

Universidad Católica de Santa María

Facultad de Ciencias e Ingenierías Físicas y Formales

Escuela Profesional de Ingeniería Electrónica



DISEÑO E IMPLEMENTACIÓN DE 2 EQUIPOS MEDIDORES DE ENERGÍA ELÉCTRICA BASADOS EN IOT PARA GESTIÓN DE CONSUMO ENERGÉTICO

Tesis presentada por la bachiller:

Pinto Guillen, Vanessa Jimena

Para optar el título profesional de:

Ingeniero Electrónico

Con especialidad en:

Automatización y Control

Asesora:

Ing. Valdivieso Herrera, Diana

Arequipa – Perú

2022

UCSM-ERP

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

INGENIERIA ELECTRONICA

TITULACIÓN CON TESIS

DICTAMEN APROBACIÓN DE BORRADOR

Arequipa, 18 de Abril del 2022

Dictamen: 003027-C-EPIE-2022

Visto el borrador del expediente 003027, presentado por:

2014221862 - PINTO GUILLEN VANESSA JIMENA

Titulado:

DISEÑO E IMPLEMENTACIÓN DE 2 EQUIPOS MEDIDORES DE ENERGÍA ELÉCTRICA BASADOS EN IOT PARA GESTIÓN DE CONSUMO ENERGÉTICO

Nuestro dictamen es:

APROBADO

**1475 - MALAGA CHAVEZ CESAR EDUARDO
DICTAMINADOR**



**1767 - SULLA TORRES RAUL RICARDO
DICTAMINADOR**



**1843 - URRUTIA ESPINOZA MARIO WILLIAM
DICTAMINADOR**



DEDICATORIA

La presente tesis va dedicada a Dios por encaminar mi vida por el bien, a mis padres Jaime y Vannia por su apoyo incondicional en esta etapa de mi vida, a mi hermanita Danika que por ella es que aun sigo luchando por mis metas, a mis abuelos que son como mis padres Feliberto y Vilma que gracias a su apoyo y confianza pude concluir mis objetivos como persona y como estudiante y a mi bebe que viene en camino, cada meta que llegue a cumplir es solo para ella.

¡Muchas gracias, este logro es gracias a ustedes!



AGRADECIMIENTO

Agradezco a Dios, por bendecir mi vida, por guiarme a lo largo de mi existencia, por ser el apoyo y fortaleza en aquellos momentos de dificultad y debilidad.

A mis padres Jaime y Vannia que han sabido darme su ejemplo de trabajo y honradez, por los consejos y principios que me han inculcado.

A mis abuelos Feliberto y Vilma por ser los principales promotores de mis sueños, por confiar y creer en mis expectativas.

A mi hermanita Danika por confiar en mí siempre, por ser mi motor y motivo para seguir luchando.

A mi bebe que viene en camino, porque tú eres la que me impulsa a seguir cumpliendo con mis metas.

RESUMEN

En el presente trabajo de investigación se muestra el diseño y la aplicación de un equipo medidor de energía eléctrica basados en Iot para gestión de consumo energético.

Este diseño cuenta con una medición exacta y precisa de la señal de voltaje, corriente y frecuencia.

También se aplican algoritmos de Procesamiento Digital de Señales para el cálculo del ángulo de fase entre el voltaje y la corriente, cálculo de potencia activa y aparente, todos estos cálculos se encuentran programados en un procesador CORTEX – ARM de la familia Texas Instrumets.

El equipo también cuenta con un módulo de comunicación WIFI que le permite transmitir las mediciones en tiempo real por medio de mensajería MQTT, siendo el software Node Red instalado en una laptop o servidor el administrador de mensajería y el que se encarga de llenar los datos en la base de datos.

El equipo desarrolla las siguientes tareas:

- Lectura real del consumo energético en las viviendas.
- Conexión a la red WIFI de las viviendas para su próxima lectura.
- Visualización de los valores medidos en una pantalla LCD mediante la red WIFI de las viviendas.
- Visualización de dashboards creados en Node Red en un celular por el usuario.

Además, el equipo también cuenta con una memoria interna donde se almacenará toda la información en una base de datos de un servidor (Laptop); este, permitirá al usuario comunicar el equipo por medio de la red WIFI para tener conocimiento de su consumo energético en su vivienda visualizando dicho consumo en su celular, el circuito fue diseñado utilizando el software EAGLE Cadsoft.

PALABRAS CLAVE:

IoT, dashboards, Node Red.

ABSTRACT

In this research work the design and application of an Iot-based electrical energy meter equipment for energy consumption management is shown.

This design features accurate and precise measurement of the voltage, current, and frequency signal.

Digital Signal Processing algorithms are also applied to calculate the phase angle between voltage and current, active and apparent power calculation, all these calculations are programmed in a CORTEX - ARM processor from the Texas Instruments family.

The equipment also has a WIFI communication module that allows it to transmit measurements in real time through MQTT messaging, with the Node Red software installed on a laptop or server being the messaging administrator and the one in charge of filling in the data in the database.

The team carries out the following tasks:

- Real reading of energy consumption in homes.
- Connection to the WIFI network of homes for your next reading.
- Visualization of the measured values on an LCD screen through the home WIFI network.
- Viewing dashboards created in Node Red on a cell phone by the user.

In addition, the equipment also has an internal memory where all the information will be stored in a database on a server (Laptop); This will allow the user to communicate the equipment through the WIFI network to have knowledge of their energy consumption in their home by viewing said consumption on their cell phone, the circuit was designed using the EAGLE Cadsoft software.

KEYWORDS:

IoT, dashboards, Node Red.

ÍNDICE GENERAL

DEDICATORIA	iii
AGRADECIMIENTO.....	iv
RESUMEN	v
ABSTRACT	vi
NOMENCLATURAS.....	xx
CAPÍTULO I	
INTRODUCCIÓN.....	1
1. IDENTIFICACIÓN DEL PROBLEMA.....	2
2. DESCRIPCIÓN DEL PROBLEMA	3
3. JUSTIFICACIÓN DEL PROBLEMA.....	3
4. OBJETIVOS.....	4
4.1 GENERALES	4
4.2 ESPECÍFICOS.....	4
5. APORTES.....	5
6. ESTRUCTURA DE TESIS	5
CAPÍTULO II.....	7
MARCO TEÓRICO.....	7
1. PRINCIPIO DE FUNCIONAMIENTO.....	8
2. ANTECEDENTES	9
3. ESPECIFICACIONES TÉCNICAS DEL EQUIPO.....	10
4. BASES TEÓRICAS	11
4.1 MICRONCOTROLADOR RM46L852	11
4.2 CÁLCULOS DE POTENCIA	14
4.3 TEORÍA SOBRE IOT	21
4.4 APLICACIONES GENERALES	30
4.5 FUENTE DE ALIMENTACIÓN	30
4.6 ETAPA ANÁLOGA.....	31

CAPÍTULO III

DISEÑO DE HARDWARE	36
1. DIAGRAMA DE BLOQUES GENERAL.....	37
2. DIAGRAMA DE BLOQUES DEL EQUIPO	38
3. DIAGRAMA DE BLOQUES DE LA ARQUITECTURA DE RED	40
4. ELEMENTOS DE APLICACIÓN GENERAL	41
4.1 CIRCUITO DE CONTROL OFFSET	41
4.2 SELECCIÓN DEL CIRCUITO AMPLIFICADOR.....	41
4.3 CARACTERÍSTICAS DE OPAMP IMPLEMENTADO	42
5. DISEÑO DE LA ETAPA ANÁLOGA DEL MEDIDOR DE ENERGÍA	43
5.1 SENSOR DE VOLTAJE	43
5.2 SENSOR DE CORRIENTE	60
5.3 SENSOR DE FRECUENCIA.....	65
6. DISEÑO DE LA ETAPA DIGITAL	69
6.1 RELOJ EN TIEMPO REAL DS1302.....	69
6.2 MEMORIA EEPROM 25LC256.....	72
6.3 ESQUEMA GENERAL DE LA ETAPA DIGITAL.....	75
7. DISEÑO DE LA ETAPA DE VISUALIZACION DE DATOS E INTERFAZ.....	75
7.1 BOTONERAS DE CONTROL	75
7.2 PANTALLA LCD SERIAL	77

CAPÍTULO IV

DISEÑO DE SOFTWARE	79
1. HALCOGEN.....	80
2. CODE COMPOSER STUDIO	80
3. PERIFÉRICOS DEL CONTROLADOR	81
3.1 CONVERTIDOR ANÁLOGO DIGITAL	81
3.2 MÓDULO TEMPORIZADOR DE ALTA PRECISIÓN GPIO (GIO).....	88
3.3 ALGORITMOS DE ANALISIS MATEMATICOS	88
3.4 MÓDULOS DE COMUNICACIÓN SERIAL (SCI).....	94
3.5 MÓDULOS DE COMUNICACIÓN SPI MEMORIA EEPROM	94
3.6 RELOJ EN TIEMPO REAL	95
4. FUNCIONES MATEMÁTICAS DE ANÁLISIS	95
4.1 CÁLCULO DEL VALOR RMS.....	95

4.2	CÁLCULO DE LAS POTENCIAS ACTIVA Y REACTIVA Y APARENTE	96
4.3	BÚSQUEDA DEL CRUCE POR CERO	98
4.4	CÁLCULO DEL FACTOR DE POTENCIA	99
5.	ESPECIFICACIONES TÉCNICAS DEL SERVIDOR.....	100
6.	CONFIGURACIÓN DEL IP DEL SERVIDOR	102
7.	NODE RED	103
7.1	NODOS DE GAUGE Y CHART	111
7.2	AEDES BROKER	112
7.3	BASE DE DATOS XAMPP	114
8.	DIAGRAMAS DE FLUJO DEL SISTEMA OPERATIVO.....	122
8.1	DIAGRAMA DE FLUJO DEL PROGRAMA PRINCIPAL	122
8.2	DIAGRAMA DE FLUJO DEL MÓDULO WIFI ESP32	125
8.3	DIAGRAMA DE FLUJO DEL MOTOR DE CÁLCULO.....	127
8.4	DIAGRAMA DE FLUJO DE NODE RED	131

CAPÍTULO V

IMPLEMENTACIÓN DEL EQUIPO	133
1. ETAPA ANÁLOGA	134
1.1 CIRCUITO DEL SENSOR DE VOLTAJE.....	134
1.2 CIRCUITO DEL SENSOR DE CORRIENTE	135
1.3 CIRCUITO DEL SENSOR DE FRECUENCIA	136
2. ETAPA DIGITAL	137
2.1 CIRCUITO DE RELOJ	137
2.2 CIRCUITO DE LA MEMORIA EEPROM 25LC256	137
3. ETAPA DE VISUALIZACIÓN DE DATOS E INTERFAZ	138
3.1 CIRCUITO PARA BOTONERAS DE CONTROL.....	138
3.2 CIRCUITO PARA LA PANTALLA LCD SERIAL	138
4. EQUIPOS MEDIDORES DE ENERGÍA ELÉCTRICA	139
5. INSTALACIÓN Y CONFIGURACIÓN DEL EQUIPO MEDIDOR DE ENERGÍA	140

CAPÍTULO VI

PRUEBAS Y RESULTADOS	144
1. CONEXIÓN DE LOS EQUIPOS A LA RED WIFI DE LAS VIVIENDAS	145

2. MEDICIÓN DE VOLTAJE Y CORRIENTE ELÉCTRICA	146
3. MEDICIÓN DE FRECUENCIA ELÉCTRICA	149
4. MEDICIÓN DE POTENCIA ELECTRICA (KW)	151
5. MEDICIÓN DE POTENCIA ELÉCTRICA (KWH)	153
6. ANÁLISIS ECONÓMICO	155

CAPÍTULO VII

CONCLUSIONES Y RECOMENDACIONES	159
CONCLUSIONES	160
RECOMENDACIONES	162
REFERENCIAS	163
ANEXOS	166
ANEXO A: PROGRAMA DE MENÚ GENERAL	166
ANEXO B: PROGRAMA DE MOTOR ELÉCTRICO.....	179
ANEXO C: PROGRAMA DE RELOJ EN TIEMPO REAL.....	186
ANEXO D: PROGRAMA DE MÓDULO WIFI ESP32.....	193
ANEXO E: PROGRAMA DE MOTOR DE CÁLCULO EN NODE RED	200
ANEXO F: PROGRAMA DE BASE DE DATOS EN NODE RED	202
ANEXO G: NORMA IEC 62053-21:2020 (Equipos de medida de electricidad. Requisitos particulares. Parte 21: Contadores estáticos de energía activa CA (clases 0,5, 1 y 2).....	203
ANEXO H: NORMA IEC 62052-11:2020 (Equipos de medida de electricidad. Requisitos generales, ensayos y condiciones de ensayo. Parte 11)	204
ANEXO I: DISEÑO DE CIRCUITO Y PLACA IMPRESA	205

ÍNDICE DE FIGURAS

<i>Figura 1.1</i> Gráfica de potencia generada durante pandemia.....	2
<i>Figura 2.1</i> Arquitectura del Bróker MQTT.....	8
<i>Figura 2.2</i> Dashboards de resultados de medición.....	9
<i>Figura 2.3</i> Diagrama fasorial de la potencia instantánea	18
<i>Figura 2.4</i> Diagrama fasorial de la potencia activa o media.....	19
<i>Figura 2.5</i> Diagrama fasorial de la potencia reactiva	20
<i>Figura 2.6</i> Representación de nodos	24
<i>Figura 2.7</i> Representación del nodo MQTT in	26
<i>Figura 2.8</i> Representación del nodo MQTT out	26
<i>Figura 2.9</i> Representación del nodo function	27
<i>Figura 2.10</i> Representación del nodo chart.....	27
<i>Figura 2.11</i> Representación del nodo gauge	28
<i>Figura 2.12</i> Representación del nodo MySQL	30
<i>Figura 2.13</i> Filtro pasa bajo RC.....	32
<i>Figura 2.14</i> OPAMP comparador de voltaje	34
<i>Figura 3.1</i> Diagrama de bloques general	37
<i>Figura 3.2</i> Diagrama de bloques de medición de energía.....	38
<i>Figura 3.3</i> Diagrama de bloques de la arquitectura de red sin máquina virtual.....	40
<i>Figura 3.4</i> Circuito offset.....	41
<i>Figura 3.5</i> Diagrama de bloques del circuito sensor de voltaje	44

<i>Figura 3.6</i> Relación de transformación en un transformador monofásico.....	45
<i>Figura 3.7</i> relación ente voltaje AC y entrada ADC.....	47
<i>Figura 3.8</i> Amplificador diferencial	48
<i>Figura 3.9</i> Onda del amplificador diferencial.....	50
<i>Figura 3.10</i> Divisor resistivo de varias impedancias	51
<i>Figura 3.11</i> Circuito de un filtro pasa bajo	57
<i>Figura 3.12</i> Circuito general del sensor de voltaje	59
<i>Figura 3.13</i> Diagrama de bloques del sensor de corriente.....	60
<i>Figura 3.14</i> Onda que ingresa al sensor de corriente.....	61
<i>Figura 3.15</i> Señal de salida del sensor.....	61
<i>Figura 3.16</i> Sensor de corriente ACS 756 – 50 A	64
<i>Figura 3.17</i> Circuito general del sensor de corriente.....	65
<i>Figura 3.18</i> Diagrama de bloques del circuito sensor de frecuencia	65
<i>Figura 3.19</i> Comparador de voltaje	66
<i>Figura 3.20</i> Comparador LM311	67
<i>Figura 3.21</i> Circuito Schmitt Trigger	68
<i>Figura 3.22</i> Circuito sensor de frecuencia	69
<i>Figura 3.23</i> Diagrama de bloques interno del reloj en tiempo real.....	70
<i>Figura 3.24</i> Circuito general del reloj en tiempo real	72
<i>Figura 3.25</i> Diagrama de bloques interno de la memoria EEPROM 25LC256.....	73
<i>Figura 3.26</i> Estructura general del protocolo SPI.....	74

<i>Figura 3.27</i> Circuito de la memoria EEPROM 25LC256.....	74
<i>Figura 3.28</i> Esquema general de la etapa digital	75
<i>Figura 3.29</i> Esquema del circuito de pulsadores	76
<i>Figura 3.30</i> Diagrama de bloques de LCD 16 x 2	77
<i>Figura 3.31</i> Circuito de la etapa de visualización	78
<i>Figura 4.1</i> Diagrama de bloques de RM46L852.....	80
<i>Figura 4.2</i> Estructura principal de Code Composer Studio	81
<i>Figura 4.3</i> Distribuciones de Canales ADC.....	82
<i>Figura 4.4</i> Diagrama de bloques específico de la RM46L852.....	83
<i>Figura 4.5</i> Configuración del ADC1.....	83
<i>Figura 4.6</i> Transformada de Fourier Radix 4	90
<i>Figura 4.7</i> Diagrama de configuración de capturado temporal.....	92
<i>Figura 4.8</i> Diagrama de bloques del RTI general	93
<i>Figura 4.9</i> Diagrama de bloques del contador 0	93
<i>Figura 4.10</i> Diagrama de configuración como I/O digital.....	94
<i>Figura 4.11</i> Representación de la potencia en 1 ciclo.....	96
<i>Figura 4.12</i> Representación del periodo T	97
<i>Figura 4.13</i> Muestreo de periodo en diferentes ciclos	98
<i>Figura 4.14</i> Detector de cruce por cero no inversor cuando E_i esta arriba de V_{ref} , $V_o = +V_{sat}$	99
<i>Figura 4.15</i> Factor de potencia en cada carga.....	99
<i>Figura 4.16</i> Tamaño de almacenamiento en la base de datos.....	100

<i>Figura 4.17</i> IP permanente del servidor de datos.....	102
<i>Figura 4.18</i> Comando “ipconfig”.....	103
<i>Figura 4.19</i> Motor de Node Red	104
<i>Figura 4.20</i> Ejecución del comando ofrecido por el fabricante	105
<i>Figura 4.21</i> Verificación de instalación exitosa de Node Red.....	105
<i>Figura 4.22</i> Ejecución del comando Node Red	106
<i>Figura 4.23</i> Entorno de desarrollo de Node Red.....	107
<i>Figura 4.24</i> Configuración de Node Red	107
<i>Figura 4.25</i> Librerías Anexas instaladas	108
<i>Figura 4.26</i> Instalación de la base de datos en Node Red.....	108
<i>Figura 4.27</i> Nodo de base de datos en Node Red	109
<i>Figura 4.28</i> Configuración de Node Red	109
<i>Figura 4.29</i> Instalación de Dashboards en Node Red	110
<i>Figura 4.30</i> Nodos de Dashboards en Node Red	110
<i>Figura 4.31</i> Visualización de Dashboards	111
<i>Figura 4.32</i> Configuración de Node Red	112
<i>Figura 4.33</i> Instalación de Bróker Aedes en Node Red.....	113
<i>Figura 4.34</i> Nodo de Bróker Aedes en Node Red	113
<i>Figura 4.35</i> Configuración del bróker MQTT Aedes	114
<i>Figura 4.36</i> Mensaje de bienvenida de XAMPP.....	115
<i>Figura 4.37</i> Selección de componentes de XAMPP	115

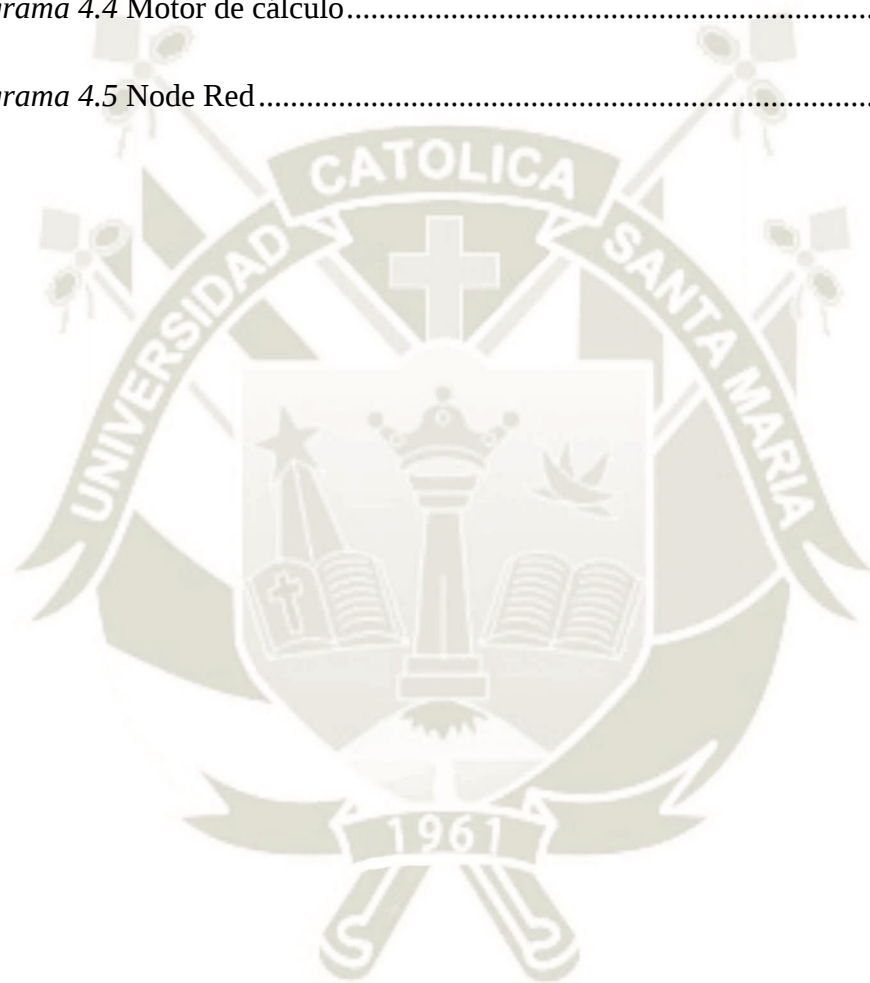
<i>Figura 4.38</i> Selección de la carpeta donde se instalará XAMPP	116
<i>Figura 4.39</i> Mensaje de Bitnami para XAMPP	116
<i>Figura 4.40</i> Instalación de XAMPP	117
<i>Figura 4.41</i> Completando el asistente de configuración de XAMPP	117
<i>Figura 4.42</i> Panel de control principal de XAMPP	118
<i>Figura 4.43</i> Panel de control principal de XAMPP	118
<i>Figura 4.44</i> Pagina web de XAMPP	119
<i>Figura 4.45</i> Creación de la base de datos en XAMPP	119
<i>Figura 4.46</i> Base de datos creada exitosamente.....	120
<i>Figura 4.47</i> Conexión de la base de datos en el entorno de Node Red.....	120
<i>Figura 4.48</i> Conexión exitosa de la base de datos en Node Red	121
<i>Figura 4.49</i> Creación de tablas en la web de XAMPP.....	121
<i>Figura 4.50</i> Visualización de la primera pantalla	122
<i>Figura 4.51</i> Visualización de la segunda pantalla.....	122
<i>Figura 4.52</i> Visualización de la tercera pantalla.....	123
<i>Figura 5.1</i> Circuito equipo sensor de voltaje armado en placa	134
<i>Figura 5.2</i> Circuito equipo sensor de corriente ACS750-50 armado en placa	135
<i>Figura 5.3</i> Señal de salida del sensor de corriente	135
<i>Figura 5.4</i> Cálculo del valor RMS	136
<i>Figura 5.5</i> Circuito equipo sensor de corriente armado en placa.....	136
<i>Figura 5.6</i> Circuito de reloj DS1302.....	137

<i>Figura 5.7</i> Circuito de memoria EEPROM 25LC256.....	137
<i>Figura 5.8</i> Circuito para botoneras de control	138
<i>Figura 5.9</i> Circuito para pantalla LCD	138
<i>Figura 5.10</i> Equipo medidor de energía eléctrica	139
<i>Figura 5.11</i> Tablero eléctrico principal de una vivienda	140
<i>Figura 5.12</i> Llave principal del tablero.....	141
<i>Figura 5.13</i> Cables a desconectar	141
<i>Figura 5.14</i> Cables del equipo conectados a la llave principal	142
<i>Figura 5.15</i> Equipo midiendo correctamente.....	143
<i>Figura 6.1</i> Reconocimiento de IP del equipo.....	145
<i>Figura 6.2</i> Reconocimiento de IP del servidor.....	145
<i>Figura 6.3</i> Conexión exitosa a la red WIFI.....	146
<i>Figura 6.4</i> Título de medición eléctrica	146
<i>Figura 6.5</i> Medición de voltaje y corriente AC equipo 1	147
<i>Figura 6.6</i> Medición de voltaje y corriente AC equipo 2	147
<i>Figura 6.7</i> Dashboard de voltaje y corriente AC equipo 1	148
<i>Figura 6.8</i> Dashboard de voltaje y corriente AC equipo 2	148
<i>Figura 6.9</i> Título de medición de frecuencia	149
<i>Figura 6.10</i> Medición de frecuencia equipo 1	149
<i>Figura 6.11</i> Medición de frecuencia equipo 2	150
<i>Figura 6.12</i> Dashboard de frecuencia equipo 1	150

