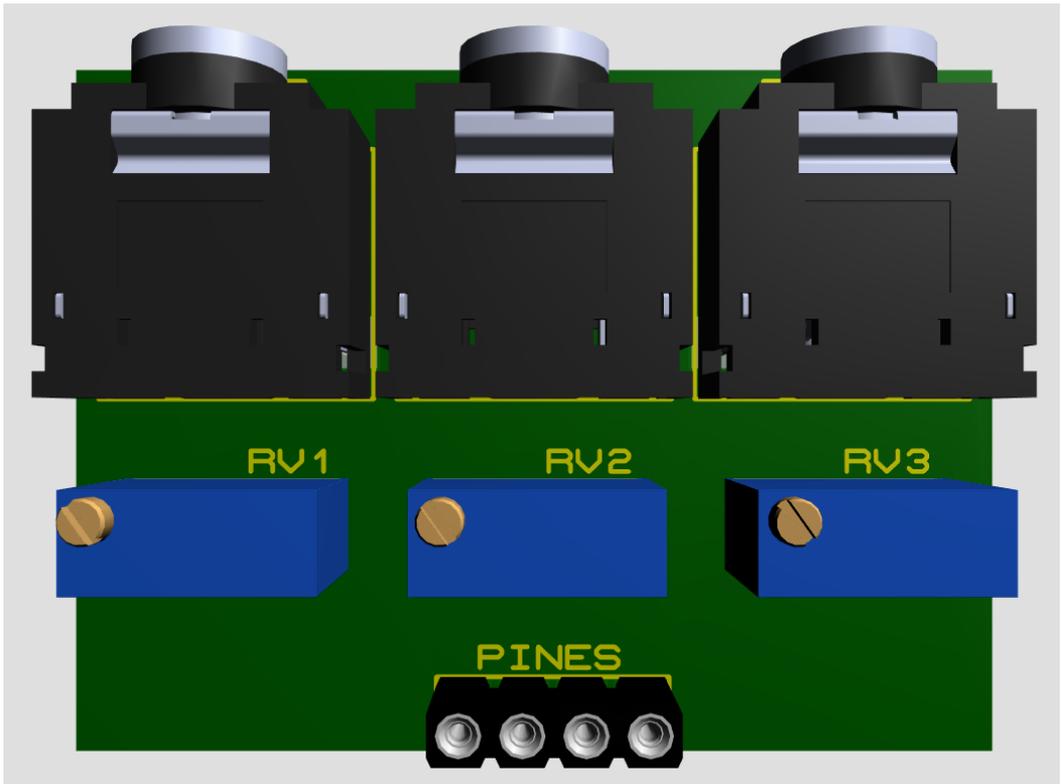




Módulo Transformador de corriente con salida de corriente