<i>Figura 6.13</i> Dashboard de frecuencia equipo 2	151
<i>Figura 6.14</i> Título de potencia eléctrica	151
<i>Figura 6.15</i> Cálculo de potencia activa y potencia aparente equipo 1.....	152
<i>Figura 6.16</i> Cálculo de potencia activa y potencia aparente equipo 2.....	152
<i>Figura 6.17</i> Dashboard de potencia eléctrica.....	152
<i>Figura 6.18</i> Medición de energía eléctrica en una vivienda	153
<i>Figura 6.19</i> Dashboard de medición de energía eléctrica	153
<i>Figura 6.20</i> Dashboard general	154
<i>Figura 6.21</i> Datos recibidos en Node Red	154

ÍNDICE DE FLUJOGRAMAS

<i>Flujograma 4.1</i> Sistema de control general	124
<i>Flujograma 4.2</i> Módulo ESP32	126
<i>Flujograma 4.3</i> Muestreo por interrupciones.....	128
<i>Flujograma 4.4</i> Motor de cálculo.....	130
<i>Flujograma 4.5</i> Node Red.....	132



ÍNDICE DE TABLAS

<i>Tabla 2.1</i> Mediciones eléctricas elementales	10
<i>Tabla 2.2</i> Mediciones eléctricas derivadas.....	10
<i>Tabla 2.3</i> Tipo de comunicación	10
<i>Tabla 2.4</i> Almacenamiento y tiempo	11
<i>Tabla 2.5</i> Procesador	11
<i>Tabla 2.6</i> Software de programación	11
<i>Tabla 2.7</i> Fuente de alimentación	11
<i>Tabla 3.1</i> Selección de opamp.....	42
<i>Tabla 4.1</i> Resolución de Voltaje y corriente	86
<i>Tabla 4.2</i> Resumen de las características de la transformada de Fourier en N puntos ..	91
<i>Tabla 4.3</i> Periodo para cada etapa.....	98
<i>Tabla 6.1</i> Análisis económico general de los componentes.....	157
<i>Tabla 6.2</i> Análisis económico de los equipos	157
<i>Tabla 6.3</i> Análisis económico total de los equipos medidores de energía	158

NOMENCLATURAS

XAMPP	:	X (para cualquiera de los diferentes sistemas operativos), Apache, MariaDB/MySQL, PHP, Perl
ESP32	:	Código del microcontrolador WIFI a utilizar
CMS	:	Content Management System (Sistema de Gestión de Contenidos)
MQTT	:	Message Queue Telemetry Transport (Transporte de Telemetría de Cola de Mensajes)
MySQL	:	My Structured Query Language (Mi Lenguaje de Consulta Estructurado)
CPU	:	Central Processing Unit (Unidad Central de Procesamiento)
GPIO	:	General Purpose Input / Output (Entrada/Salida de Uso General)
UART	:	Universal Asynchronous Receiver / Transmitter (Receptor/Transmisor Asíncrono Universal)
RMS	:	Root Medium Square (Raíz Media Cuadrada)
IoT	:	Internet of Things (Internet de las Cosas)
M2M	:	Machine – to – Machine (Máquina a Máquina)
QoS	:	Quality of Service (Calidad de Servicio)
TCP	:	Transmission Control Protocol (Protocolo de Control de Transmisión)
UDP	:	User Datagram Protocol (Protocolo de Datagramas de Usuario)
PHP	:	Hypertext Pre – Processor (Procesador de Hipertexto)
Perl	:	Practical Extraction and Report Language (Lenguaje Practico de Extracción e Informe)
ADC	:	Analog – to – Digital Converter (Convertidor Análogo/Digital)
RAM	:	Random Access Memory (Memoria de Acceso Aleatorio)



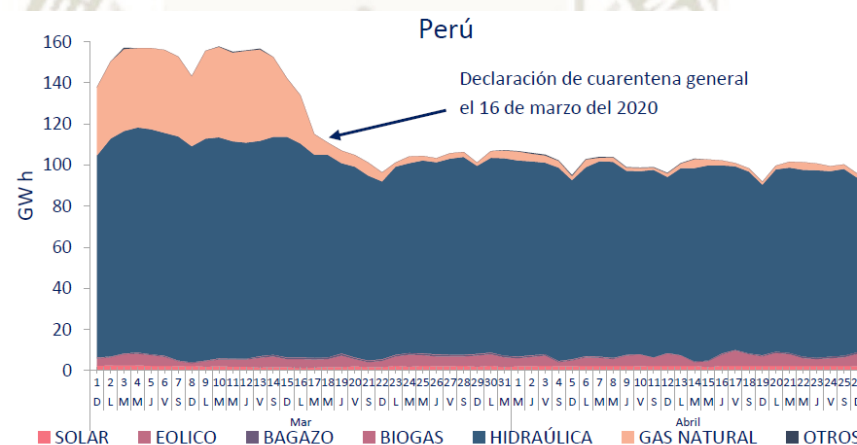
CAPÍTULO I

INTRODUCCIÓN

1. IDENTIFICACIÓN DEL PROBLEMA

La pandemia del COVID-19 desatada a inicios del año 2020, obligó a casi todos los países del mundo a tomar medidas de emergencia sanitaria, incluyendo el aislamiento social, lo que llevó a una paralización de gran parte de sus sistemas productivos, con el consecuente impacto sobre los ingresos nacionales. (OLADE, 2020)

Así mismo la pandemia ocasiono que la producción de energía eléctrica en Perú se reduzca en promedio un 32% y con una caída de demanda máxima con un 36%, desde la declaración de cuarentena general del pasado 16 de marzo del 2020. (OLADE, 2020)



Fuente: Comité de Operaciones Económica del Sistema Interconectado Nacional.

Figura 1.1 Gráfica de potencia generada durante pandemia

Fuente: Elaboración propia

Por consiguiente, el gobierno peruano publicó un Decreto de Urgencia 035-2020 y su modificatoria declaró la continuidad de la prestación de los servicios públicos básicos de energía eléctrica para la población vulnerable, fraccionando el pago de recibos pendientes, así mismo se suspendió la toma de lectura en los medidores y las empresas fueron autorizadas a emitir recibos de consumo de electricidad utilizando un sistema de promedios, teniendo como consecuencia algunos errores como cobros excesivos,

lo que llevo a las personas a reunirse en los centro de atención al público, exponiéndose al contagio del virus. (Presidente de la Republica Martin Vizcarra, 2020)

2. DESCRIPCIÓN DEL PROBLEMA

Las empresas distribuidoras de energía debido a que se rigen al Decreto de Urgencia 035-2020, seguirán fraccionando los cobros por lo que se podrían realizar cobros excesivos por no ser lecturas reales.

Debido al Estado de Emergencia los ciudadanos se encuentran en cuarentena, por lo que vendrán realizando labores desde sus viviendas lo que genera un mayor consumo de kW.h.

Al mismo tiempo al estar más tiempo en casa es común utilizar los artefactos eléctricos con mayor frecuencia, esto podría ocasionar que la facturación del recibo se incremente por el cambio de hábitos de uso de energía eléctrica al no poder salir por la cuarentena.

3. JUSTIFICACIÓN DEL PROBLEMA

Las empresas prestadoras de servicios básicos actualmente se rigen bajo los Decretos de Urgencia del Gobierno del Perú por ende la idea de construir un equipo que sea preciso realizando mediciones eléctricas y cuenta con comunicación inalámbrica es de mucha ayuda durante la suspensión de toma de lectura en los medidores.

Por lo que es importante diseñar e implementar un medidor de energía eléctrica accesible para el ciudadano que cuente con comunicación inalámbrica a internet y pueda subir las mediciones de voltaje, corriente, frecuencia y las potencias a la nube,

lo que permite una mejor administración de datos sin la necesidad de recurrir a un personal para la toma de lectura de medidores de energía eléctrica.

Al contar con la información de consumo energético en la nube, Osinergmin, las empresas distribuidoras de energía y el consumidor; pueden acceder a esta información permitiendo una correcta facturación de energía.

4. OBJETIVOS

4.1 GENERALES

Diseñar e implementar 2 equipos medidores de energía eléctrica los cuales realizarán mediciones de voltaje, corriente y frecuencia y enviarán su información de forma inalámbrica (WIFI) a una base de datos en un servidor con la finalidad de generar reportes a los usuarios que les permitirá gestionar mejor su consumo eléctrico.

Contar con datos reales para actuar con pruebas frente a un cobro excesivo, debido a la suspensión de toma de lectura de medidores.

4.2 ESPECÍFICOS

- Realizar la medición de voltaje y corriente eléctrica alterna respetando los estándares internacionales IEC 62053-21:2020 (Equipos de medida de electricidad. Requisitos particulares. Parte 21: Contadores estáticos de energía activa CA (clases 0,5, 1 y 2) y el IEC 62052-11:2020 (Equipos de medida de electricidad, 2020).
- Realizar la implementación de una comunicación WIFI entre el equipo y la Red inalámbrica del domicilio.
- Establecer un protocolo de comunicación MQTT (Message Queue Telemetry Transport), con un servidor de datos.

- Establecer una base de datos del tipo MySQL (My Structured Query Language) en el servidor de datos.
- Establecer una conexión entre el Servidor de mensajería MQTT y la base de datos para generar reportes y graficas del consumo de energía.

5. APORTES

Los aportes al desarrollo de la ingeniería nacional brindadas en la tesis presente son las siguientes:

- Un equipo prototipo de bajo costo comparado con los existentes en el mercado.
- Metodología para la aplicación de Procesamiento Digital de Señales.
 - Algoritmos para la detección de Transitorios
 - Aplicación de la FFT, análisis de magnitud y fase.
 - Análisis en el dominio de frecuencia.

6. ESTRUCTURA DE TESIS

Esta tesis está conformada por 7 capítulos organizados de la siguiente forma:

CAPÍTULO 1

En este capítulo se da una introducción a los diferentes tipos de problemas que se pueden identificar en una red eléctrica, su descripción, también se presenta la justificación, objetivos y aportes de la tesis.

CAPÍTULO 2

En este capítulo se da a conocer sobre lo que es un medidor de energía eléctrica y su funcionamiento, también se expone las diferentes características técnicas del equipo.

CAPÍTULO 3

En este capítulo se presenta el esquema circuital, dando a conocer la disposición de los circuitos, especificaciones, funcionamiento, selección y cálculo respectivo para su diseño.

CAPÍTULO 4

En este capítulo se presenta los diagramas de flujo del software a implementar. Se profundiza cada etapa del programa con diagramas de flujo y su respectiva explicación.

CAPÍTULO 5

En este capítulo se muestra la implementación y pruebas realizadas a cada parte del esquema circuital que conforman los equipos medidores de energía eléctrica.

CAPÍTULO 6

En este capítulo se muestran las pruebas realizadas en diferentes viviendas, donde se podrán constatar el consumo de energía, así como los dashboards creados que podrán ser visualizados en un dispositivo móvil.

CAPÍTULO 7

En este capítulo se presentan las conclusiones y recomendaciones



CAPÍTULO II

MARCO TEÓRICO

En este capítulo se dará a conocer las características de los medidores de energía, su funcionamiento y sus especificaciones técnicas del equipo.

1. PRINCIPIO DE FUNCIONAMIENTO

El medidor de energía se conecta a la llave principal y por medio de un teclado se ingresa los datos de la Red Wi-Fi del domicilio.

Una vez instalado el equipo procederá a realizar las mediciones de voltaje y corriente con lo que calculará la potencia consumida de la vivienda.

El microcontrolador consulta constantemente si hay comunicación por protocolo MQTT con el Bróker como se muestra en la figura 2.1.

Si existe una respuesta afirmativa, la conexión se establece y el microcontrolador empieza a PUBLICAR la información en el Bróker; en caso de no existir conectividad, el microcontrolador cuenta con una memoria interna y un reloj en tiempo real, donde se almacenará los resultados del procesamiento hasta que vuelva a existir conexión con el Bróker y retransmitir dicha información para que no haya pérdida de datos.

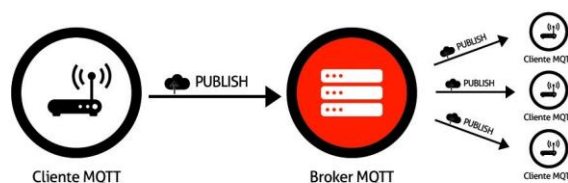


Figura 2.1 Arquitectura del Bróker MQTT

Fuente: <https://www.codigoiot.com/base-de-conocimiento/mqtt/>

También se establece el Bróker de MQTT en un servidor que cuenta con MySQL como base de datos y Node Red como software de interfaz.

Node Red se conecta al Bróker de MQTT como CLIENTE, leyendo la data que es enviada por el equipo Medidor de Energía Eléctrica, luego almacena la información en la base de datos (MySQL), para generar dashboards como se muestra en la figura

2.2 (pantallas de visualización y alamas), utilizando las mismas librerías del Node Red.

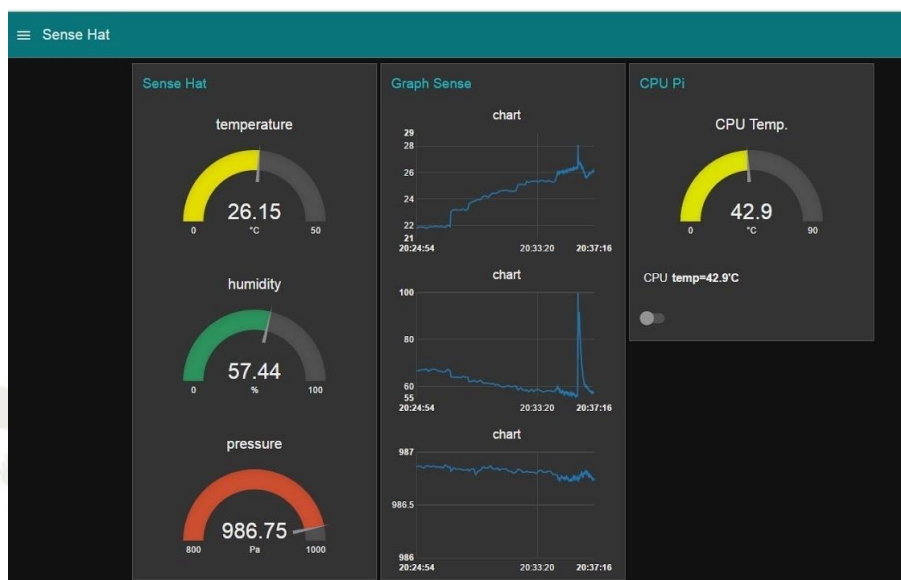


Figura 2.2 Dashboards de resultados de medición

Fuente: <https://blog.bricogeek.com/noticias/raspberry-pi/tutorial-de-node-red-con-raspberry-pi-y-esp8266/>

2. ANTECEDENTES

La presente tesis se elaboró tomando en cuenta las siguientes investigaciones realizadas:

- **Diseño e Implementación de una red de medidores de energía para artefactos domésticos – Nelson Eduardo Huiman Tocto / Pontificia Universidad Católica del Perú (17 de noviembre del 2016).**

El objetivo general de la tesis es el desarrollo de la red de los medidores de energía eléctrica para uso doméstico y que permita obtener los datos del consumo de cada dispositivo que se encuentre conectado a la red (Nelson Eduardo Huiman Tocto, 2016).

En mi propuesta de tesis, la diferencia es que se implementará un solo equipo de medición para toda la red eléctrica de cada vivienda y este cuenta con conexión a WIFI.

- **Diseño e Implementación de un analizador de calidad de energía eléctrica – Martin Valencia Alejo / Universidad Católica de Santa María (22 de abril del 2016).**

El objetivo general de esta tesis es diseñar un equipo analizador de energía eléctrica de alta precisión en sus mediciones de corriente, voltaje y frecuencia y registrarlos en una memoria SD del mismo equipo (Martin Valencia Alejo, 2016)

En mi propuesta de tesis, utilizare el mismo microcontrolador que se menciona en esta tesis con el fin de tener tal precisión en mis mediciones de corriente, voltaje y frecuencia con la diferencia que el almacenamiento de la información será en una base de datos de un servidor (Laptop/PC).

3. ESPECIFICACIONES TÉCNICAS DEL EQUIPO

Mediciones Eléctricas Elementales	
Voltaje	0 ... 300 VAC según IEC62052-11:2020
Corriente	0 ... 50 Amperios
Frecuencia	60 Hz según IEC 62053-21:2020

Tabla 2.1 Mediciones eléctricas elementales

Fuente: Elaboración propia

Mediciones Eléctricas Derivadas	
Potencias	Activa (W), Reactiva (VAR), Aparente (VA), KW-H, KVAR-H.

Tabla 2.2 Mediciones eléctricas derivadas

Fuente: Elaboración propia

Tipo de Comunicación	
Wi-Fi	802.11 b/g/n/e/i (802.11n @ 2.4 GHz hasta 150 Mbit/s)
Serial	9600 baudios
Modelo	ESP32

Tabla 2.3 Tipo de comunicación

Fuente: Elaboración propia

Almacenamiento y Tiempo	
Memoria EEPROM	256 KBytes / 25LC256

Reloj en tiempo Real	DS1307
----------------------	--------

Tabla 2.4 Almacenamiento y tiempo

Fuente: Elaboración propia

Procesador	
Microcontrolador	ARM Cortex - R5F
Marca	Texas Instruments
Familia	Hercules RMx
Programador	XDS110
Frecuencia	300 MHz
Certificación Industrial	Si
Numero de Bits	32 bits
Memoria de Programa	4MB
Memoria RAM	512 KB

Tabla 2.5 Procesador

Fuente: Elaboración propia

Software de Programación	
Programador de Microcontrolador	XDS110 Debugger
Software de desarrollo	Code Composer Studio 10.2.0 (Texas Instruments)
Software de Configuración de microcontrolador	Halcogen 04.07.01 (Texas Instruments)
Software de procesamiento	Cortex-R DSP Software Library
Programador	XDS110
Lenguaje de Programación	C++, Nativo para ARM Cortex

Tabla 2.6 Software de programación

Fuente: Elaboración propia

Fuente de Alimentación
Alimentación desde la misma Red de Eléctrica.

Tabla 2.7 Fuente de alimentación

Fuente: Elaboración propia

4. BASES TEÓRICAS

4.1 MICRONCOTROLADOR RM46L852

El procesador RM45L852 es una familia de microcontroladores de alto rendimiento para sistemas de seguridad.

4.1.1 UNIDAD DE PUNTO FLOTANTE

Una unidad de punto flotante es un componente de la CPU especializado en el cálculo de operaciones en coma flotante. Las operaciones básicas que todo punto flotante puede realizar son las aritméticas (suma y multiplicación). (Hércules RM46L852, 2020)

El procesador RM46L852 integra la CPU de punto flotante que ofrece una eficacia de 1.66 DMIPS/MHz y puede funcionar hasta 220 MHz proporcionando hasta 365 DMIPS. (Hércules RM46L852, 2020)

4.1.2 COVERTIDOR ANÁLOGO DIGITAL

Un convertidor análogo digital (ADC) es un dispositivo que convierte una señal analógica variable en el tiempo a una representación de números digitales de la amplitud de esa señal (Hércules RM46L852, 2020).

Uno de los principales parámetros de un ADC es la velocidad y la resolución de muestra. La tasa de muestra del ADC es la frecuencia en la que la señal de entrada se prueba, esta velocidad de muestreo determina el ancho de banda máximo de la señal de entrada y normalmente es mayor que el doble de la señal de entrada más alta, también conocida con la tasa de Nyquist. La resolución del ADC determina la precisión de la representación digital discreta que se puede producir sobre el rango de valores analógicos. (Hércules RM46L852, 2020)

El procesador RM46L852 tiene una resolución de ADC de 12 bits con 24 entradas en total y 64 palabras de memoria RAM de buffer protegida por paridad cada una. Los canales ADC se pueden convertir individualmente o se pueden agrupar por software para secuencias de conversión secuencia. El ADC tiene un modo de 10 bits para usar cuando se desea compatibilidad con dispositivos más o antiguos o un tiempo de conversión más rápido.

El módulo ADC admite 3 grupos de conversión:

- Grupo de eventos
- Grupo 1
- Grupo 2

Cada uno de estos tres grupos se pueden configurar para que se activen mediante eventos de hardware. En ese caso la aplicación puede seleccionar entre 8 fuentes de eventos para que sean el desencadenante de las conversiones en grupo (Hércules RM46L852, 2020).

4.1.3 ENTRADAS Y SALIDAS DIGITALES

Las entradas y salidas digitales (nivel lógico) son los bloques más básicos de los sistemas de hardware digitales. También se les conoce comúnmente como GPIO (Hércules RM46L852, 2020).

El módulo GPIO de este dispositivo admite dos puertos, GIOA y GIOB. Los pines de E/S son bidireccionales y programables por bits. Tanto GIOA como GIOB admiten la capacidad de interrupción externa (Hércules RM46L852, 2020).

El módulo GPIO tiene las siguientes características:

- Cada pin IO se puede configurar como aporte, producción o drenaje abierto.
- Detección de interrupciones programables y ambos bordes o en un solo borde.
- Polaridad de detección de flanco programable, ya sea flanco ascendente o descendente (establecido en registro GIOPOL)
- Banderas de interrupción individuales (establecidas en el registro GIOFLG)
- La interrupción individual habilita, establece y borra a través de los registros GIOENASET y GIONACLR respectivamente.
- Prioridad de interrupción programable, configurada mediante los registros GIOVLSET y GIOVLCLR
- Pullup / pulldown interno que permite que los pines de E/S no utilizados queden desconectados.

4.1.4 MÓDULO DE COMUNICACIÓN SERIAL

La comunicación serial puede transmitir o recibir datos bit a bit, llamado comunicación UART (Standard universal asynchronous receiver-transmitter). A continuación, se describen las características del módulo UART (Hoja de datos del fabricante).

- Admite la operación de dúplex completo o medio.
- Formato de transmisión de recepción de doble buffer.
- Formato de trama configurable de 3 a 13 bits por carácter
- Longitud de la palabra de datos programable de uno a ocho bits.
- Bit de dirección adicional en modo de bit de dirección
- Paridad programable para cero o un bit de paridad, paridad par o impar.
- Stop programable para uno o dos bits de parada.
- Modos de comunicación asíncrona.

4.2 CÁLCULOS DE POTENCIA

4.2.1 MEDICIÓN DE ENERGÍA ELÉCTRICA

El medidor de energía eléctrica o medidor eléctrico es el instrumento de medición comercial más utilizado en el mundo. Es un dispositivo que mide el consumo de energía eléctrica de un circuito o servicio eléctrico (<https://jdelectricos.com.co/medidor-de-energia-electrica/>).

- **NORMA IEC 62053-21:2020 (Equipos de medida de electricidad. Requisitos particulares. Parte 21: Contadores estáticos de energía activa CA (clases 0,5, 1 y 2)**

Se aplica solo a los medidores de vatios-hora estáticos de clases de precisión 0.5 1 y 2 para la medición de energía eléctrica activa de corriente alterna en redes de 50 Hz o 60 HZ.

Se aplica a los equipos de medición de electricidad diseñados para:

- Medir y controlar la energía eléctrica en redes eléctricas con voltaje de hasta 1000 V CA.
- Operar con pantallas indicadoras integradas o separadas, o sin pantalla indicadora.
- Instalarse en un zócalo o bastidor correspondiente.

No se aplica a:

- Medidores para los cuales el voltaje de línea a neutro derivado de los normales excede 1000 VCA.
 - Medidores destinados a la conexión con transformadores de instrumentos de baja potencia.
 - Medidores portátiles.
 - Medidores utilizados en vehículos, barcos y aviones.
- **NORMA IEC 62052-11:2020 (Equipos de medida de electricidad. Requisitos generales, ensayos y condiciones de ensayo. Parte 11)**

Esta norma detalla los requisitos funcionales, mecánicos, eléctricos y de marcado, los métodos de prueba y las condiciones de prueba, incluida la inmunidad a las influencias externas que cubren los entornos electromagnéticos y climáticos.

Se aplica a los equipos diseñados para:

- Medir y controlar la energía eléctrica en redes eléctricas con un voltaje de hasta 1000 VCA o 1500 VCC.
- Operar con pantallas integradas.
- Operar con o sin pantallas indicadoras
- Estar instalado en enchufes o racks.

No aplica a:

- Medidores para los cuales el voltaje de línea a neutro derivado de os voltajes nominales excede a 1000 VCA o 1500 VCC.
- Medidores destinados a la conexión con transformadores de instrumentos de baja potencia.
- Medidores portátiles
- Medidores utilizados en vehículos, barcos y aviones.

4.2.2 TEORÍA DE MUESTREO

- **TEOREMA DE MUESTREO DE NYQUIST – SHANNON**

El teorema de muestreo de Nyquist se refiere a la relación de la velocidad de muestreo y la frecuencia de la señal medida. Establece que la tasa de muestreo (f_s) debe ser mayor que el doble del componente de frecuencia más alto en interés de la señal medida. Esta frecuencia se denomina comúnmente como frecuencia de Nyquist f_n .

$$f_s > 2 * f_n$$

Este teorema no debe ser confundido o asociado con la cuantificación, este es el proceso de seguimiento del muestreo en la digitalización de señales y a diferencia les muestreo este es irreversible. En otras palabras, desde el punto de vista del teorema, las muestras discretas de una señal son valores precisos que no han sufrido ningún redondeo o truncamiento con una precisión determinada, es decir, no han sido cuantificados. (Redes Cisco. 2018)

4.2.3 CÁLCULO DEL RMS

El valor eficaz de voltaje o corriente se denomina valor cuadrático medio o RMS. El valor eficaz de una onda de voltaje periódica se basa en la potencia promedio entregada a una resistencia (Daniel W. Hart, 2001).

$$P = \frac{V_{cc}^2}{R}$$

Para una tensión periódica sobre una resistencia, la tensión eficaz se define como un voltaje que proporciona la misma potencia media que el voltaje continuo. El voltaje eficaz puede calcularse así:

$$P = \frac{V_{cf}^2}{R}$$

Si calculamos la potencia media de una resistencia a partir de las ecuaciones anteriores obtenemos:

$$\begin{aligned} P &= \frac{1}{T} \int_0^T p(t) dt \\ &= \frac{1}{T} \int_0^T v(t)i(t) dt \\ &= \frac{1}{T} \int_0^T \frac{v^2(t)}{R} dt \\ &= \frac{1}{R} \left[\frac{1}{T} \int_0^T v^2(t) dt \right] \end{aligned}$$

Igualando las expresiones para la potencia media dadas por las ecuaciones anteriores, obtenemos:

$$\begin{aligned} P &= \frac{V_{cf}^2}{R} \\ &= \frac{1}{R} \left[\frac{1}{T} \int_0^T v^2(t) dt \right] \\ V_{cf}^2 &= \frac{1}{T} \int_0^T v^2(t) dt \end{aligned}$$

Obteniéndose la siguiente ecuación para la tensión eficaz o RMS:

$$V_{ef} = V_{rms} = \sqrt{\frac{1}{T} \int_0^T v^2(t) dt}$$

Donde se sabe que el valor eficaz es la raíz cuadrada del valor medio del cuadrado de la tensión, expresión que da el nombre de RMS (Root Mean Square).

De la misma manera, la corriente eficaz se obtiene a partir de la ecuación dada por $p = I_{rms}^2 R$.

$$I_{rms} = \sqrt{\frac{1}{T} \int_0^T i^2(t) dt}$$

El valor eficaz de tensión y corriente es útil para calcular la potencia absorbida por la resistencia. Además, las clasificaciones de los dispositivos generalmente se especifican en forma de tensión y corriente eficaz o RMS (Daniel W. Hart, 2001).

4.2.4 CÁLCULO DE LA POTENCIA ELÉCTRICA

La potencia instantánea de cualquier dispositivo eléctrico se calcula a partir de la tensión y la corriente que le atraviesa $p(t) = v(t)i(t)$.

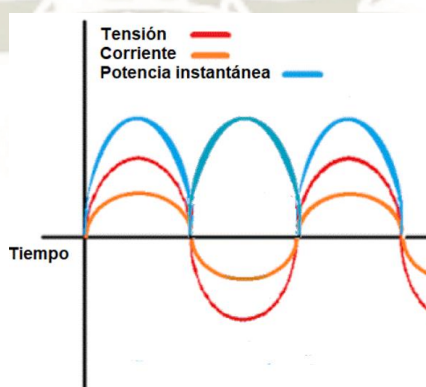


Figura 2.3 Diagrama fasorial de la potencia instantánea

Fuente: <https://instrumentacionhoy.blogspot.com/2019/08/potencia-activa-y-reactiva.html>

- **CÁLCULO DE LA POTENCIA ACTIVA**

La potencia activa también llamada potencia media es el valor medio de la potencia instantánea. Su unidad de medida es Vatios. Se calcula como:

$$P = U * I * \cos\varphi$$

Donde:

U = Tensión eficaz

I = Corriente eficaz

$\cos\varphi$ = Coseno del desfase

Siendo φ :

$$\varphi = \arctg \frac{X}{R}$$

Donde:

X= reactancia

R=resistencia de la carga

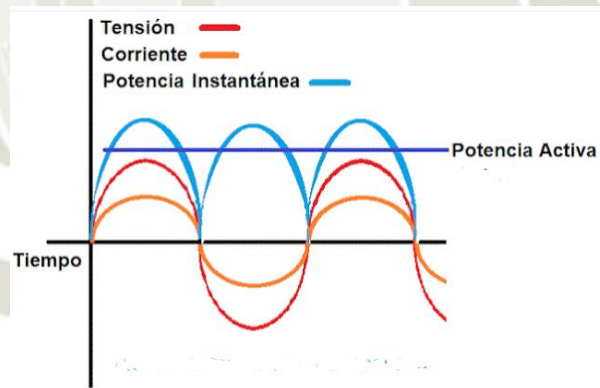


Figura 2.4 Diagrama fasorial de la potencia activa o media

Fuente: <https://instrumentacionhoy.blogspot.com/2019/08/potencia-activa-y-reactiva.html>

- **CÁLCULO DE LA POTENCIA REACTIVA**

Es la potencia en la cual la corriente está fuera de fase con respecto al voltaje.

Esta dada por la siguiente ecuación:

$$P = U * I * \sen\varphi$$

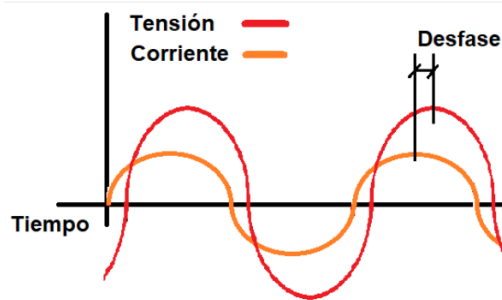


Figura 2.5 Diagrama fasorial de la potencia reactiva

Fuente: <https://instrumentacionhoy.blogspot.com/2019/08/potencia-activa-y-reactiva.html>

- **CÁLCULO DE LA POTENCIA APARENTE**

La potencia aparente es el producto de la tensión eficaz y la corriente eficaz. Se utiliza mayormente para especificar el valor nominal de los equipos. Viene expresada por la siguiente fórmula:

$$S = V_{RMS} I_{RMS}$$

Nota: en los circuitos de corriente alterna la potencia aparente S es la magnitud de la potencia compleja.

- **CÁLCULO DEL FACTOR DE POTENCIA**

El factor de potencia de una carga es el cociente de la potencia media y la potencia aparente, se define como:

$$fp = \frac{P}{S} = \frac{P}{V_{RMS} I_{RMS}}$$

4.2.5 CÁLCULO DE ENERGIA – KILO WATT HORA

- **KILO WATT HORA (kWh)**

El kWh es una unidad de medida que se utiliza para contabilizar el consumo eléctrico que ha realizado durante un periodo de tiempo.

Actualmente se utiliza en el sector energético para facturar el consumo de energía eléctrica realizado por los consumidores.

Se calcula mediante la siguiente formula extraída de la página oficial OSINERG, pág. 19:

$$CONSUMO\ MENSUAL = P * horas\ de\ uso\ por\ día * días\ de\ uso\ al\ mes$$

4.3 TEORÍA SOBRE IOT

El internet de las cosas IoT (Internet of Things) describe la red de objetos que tienen incorporados sensores, software u otras tecnologías con el fin de conectar e intercambiar datos con otros dispositivos y sistemas por medio de internet.

En los últimos años el IoT se ha convertido en una de las tecnologías más importantes del siglo XXI ya que permite la comunicación perfecta entre personas, procesos y cosas.

Si bien es cierto la idea de IoT existe desde hace mucho tiempo, existen una serie de tecnologías que han hecho posible que hoy en día sea una herramienta para muchas personas (Oswaldo Quiñones, 2019).

4.3.1 MÓDULO DE COMUNICACIÓN WIFI - ESP32

El módulo ESP32 es un procesador integrado con interfaces para conectarse con varios periféricos. Tiene dos núcleos de procesamiento cuyas frecuencias operativas pueden controlarse independientemente entre 80 MHz y 240 MHz. Los periféricos del procesador facilitan la conexión a una variedad de interfaces externas como:

- Interfaz periférica serial (SPI)
- I^2C
- Transmisor receptor asíncrono universal (UART)
- I^2S

- Tarjetas SD
- Ethernet
- Interfases táctiles y capacitivas. (Módulo WIFI ESP32, 2020)

4.3.2 DEFINICIÓN DEL PROTOCOLO DE COMUNICACIÓN MQTT

MQTT (Message Queue Telemetry Transport) es un protocolo usado para la comunicación machine-to-machine (M2M) en el IoT. Está orientado a la comunicación de sensores, consume muy poco ancho de banda y puede ser utilizado en la mayoría de dispositivos

(<https://www.murkyrobot.com/guias/comunicacion/mqtt>).

Su función básica es hacer posible la lectura de datos y envío de comandos a dispositivos que requieren bajo ancho de banda y menor consumo energético (<https://www.murkyrobot.com/guias/comunicacion/mqtt>).

- **PUBLISH / SUSCRIBE**

En el protocolo MQTT existen publicadores de datos o suscriptores. El publicador puede ser un sensor de temperatura que cada 90 segundos mide la temperatura de una cámara fría y el suscriptor o consumidor es el destinatario del dato, el cual debe indicar al bróker su interés en recibir ese dato específico. (<https://www.murkyrobot.com/guias/comunicacion/mqtt>)

- **TÓPICO**

Se refiere al que publica los mensajes de determinado tema y suscribe solo los que le interesan (<https://www.murkyrobot.com/guias/comunicacion/mqtt>).

- **QoS**

Es la parte más esencial del protocolo MQTT ya que define la calidad de los mensajes. (<https://www.luisllamas.es/que-es-mqtt-su-importancia-como-protocolo-iot/>)

Existen 3 niveles de la calidad de servicio:

- QoS 0: el mensaje se envía solamente una vez. En caso de que falle puede que algún mensaje no se entregue.
- QoS 1: el mensaje se envía hasta que se compruebe la entrega. En caso de que falle el suscriptor o consumidor puede recibir algún mensaje duplicado.
- QoS 2: se garantiza que cada mensaje se entregue al consumidor, y solamente una vez. (<https://www.luisllamas.es/que-es-mqtt-su-importancia-como-protocolo-iot/>)

4.3.3 BROKER PARA MQTT

Es un gestor de datos que se encuentra en la nube. Recibe todos los datos que están siendo publicados y decide que suscriptores o consumidores deben recibirlos de acuerdo al interés de cada uno.

4.3.4 BROKER AEDES

Aedes es un servidor bróker MQTT OpenSource diseñado exclusivamente para reemplazar al bróker Mosquitto. Su funcionamiento consta en la conexión interna de diferentes servidores. Una de sus ventajas es que si se pierde la conexión eléctrica el bróker sigue recibiendo datos normales sin la pérdida de estos comparando con el bróker anteriormente mencionado. (JMOT, 2020)

4.3.5 NODE RED

Node Red es una herramienta de programación visual que puede mostrar visualmente las relaciones y funciones a los usuarios sin tener que aprender un nuevo lenguaje de programación. Su objetivo principal es conectar dispositivos de hardware, API y servicios en línea.

Fue creada en el 2013 por el grupo de Tecnologías Emergentes de IBM, su objetivo fue ofrecer una solución cuando se trata de integrar software con otros servicios. En otras palabras, es un sistema que permite simplificar la programación, la conectividad y los servicios, distribuyéndolos de forma eficiente para ganar tiempo y espacio.

Node Red esta creado a partir de NodeJS y la librería de JavaScript D3-js (Taiji Hagino, 2021).

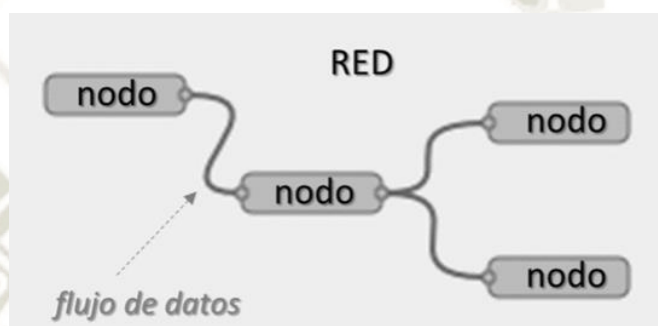


Figura 2.6 Representación de nodos

Fuente: <https://www.techedgegroup.com/es/blog/fundamentos-node-red>

- **PUERTOS TCP/UDP**

En la actualidad casi todas las personas en su domicilio, trabajo o lugar de ocio dispone de conexión a internet a través de un router. Para que exista la comunicación de extremo a extremo se necesita usar un puerto ya sean TCP o UDP. Cualquier aplicación o servicio que se utilice, desde un correo electrónico hasta un video juego o mensajería instantánea como tienen una serie de puertos abiertos transmitiendo o escuchando lo que sucede alrededor. (<https://www.hostdime.com.pe>)

Los puertos pueden ser de 2 tipos diferentes:

- **Protocolo TCP**

Es un protocolo conectivo, fiable y orientado a conexión, es el encargado de transmitir todos los segmentos de manera correcta, y utiliza las retransmisiones necesarias en caso de que ocurra algún tipo

de problema de conexión, por otro lado, garantiza el orden, por lo que las capas superiores no tienen que encargarse de ello.
(<https://www.hostdime.com.pe>)

Entre los principales se encuentran:

- ✓ Puerto TCP 1880: Libre asignación
- ✓ Puerto TCP 1883: Protocolo bróker MQTT
- ✓ Puerto TCP 3306: Sistema de gestión de base de datos MySQL.

○ **Protocolo UDP**

Es un protocolo no orientado a conexión, no es muy fiable y no garantiza que lleguen los paquetes.

Cuando se hace uso de UDP, los paquetes se envían al destinatario, el remitente no puede asegurarse de que el destinatario haya recibido el paquete, entonces este continuara enviando los siguientes paquetes. Si el destinatario pierde alguno de los paquetes UDP, no podrá solicitar nuevamente esos paquetes y se dan por perdidos.
(<https://www.hostdime.com.pe>)

● **NodeJS**

Es un software muy potente que permite programar en JavaScript. Esta herramienta proporciona la potencia suficiente para que Node Red sea fiable y escalable. Una de sus ventajas es que esta optimizado para poder tratar múltiples conexiones concurrentes de una mas muy optima. (Claudia Alves, 2021)

● **JavaScript D3-js**

Es el encargado de la interfaz web. Node Red es capaz de crear sus propios sistemas a través de una página web, es decir, no necesita un programa extra para poder utilizar esta herramienta (Claudia Alves, 2021).

- **NODO MQTT**

Protocolo de comunicación que funciona con un servidor central que es el bróker y con suscriptores a tópicos que serían los mensajes que nosotros enviamos a través del bróker. (Node Red, 2021). Existen dos nodos con MQTT que permiten una comunicación y son:

- **MQTT in**

Se conecta a un agente MQTT a través del bróker, suscribe los mensajes de un tema en específico (topic) y lee dichos mensajes (Node Red, 2021).

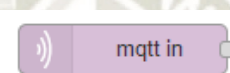


Figura 2.7 Representación del nodo MQTT in

Fuente: Node Red, 2021

- **MQTT out**

Se conecta a un agente MQTT a través de un bróker y publica los mensajes (Node Red, 2021).

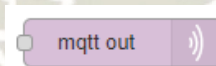


Figura 2.8 Representación del nodo MQTT out

Fuente: Node Red, 2021

- **NODO DE FUNCIÓN**

Es un bloque de funciones de JavaScript que se ejecutan contra los mensajes que recibe dicho nodo. Estos mensajes pasan como un objeto en JavaScript llamado msg.

Se espera que esta función devuelva un objeto de mensaje (o varios objetos de mensaje), pero también puede elegir no devolver nada para detener un flujo determinado (Node Red, 2021).

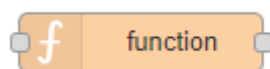


Figura 2.9 Representación del nodo function

Fuente: Node Red, 2021

- **NODO DE DASHBOARD**

Node Red presenta diferentes Dashboards entre los cuales se encuentran los siguientes:

- **Chart**

Es un nodo para hacer posible la gráfica de datos. Tiene diferentes modos de gráficos como: gráficos lineales basados en el tiempo, gráficos de barras (verticales u horizontales) y gráficos circulares. Además, las etiquetas del eje X pueden ser personalizadas utilizando una cadena de formato de fecha. (Node Red, 2021)

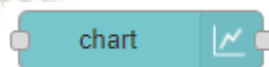


Figura 2.10 Representación del nodo chart

Fuente: Node Red, 2021

- **Gauge**

Adiciona un widget indicador a la interfaz del usuario, tiene 4 modos: estándar (indicador simple), circular (360° completo), brújula y onda. También se puede especificar la gama de colores de los indicadores (Node Red, 2021).

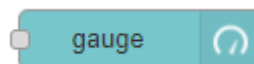


Figura 2.11 Representación del nodo gauge

Fuente: Node Red, 2021

- **NODO DE BASE DE DATOS**

Nodo que permite el acceso básico a una base de datos MySQL. Su función es una vez registrada la base de datos con el usuario y contraseña que se le asignó, ejecuta la sentencia SQL que contiene el tópico del mensaje. (Node Red, 2021)

Node Red permite una amplia integración con otros servicios como:

- Integración con MQTT, son los nodos que permite la publicación y suscripción del mensaje.
- Integración con API REST, se utiliza el protocolo HTTP teniendo una opción para permitir la creación de una cadena de consulta de mg. Payload.

En Node Red también se pueden enviar emails, pero se tiene que habilitar una contraseña de aplicación, envío de mensajes de texto o realiza una llamada de voz, que está configurado para hacer dichas tareas (Node Red, 2021).

4.3.6 SOFTWARE XAMPP

XAMPP es una herramienta de servidor independiente de código libre, que permite probar el desarrollo web basado en PHP en un ordenador sin necesidad de tener acceso a internet (Christophe Aubry, 2015).

XAMPP incluye diferentes aplicaciones libres. Su nombre deriva de los siguientes programas:

- **Linux**

Sistema operativo donde se instala la aplicación. Linux es libre, segura, no requiere pago de licencias y tiene alto rendimiento (Christophe Aubry, 2015).

- **Apache**

Es un servidor web de código abierto usado globalmente para la entrega de contenidos web (Christophe Aubry, 2015).

- **MySQL**

MySQL sirve para el almacenamiento de datos para servicios web, esto se da gracias a la combinación con el servidor web Apache y el lenguaje PHP (Christophe Aubry, 2015).

- **PHP**

Es un lenguaje de programación que permite crear páginas web o aplicaciones dinámicas, soporta varios sistemas de datos y es independiente de la plataforma (Christophe Aubry, 2015).

- **Perl**

Es un lenguaje de programación que se usa en la administración del sistema, desarrollo web y programación de red. Este lenguaje también permite programar aplicaciones web dinámicas (Christophe Aubry, 2015).

4.3.7 BASE DE DATOS MYSQL

Permite el acceso rápido a una base de datos MySQL. Este nodo usa la operación de consulta contra la base de datos configurada. Permite insertar y eliminar de manera muy fácil.

Si se pierde la conexión, no hay problema porque la reconexión es en milisegundos (Node Red, 2021).

Trabaja con las siguientes instrucciones:

- i. Msg.topic: debe contener la consulta de la base de datos y el resultado se devuelve en msg.payload.
- ii. Msg.payload: puede contener una matriz de valores para enlazar el tema.

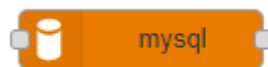


Figura 2.12 Representación del nodo MySQL

Fuente: Node Red, 2021

4.4 APLICACIONES GENERALES

4.4.1 CIRCUITO DE CONTROL OFFSET

El circuito está conformado por dos resistencias, dos capacitores y un OPAMP en configuración seguidor de voltaje. Este circuito permite el funcionamiento con un OPAMP de suministro único, generando así una tierra virtual para el amplificador. El amplificador operacional de fuente única necesita una tierra virtual, generalmente el voltaje es igual a $V_{cc} / 2$.

4.5 FUENTE DE ALIMENTACIÓN

4.5.1 FUENTE DE ALIMENTACIÓN DE ESTADO SÓLIDO (SWITCHING)

La fuente de alimentación de estado sólido también llamada fuente de alimentación conmutada se basa en la conmutación del transistor, su principal funcionamiento es solucionar los problemas de disipación térmica que tienen las fuentes lineales o con transformador

Por otro lado, las fuentes de alimentación con transformador se caracterizan por una eficiencia mucho más baja, es decir, la relación entre la potencia de salida y la potencia consumida, esto se debe a la estructura del transformador. Los materiales utilizados, así como el estabilizador en una parte de la potencia se pierden en forma de calor emitido. Por lo tanto, la desventaja de estas fuentes de alimentación es el gran peso y gran tamaño en comparación con las fuentes conmutadas, el transformador también consume corriente que incluso puede alcanzar el 20% de la corriente nominal de la fuente de alimentación, el precio, etc.

Entonces, se selecciona una fuente de alimentación conmutada dado que entrega más corriente a la carga ocupando un menor espacio, teniendo un buen rendimiento.

4.6 ETAPA ANÁLOGA

4.6.1 SENSOR DE VOLTAJE

- **CIRCUITO DE ALTA IMPEDANCIA**

El circuito de alta impedancia consta de OPAMP en una configuración de seguidor de voltaje.

Debido a que su entrada tiene un consumo de corriente mínimo, puede generar niveles de protección y alta impedancia para el circuito del sensor de voltaje.

- **CIRCUITO OFFSET**

El offset de una señal alterna es el nivel de señal continua sumada a una señal alterna. Por lo tanto, si la señal está centrada en el origen, se dice que el nivel de offset es 0. Si se desplaza hacia arriba, el offset es positivo, mientras que, si se desplaza hacia abajo, será negativo.

- **AMPLIFICADOR DIFERENCIAL**

Este circuito se puede producir con entradas inversoras y no inversoras que genera una salida de voltaje que esta es igual a la diferencia entre las dos entradas. Se aplica un voltaje de offset, lo que nos permite mover la salida de voltaje.

- **FILTRO DE SALIDA RC PASABAJO**

Un filtro RC pasa bajo es un circuito de filtro compuesto por una resistencia y un condensador que pasa a través de señales de baja frecuencia, mientras que el bloqueo pasa por señales de alta frecuencia.

Para crear un filtro RC pasa bajo, la resistencia se coloca en serie en la señal de entrada y el condensador se coloca en paralelo a la señal de entrada, tal como se muestra en el siguiente esquema

(<http://www.learningaboutelectronics.com/Articulos/Filtro-paso-bajo.php>).

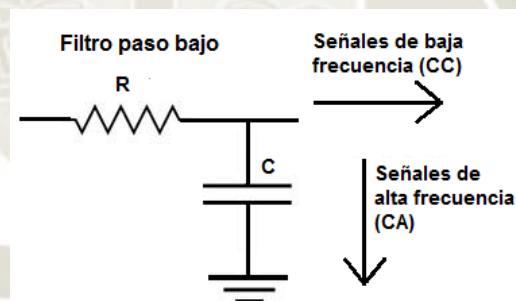


Figura 2.13 Filtro pasa bajo RC

Fuente: <http://www.learningaboutelectronics.com/Articulos/Filtro-paso-bajo.php>

Un condensador es un dispositivo reactivo que ofrece una resistencia muy alta a señales de baja frecuencia, y ofrece baja resistencia a las señales de alta frecuencia. Las señales de alta frecuencia pasarán a través del condensador, puesto que el condensador les ofrece una trayectoria de muy baja resistencia. Teniendo en cuenta que la corriente siempre toma el camino de menor

resistencia. (<http://www.learningaboutelectronics.com/Articulos/Filtro-paso-bajo.php>)

4.6.2 SENSOR DE CORRIENTE

- **SENSOR ACS756-050**

El sensor proporciona una medición de corriente precisa para señales de corriente alterna y corriente continua. El conductor de cobre grueso tiene la capacidad de soportar hasta 5 veces en condiciones de sobre corriente. El ACS756-050 emite una señal de salida de voltaje analógico que varía linealmente con la corriente detectada.

Este dispositivo consta de un circuito de sensor Hall lineal de baja compensación y precisión con una ruta de conducción de cobre ubicada cerca de la matriz. La corriente aplicada que fluye a través de esta ruta de conducción de cobre genera un campo magnético que es detectado por el Hall IC integrado y convertido en un voltaje proporcional. La precisión del dispositivo se optimiza gracias a la proximidad de la señal magnética al transductor Hall.