EQUIPO DE MONITOREO DE CONSUMO ELÉCTRICO



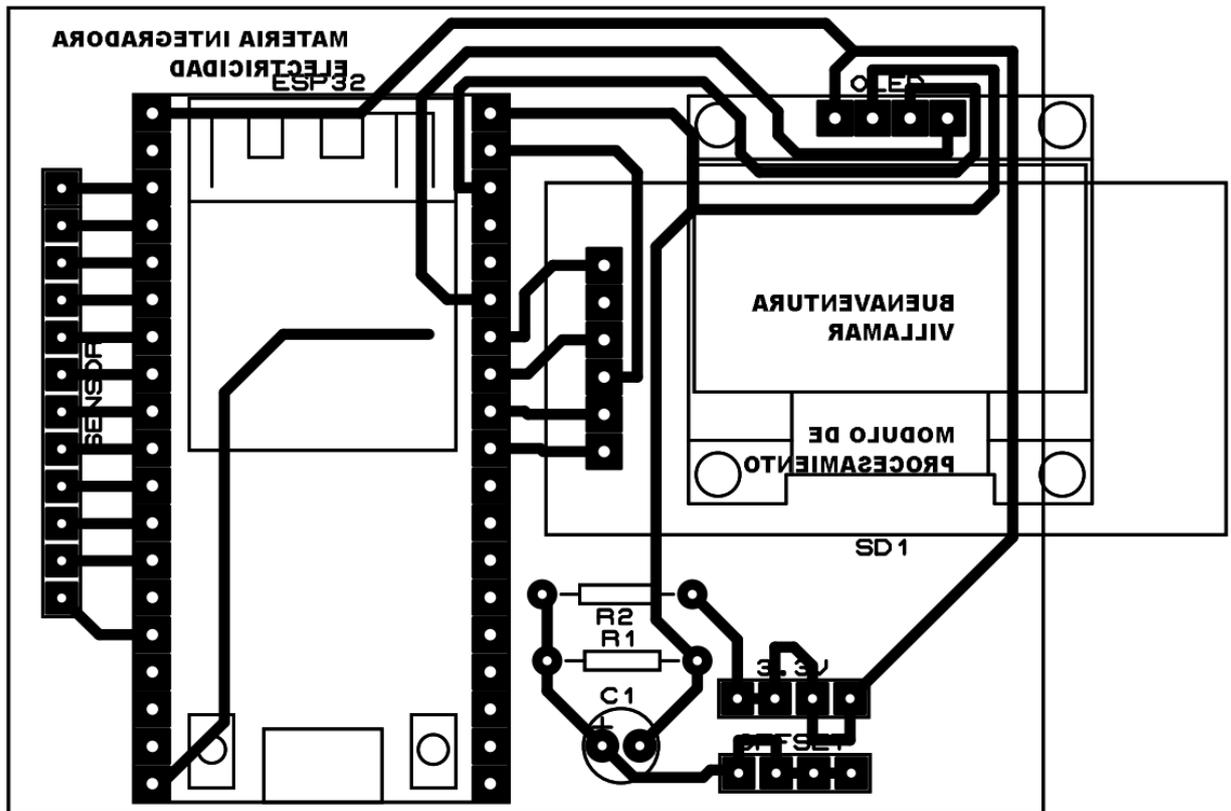
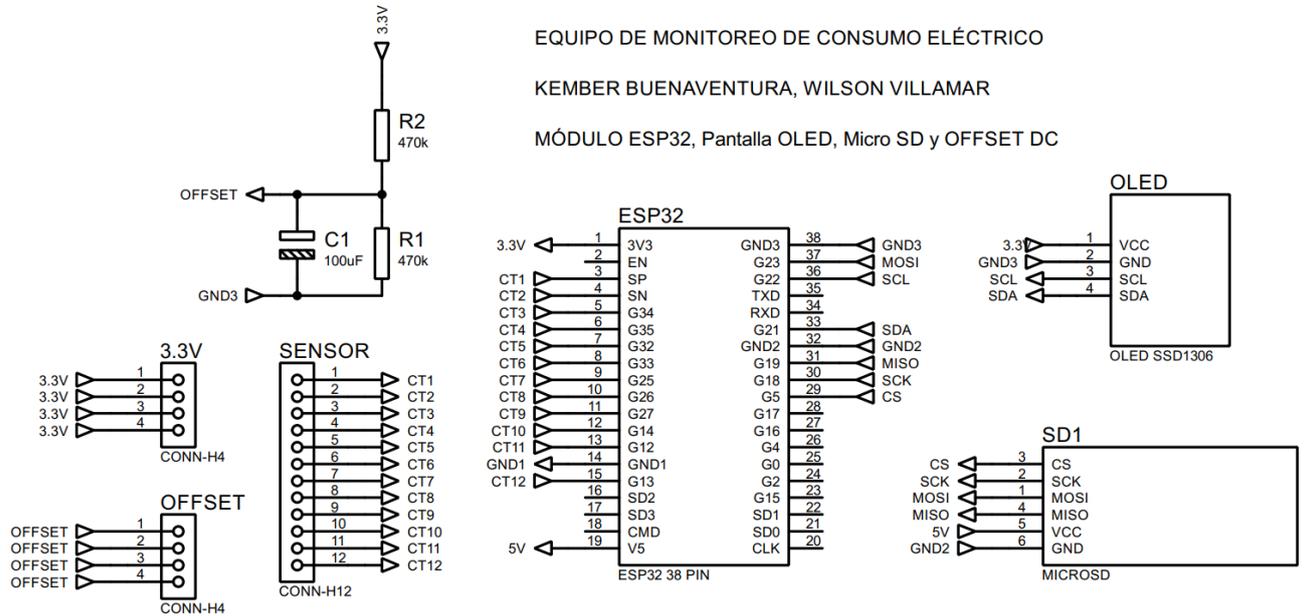