El sensor de corriente ACS756-050 opera desde 5V y emite voltaje analógico proporcional a la corriente medida en los terminales de detección. El procesador RM46Lx será el encargado de leer los valores proporcionados por dicho sensor. El terminal de detección puede incluso medir la corriente para cargas que operan a altos voltajes, como la red de 230 VCA, mientras que el voltaje de detección de salida está aislado de la parte de medición.

El sensor ACS56-050 puede detectar hasta 50 A de flujo de corriente. Detectar y controlar el flujo de corriente es un requisito fundamental en una amplia variedad de aplicaciones, que incluyen circuitos de protección contra sobre corriente, cargadores de batería, fuentes de alimentación de modo de conmutación, vatímetros digitales, fuentes de corriente programables, etc. (Hoja de datos del fabricante)

- **Características:**

- ✓ Alimentación de 5 V.
- ✓ Indicador de alimentación integrado.
- ✓ Voltaje de salida proporcional a las corrientes de CA o CC.
- ✓ Sensibilidad de salida de 40 mV / A.

4.6.3 SENSOR DE FRECUENCIA

- **COMPARADOR DE VOLTAGE**

Un comparador es un circuito electrónico que compara dos entradas y produce una salida. El valor de salida del comparador indica qué entrada es mayor o menor.

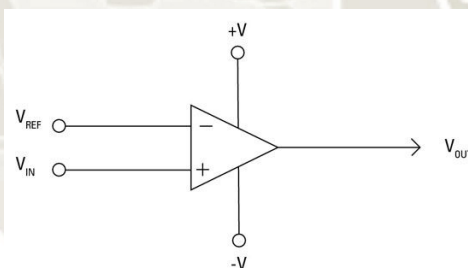


Figura 2.14 OPAMP comparador de voltaje

Fuente: <https://maniqui.ru/electronica-de-consumo/electronica/componentes-4/12522-componentes-de-la-electronica-cmo-utilizar-un.html>

Se expresa a la ecuación del comparador como:

$$V_o = -V_{cc} \quad \text{si} \quad V_1 > V_2$$

$$V_o = V_{cc} \quad \text{si} \quad V_1 < V_2$$

- **COMPUERTA LÓGICA SCHMITT TRIGGER**

Un Schmitt Trigger es un tipo de entrada lógica que nos proporcionará histéresis o dos niveles de umbral: alto y bajo. Esto nos permitirá reducir los errores producidos por señales ruidosas, dando como resultado una onda cuadrada.

- Reducir el ruido de la señal
- Acomoda la señal de la salida del comparador LM311.

- **MÓDULO CAP DEL MICROCONTROLADOR HERCULES RM46**

La función del módulo cap es medir el tiempo entre cada flanco de subida, este tiempo se incrementa cada $1\mu s$.

CAPÍTULO III
DISEÑO DE HARDWARE

En este capítulo se describe el proceso de diseño de diferentes circuitos que conforman cada etapa de los 2 medidores de energía eléctrica. Se muestra un diagrama de bloques de las diferentes etapas de diseño.

1. DIAGRAMA DE BLOQUES GENERAL

En el siguiente diagrama se presentan 3 bloques casa 1 y casa 2 que representan las mediciones que se realizan y un diagrama para el servidor de datos, como se muestra en la figura 3.1.

- Casa 1: Vivienda que se ubica en el distrito de Mariano Melgar de la ciudad de Arequipa.
- Casa N: Vivienda que se ubica cualquier distrito de la ciudad de Arequipa.
- Servidor de Datos: se ubica en una zona X, el cual va a recibir la información enviada de ambas viviendas por WIFI.

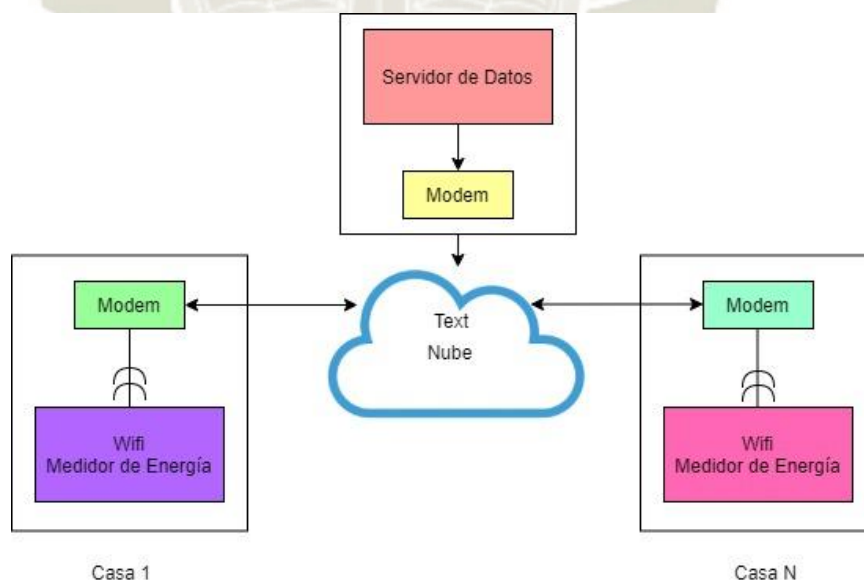


Figura 3.1 Diagrama de bloques general

Fuente: Elaboración propia

- Servidor: El servidor puede ubicarse en cualquiera de las viviendas o en algún lugar diferente por ende este bloque no se encuentra dentro de alguna vivienda o lugar en específico. También cuenta con su modem conectado a la red, el cual se encargará de enviar datos a la nube para conectar las diferentes viviendas con su respectivo medidor.
- CASA 1, CASA N: Según los bloques con el nombre de Casa 1 y Casa N, indican que las viviendas de los usuarios contarán cada una con un equipo de medición de energía eléctrica con conectividad IoT enviado por el servidor de datos, el cual enviara los datos que se midan a la nube.

2. DIAGRAMA DE BLOQUES DEL EQUIPO

En el siguiente diagrama se desglosa cada diagrama de bloques del circuito que conforman cada etapa de los 2 medidores de energía, se pueden diferenciar 3 etapas en la etapa de medición que consta del circuito de voltaje, corriente y frecuencia; la etapa de control consta de visualizaciones e interfaz, etapa de memoria y etapa de comunicación, como se muestra en la figura 3.2.

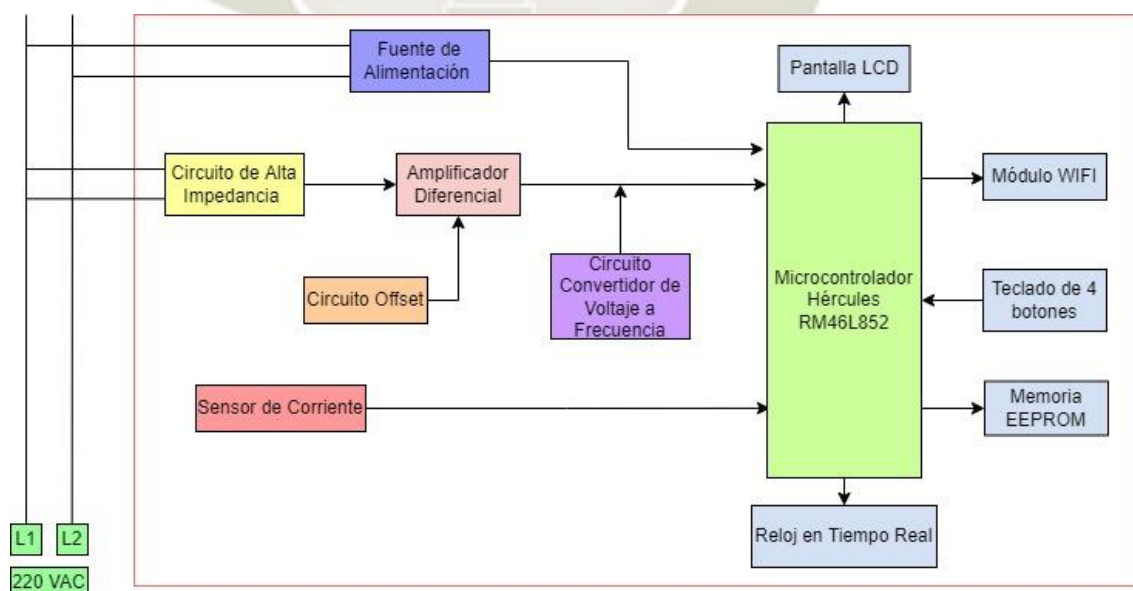
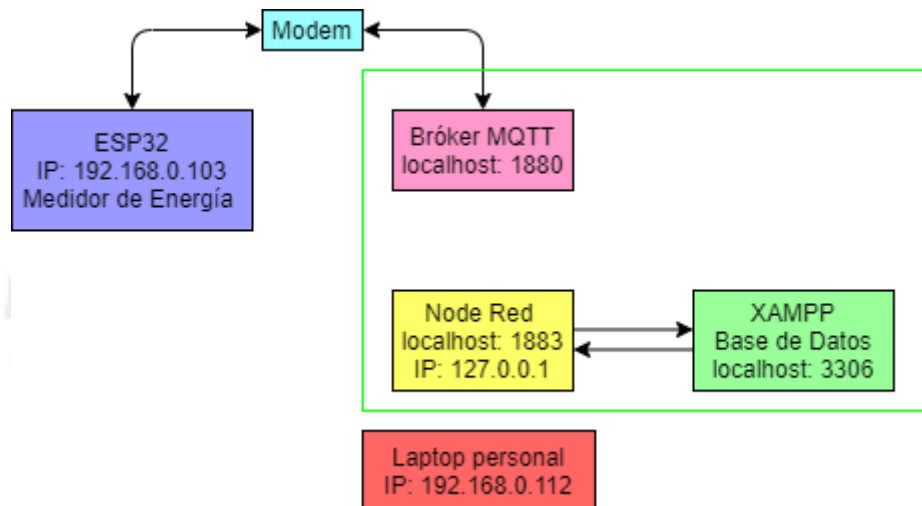


Figura 3.2 Diagrama de bloques de medición de energía

Fuente: Elaboración propia

- i. Bloque Fuente de Alimentación: Este bloque indica que el equipo se alimentará de la misma red eléctrica de donde se realiza la medición para alimentar al procesador.
- ii. Bloque Sensor de Voltaje: Es un circuito que va a medir el voltaje a través de una red de resistencias que más adelante se calcularán para demostrar que se puede medir tensión sin alterar su fase ni quitar información de sus armónicos que componen la señal y acondicionará la señal para ser medida por el microcontrolador. Recibe una señal de 5 V del procesador y le da una ganancia de 833.33×10^6 a la entrada.
- iii. Bloque Sensor de Corriente: En este bloque se implementa un circuito acondicionador con un sensor de corriente tipo Hall que permitirá obtener la señal de corriente y luego ser medida por el microcontrolador.
- iv. Bloque Sensor de Convertidor Voltaje – Frecuencia: En este bloque se utiliza un circuito que convierta la señal análoga del voltaje en una señal digital para poder medir su ancho de pulso y calcular la frecuencia de línea.
- v. Bloque Microcontrolador, Pantalla Lcd, Modulo Wifi: El bloque del microcontrolador, se encargará de realizar las mediciones de las señales de voltaje, corriente y frecuencia de los mencionados anteriormente, para luego mostrar en la pantalla LCD los resultados y enviar por el módulo WIFI la data a una base de datos en el Bloque del Servidor.
Además, el bloque de la pantalla LCD también permite visualizar las configuraciones del equipo.
- vi. Bloque Reloj en Tiempo Real y Memoria EEPROM: El bloque de reloj en tiempo real, permite al equipo registrar la hora y fecha de las mediciones en tiempo real, también se envía esta data por WIFI.
- vii. El bloque de memoria EEPROM, permite almacenar las configuraciones del equipo como, por ejemplo; voltaje máximo de medición, corriente máxima, numero de muestras, etc.

En el siguiente diagrama se puede observar 2 etapas, la primera que es el ESP32 que sería el medidor de energía, la segunda etapa que sería el bróker incluido en la laptop personal y ambos conectados al modem.



Fuente: Elaboración propia

- i. Bloque Modem: El modem se encargará de recibir la información de internet a través de un cable coaxial, el cual podrá emitir internet por medio de WIFI.
- ii. Bloque ESP32: En esta etapa el módulo de WIFI ESP32 se conectará al modem mediante WIFI para que trabaje como el medidor de energía.
- iii. Bloque XAMPP: Bloque que trabaja como el administrador de la base de datos bajo el protocolo MySQL.
- iv. Bloque NODE RED: El bloque de Node Red permite crear los dashboards y la base de datos para poder visualizarlos.

4. ELEMENTOS DE APLICACIÓN GENERAL

4.1 CIRCUITO DE CONTROL OFFSET

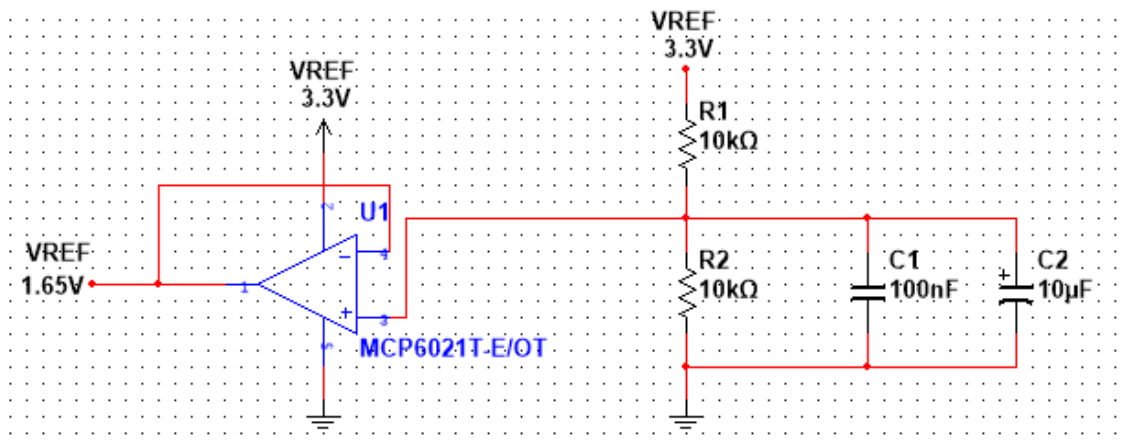


Figura 3.4 Circuito offset

Fuente: Elaboración propia

El condensador no polarizado de 100 nF forma un filtro de paso bajo para eliminar el ruido, eso quiere decir que elimina todas las señales no deseadas y solo dejar pasar las señales de entrada sinusoidales en función de su frecuencia.

El condensador de 10uF se encarga de mantener la línea de voltaje $VCC/2$, estabilizada, dado que el condensador se queda cargado y de existir alguna suba o baja de tensión empieza a entregar energía al condensador, de forma que no se note dicha alteración.

4.2 SELECCIÓN DEL CIRCUITO AMPLIFICADOR

Entre los diferentes opamps que encontramos en el mercado, se tienen los siguientes que se presentan en la tabla 3.1

Código	Rail to Rail	Voltaje Offset	Rechazo al Ruido Blanco	Ancho de Banda	Rango de Voltaje
LM358	NO	2mV	85	1MHz	3-32V
MCP6022	SI	500uV	90	10MHz	2.5-5V
OPA316	SI	0.5mV	90	10MHz	1.8-5V

Tabla 3.1 Selección de opamp

Fuente: Elaboración propia

En la tabla 3.1 se presentan las características más importantes de los opamps para su selección.

Entre sus principales características tenemos:

- **RAIL TO TAIL:** El opamp tiene la capacidad de salida de voltaje $V_0 = V_{CC}$
- **Voltaje offset:** es el voltaje contaminante que todos los opamps traen por fabricación, mientras más cercano a cero sea este valor el opamp es de mejor calidad
- **Rechazo al ruido blanco:** Es una característica del opamp dada por su fabricación, el cual le permite diferenciar entre la señal que se va a amplificar y el ruido del ambiente, mientras este valor sea más aproximado a los 100dB es un opamp de mejor calidad.
- **Ancho de banda:** El ancho de banda es la velocidad que el opamp tiene para dar seguimiento a la señal que ingresa en este, mientras más alto sea el ancho de banda el opamp es de mejor calidad.
- **Rango de voltaje:** Son valores de operación de voltaje que tiene el opamp, para el diseño de este equipo se prefiere trabajar con un voltaje de 3.3v debido ya que el convertidor análogo digital del microcontrolador es de 0 hasta 3.3v.

4.3 CARACTERÍSTICAS DE OPAMP IMPLEMENTADO

El amplificador MCP6022 es un OPAMP que utiliza la tecnología CMOS avanzada de Microchip, que proporciona corriente de baja polarización, operación de alta velocidad, alta impedancia de entrada, baja impedancia de salida, etc.

además, tiene un ancho de banda de ganancia de 10 MHz con una corriente de operación típica de 1 mA y un voltaje de compensación que es menor a 0.5 mV. Funciona con un solo voltaje de entrada que puede ser tan bajo como 2.5 V. (Hoja de datos del fabricante)

- Rail – to – Rail Input/Output
- Wide Bandwidth: 10 MHz (typical)
- Low Noise: 8.7 nV/ $\sqrt{\text{Hz}}$ at 10 kHz (typical)
- Low Offset Voltage:
 - Industrial Temperature: $\pm 500 \mu\text{V}$ (max.)
 - Extended Temperature: $\pm 250 \mu\text{V}$ (max.)
- Low Supply Current: 1 mA (typical)
- Power Supply Range: 2.5V to 5.5V
- Temperature Range:
 - Industrial: -40°C to $+85^{\circ}\text{C}$
 - Extended: -40°C to $+125^{\circ}\text{C}$

5. DISEÑO DE LA ETAPA ANÁLOGA DEL MEDIDOR DE ENERGÍA

5.1 SENSOR DE VOLTAJE

El proyecto se compone de dos equipos medidores de energía eléctrica que enviarán información por WIFI a la red local de la vivienda. Por lo que los equipos requieren una medición de voltaje de mucha precisión en su forma de onda, puesto que la única manera en la que se puede utilizar este tipo de trabajo es una red resistiva; por lo que se detallará más adelante. Utiliza una red de resistencias la cual está encargada de reducir la tensión de entrada, su rango de medición es de 0-300 Vrms y entrega una señal senoidal teniendo un rango de variación de 0–3.3 V.

Los equipos medidores de energía están compuestos por un circuito de alta impedancia, un amplificador diferencial y un circuito offset respectivamente.

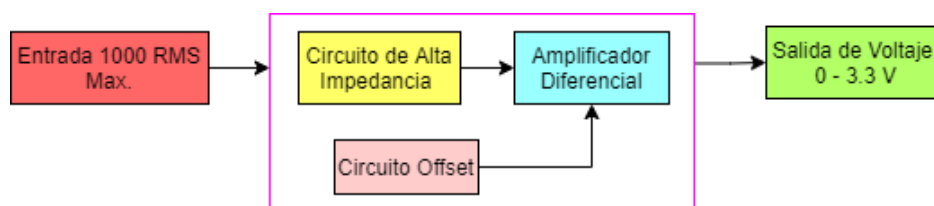


Figura 3.5 Diagrama de bloques del circuito sensor de voltaje

Fuente: Elaboración propia

A continuación, se presentan las especificaciones técnicas del sensor de voltaje:

Voltaje de Entrada Máxima	: 1500 Vac
Frecuencia de Trabajo	: 60 Hz
Resolución de ADC	: 12 bits

5.1.1 CÁLCULOS Y DISEÑO DEL CIRCUITO

- ETAPA DE DISEÑO DEL CIRCUITO DE ATENUACIÓN PARA SEÑALES DE LÍNEA NEUTRO**

Según las normas IEC 62053 – 31:2020 y IEC 62052 – 11:2020, el equipo debe sensor hasta un voltaje máximo de 300 Vrms.

Calculando:

$$V_{max} = V_{rms} * \sqrt{2}$$

$$V_{max} = 300 * \sqrt{2}$$

$$V_{max} = 424 \text{ Voltios}$$

Donde:

V_{max}: Voltaje máximo en un instante de tiempo

V_{rms}: Energía entregada por la señal senoidal en un periodo de tiempo

Se concluye que, para 300 Voltios de medición, se debe tener un voltaje instantáneo máximo de 424 Voltios.

- **MÉTODOS DE ATENUACIÓN DE SEÑAL ALTERNA**

Para realizar la medición de voltaje se tienen dos métodos, por los que serían:

- **CONVERSIÓN POR TRANSFORMADOR**

Se puede utilizar un transformador de tensión alterna con una relación $N_1:N_2$, en el que el voltaje de entrada máximo sea 300 Voltios en RMS y el voltaje de salida sea de 3.3 Voltios RMS.

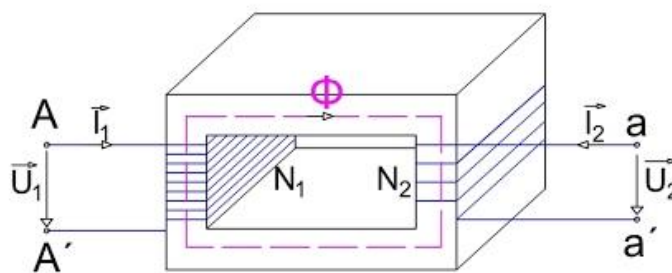


Figura 3.6 Relación de transformación en un transformador monofásico

Fuente: https://www.youtube.com/watch?v=UbalPc_5sdk

Se procede a calcular la relación de transformación, basándonos en las ecuaciones ya conocidas de un transformador de voltaje.

$$\begin{aligned}\frac{V_1}{V_2} &= \frac{n_1}{n_2} \\ \frac{300}{3.3} &= \frac{n_1}{n_2} \\ 90 &= \frac{n_1}{n_2} \\ n_1 &= 90 * n_2\end{aligned}$$

Donde:

V1: Voltaje de entrada del bobinado primario

V2: Voltaje de salida del bobinado secundario

n_1 y n_2 : Numero de vueltas en cada bobinado

Donde se tiene que el voltaje primario máximo es de 300 Vrms y el voltaje deseado del secundario es de 3.3 Vrms.

Nota: al aumentar el bobinado, estamos teniendo una carga altamente inductiva en el lado de N_1

Ejemplo:

Se plantea que N_2 debe tener 40 vueltas por lo tanto para N_1 se calcula lo siguiente

$$n_1 = 303 * 40$$

$$n_1 = 12120$$

Dada esta relación de bobinas, el transformador resulta de un gran tamaño, pero para tener en cuenta que el transformador actúa como un filtro, por lo que nos limitara el ancho de banda de la señal de voltaje y modificara el ángulo de fase de la señal, lo que alteraría la medición del factor de potencia.

Lo que lleva a que el recibo de energía por potencia se vea afectado, por lo tanto, el uso de un transformador no nos conviene.

- **REDUCCIÓN POR RESISTENCIAS**

Las resistencias actúan como un divisor de voltaje, lo que nos permite reducir en 100 de veces una señal sin alterar su fase dado que el convertidor análogo digital del microcontrolador cuenta con un voltaje máximo de entrada de 3.3 Voltios.

Pero se complica al tratarse de voltajes muy altos por el consumo de corriente que pasan por las resistencias.

Entonces procedemos a calcular teniendo en cuenta las normas establecidas.

$$V_{rms} = 300 V \rightarrow \text{según norma IEEE}$$

$$V_{pico} = V_{rms} * \sqrt{2}$$

Sustituimos los valores para poder hallar el voltaje pico:

$$V_{pico} = 300 * \sqrt{2}$$

Obteniendo como resultado:

$$V_{pico} = 424 V$$

Dado que el ADC tiene voltaje máximo de 3.3 V y la onda ADC es alterna, entonces:

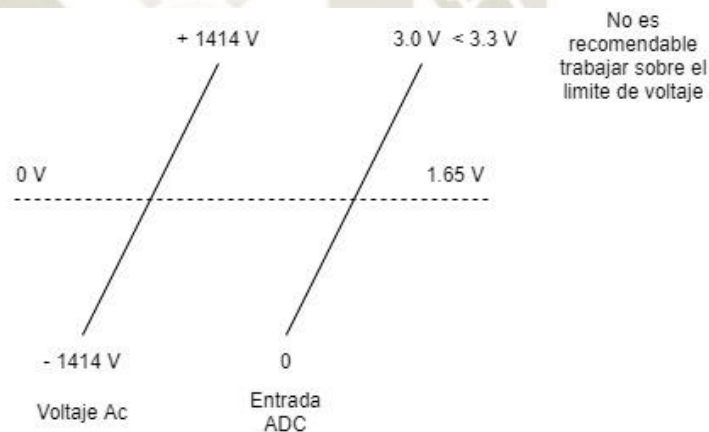


Figura 3.7 relación ente voltaje AC y entrada ADC

Fuente: Elaboración propia

De acuerdo a la figura 3.7, se tiene los puntos:

P1 (1414, 3.0)

P0 (0, 1.65)

Entonces utilizamos la ecuación de la recta

$$m = \frac{y_1 - y_0}{x_1 - x_0}$$

Sustituyendo valores tenemos:

$$m = \frac{3.0 - 1.65}{1414 - 0}$$

Obteniendo como resultado:

$$m = 954.738 \times 10^{-6}$$

Entonces, sustituimos en la siguiente ecuación:

$$y - y_0 = x_1 - x_0$$

$$y - 1.65 = m (x_1 - 0)$$

$$y = mx + 1.65$$

Donde:

$$y = 954.738 * 10^{-6} x + 1.65$$

Ahora procedemos a hacer uso de la ecuación de OPAMP diferencial tomando en cuenta la figura 3.8.

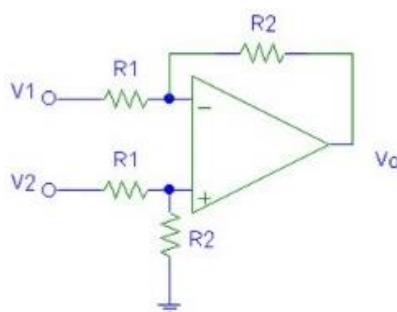


Figura 3.8 Amplificador diferencial

Fuente: <http://amplificadoroperacional.blogspot.com>

$$V_o = G V_{in} + V_{ref}$$

Si $Y = V_o$ y $V_{in} = x$

Ahora procedemos a realizar el diseño:

$$G = \frac{R_f}{R}$$

$$R_f = 954.738 * 10^{-6} R$$

Siendo $R = 18 \text{ M}\Omega$, dándole un valor alto para una impedancia alta de entrada.

$R_f = 17.18 \text{ k}\Omega$, como no es comercial entonces se ajusta a la más cercana que sería $R_f = 15 \text{ k}\Omega$.

Análisis:

$V_{rms} = 300 \text{ V}$

$V_{pico} = 424 \text{ V}$

Tomando en cuenta la figura 3.8 sustituimos en la fórmula:

$$V_o = V_{in} * G$$

$$V_o = \left(\frac{15 \text{ K}}{18 \text{ M}\Omega} \right) V_{in}$$

$$V_o = 833.33 * 10^{-6} (1500)$$

$$V_o = \pm 1.25 \text{ V}$$

Ecuación completa

$$V_o = G V_{in} + V_{ref}$$

$$V_o = \pm 1.25 + 1.65$$

$$V_o \text{ máx.} = 2.9 \text{ V} \leftarrow 424 \text{ Vpico}$$

$$V_o \text{ mín.} = 0.24 \text{ V} \leftarrow -424 \text{ Vpico}$$

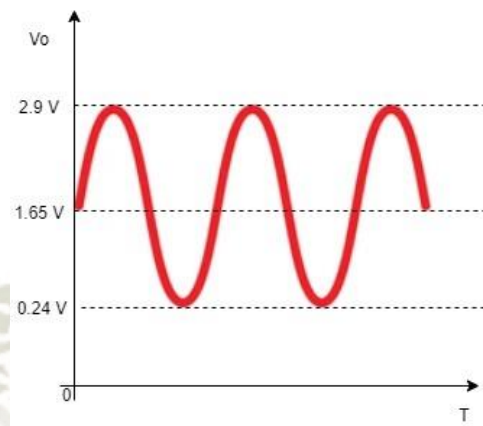


Figura 3.9 Onda del amplificador diferencial

Fuente: Elaboración propia

Por otro lado, sabemos que el valor límite de potencia de las resistencias esta ya establecido, por ende, seleccionamos la resistencia inmediata superior

Potencias comerciales:

$$\frac{1}{8}, \frac{1}{4}, \frac{1}{2}, 1 \text{ watt}$$

$$\frac{1}{8} = 0.125 \text{ watt}$$

$$\frac{1}{4} = 0.25 \text{ watt}$$

$$\frac{1}{2} = 0.5 \text{ watt}$$

Se selecciona el valor de 0.5 watt ya que es el inmediato superior de 0.36. Pero al tener un divisor de voltaje, se tiene conocimiento de que en la resistencia de $4.5\text{M}\Omega$, caerá una tensión de $300\text{V} - 3.3\text{v} = 297$ Voltios, entonces, calculamos el valor de la corriente.

$$I = \frac{997V}{4.5M\Omega}$$

Teniendo como resultado:

$$I = 0.06 \text{ mA}$$

Nota: el solo hecho de que una resistencia de medio watt soporte una tensión de 997 Voltios es razonablemente peligrosa, por lo tanto, se opta por dividir la carga de tensión de una sola resistencia en pequeñas resistencias que en su totalidad sumaran la de una de alta impedancia, conociendo ya porque es razonable dividir las dado que una resistencia con tan poca potencia sea capaz de soportar una sobretensión, dé modo que nos facilita el poder conseguir resistencias comerciales.

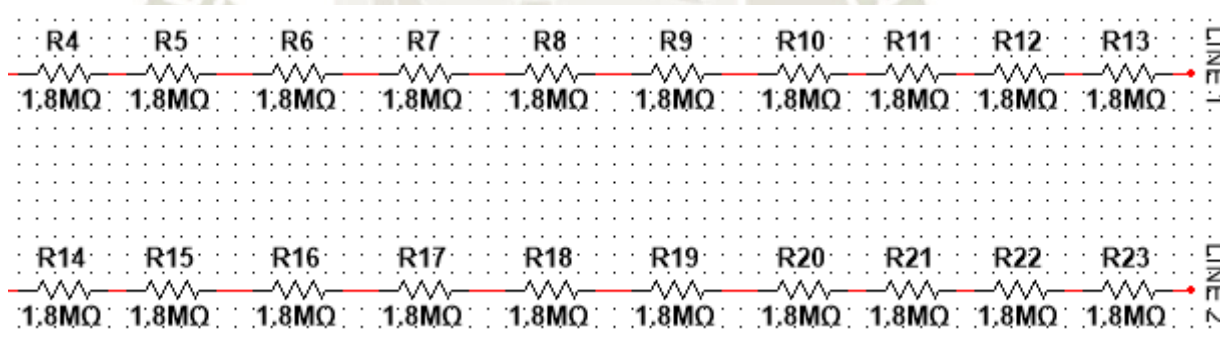


Figura 3.10 Divisor resistivo de varias impedancias

Fuente: Elaboración propia

Por lo tanto, se deduce que:

- El uso de un transformador como reductor de voltaje para el cumplimiento de las normas IEC 62053 – 31:2020 y IEC 62052 – 11:2020 nos llevaría a tener un transformador de gran tamaño y sería altamente inductivo lo que llevaría a filtrar la señal de entrada y mover su fase con respecto a la corriente sensada.

- El uso de resistencia de reducción de voltaje no modifica la fase de señal de voltaje sentido y mantienen toda la información de onda requerida.
- Por lo tanto, se escoge como método de sentido de voltaje el de “reducción por resistencias”.

Ahora se procede a realizar los cálculos con el método de sentido escogido que sería el de reducción por resistencias con lo que se calcula para un voltaje de entrada de 220 Vrms a continuación:

$$V_p = 220V_{rms} * \sqrt{2}$$

Obteniendo como resultado:

$$V_p = 311 \text{ Voltios}$$

Ahora procedemos a realizar el cálculo para el voltaje de salida mediante la siguiente ecuación:

$$V_o = V_{in} * K$$

- a) Primero calculamos para un voltaje de entrada máximo 311 Voltios.

$$V_o = 311 * 3.14136e - 3$$

$$V_o = 976.943 \text{ mV}$$

- b) Ahora hallamos para un voltaje de 220 Vrms utilizando la ecuación 3.9:

$$V_o = 220V_{rms} * 3.14136e - 3$$

$$V_o = 690.8 \text{ mV}$$

- **CÁLCULO PARA VERIFICAR QUE LA RESOLUCIÓN DE 12 BITS ES CORRECTA**

El convertidor análogo digital es quien por diseño especifica la resolución del voltaje, esto quiere decir que el elemento que detecta el voltaje es el convertidor ADC.

Por lo tanto:

$$ResolVADC = \frac{3.3}{4095} = \frac{V_{cc}}{12Bits}$$

Esto quiere decir que $\frac{V_{cc}}{12Bits}$ es el menor voltaje que el convertidor ADC puede cuantificar.

Por ende:

$$ResolADC = 805\mu V$$

Entonces siendo el voltaje pico para una señal de 220Vrms es de 311 Voltios a la entrada del circuito atenuador a la salida de este es de 976.943MV.

Entonces calculamos el número de pasos a continuación:

$$Número\ de\ pasos = \frac{976.943e - 3}{805e - 6}$$

$$Número\ de\ pasos = 1214$$

El convertidor análogo digital para cuantificar una señal de 311 Voltios, este utilizará 1214 pasos.

Por lo tanto, no es necesario un amplificador.

Recalculando con un convertidor ADC de 10 bits

$$ResolADC = \frac{3.3}{1024} = 3.22mV$$

Entonces calculamos el número de pasos a continuación:

$$Número\ de\ pasos = \frac{976.943e - 3}{3.22e - 3}$$

$$Número\ de\ Pasos = 303.39$$

Entonces decir que con 12 bits tenemos una mejor señal de resolución de 300 pasos, esto para tener más precisión en la captura de forma de onda, de tal modo que esto conviene porque no se necesitaría hacer un amplificador de señal ya que con 10 bits llegamos a una resolución de 303.39.

Nota: con el circuito diseñado hemos reducido 300 Voltios a 3.15 v con una ganancia de $3.14136e-3$, debido que tenemos un voltaje offset de 1.65 Voltios la señal se corta en 3.3 Voltios ya que el Opamp es de rail to rail

Entonces, se tomó como voltaje de entrada el valor RMS de 300 Voltios y se calculó resistencias de reducción de $4.7\ M\Omega$ y $15K\Omega$, por lo tanto, en el voltaje de salida se obtuvo una señal de 3.15 Voltios RMS

Pero al ingresar el voltaje offset de 1.65 Voltios, el Opamp se saturaba, esto sucedió por que no se considero que el valor pico de la señal es de 4.45 Voltios en zona positiva y en zona negativa es de -4.45 Voltios, lo que da una amplitud total de 8.9 Voltios

Esto significa que al ingresar una señal de 300 Voltios se tiene un voltaje pico positivo de $300 * \sqrt{2} = 424$ Voltios y un valor negativo maximo de -424 Voltios.

Lo que tiene una amplitud maxima de $v_{pp} = 2 * v_p = 424 * 2 = 848$ Voltios.

- **RECALCULANDO LA ATENUACIÓN DEL CONJUNTO DE RESISTENCIAS**

Calcularemos nuevamente la atenuación del conjunto de resistencias, entonces:

$$V_o = v_{in} * \frac{R_3}{R_3 + R_2}$$

$$3.3 = 2800 * \frac{15e3}{15e3 + R_2}$$

$$R_2 = 12.7 \text{ M}\Omega$$

Resistencias de distribución $\frac{R_2}{7} = 1.81 \text{ M}\Omega$

Señal para un voltaje de entrada de 220 VRMS, lo que nos da una señal de salida de 260mV

Siendo el voltaje maximo de 368 mV

Pero una señal de $V_{pp} = 736 \text{ mV}$

ResolADC = $805e-6$

$$\text{Numero de pasos} = \frac{736e-3}{805e-6}$$

$$\text{Numero de pasos} = 914$$

No se requiere amplificar porque tiene cerca de 300 valores que representa la señal senoidal.

Porque el convertidor análogo digital es de 12 Bits

- **CÁLCULO DE LA POTENCIA DE LAS RESISTENCIAS CON LOS NUEVOS VALORES**

Procedemos a calcular la potencia partiendo de ecuaciones ya conocidas como:

$$I = \frac{V}{R}$$

Donde el $V=220 \text{ Vrms}$

$$R = 1.8 * 7$$

$$R = 12.6 \text{ M}\Omega$$

Hallamos I_{rms} :

$$I_{rms} = \frac{220V_{rms}}{12.6M\Omega}$$

$$I_{rms} = 0.17 \text{ uA rms}$$

Hallamos la potencia mediante la siguiente formula:

$$P_{rms} = I_{rms}^2 * R$$

$$P_{rms} = (0.17E-6)^2 * (12.6E6)$$

$$P_{rms} = 0.364 \text{ uWatt}$$

Entonces decimos que 7 resistencia en serie de $\frac{1}{4}$ watt, pueden soportar una potencia promedio de $7 * 0.25 \text{ Watt} = 1.75 \text{ Watt}$. Entonces decimos que el repartir la potencia entre varias resistencias mejora la distribución de potencia entre ellas calcular $P_{max} = 0.634 \mu\text{Watt} * \sqrt{2} = 0.8 \mu\text{watt}$, lo que indica que una resistencia de $\frac{1}{4}$ de Watt puede soportar $0.25 \text{ Watt} > 0.8 \mu\text{Watt}$

Sensor de voltaje diseñado para soportar hasta 300 Vrms.

- CÁLCULOS PARA EL FILTRO PASABAJO

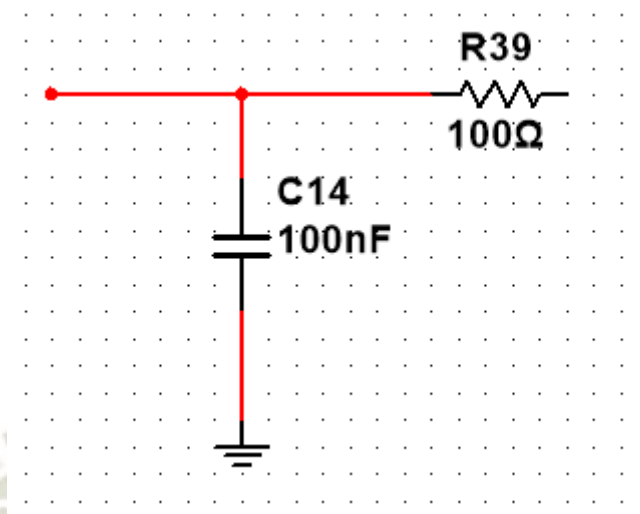


Figura 3.11 Circuito de un filtro pasa bajo

Fuente: Elaboración propia

Calculamos la frecuencia mediante la siguiente formula:

$$F = \frac{1}{2 * \pi RC}$$

$$F = \frac{1}{2 * \pi * 100 * 100E - 9}$$

$$F = 15915 \text{ Hz}$$

El valor aproximado de la frecuencia seria:

$$F = 15.9 \text{ KHz}$$

En este caso el filtro pasa bajo elimina las frecuencias de 16 KHz para arriba, lo que evita señales de ruido de alta frecuencia

5.1.2 DISPOSITIVOS SELECCIONADOS

- **CIRCUITO ATENUADOR DE VOLTAJE Y DIVISOR RESISTIVO**

Para el circuito reductor de potencia se aplicaron resistencia de precisión %1, con una potencia de $\frac{1}{4}$ W con una variación de 5 % ya que son resistencias de carbón.

- **AMPLIFICADOR DIFERENCIAL**

Es muy importante utilizar un OPAMP con una salida de tensión Rail to Rail y un voltaje offset bajo.

También se requiere que el Opamp trabaje con una sola fuente de alimentación.

5.1.3 ESQUEMÁTICO DEL CIRCUITO

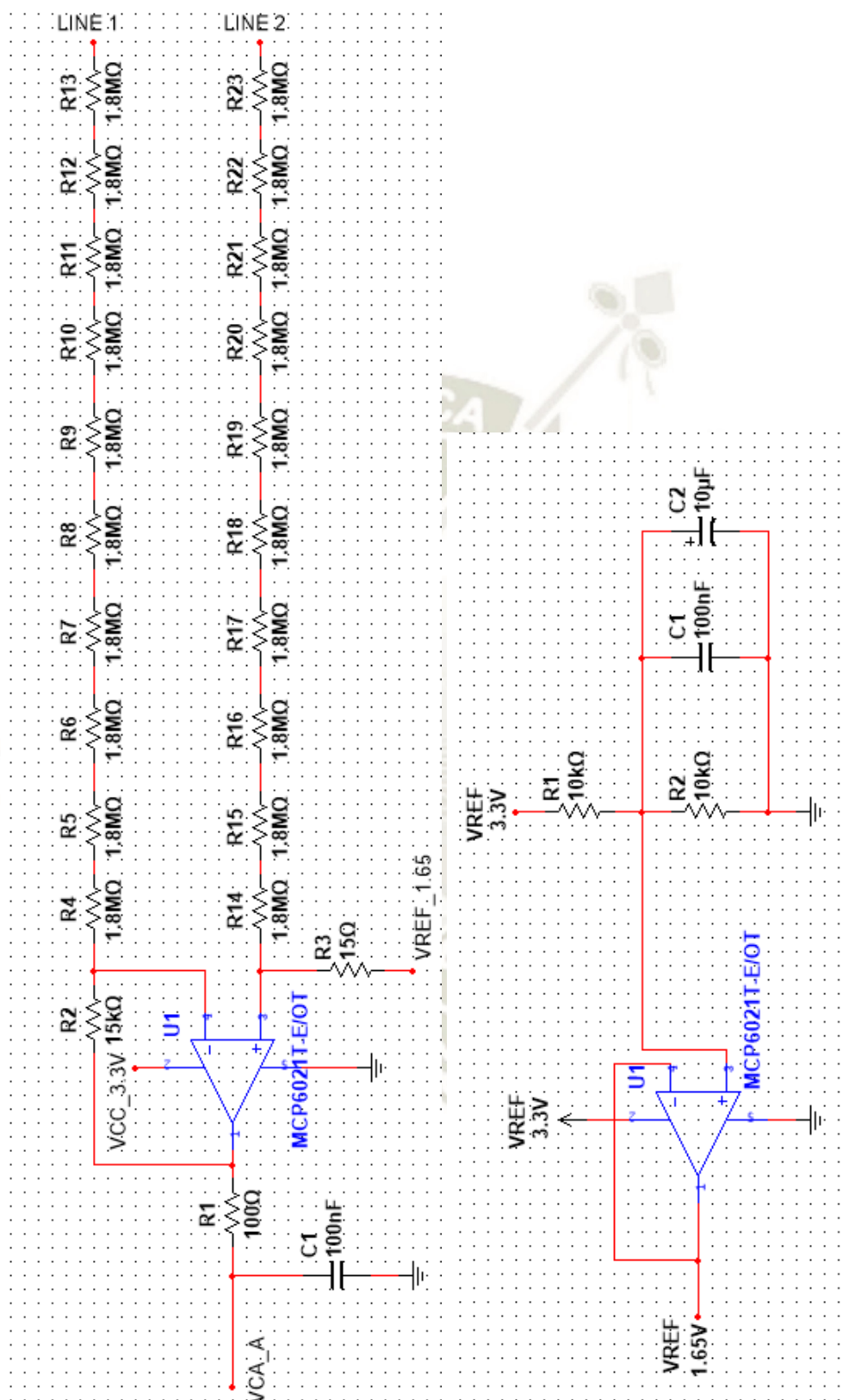


Figura 3.12 Circuito general del sensor de voltaje

Fuente: Elaboración propia

5.2 SENSOR DE CORRIENTE

Este circuito se encarga de sensar la corriente que consume la carga eléctrica de la vivienda, esto es gracias al sensor de corriente ACS 756-050 el cual entrega una señal senoidal con un rango de variación de 0-50 A que debe ser acondicionada para ser sensada por el procesador.

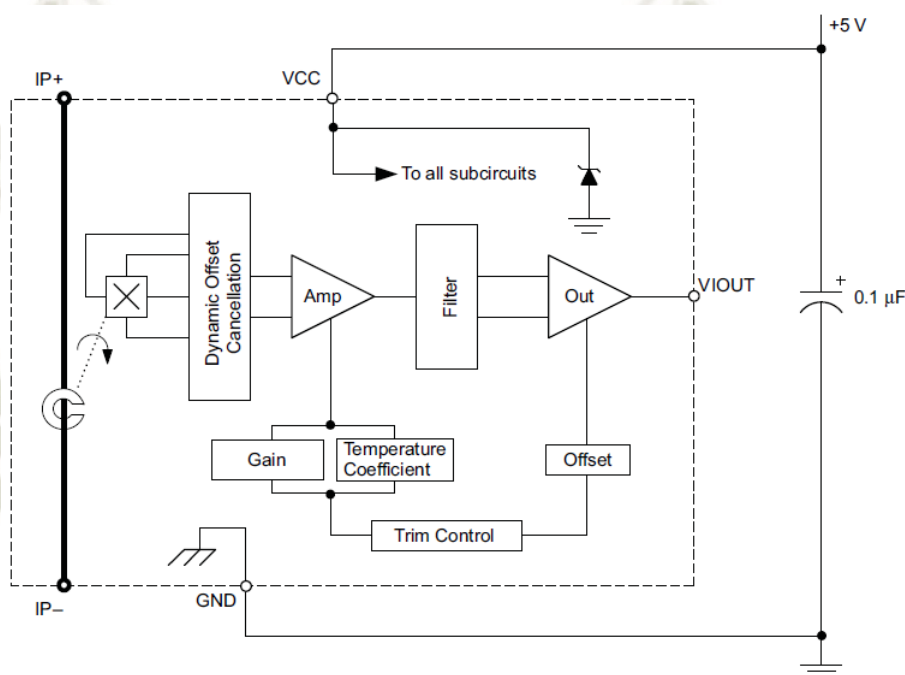


Figura 3.13 Diagrama de bloques del sensor de corriente

Fuente: Hoja de datos del fabricante

A continuación, se presentan las especificaciones técnicas del sensor de corriente:

Voltaje de Entrada	: 3 V – 5.5 V
Corriente Máxima	: 50 Arms
Sensibilidad	: 40 m V/A
Impedancia de Entrada	: 100 u ohm
Frecuencia Corriente Continua	: 120Hz

5.2.1 CÁLCULOS Y DISEÑO DEL CIRCUITO

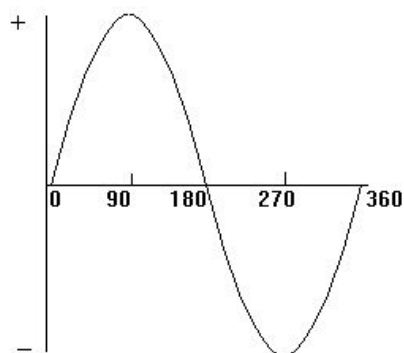


Figura 3.14 Onda que ingresa al sensor de corriente

Fuente: <https://www.diarioelectronico hoy.com/blog/leccion-9-conocimientos-basicos>

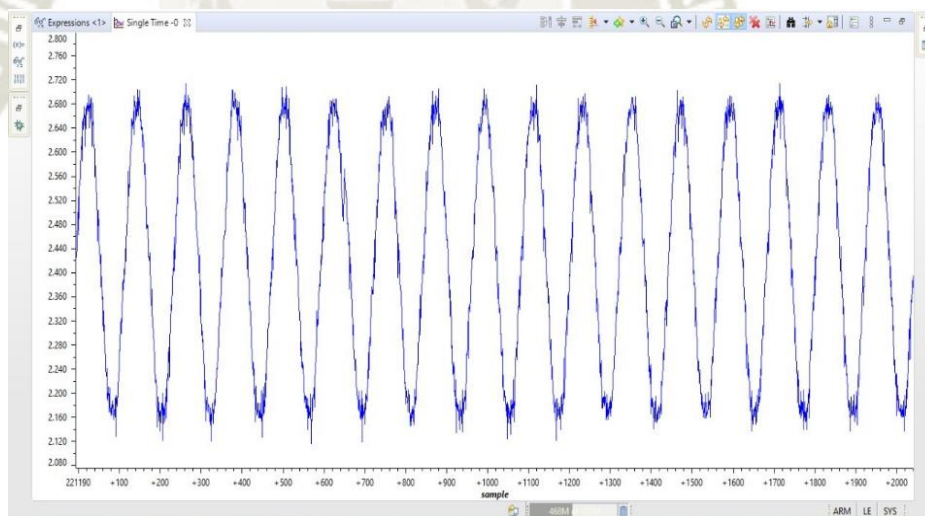


Figura 3.15 Señal de salida del sensor

Fuente: Elaboración propia

- **CORRIENTE MÁXIMA**

Procedemos a calcular la corriente máxima que pasa por el sensor mediante la siguiente formula:

$$50 \text{ rms} \rightarrow A(T) = A \text{ max} * \sin (wt)$$

$$A \text{ max} = A \text{ rms} * \sqrt{2}$$

$$A \text{ max} = 50 * \sqrt{2}$$

Obteniendo como resultado de la corriente máxima:

$$A \text{ max} = 70.71 \text{ A}$$

Para conocer el voltaje pico pico del sensor, utilizamos la ecuación anterior:

$$A(T) = A \text{ max} * \sin (wt)$$

$$A(T) = 70.71 * \sin (wt)$$

El valor pico máximo lo calculamos así:

$$A(T) = 70.71 * \sin (90)$$

$$A(T) = 70.71 \text{ A}$$

El valor pico mínimo lo calculamos así:

$$A(T) = 70.71 * \sin (-90)$$

$$A(T) = -70.71 \text{ A}$$

El sensor de corriente reduce la señal para convertirlo en voltaje y reducir su tamaño de la señal desde 70.71 A hasta -70.71 A pico pico.