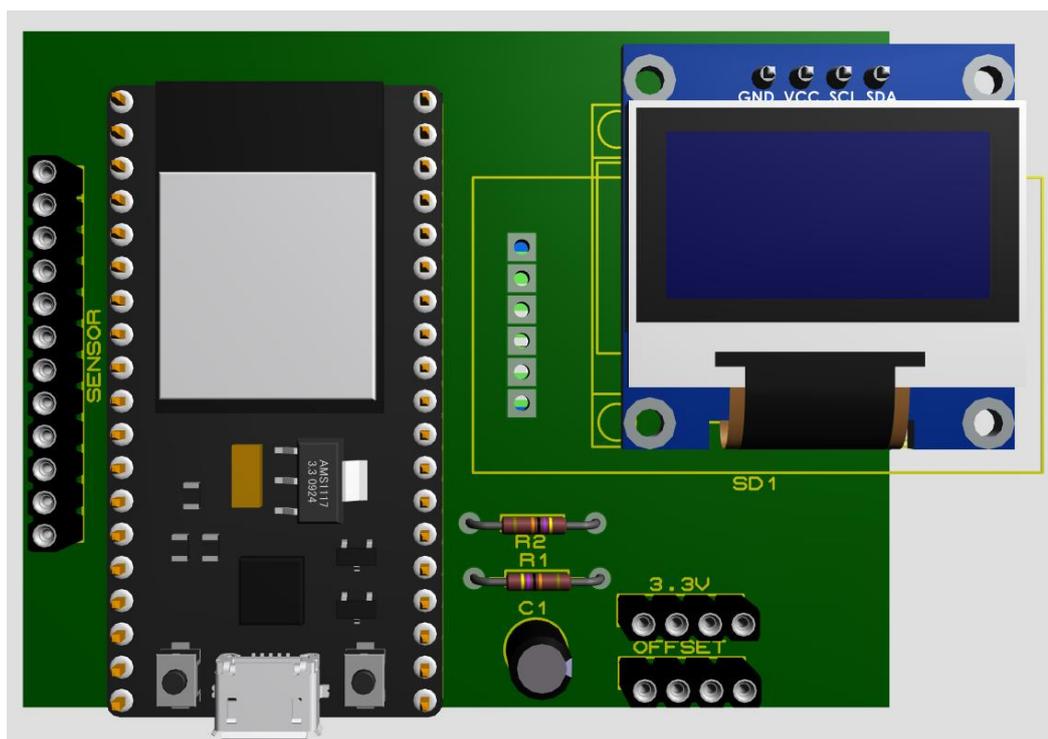
Módulo ESP32, Pantalla OLED, Micro SD y OFFSET DC