- **CÁLCULO DE LA FUNCION DE TRANSFERENCIA**

Conociendo la sensibilidad del sensor:

$$Sensibilidad = 40 \frac{[mV]}{[A]}$$

Procedemos a calcular la función de transferencia de acuerdo a la siguiente ecuación:

$$Signal_{out} = Signal_{In} * Sencibility$$

$$Signal_{In} = Amax * \sin(wt)[A]$$

$$Signal_{out} = Amax * \sin(wt)[A] * \frac{40[mV]}{[A]}$$

$$Signal_{out} = (Amax * \sin(wt) * 40m)[V]$$

$$Signal_{out} = (Amax * \sin(wt) * 40) * (1E - 3)[V]$$

$$Signal_{out} = \frac{(40 * Signal_{In})}{300} [V] + 2.5$$

$$\frac{Signal_{out}}{Signal_{In}} = \frac{40}{300} [V] + 2.5$$

$$TF = \frac{40}{300} [V] + 2.5$$

5.2.2 DISPOSITIVOS SELECCIONADOS

- **RESISTENCIAS DE SENSIBILIDAD**

Para el circuito reductor de potencia, se aplicaron resistencias de carbón con una variación de 5 % y a $\frac{1}{4}$ de Watt.

- **SENSOR DE CORRIENTE ACS 756 – 50 A**



Figura 3.16 Sensor de corriente ACS 756 – 50 A

Fuente: Hoja de datos del fabricante

5.2.3 ESQUEMÁTICO DEL CIRCUITO

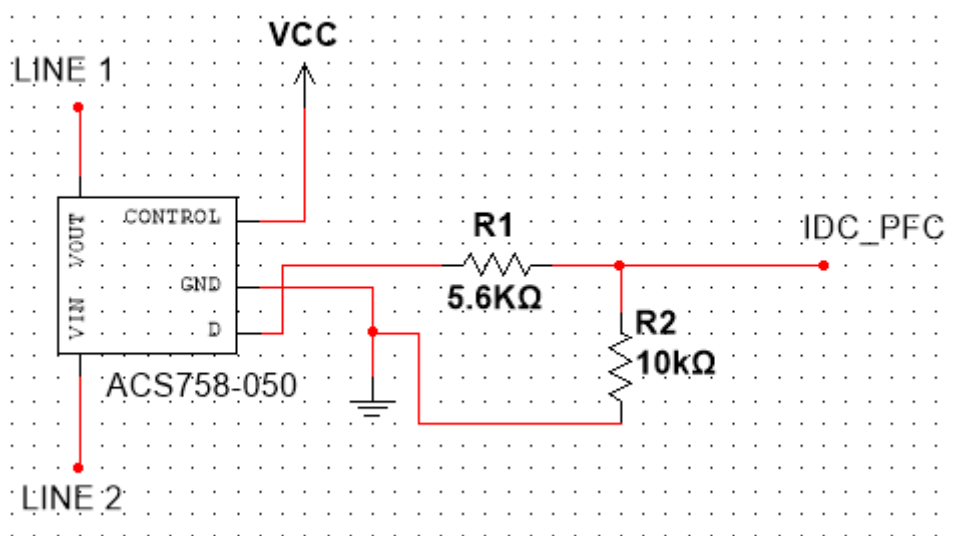


Figura 3.17 Circuito general del sensor de corriente

Fuente: Elaboración propia

5.3 SENSOR DE FRECUENCIA

Este circuito tiene como función tomar la señal de voltaje sensado por el divisor resistivo y compararlo con una señal de referencia para poder obtener una señal cuadrada con la misma frecuencia que tiene la señal de voltaje de línea con un rango de frecuencia de 40 – 70 Hz.

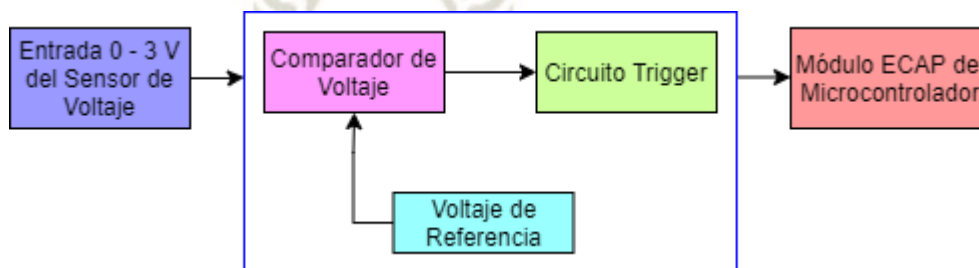


Figura 3.18 Diagrama de bloques del circuito sensor de frecuencia

Fuente: Elaboración propia

A continuación, se presentan las especificaciones técnicas del sensor de frecuencia:

Voltaje de Entrada	: 0 – 3.3 V
Voltaje de Salida	: 0 – 3.3 V
Rango de Frecuencia	: 40 – 70 Hz

5.3.1 CÁLCULOS Y DISEÑO DEL CIRCUITO

- CÁLCULOS PARA EL COMPARADOR DE VOLTAJE**

Teniendo conocimiento del circuito de control offset del sensor de voltaje, concluimos que para el comparador tiene un voltaje $\frac{V_{cc}}{2}$.

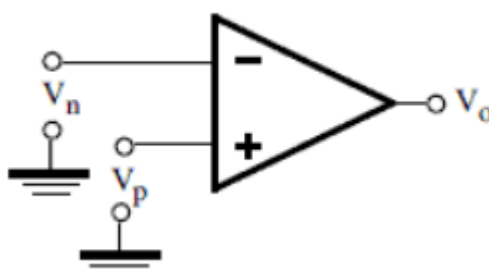


Figura 3.19 Comparador de voltaje

Fuente: <https://unicrom.com/comparadores-de-voltaje/>

- CIRCUITO SCHMITT TRIGGER**

La salida del comparador de voltaje no nos entrega una señal cuadrada bien definida, por lo que tendríamos que hacer pasar la señal por un juego de compuertas Schmitt Trigger, con el único fin de que entreguen una señal cuadrada bien definida manteniendo la frecuencia de la señal análoga original.

5.3.2 DISPOSITIVOS SELECCIONADOS

- COMPARADOR DE VOLTAJE**

Utilizaremos el circuito integrado LM311 con son comparadores de voltaje de alta velocidad. Está diseñado para funcionar con una amplia gama de voltajes de entrada como ± 15 V y puede funcionar con tan solo 5 V, su mínima corriente de consumo es de 250 nA. (Hoja de datos del fabricante)

Tiene un rango de temperatura de 0°C a $+70^{\circ}\text{C}$.

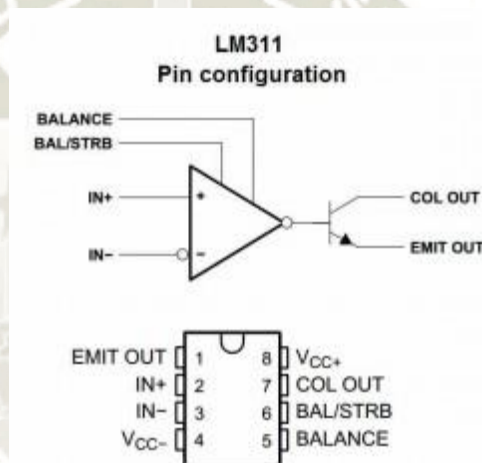


Figura 3.20 Comparador LM311

Fuente: <https://ssdielect.com/cb/comparadores-de-voltaje/77-lm311.html>

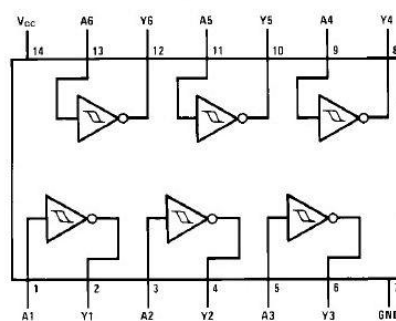
- CIRCUITO SCHMITT TRIGGER**

Para el circuito de Schmitt Trigger se utilizará el circuito integrado 74HC14 perteneciente a la familia de los CMOS, posee seis inversores, la histéresis de las entradas incrementa la inmunidad al ruido y elimina el cambio abrupto tanto de fase como de frecuencia. (Hoja de datos del fabricante)

Su principal funcionamiento es recuperar señales de onda cuadradas que se deformaron.

- Voltaje de alimentación: 2 Vcc a 6 Vcc
- Voltaje de salida: 0 Vcc
- Numero de compuertas: 6 compuertas inversoras
- Rango de temperatura de operación: -40°C a +85°C.

Connection Diagram



Function Table

$$Y = \bar{A}$$

Input	Output
A	Y
L	H
H	L

H = HIGH Logic Level
L = LOW Logic Level

Figura 3.21 Circuito Schmitt Trigger

Fuente: Hoja de datos On Semiconductor 74HC14

5.3.3 ESQUEMÁTICO DEL CIRCUITO

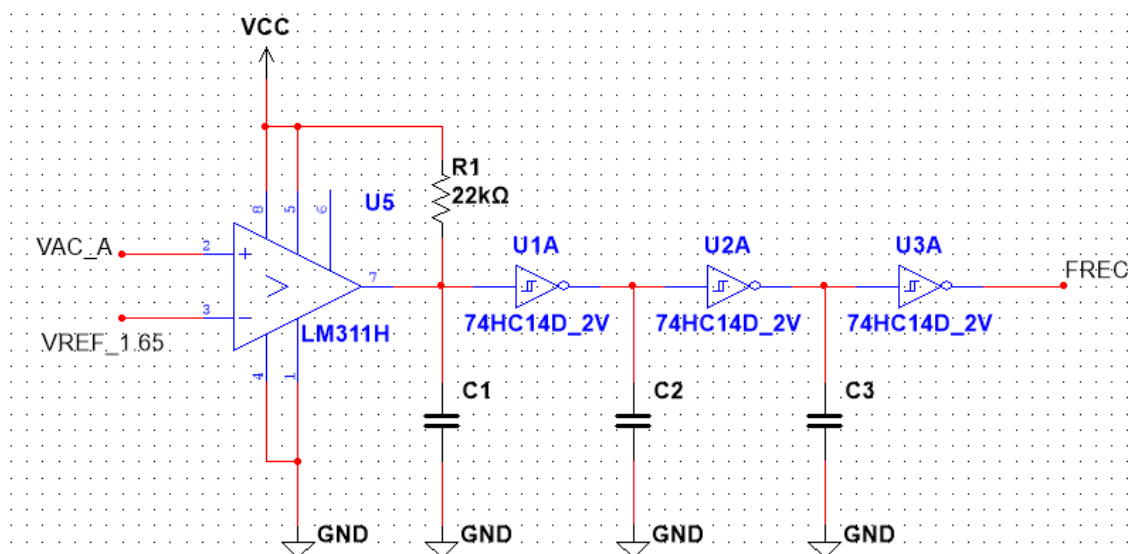


Figura 3.22 Circuito sensor de frecuencia

Fuente: Elaboración propia

6. DISEÑO DE LA ETAPA DIGITAL

6.1 RELOJ EN TIEMPO REAL DS1302

Consiste en un circuito integrado alimentado por una batería el cual permite registrar en todo momento la fecha, la semana, el día y hora.

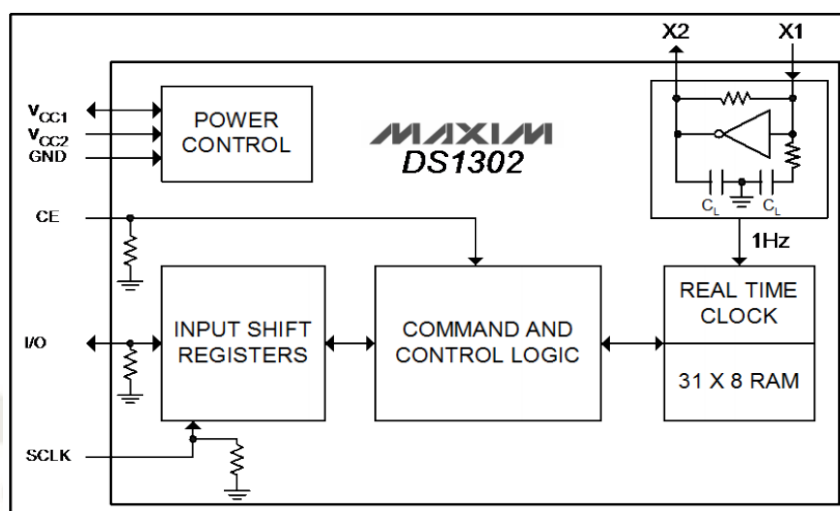


Figura 3.23 Diagrama de bloques interno del reloj en tiempo real

Fuente: Hoja de datos del fabricante

- i. Etapa de control de energía (Power Control)
- ii. Circuito oscilador (X1-X2)
- iii. Entrada de registros desplazados
- iv. Comando y control lógico
- v. Reloj en tiempo real
- vi. Registros 31x8 de memoria RAM

A continuación, se presentan las especificaciones técnicas del Reloj en Tiempo Real

Voltaje de Operación : 2 V – 5.5 V
 Temperatura de Operación : 0°C – 70°C
 RAM de almacenamiento : 31 8

6.1.1 PROTOCOLOS DE COMUNICACIÓN

- **ENTRADA DE DATOS**

Después de los ocho ciclos SCLK que ingresan un byte de comando de escritura, se ingresa un byte de datos en el flanco ascendente de los

siguientes ocho ciclos SCLK. Los ciclos SCLK adicionales se ignoran si ocurren inadvertidamente. Los datos se ingresan comenzando con el bit 0.

- **SALIDA DE DATOS**

Después de los ocho ciclos SCLK que ingresan un byte de comando de lectura, se envía un byte de datos en el flanco descendente de los siguientes ocho ciclos SCLK. Tenga en cuenta que el primer bit de datos que se va a transmitir se produce en el primer flanco descendente después de que se escribe el último bit del byte de comando. Los ciclos de SCLK adicionales retransmiten los bytes de datos en caso de que ocurran inadvertidamente. Esta operación permite la capacidad de lectura en modo de ráfaga continua. Además, el pin de E / S está triestado en cada flanco ascendente de SCLK. La salida de datos comienza con el bit 0.

6.1.2 DISPOSITIVOS SELECCIONADOS

- **RELOJ EN TIEMPO REAL DS132**

Circuito que permite registrar todo en tiempo real, lleva integrado un cristal de cuarzo de 32.768kHz para contar segundos con excelente precisión.

- Reloj en tiempo real valido hasta 2100.
- Sencillo protocolo de comunicación para el microcontrolador
- Operación entre 2 V a 5.5 V
- Utiliza menos de 300 nA en 2 V.
- Rango de temperatura 0°C a 70°C

6.1.3 ESQUEMÁTICO DEL CIRCUITO

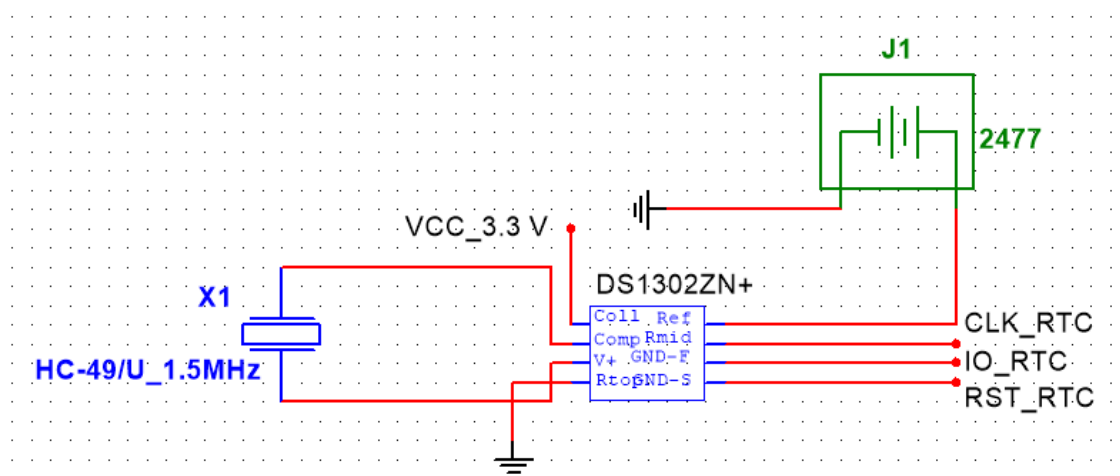


Figura 3.24 Circuito general del reloj en tiempo real

Fuente: Elaboración propia

6.2 MEMORIA EEPROM 25LC256

La interfaz periférica se utiliza para proporcionar las señales de entrada del reloj, entrada de datos y salida de datos necesarias.

La comunicación con el dispositivo se puede detener mediante el pasador de mantenimiento que hace que se ignoren las entradas excepto las interrupciones de mayor prioridad definidas (Hoja de datos del fabricante)

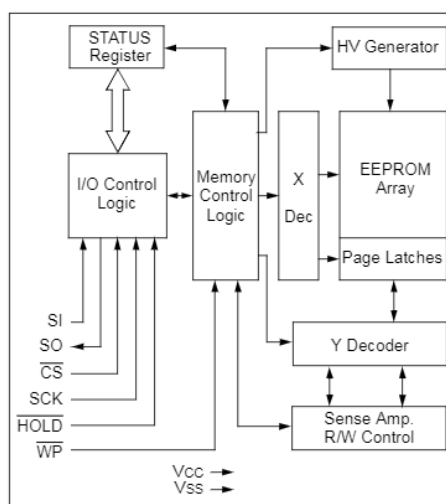


Figura 3.25 Diagrama de bloques interno de la memoria EEPROM 25LC256

Fuente: Hoja de datos del fabricante

A continuación, se presentan las especificaciones técnicas de la memoria EEPROM:

Memoria	: 32 k x 8 bit
Frecuencia	: 10 MHz
Capacidad de Memoria	: 256k bit
Temperatura de Trabajo	: -40°C – 85°C
Número Máximo de Ciclos	: 1 M
Tiempo de Acceso	: 5 ms
Voltaje de Entrada	: 2.5 V – 5.5 V

• PROTOCOLOS DE COMUNICACIÓN SPI

Es un protocolo de comunicación que trabaja en modo full dúplex para recibir y transmitir información, permitiendo así que dos dispositivos puedan comunicarse entre sí al mismo tiempo utilizando canales diferentes.

Dentro de este protocolo se define maestro a aquel dispositivo capaz de transmitir información a sus esclavos. El maestro también puede recibir y enviar información de sus esclavos. Para que todo este procedimiento suceda

es necesario tener registros de desplazamiento, uno para el maestro y uno para el esclavo respectivamente. Estos registros de desplazamiento son los encargados de almacenar los bits de manera paralela para realizar una conversión paralela a serial para la transmisión de información.

Una de sus ventajas más resaltantes de este protocolo es la velocidad de transmisión ya que es configurable a través de software y dependerá también de los dispositivos utilizados (<http://panamahitek.com/como-funciona-el-protocolo-spi/>).

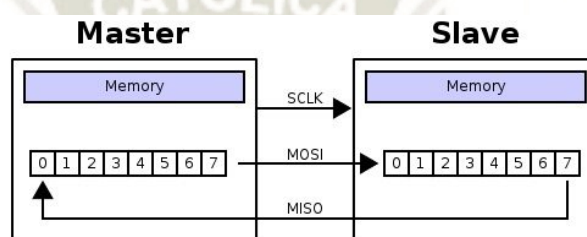


Figura 3.26 Estructura general del protocolo SPI

Fuente: <http://panamahitek.com/como-funciona-el-protocolo-spi/>

6.2.1 ESQUEMÁTICO DEL CIRCUITO

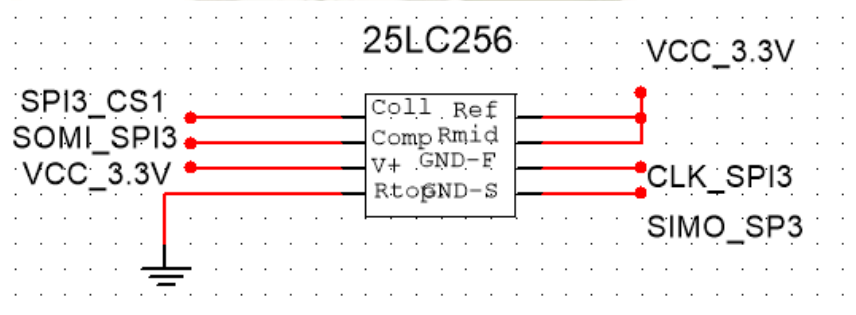


Figura 3.27 Circuito de la memoria EEPROM 25LC256

Fuente: Elaboración propia

6.3 ESQUEMA GENERAL DE LA ETAPA DIGITAL

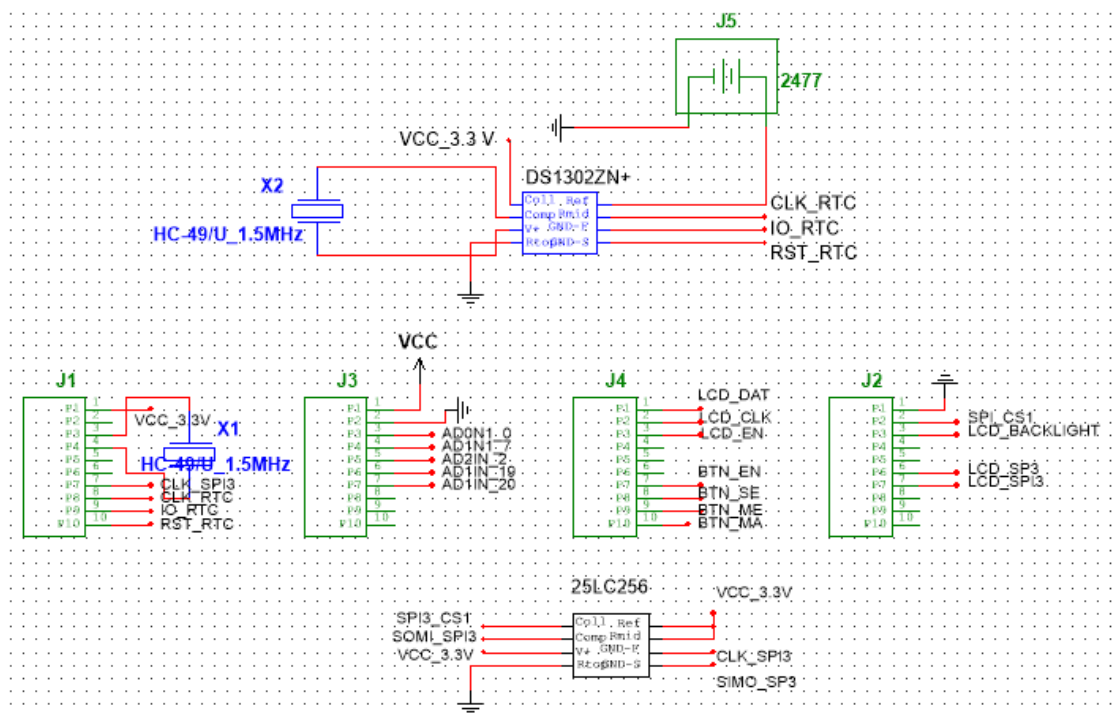


Figura 3.28 Esquema general de la etapa digital

Fuente: Elaboración propia

7. DISEÑO DE LA ETAPA DE VISUALIZACION DE DATOS E INTERFAZ

7.1 BOTONERAS DE CONTROL

- **RESISTENCIAS PULL DOWN**

Cuando el circuito está en reposo, la caída de tensión en la resistencia es prácticamente 0 V, en cambio si pulsamos P1, este dejara pasar la corriente y tendremos una diferencia de potencia de 5 V.

- **RESISTENCIAS PULL UP**

Cuando el circuito está en reposo, P1 sin pulsar, la caída de tensión es de 5 V, en cambio sí pulsamos P1 la caída de tensión es 0 V.

A continuación, se presentan las especificaciones técnicas de las botoneras de control:

Resistencia de Aislamiento	: 100 M ohm Min.
Fuerza Dieléctrica	: 250 VAC
Temperatura de Funcionamiento	: -20 °C a 70 °C

7.1.1 DISPOSITIVOS SELECCIONADOS

- **PULSADORES DE 4 PINES**

El pulsador tiene 4 pines que están conectadas a pares. Cuando pulsamos el interruptor se cierra el circuito y la corriente empieza a fluir. Esto nos permite controlar un LED, un motor o cualquier otro circuito.