EQUIPO DE MONITOREO DE CONSUMO ELÉCTRICO

KEMBER BUENAVENTURA, WILSON VILLAMAR

MÓDULO ESP32, Pantalla OLED, Micro SD y OFFSET DC





APÉNDICE B

Código final del equipo de monitoreo

```
// Incluir librería WiFi y MQTT
#include <WiFi.h>
#include <PubSubClient.h>

// Colocar credenciales WiFi
const char* ssid = "NETLIFE-VILLAMAR";
const char* password = "0918998683";

// Colocar dirección IP del MQTT Broker
const char* mqtt_server = "192.168.100.74";
const int mqtt_port = 1883;
const char* topic = "casa/cuarto/medidor1";

// Crear cliente WiFi
WiFiClient espClient;
PubSubClient client(espClient);

// Incluir Librería Open Energy Monitor
#include "EmonLib.h"

// Crear instancia de medición EnergyMonitor
EnergyMonitor energyMonitor;

// Incluir Librería microSD
#include "FS.h"
#include "SD.h"
#include "SPI.h"
String datoSD;

// Crear funciones microSD
void listDir(fs::FS &fs, const char * dirname, uint8_t levels){
    Serial.printf("Lista de directorios: %s\n", dirname);

    File root = fs.open(dirname);
    if(!root){
        Serial.println("No se pudo abrir el directorio");
        return;
    }
```

```

}
if(!root.isDirectory()){
    Serial.println("No es un directorio");
    return;
}

File file = root.openNextFile();
while(file){
    if(file.isDirectory()){
        Serial.print(" DIR : ");
        Serial.println(file.name());
        if(levels){
            listDir(fs, file.name(), levels -1);
        }
    } else {
        Serial.print(" FILE: ");
        Serial.print(file.name());
        Serial.print(" SIZE: ");
        Serial.println(file.size());
    }
    file = root.openNextFile();
}
}

void createDir(fs::FS &fs, const char * path){
    Serial.printf("Creando Directorio: %s\n", path);
    if(fs.mkdir(path)){
        Serial.println("Directorio creado");
    } else {
        Serial.println("Fallo creación de directorio mkdir");
    }
}

void removeDir(fs::FS &fs, const char * path){
    Serial.printf("Eliminando Directorio: %s\n", path);
    if(fs.rmdir(path)){
        Serial.println("Directorio eliminado");
    } else {
        Serial.println("Fallo eliminación directorio");
    }
}

void readFile(fs::FS &fs, const char * path){
    Serial.printf("Leyendo archivo: %s\n", path);

    File file = fs.open(path);
    if(!file){

```

```

        Serial.println("No se pudo abrir el archivo para leer");
        return;
    }

    Serial.print("Leyendo desde archivo: ");
    while(file.available()){
        Serial.write(file.read());
    }
    file.close();
}

void writeFile(fs::FS &fs, const char * path, const char * message){
    Serial.printf("Escribiendo archivo: %s\n", path);

    File file = fs.open(path, FILE_WRITE);
    if(!file){
        Serial.println("No se pudo abrir el archivo para escribir");
        return;
    }
    if(file.print(message)){
        Serial.println("archivo escrito");
    } else {
        Serial.println("Escritura fallida");
    }
    file.close();
}

void appendFile(fs::FS &fs, const char * path, const char * message){
    Serial.printf("Anexando a archivo: %s\n", path);

    File file = fs.open(path, FILE_APPEND);
    if(!file){
        Serial.println("Error al abrir el archivo para anexar");
        return;
    }
    if(file.print(message)){
        Serial.println("Mensaje anexado");
    } else {
        Serial.println("Anexar fallido");
    }
    file.close();
}

void renameFile(fs::FS &fs, const char * path1, const char * path2){
    Serial.printf("Renombrando archivo %s to %s\n", path1, path2);
    if (fs.rename(path1, path2)) {
        Serial.println("Archivo renombrado");
    }
}

```

```

    } else {
        Serial.println("Cambio de nombre fallido");
    }
}
void deleteFile(fs::FS &fs, const char * path){
    Serial.printf("Eliminando archivo: %s\n", path);
    if(fs.remove(path)){
        Serial.println("Archivo eliminado");
    } else {
        Serial.println("Error al eliminar");
    }
}
void testFileIO(fs::FS &fs, const char * path){
    File file = fs.open(path);
    static uint8_t buf[512];
    size_t len = 0;
    uint32_t start = millis();
    uint32_t end = start;
    if(file){
        len = file.size();
        size_t flen = len;
        start = millis();
        while(len){
            size_t toRead = len;
            if(toRead > 512){
                toRead = 512;
            }
            file.read(buf, toRead);
            len -= toRead;
        }
        end = millis() - start;
        Serial.printf("%u bytes read for %u ms\n", flen, end);
        file.close();
    } else {
        Serial.println("No se pudo abrir el archivo para leer");
    }

    file = fs.open(path, FILE_WRITE);
    if(!file){
        Serial.println("No se pudo abrir el archivo para escribir");
        return;
    }

    size_t i;

```

```

start = millis();
for(i=0; i<2048; i++){
    file.write(buf, 512);
}
end = millis() - start;
Serial.printf("%u bytes written for %u ms\n", 2048 * 512, end);
file.close();
}

void initSDCard(){
    if (!SD.begin()) {
        Serial.println("Falló el montaje de la tarjeta");
        return;
    }
    uint8_t cardType = SD.cardType();

    if(cardType == CARD_NONE){
        Serial.println("Sin tarjeta SD adjunta");
        return;
    }
    Serial.print("Tipo de tarjeta SD: ");
    if(cardType == CARD_MMC){
        Serial.println("MMC");
    } else if(cardType == CARD_SD){
        Serial.println("SDSC");
    } else if(cardType == CARD_SDHC){
        Serial.println("SDHC");
    } else {
        Serial.println("DESCONOCIDO");
    }
    uint64_t cardSize = SD.cardSize() / (1024 * 1024);
    Serial.printf("Tamaño de la tarjeta SD: %lluMB\n", cardSize);
}

// Crear variables
unsigned long tiempo = millis();
String activo = "";
String VRMS = "";
String IRMS = "";
String PINST = "";
int segundos = 0;
int minutos = 0;
int horas = 0;
int dias = 0;
float periodo = 16.666667;

```

```

int muestreo = 100;
float kWh = 0;
float consumo = 0;

// Incluir Librerías de pantalla OLED
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

// Declarar pantalla SSD1306 conectada a I2C (pin SDA, pin SCL)
#define SCREEN_WIDTH 128 // ancho pantalla OLED en pixeles
#define SCREEN_HEIGHT 64 // altura pantalla OLED en pixeles
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, -1);

// Colocar logo BMP en formato ARRAY
const unsigned char logo[] = {
    // 'logo, 128x64px
    0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0f, 0xff, 0xff, 0xfc, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x1f, 0xff, 0xff, 0xfe, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3f, 0xff, 0xff, 0xff, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x78, 0x00, 0x00, 0x07, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xf0, 0x00, 0x00, 0x03, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xe0, 0x03, 0xff, 0x01, 0xc0, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xe0, 0x07, 0xff, 0x01, 0xc0, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xe0, 0x07, 0x0e, 0x01, 0xc0, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xe0, 0x07, 0x0e, 0x01, 0xc0, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xe0, 0x07, 0x1c, 0x01, 0xc0, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xe0, 0x0e, 0x1c, 0x01, 0xc0, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xe0, 0x1c, 0x1c, 0x01, 0xc0, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xe0, 0x1c, 0x38, 0x01, 0xc0, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xe0, 0x38, 0x38, 0x01, 0xc0, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xe0, 0x38, 0x7f, 0x81, 0xc0, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xe0, 0x38, 0x7f, 0x81, 0xc0, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xe0, 0x7e, 0x0f, 0x81, 0xc0, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xe0, 0x7e, 0x0f, 0x01, 0xc0, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xe0, 0x7f, 0x1e, 0x01, 0xc0, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xe0, 0x7e, 0x3c, 0x01, 0xc0, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xe0, 0x0e, 0x78, 0x01, 0xc0, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xe0, 0x1c, 0xf0, 0x01, 0xc0, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xe0, 0x1d, 0xe0, 0x01, 0xc0, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xe0, 0x1f, 0xc0, 0x01, 0xc0, 0x00, 0x00, 0x00, 0x00, 0x00,