7.1.2 ESQUEMÁTICO DEL CIRCUITO

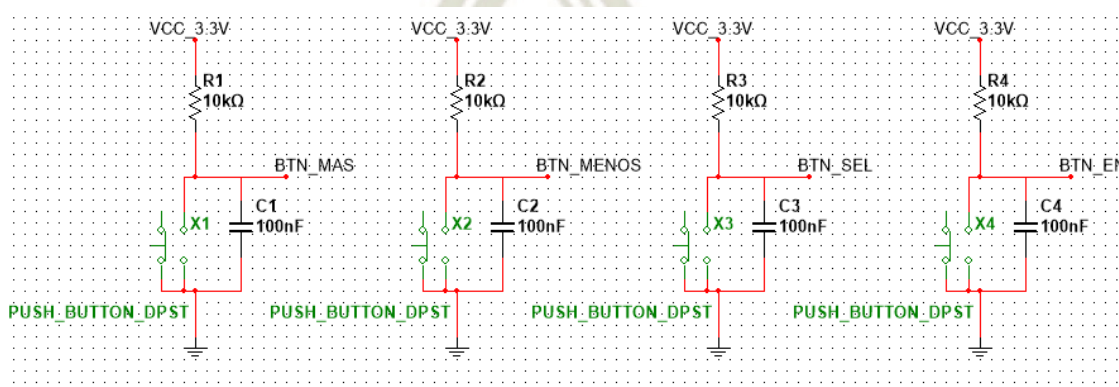


Figura 3.29 Esquema del circuito de pulsadores

Fuente: Elaboración propia

7.2 PANTALLA LCD SERIAL

El equipo cuenta con una pantalla LCD programada para que puedan visualizarse los datos que se requieran como:

- Voltaje
- Corriente
- Frecuencia

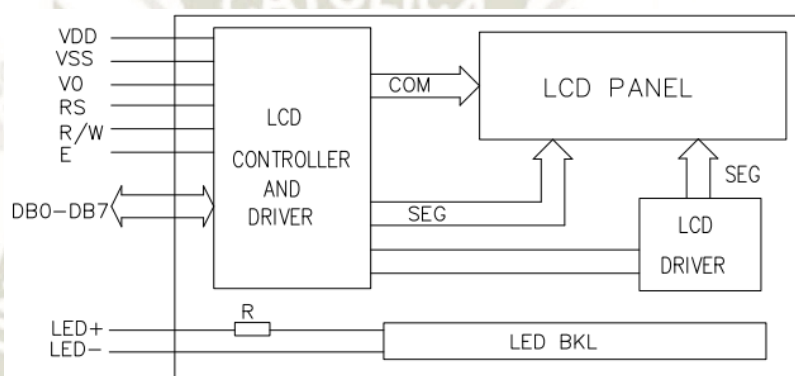


Figura 3.30 Diagrama de bloques de LCD 16 x 2

Fuente: Hoja de datos del fabricante

7.2.1 PROTOCOLO DE COMUNICACIÓN

- **COMUNICACIÓN SERIAL**

Es un bus de comunicaciones en serie que se utiliza para transmitir la información: datos y reloj.

Su función principal es que puede transmitir y recibir datos binarios seriales. Como resultado, convierte un dato paralelo (byte) a un conjunto de pulsos seriales que pueden ser recibidos y enviados por una línea de transmisión.

Este protocolo opera mediante tres condiciones digitales básicas: inicio de transmisión, paridad y fin de transmisión, estas condiciones son sincronizadas mediante un sincronizador internos.

A continuación, se presentan las especificaciones técnicas de la Pantalla LCD:

Voltaje de Entrada	: 5 V
Rango de Temperatura de Operación	: 0 °C - +50°C
Rango de Temperatura de Almacenamiento	: -10 °C - +60 °C

Entonces se procede a utilizar una pantalla de cristal líquido que cuenta con dos filas de 16 caracteres cada una, su función es mostrar información procesada alfanumérica. Su consumo de energía es mínimo y su programación es sencilla.

7.2.2 ESQUEMÁTICO DEL CIRCUITO

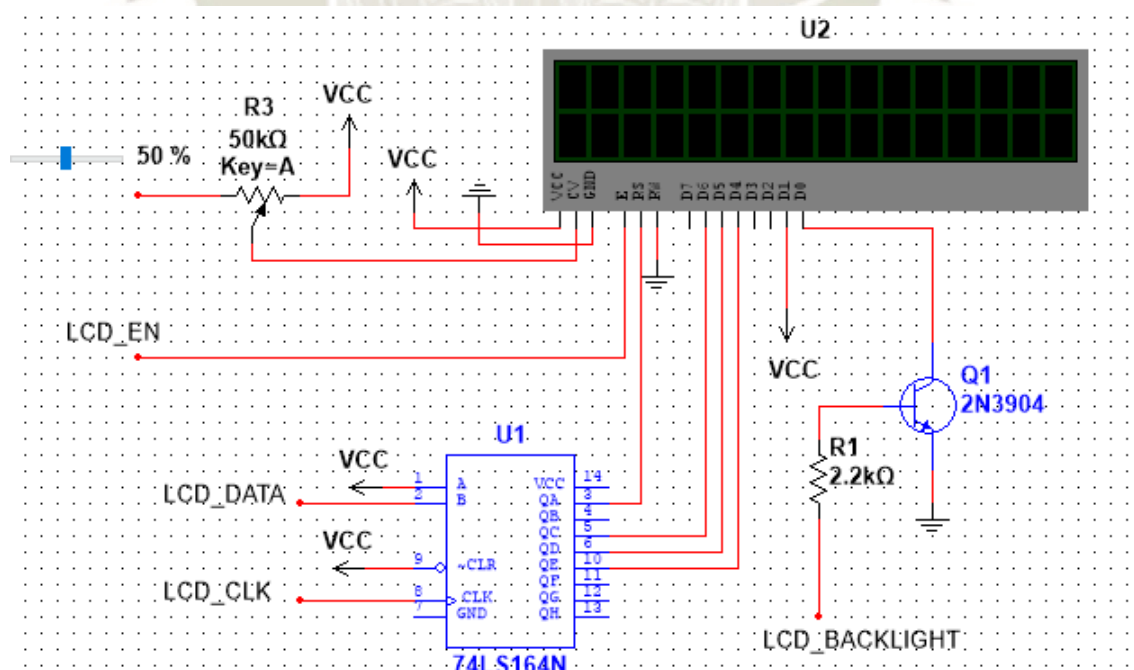


Figura 3.31 Circuito de la etapa de visualización

Fuente: Elaboración propia

CAPÍTULO IV
DISEÑO DE SOFTWARE

En este capítulo se presentan los esquemas generales de software, diagramas de flujo y ecuaciones para poder comprender de manera eficiente el software que el microcontrolador ejecuta para realizar cada una de las tareas propuestas.

1. HALCOGEN

HALCoGen es un software libre que proporciona a los usuarios una interfaz gráfica permitiendo así configurar periféricos, interrupciones, relojes y otros. Este software permite generar librerías en las cuales se encuentran funciones que permiten al usuario habilitar una entrada digital o una salida digital. Además, se pueden adquirir señales por el ADC. (Texas Instrument – Halcogen, 2021)

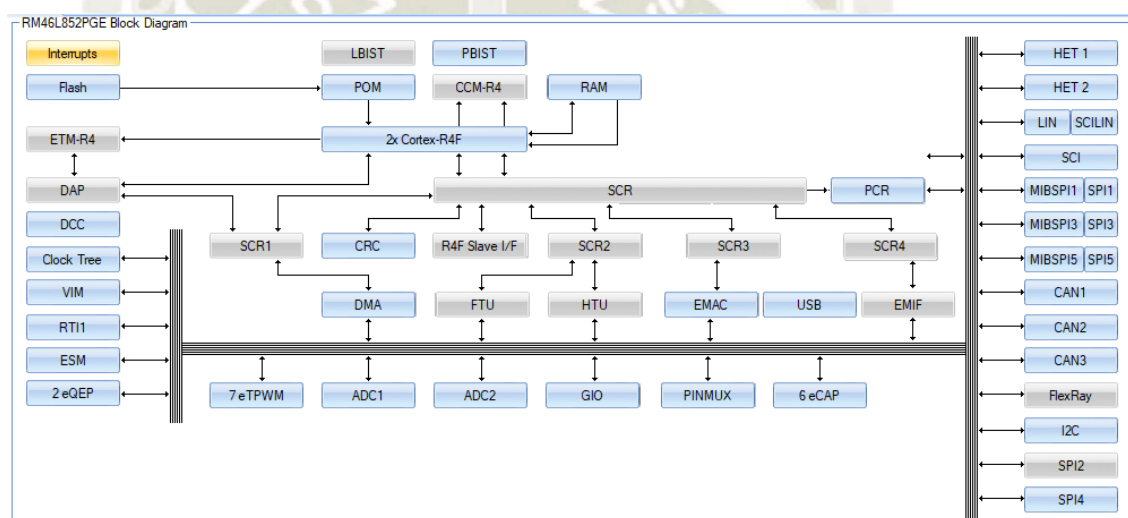


Figura 4.1 Diagrama de bloques de RM46L852

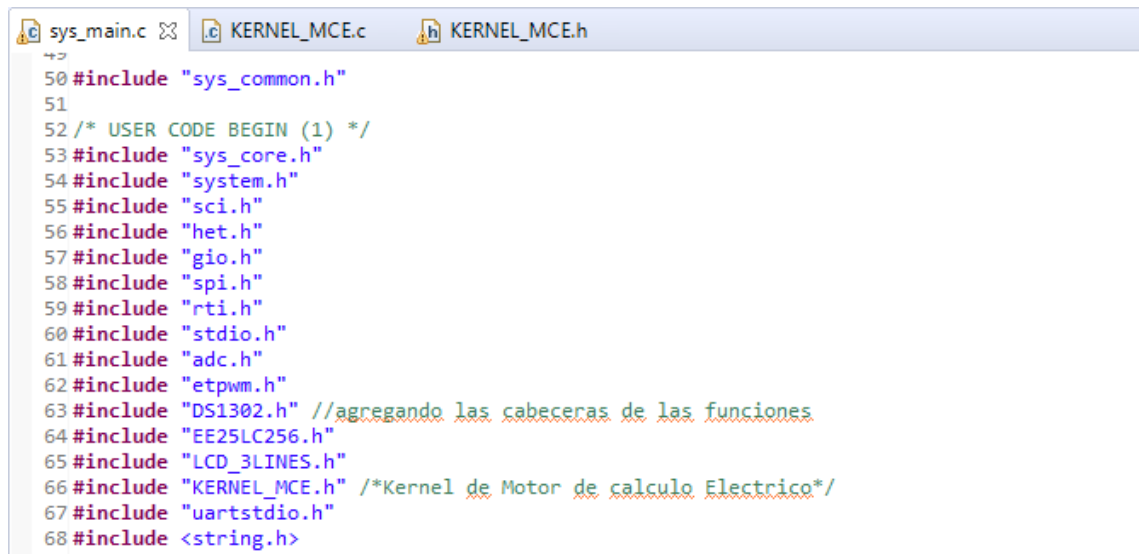
Fuente: HALCoGen

En la figura 4.1 se observa los diferentes módulos internos que tiene el microcontrolador.

2. CODE COMPOSER STUDIO

Code Composer Studio es un software libre que comprende un conjunto de herramientas que se utilizan para desarrollar una programación detallada el cual incluye un compilador C++, un editor de código fuente, un entorno para la creación

del proyecto, depurador, generador de perfiles, etc. (Texas Instrument – CodeComposer, 2021)



```

50 #include "sys_common.h"
51
52 /* USER CODE BEGIN (1) */
53 #include "sys_core.h"
54 #include "system.h"
55 #include "sci.h"
56 #include "het.h"
57 #include "gio.h"
58 #include "spi.h"
59 #include "rti.h"
60 #include "stdio.h"
61 #include "adc.h"
62 #include "etpwm.h"
63 #include "DS1302.h" //agregando las cabeceras de las funciones
64 #include "EE25LC256.h"
65 #include "LCD_3LINES.h"
66 #include "KERNEL_MCE.h" /*Kernel de Motor de calculo Electrico*/
67 #include "uartstdio.h"
68 #include <string.h>

```

Figura 4.2 Estructura principal de Code Composer Studio

Fuente: Code Composer Studio

3. PERIFÉRICOS DEL CONTROLADOR

3.1 CONVERTIDOR ANÁLOGO DIGITAL

El convertidor análogo digital se utilizó para adquirir señales de voltaje y corriente, como se presentó anteriormente en el capítulo 3 Diseño de Hardware, se requiere una resolución de 3.1413 mV para el sensor de voltaje y 40 mV/A para el sensor de corriente, por lo que vi como mejor opción utilizar un ADC de 12 bits que tiene como resolución 860 μ V y un tiempo de muestreo de 0.56 μ s. (Texas Instrument – CodeComposer, 2021)

El equipo trabaja con 3 canales análogo digital, con la siguiente distribución:

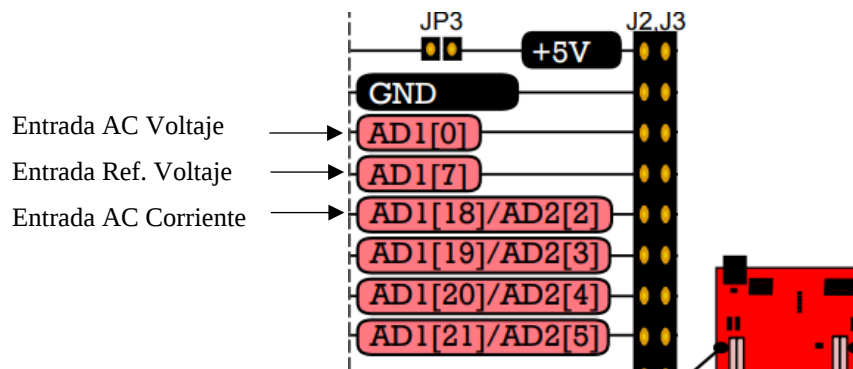


Figura 4.3 Distribuciones de Canales ADC

Fuente: Hoja de datos del fabricante

Los pines operativos serían:

- AD1 [0]: Canal que ingresa la señal de voltaje alterno.
- AD1 [7]: Canal que ingresa la señal de referencia solo voltaje.
- AD2 [2]: Canal que ingresa la señal de corriente alterna.

La señal de referencia, es el voltaje offset enviado por la configuración de los OPAMP, esta señal tiene un voltaje offset único por lo que es de mucha necesidad obtenerlo durante el proceso, también permite que pueda volver discreta a la señal de voltaje que ingresa con valores netamente positivos a la memoria del controlador al realizar el muestreo. Esta señal muestreada me permite mover dicha señal netamente positiva a cero y darle valores positivos y negativos, lo que les permite a las librerías DSP un cálculo mucho más preciso del RMS, FFT, MAX, MIN, etc. (HALCoGen, 2020)

En el caso de la señal de referencia de la corriente no es necesario ya que se utiliza el que sale del integrado ACS756, el cual es estable.

Estas señales de referencia nos permiten tener una señal con valores positivos y negativos en la memoria del microcontrolador. Ayudando también en el procesamiento digital de señales. (HALCoGen, 2020)

La configuración del ADC1 y ADC2 son muy similares:

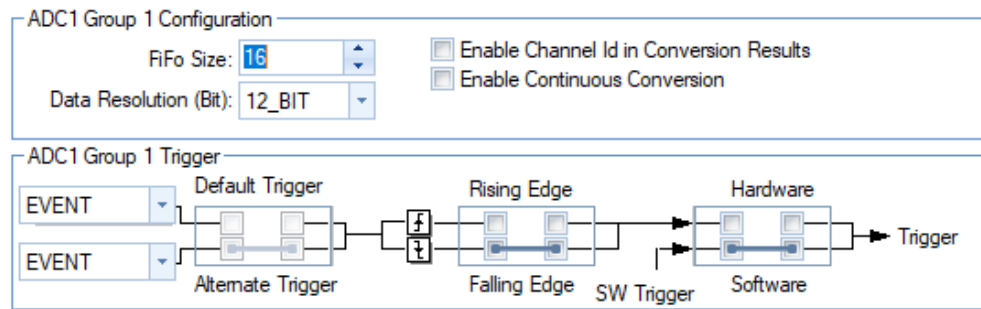


Figura 4.4 Diagrama de bloques específico de la RM46L852

Fuente: HALCoGen

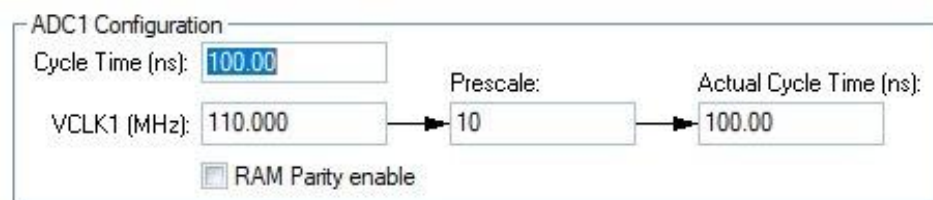


Figura 4.5 Configuración del ADC1

Fuente: HALCoGen

Cada canal utiliza un tiempo de 100 nanosegundos. Por lo tanto, para 3 canales utilizan un tiempo total de

$$Tx3ch = Nchadc * tTotal \text{ (Ecuación 4.1)}$$

$$Tx3ch = 3 * 100 \text{ ns}$$

$$Tx3ch = 300 \text{ ns}$$

La conversión de los 3 canales ADC, tienen un tiempo de retardo de 300 nanosegundos, lo que nos da una frecuencia de muestreo de:

$$Fx3ch = \frac{1}{Tx3ch}$$

$$Fx3ch = \frac{1}{300 \text{ ns}}$$

$$F_{x3ch} = 33.3 \text{ KHz}$$

Lo que significa que 33.3 KHz es la frecuencia máxima de muestreo del ADC del microcontrolador.

El ancho de banda es hasta el armónico 59 de la frecuencia de 60 Hz, entonces sería:

$$Bw = (\text{Numero de armónicos}) \times (\text{frecuencia de línea})$$

$$Bw = 59 \times 60 \text{ Hz}$$

$$Bw = 3.540 \text{ KHz}$$

Por lo tanto,

$$F_{x3ch} \gg BW$$

Entonces, podemos decir que la frecuencia de muestreo debe ser mucho mayor al ancho de banda de la señal.

Fs: Frecuencia de muestreo

$$Fs \gg Bw$$

Según Nyquist Shannon debe ser 2 veces más grande.

$$Fs = 2 \times 3.540 \text{ KHz}$$

$$Fs = 7.08 \text{ KHz}$$

Entonces $Fs < 33.3 \text{ KHz}$, es posible utilizar esta frecuencia de muestreo.

Sabemos que capturamos (2048 valores) de la señal de voltaje y corriente.

T_s : periodo de muestreo

$$T_s = \frac{1}{F_s}$$

$$T_s = \frac{1}{7.08 \text{ kHz}}$$

$$T_s = 141.24 \mu\text{s}$$

Entonces el tiempo total de muestra (2048 valores), equivale a:

T_t : tiempo total

$$T_t = \text{número de muestras} \times \text{periodo de muestreo}$$

$$T_t = 2048 \times 141.24 \mu\text{s}$$

$$T_t = 2.8925 \text{ ms}$$

Para utilizar el método matemático Radix 4, la frecuencia de muestreo se divide entre 4 y es el nuevo ancho de banda que aparece en el dominio de la frecuencia.

Por lo tanto:

$$B_{w\text{radix}4} = \frac{F_s}{4}$$

$$B_{w\text{radix}4} = 1.77 \text{ kHz}$$

Se desea un ancho de banda de señal = 3.540 KHz, por lo tanto, el ancho de banda de $B_{w\text{radix}4}$ no es aceptable, por ende, se debe volver a calcular.

Entonces seleccionado un $F_s = 25 \text{ KHz} \gg B_w$

Recalculamos:

$$T_s = \frac{1}{F_s}$$

$$T_s = \frac{1}{25 \text{ kHz}}$$

$$T_s = 40 \mu\text{s}$$

Calculamos el tiempo total

$$Tt = 2048 \times 40 \mu s$$

$$Tt = 81.92 ms$$

Según Radix4

$$Bw_{radix4} = \frac{Fs}{4}$$

$$Bw_{radix4} = \frac{25 kHz}{4}$$

$$BW_{radix4} = 6.25 KHz$$

Entonces es posible trabajar con una frecuencia de 25KHz.

Como se cumple la condición, el conversor ADC es perfecto para el muestreo de señal alterna en nuestro equipo.

La resolución del conversor ADC es de 12 bits y un voltaje de 3.3 Voltios, entonces se tiene un LSB de:

$$LSB = \frac{V_{max}}{2^n}$$

$$LSB = \frac{3.3}{2^{12}}$$

$$LSB = 805.66 \mu V$$

TIPO DE SENSOR	SENSIBILIDAD	ADC RESOL.	SENSIBILIDAD > ADC RESOL.	PASOS MÍNIMOS
VOLTAJE	3.1415 mV	806 μV	Si	4
CORRIENTE	40 mV	806 μV	Si	50

Tabla 4.1 Resolución de Voltaje y corriente

Fuente: Elaboración propia

Este último dato y conociendo la atenuación dada por la red resistiva para el muestreo de voltaje nos da la facilidad de calcular la sensibilidad del equipo ante los cambios de voltaje y corriente que puedan suscitarse.

3.1.1 SENSIBILIDAD DEL VOLTAJE (V_{acxsbd})

Tomando como referencia el dato del capítulo 3, ecuación 3.16. Atenuación para resistencias (A_{txR}) = 3.14136^{-3}

$$V_{acxsbd} = \frac{1}{A_{txR}} * LSB$$

$$V_{acxsbd} = \frac{1}{1.049^{-3}} * 805.66 \mu V$$

$$V_{acxsbd} = 0.2497 V$$

Por ende, el equipo puede capturar variaciones de voltaje de 0.76 Voltios.

3.1.2 SENSIBILIDAD DE LA CORRIENTE (A_{acxsbd})

Haciendo mención a la fórmula del capítulo 3 diseño de hardware “sensor de corriente”, deducimos que:

$$50 \text{ rms} \rightarrow A(T) = A_{max} * \sin(wt)$$

$$A_{max} = A_{rms} * \sqrt{2}$$

$$A_{max} = 50 * \sqrt{2}$$

$$A_{max} = 70.71 A$$

Donde;

A_{max} = Corriente máxima

$\sin(wt)$ = Seno del ángulo donde deseamos entender el valor instantáneo.

Conociendo la sensibilidad del sensor:

$$Sencibilidad = 40 \frac{[mV]}{[A]}$$

El equipo entrega una corriente máxima de 70.71 A con una sensibilidad de $40 \frac{[mV]}{[A]}$ entregada por el sensor, entonces si es posible pueda medir la señal de corriente.

3.2 MÓDULO TEMPORIZADOR DE ALTA PRECISIÓN GPIO (GIO)

El módulo GIO está diseñado para ser utilizado como entrada y salida digital, su única diferencia es que se compone de un registro de 8 bits y su interrupción es más fácil de definir en programación (Texas Instruments, 2021).

Los pines de entrada y salida son bidireccionales y programables por bits. El módulo GIO también admite la capacidad de interrupción externa (Texas Instruments, 2021).

Este módulo temporizador se está utilizando en conjunto con las interrupciones “real time interruption” con el fin de medir la frecuencia de la línea que viene del sensor de frecuencia, la cual entrega una señal cuadrada con el mismo periodo que la señal de voltaje. (Texas Instruments, 2021)

3.3 ALGORITMOS DE ANALISIS MATEMATICOS

Su funcionamiento es en nanosegundos.