```



```

    0x00, 0x00, 0x03, 0xc0, 0x78, 0x00, 0x00, 0x01, 0xc0, 0x70, 0x01, 0xff, 0x80, 0xe0, 0x70, 0x03,
    0xff, 0x80, 0xe0, 0x70, 0x03, 0x87, 0x00, 0xe0, 0x70, 0x03, 0x87, 0x00, 0xe0, 0x70, 0x03, 0x8e,
    0x00, 0xe0, 0x70, 0x07, 0x0e, 0x00, 0xe0, 0x70, 0x0e, 0x0e, 0x00, 0xe0, 0x70, 0x0e, 0x1c, 0x00,
    0xe0, 0x70, 0x1c, 0x3c, 0x00, 0xe0, 0x70, 0x1c, 0x3f, 0xc0, 0xe0, 0x70, 0x1c, 0x3f, 0xc0, 0xe0,
    0x70, 0x3f, 0x07, 0xc0, 0xe0, 0x70, 0x3f, 0x07, 0x80, 0xe0, 0x70, 0x3f, 0x8f, 0x00, 0xe0, 0x70,
    0x3f, 0x1e, 0x00, 0xe0, 0x70, 0x07, 0x3c, 0x00, 0xe0, 0x70, 0x0e, 0x78, 0x00, 0xe0, 0x70, 0x0e,
    0xf0, 0x00, 0xe0, 0x70, 0x0f, 0xe0, 0x00, 0xe0, 0x70, 0x1f, 0xc0, 0x00, 0xe0, 0x70, 0x1f, 0x80,
    0x00, 0xe0, 0x70, 0x1f, 0x00, 0x00, 0xe0, 0x70, 0x1e, 0x00, 0x00, 0xe0, 0x70, 0x3c, 0x00, 0x00,
    0xe0, 0x70, 0x38, 0x00, 0x00, 0xe0, 0x70, 0x30, 0x00, 0x00, 0xe0, 0x78, 0x00, 0x00, 0x01, 0xc0,
    0x3c, 0x00, 0x00, 0x03, 0xc0, 0x1f, 0xff, 0xff, 0xff, 0x80, 0x0f, 0xff, 0xff, 0xff, 0x00, 0x03,
    0xff, 0xff, 0xfc, 0x00
};

```

```

void setup()
{
  Serial.begin(115200);
  // Verificar conexión correcta de la pantalla SSD1306
  if(!display.begin(SSD1306_SWITCHCAPVCC, 0x3C)) { // Dirección 0x3D para OLED 128x64
    Serial.println(F("SSD1306 Alocación fallida"));
    for(;;);
  }

  // Visualización inicial pantalla
  display.clearDisplay();
  display.drawBitmap(0, 0, logo, 128, 64, 1);
  display.display();
  delay(4000);

  // Visualización inicial pantalla
  display.clearDisplay();
  display.drawBitmap(90, 14, icono, 36, 36, 1);
  display.drawRect(0, 0, 128, 12, WHITE);
  display.drawRect(0, 52, 128, 12, WHITE);
  display.setTextSize(1);
  display.setTextColor(WHITE);
  display.setCursor(5, 2);
  display.println("Monitoreo Energetico");
  display.setCursor(0, 15);
  display.println("Conexion SD");
  display.display();

  initSDCard();
  File file = SD.open("/mediciones.txt");
  if(!file) {

```

```

    Serial.println("Archivo no existe");
    Serial.println("Creando archivo...");
    writeFile(SD, "/mediciones.txt", "tiempo, Voltaje, corriente, potencia, energía, consumo \r\n");
}
else {
    Serial.println("El archivo ya existe");
}
file.close();

delay(1000);
display.setCursor(0, 24);
display.println("OK");
display.setCursor(0, 33);
display.println("Conexion MQTT");
display.display();

// Iniciar conexión WiFi
setup_wifi();
client.setServer(mqtt_server, mqtt_port);

display.setCursor(0, 42);
display.println("OK");
display.setCursor(5, 54);
display.println(String(topic));
display.display();
delay(1000);

// Definir entradas de voltaje y corriente Open Energy Monitor
energyMonitor.voltage(32, 106.8, 1.7); // (pin, calibración, diferencia de fase)
energyMonitor.current(33, 100); //(pin, calibración)
}

void setup_wifi() {
    delay(10);
    // Conectar a la red WiFi
    Serial.println();
    Serial.print("Conectando a ");
    Serial.println(ssid);
    WiFi.begin(ssid, password);
    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print(".");
    }
}

```

```

Serial.println("");
Serial.println("WiFi conectado");
Serial.println("Dirección IP: ");
Serial.println(WiFi.localIP());
}

void reconnect() {
  Serial.println("Conectando a MQTT Broker...");
  while (!client.connected()) {
    Serial.println("Reconectando a MQTT Broker..");
    String clientId = "Medidor1-";
    clientId += String(random(0xffff), HEX);

    if (client.connect(clientId.c_str())) {
      Serial.println("Conectado.");
    }
    client.publish(topic, "Medidor Conectado");
  }
}

void loop()
{
  if (!client.connected()) {
    reconnect();
  }
  client.loop();

  // Marcar el tiempo antes de medición
  tiempo = millis();

  // Calcular valores VI con función de Open Energy Monitor
  energyMonitor.calcVI(periodo, muestreo); // (longitud de onda, tiempo de medición ms)

  // Realizar Cálculos de consumo
  kWh = kWh + energyMonitor.apparentPower*(millis()-tiempo)/3600000000;
  consumo = kWh*0.09;

  int segundos = millis()/1000;
  int minutos = segundos/60;
  int horas = minutos/60;
  int dias = horas/24;

```

```

activo = String(dias%30) + "d " + String(horas%24) + "h " + String(minutos%60) + "m " + String(segundos%60) +
"s";
VRMS = String(energyMonitor.Vrms, 3);
IRMS = String(energyMonitor.Irms, 3);
PINST = String(energyMonitor.apparentPower, 3);

datoSD = activo + "," + VRMS + "," + IRMS + "," + PINST + "," + String(kWh, 3) + "," + String(consumo, 3) + "\r\n";
appendFile(SD, "/mediciones.txt", datoSD.c_str());
readFile(SD, "/mediciones.txt");

// Enviar datos por MQTT
char voltageString[10];
dtostrf(energyMonitor.Vrms, 11, 4, voltageString);
client.publish("casa/cuarto/medidor1/voltaje", voltageString);

char corrienteString[10];
dtostrf(energyMonitor.Irms, 11, 4, corrienteString);
client.publish("casa/cuarto/medidor1/corriente", corrienteString);

char potenciaString[10];
dtostrf(energyMonitor.apparentPower, 11, 4, potenciaString);
client.publish("casa/cuarto/medidor1/potencia", potenciaString);

char energiaString[10];
dtostrf(kWh, 11, 4, energiaString);
client.publish("casa/cuarto/medidor1/energia", energiaString);

char consumoString[10];
dtostrf(consumo, 11, 4, consumoString);
client.publish("casa/cuarto/medidor1/consumo", consumoString);

char activoString[10];
activo.toCharArray(activoString, 14);
client.publish("casa/cuarto/medidor1/activo", activoString);

// Mostrar datos en pantalla OLED
display.clearDisplay();
display.drawBitmap(90, 14, icono, 36, 36, 1);
display.drawRect(0, 0, 128, 12, WHITE);
display.drawRect(0, 52, 128, 12, WHITE);
display.setTextSize(1);
display.setTextColor(WHITE);
display.setCursor(5, 2);

```

```
display.println("Monitoreo Energetico");
display.setCursor(0, 15);
display.println("Vrms:" + VRMS + " V");
display.setCursor(0, 24);
display.println("Irms:" + IRMS + " A");
display.setCursor(0, 33);
display.println("Pnst:" + PINST + " W");
display.setCursor(0, 42);
display.println("Cons: $" + String(consumo, 3));
display.setCursor(5, 54);
display.println("Tiempo:" + activo);
display.display();
}
```