3.3.1 RADIX 4

El algoritmo radix 4 divide la transformada discreta de Fourier en N/4 puntos. La ecuación de la transformada de Fourier se expresa como la suma de cálculo para la primera mitad y una suma de cálculo para la segunda mitad de la secuencia de entrada. De manera similar radix 4 expresa la transformada de Fourier como

cuatro sumas, luego la divide en cuatro ecuaciones, cada una de las cuales calcula cada cuarta muestra de salida. (Texas Instruments, 1998)

$$\begin{aligned}
 X(k) &= \sum_{n=0}^{N-1} x(n) W_N^{nk} \\
 &= \sum_{n=0}^{N/4-1} x(n) W_N^{nk} + \sum_{n=N/4}^{2N/4-1} x(n) W_N^{nk} + \sum_{n=2N/4}^{3N/4-1} x(n) W_N^{nk} + \sum_{n=3N/4}^{N-1} x(n) W_N^{nk} \\
 &= \sum_{n=0}^{N/4-1} x(n) W_N^{nk} + \sum_{n=0}^{N/4-1} x(n + N/4) W_N^{(n+N/4)k} + \sum_{n=0}^{N/4-1} x(n + N/2) W_N^{(n+N/2)k} \\
 &\quad + \sum_{n=0}^{N/4-1} x(n + 3N/4) W_N^{(n+3N/4)k} \\
 &= \sum_{n=0}^{N/4-1} \left[x(n) + x(n + N/4) W_N^{(n+N/4)k} + x(n + N/2) W_N^{(n+N/2)k} \right. \\
 &\quad \left. + x(n + 3N/4) W_N^{(n+3N/4)k} \right] W_N^{nk}
 \end{aligned}$$

Los tres coeficientes se pueden expresar de la siguiente manera:

$$W_N^{(N/4)k} = \left[\cos(\pi/2) - j \sin(\pi/2) \right]^k = (-j)^k$$

$$W_N^{(N/2)k} = \left[\cos(\pi) - j \sin(\pi) \right]^k = (-1)^k$$

$$W_N^{(3N/4)k} = \left[\cos(3\pi/2) - j \sin(3\pi/2) \right]^k = j^k$$

Entonces, la ecuación se expresa así:

$$X(k) = \sum_{n=0}^{N/4-1} \left[x(n) + (-j)^k x(n + N/4) + (-1)^k x(n + N/2) \right. \\
 \left. + (j)^k x(n + 3N/4) W_N^{nk} \right] W_N^{nk}$$

Fórmula para llegar a una descomposición de Fourier de cuatro puntos

$$\begin{aligned}
 X(4k) &= \sum_{n=0}^{N/4-1} \begin{bmatrix} x(n) + x(n + N/4) + x(n + N/2) \\ + x(n + 3N/4) \end{bmatrix} W_{N/4}^{nk} \\
 X(4k+1) &= \sum_{n=0}^{N/4-1} \begin{bmatrix} x(n) - jx(n + N/4) - x(n + N/2) \\ + jx(n + 3N/4) \end{bmatrix} W_N^n W_{N/4}^{nk} \\
 X(4k+2) &= \sum_{n=0}^{N/4-1} \begin{bmatrix} x(n) - x(n + N/4) - x(n + N/2) \\ - x(n + 3N/4) \end{bmatrix} W_N^{2n} W_{N/4}^{nk} \\
 X(4k+3) &= \sum_{n=0}^{N/4-1} \begin{bmatrix} x(n) + jx(n + N/4) - x(n + N/2) \\ - jx(n + 3N/4) \end{bmatrix} W_N^{3n} W_{N/4}^{nk}
 \end{aligned}$$

$X(4k)$, $X(4K + 1)$, $X(4K + 2)$ y $X(4K + 3)$ cada uno de los puntos es la suma de cuatro muestras de entrada $X(n)$, $X(n + N/4)$, $X(n + N/2)$ y $X(n + 3N/4)$ cada uno multiplicado por $+1$, -1 , j , o $-j$.

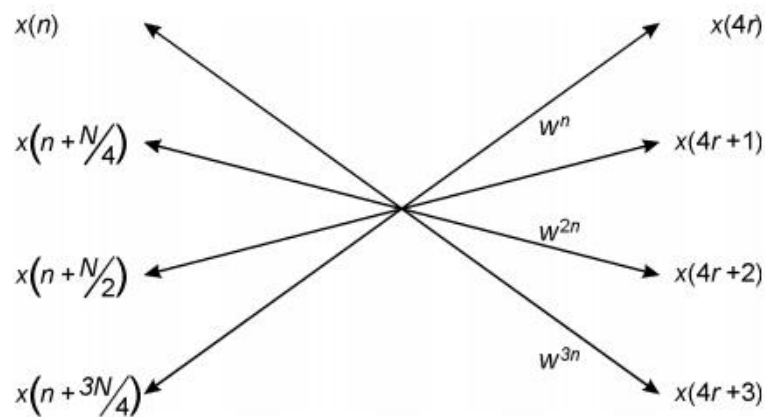


Figura 4.6 Transformada de Fourier Radix 4

Fuente: Texas Instruments, Radix 4

Los valores de salida reales e imaginarios son dados por las siguientes ecuaciones:

$$\begin{aligned}x_a' &= x_a + x_b + x_c + x_d \\y_a' &= y_a + y_b + y_c + y_d \\x_b' &= (x_a + y_b - x_c - y_d)C_b - (y_a - x_b - y_c + x_d)(-S_b) \\y_b' &= (y_a - x_b - y_c + x_d)C_b + (x_a + y_b - x_c - y_d)(-S_b) \\x_c' &= (x_a - x_b + x_c - x_d)C_c - (y_a - y_b + y_c - y_d)(-S_c) \\y_c' &= (y_a - y_b + y_c - y_d)C_c + (x_a - x_b + x_c - x_d)(-S_c) \\x_d' &= (x_a - y_b - x_c + y_d)C_d - (y_a + x_b - y_c - x_d)(-S_d) \\y_d' &= (y_a + x_b - y_c - x_d)C_d + (x_a - y_b - x_c + y_d)(-S_d)\end{aligned}$$

Mayormente se necesitan más de cien operaciones para el cálculo de Radix 4

Stage	1	2	3		$\log_2 N/2$
Butterfly Group	1	4	16		$N/4$
Butterflies per Group	$N/4$	$N/16$	$N/64$		1
Dual Node Spacing	$N/4$	$N/16$	$N/64$		1
Twiddle	leg0	0	0		0
Factor	leg2	n	$4n$		$(N/4)n$
Exponent	leg3	$2n$	$8n$		$(N/2)n$
	leg4	$3n$	$12n$		$(3N/4)n$
	$n = 0 - N/4 - 1$	$n = 0 - N/16 - 1$	$n = 0 - N/64 - 1$		$n=0$

Tabla 4.2 Resumen de las características de la transformada de Fourier en N puntos

Fuente: Texas Instruments, Radix 4

Para este caso se está utilizando $N = 2048$ puntos, lo que entrega un vector con 1024 valores reales y 1024 valores imaginarios que al calcular la magnitud de estos valores se determina solo 512 valores de magnitud de señal, estos cálculos permiten encontrar el THD y la fase de la señal.

3.3.2 MÓDULO CAPTURA DE ANCHO DE PULSO

El módulo de temporizador se utiliza específicamente para medir el ancho de pulso proveniente del sensor de frecuencia. Cuando se detecta un flanco ascendente o un flanco descendente, el temporizador comienza a contar de acuerdo con la programación y luego termina cuando se recibe un nuevo flanco. Cuando el buffer carga de datos con el valor equivalente en decimal este es equivalente en microsegundos a su equivalente. (HALCoGen, 2021)

Es decir:

Registro (valor decimal) = 16736, es equivalente a 16736 microsegundos.

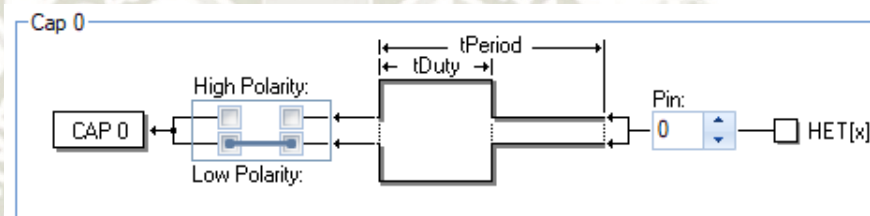


Figura 4.7 Diagrama de configuración de capturar temporal

Fuente: HALCoGen

3.3.3 MÓDULO RTI GENERAL

El módulo RTI está compuesto por dos submódulos, dos contadores y un comparador, para definir la frecuencia de trabajo se debe hacer mediante el registro RTI1CLK que está conformado por 75 MHz. Una vez que el temporizador culmina se activa una bandera de interrupción y ejecuta la tarea que fue programada. (HALCoGen, 2021)

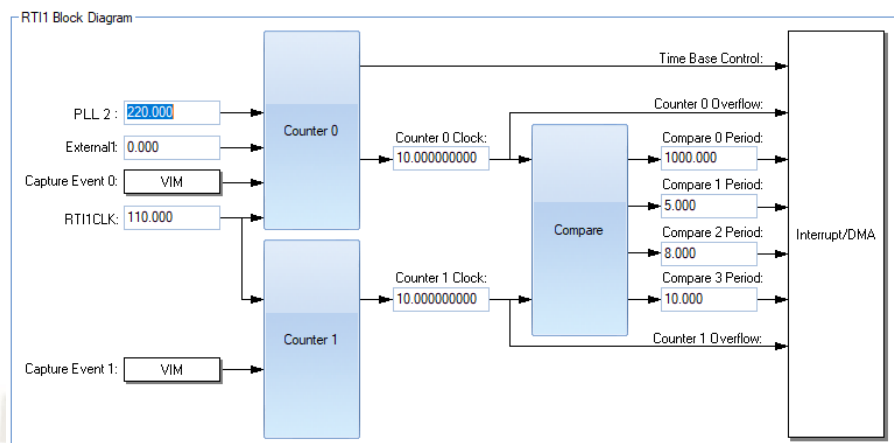


Figura 4.8 Diagrama de bloques del RTI general

Fuente: HALCoGen

• DIAGRAMA DE CONTADORES

Este módulo nos permite configurar la frecuencia base del tiempo de conteo, que es equivalente prescaler en otras series de microcontroladores. El módulo RTI tiene dos módulos que funciona de forma independiente, por lo que uno se puede configurar con una frecuencia de conteo básica y el otro se puede configurar con un valor completamente diferente sin afectar el trabajo entre ellos. esto es favorable porque un contador se puede configurar durante mucho tiempo, mientras podemos configurar otro temporizador durante poco tiempo. (HALCoGen, 2021)

Solo se utiliza el contador 0 que es el encargado de generar una interrupción cada 1 segundo que permite medir la frecuencia de línea que viene del sensor de frecuencia (HALCoGen, 2021).

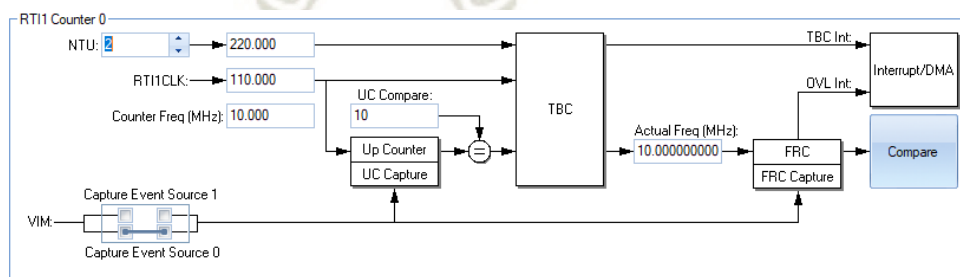


Figura 4.9 Diagrama de bloques del contador 0

Fuente: HALCoGen

3.3.4 ENTRADA Y SALIDA DIGITAL

El MCU RM46 puede multiplexar su modulo NHET y configurarlos como datos de salida y entradas digitales.

Es el encargado de controlar la pantalla LCD, los botones de configuración, el reloj interno del microcontrolador RTC DS1302.

Cuenta con 64 pines I/O, todos pueden configurarse para que trabaje como salidas PWM, temporizadores o contadores de pulso. Así como también, cuenta con compuertas lógicas en sus salidas y entradas como AND y XOR combinables entre sí (HALCoGen, 2021).

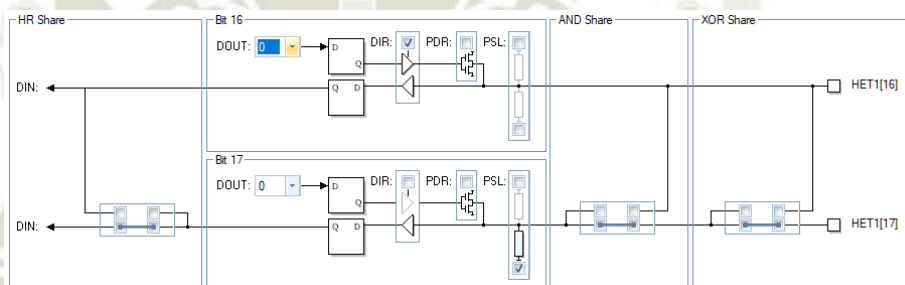


Figura 4.10 Diagrama de configuración como I/O digital

Fuente: HALCoGen

3.4 MÓDULOS DE COMUNICACIÓN SERIAL (SCI)

Este módulo se utiliza para la comunicación serial entre el procesador Hércules RM46 y el módulo WIFI ESP32, ambos se comunican a una velocidad de 115200 Baudios con 1 bit de paridad.

El sistema utiliza dos módulos de comunicación serial en el equipo, que son:

3.5 MÓDULOS DE COMUNICACIÓN SPI MEMORIA EEPROM

La memoria EEPROM solo tiene 2 funciones: escribir palabras en 8 bits y leer palabras en 8 bits.

3.6 RELOJ EN TIEMPO REAL

Las interrupciones en tiempo real proporcionan funciones de temporización en el sistema, el sistema RTI este compuesto por varios contadores que se pueden configurar según lo necesitado.

El reloj en tiempo real nos entrega la hora y fecha al controlador los que permite sincronizar la comunicación con la red WIFI.

4. FUNCIONES MATEMÁTICAS DE ANÁLISIS

4.1 CÁLCULO DEL VALOR RMS

El valor RMS de una onda se relaciona con su capacidad de suministro de energía. Es el valor definitorio más importante de las funciones periódicas, se trata del valor que está directamente relacionado con la potencia eléctrica. Corresponde al valor de corriente continua que proporcionaría a una resistencia el mismo trabajo por ciclo. (Medidas eléctricas para ingenieros, 2007)

Matemáticamente corresponde a la raíz cuadrada del valor medio total de la función cuadrática, así:

$$A = \sqrt{\frac{1}{T} * \int_0^T a^2(t) * dt}$$

Para calcular el RMS en el software Code Composer Studio de los elementos de un vector de entrada, se utiliza la siguiente ecuación:

```
Result = sqrt(((pSrc[0] * pSrc[0] + pSrc[1] * pSrc[1] + ... + pSrc[blockSize-1] *  
pSrc[blockSize-1]) / blockSize))
```

Donde:

[in] * pSrc : Apunta al vector de entrada

[in] blockSize : Longitud del vector de entrada

[out] * pResult : Valor RMS devuelto aquí

La función se implementa mediante un acumulador interno de 64 bits. El acumulador mantiene la precisión total de los resultados de multiplicación intermedia, pero solo proporciona un bit. Si el acumulador falla se distorsiona el resultado. Para evitar dichas fallas, la señal de entrada debe reducirse en $\log_2(\text{blockSize})$, ya que dicho parámetro se realiza internamente.

4.2 CÁLCULO DE LAS POTENCIAS ACTIVA Y REACTIVA Y APARENTE

4.2.1 POTENCIA ACTIVA

También llamada potencia activa o potencia promedio, es el valor promedio de la potencia instantánea.

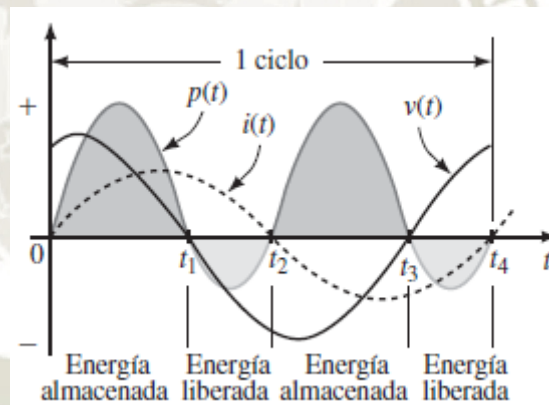


Figura 4.11 Representación de la potencia en 1 ciclo

Fuente: Análisis de circuitos, Allan H. Robbins – Wilhelm C. Miller, Primera edición

Como se observa en la figura 4.11 La letra p representa la potencia que fluye hacia la carga, su promedio será la potencia promedio hacia la carga. Si tiene un valor positivo, entonces, en promedio, más potencia fluye hacia la carga que la que retorna de ella. Entonces si p tiene un valor positivo, representa la potencia que en realidad es disipada por la carga. (Allan H. Robbins – Wilhelm C. Miller, 1995)

4.2.2 POTENCIA REACTIVA

Tomando como referencia la figura 4.11 durante los intervalos en que p es negativa, la potencia está siendo retornada desde la carga. La porción de la potencia que fluye hacia la carga y luego regresa hacia fuera se llama potencia reactiva, ya que primero fluye en un sentido y luego en otro, su valor promedio es cero. Aunque la potencia reactiva no realiza ningún trabajo útil, no puede ignorarse, se requiere corriente adicional para crearla y dicha corriente debe ser suministrada desde la fuente. (Allan H. Robbins – Wilhelm C. Miller, 1995)

4.2.3 POTENCIA APARENTE

Se conoce como potencia aparente a la combinación matemática de la potencia activa y reactiva.

4.2.4 CÁLCULO DEL PERIODO

El periodo T de una forma de onda es la duración de un ciclo medida en segundos. Es el inverso de la frecuencia.

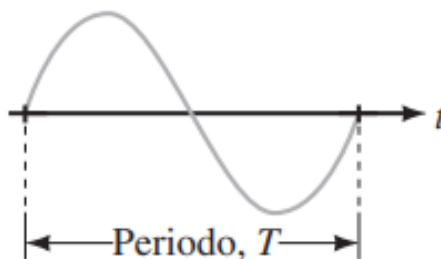


Figura 4.12 Representación del periodo T

Fuente: Análisis de circuitos, Allan H. Robbins – Wilhelm C. Miller, 1995

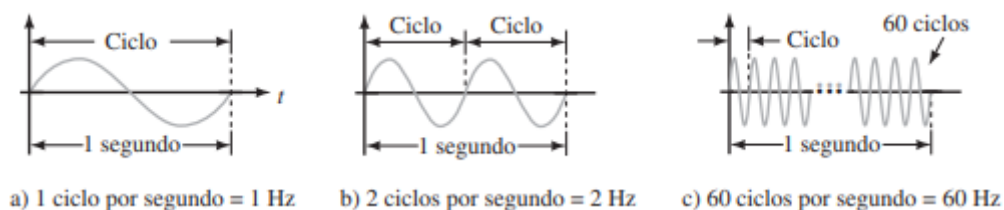


Figura 4.13 Muestreo de periodo en diferentes ciclos

Fuente: Análisis de circuitos, Allan H. Robbins – Wilhelm C. Miller, 1995

Donde dice en la imagen a) que la frecuencia es 1 ciclo por segundo, por lo que la duración de cada ciclo es $T=1$, así mismo en b) la frecuencia es dos ciclos por segundo, entonces la duración de cada ciclo es $T=1/2$ s, etc. (Allan H. Robbins – Wilhelm C. Miller, 1995)

PERIODO DE MUESTREO	PERIODO DE PUBLICACIÓN MQTT	PERIODO DE ALMACENAMIENTO
40 us	1 segundo	10 segundos

Tabla 4.3 Periodo para cada etapa

Fuente: Elaboración propia

4.3 BÚSQUEDA DEL CRUCE POR CERO

4.3.1 DETECTOR NO INVERSOR DE CRUCE POR CERO

El amplificador operacional funciona como un comparador. Su entrada (+) contrasta el voltaje E_i con un voltaje de referencia de 0 V ($V_{ref} = 0$ V). pero cuando el voltaje es mayor que el V_{ref} , el voltaje de entrada es igual a voltaje de saturación. Esto se debe a que el voltaje de entrada (+) es más positivo que el voltaje de entrada (-). Por lo tanto, el signo del voltaje es positivo. (Robert F. Coughlin, Frederick F. 1977)

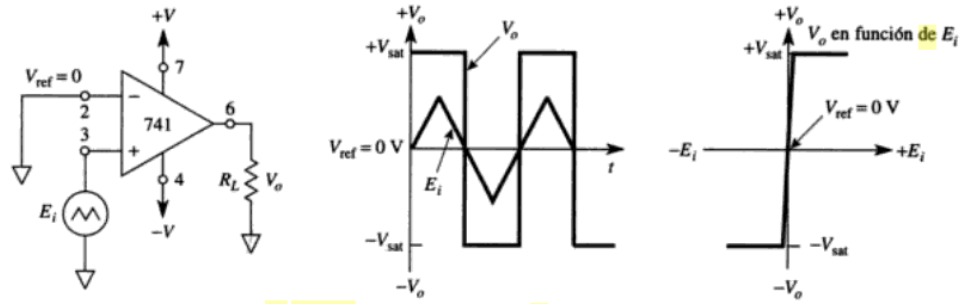


Figura 4.14 Detector de cruce por cero no inversor cuando E_i esta arriba de V_{ref} , $V_o = +V_{sat}$.

Fuente: Operational Amplifiers & Linear Integrated Circuits, Robert F. Coughlin, Frederick F. 1977

- **CRUCE POR CERO DIGITAL**

Cuando se tiene 2 señales en este caso señal de voltaje y señal de corriente, la función del algoritmo matemático desarrollado por el procesador hace que cada vez que las señales cambien de sentido positivo o negativo lo tome como valor.

4.4 CÁLCULO DEL FACTOR DE POTENCIA

El factor de potencia es el ángulo de desfase entre el voltaje y corriente. Y esta entre 0 y 1.



Figura 4.15 Factor de potencia en cada carga

Fuente: <http://blog.fullwat.com/que-es-el-factor-de-potencia/>

5. ESPECIFICACIONES TÉCNICAS DEL SERVIDOR

Para la implementación del sistema de almacenamiento y administración de servicios en la nube, se tiene que cumplir los siguientes parámetros.

La capacidad de almacenamiento del disco duro, parte del volumen de datos adquiridos durante un día de trabajo.

Tabla	Acción	Filas	Tipo	Cotejamiento	Tamaño	Residuo a depurar
☐ voltaje	Examinar Estructura Buscar Insertar Vaciar Eliminar	28,257	InnoDB	utf8_general_ci	1.5 MB	-
1 tabla	Número de filas	28,257	InnoDB	utf8_general_ci	1.5 MB	0 B

Figura 4.16 Tamaño de almacenamiento en la base de datos

Fuente:

http://localhost/phpmyadmin/index.php?route=/database/structure&server=1&db=db_energia

Durante las pruebas realizadas se obtuvo los siguientes datos:

- Inicio de la base datos: 12/12/2021
- Fin del almacenamiento de datos: 16/12/2021
- Días de funcionamiento: 5
- Tamaño de la información: 1.5 MB

$$\text{NroBytes}_{\text{Dia}} = \frac{\text{TamañoInfo}}{\text{Dias}}$$

$$\text{NroBytes}_{\text{Dia}} = \frac{1.5\text{MB}}{5}$$

$$\text{NroBytes}_{\text{Dia}} = 300 \text{ KB}$$

Por día, la base datos almacena información de 300KB por viviendas, lo que nos lleva a realizar la siguiente estimación de memoria.

El año cuenta con 365 días, por lo tanto, se tiene:

$$Capacidad_{Año} = 365 * NroBytes_{dia}$$

$$Capacidad_{Año} = 365 * 300kB$$

$$Capacidad_{Año} = 109.5 MB$$

Con el último cálculo se estima que una vivienda por año, ocupa un tamaño de memoria de 109.5 MB

Se estima que cada servidor pueda almacenar y administrar información de 50 viviendas como mínimo, se tendría el siguiente cálculo de memoria:

$$Capacidad_{DiscoDuro} = Capacidad_{Año} * NroCasas$$

$$Capacidad_{DiscoDuro} = 109.5 MB * 50$$

$$Capacidad_{DiscoDuro} = 5.475 GB$$

Como se puede concluir con este cálculo para 50 viviendas, se ocupa una memoria de 5.5 GB, pero en la actualidad se tiene capacidad de discos duros en TeraBytes para servidores por lo tanto el límite de viviendas, lo define el tamaño del disco duro.

$$Capacidad_{DiscoDuro} = 1 TeraByte$$

$$Capacidad_{DiscoDuro} = 1000 GB$$

$$Capacidad_{DiscoDuro} = Capacidad_{Año} * NroCasas$$

Despejando el número de viviendas se tiene:

$$NroCasas = \frac{Capacidad_{DiscoDuro}}{Capacidad_{Año}}$$

$$NroCasas = \frac{1000\text{ GB}}{109.5\text{ MB}}$$

$$NroCasas = 9132.42$$

El servidor de datos puede almacenar la información de hasta 9132 viviendas.

Por lo que se recomienda la implementación del equipo con las siguientes especificaciones:

- Procesador: Core I7 – 10 generación, 16MB cache/ 2.9 GHz
- Memoria RAM: 16 GB/ DDR4, 2933 MHz
- Disco Duro: 256 GB M.2 PCI Express (OS, Node Red, MySQL) + 1 TB 7200rpm
- 3.5" SATA (Almacenamiento).

6. CONFIGURACIÓN DEL IP DEL SERVIDOR

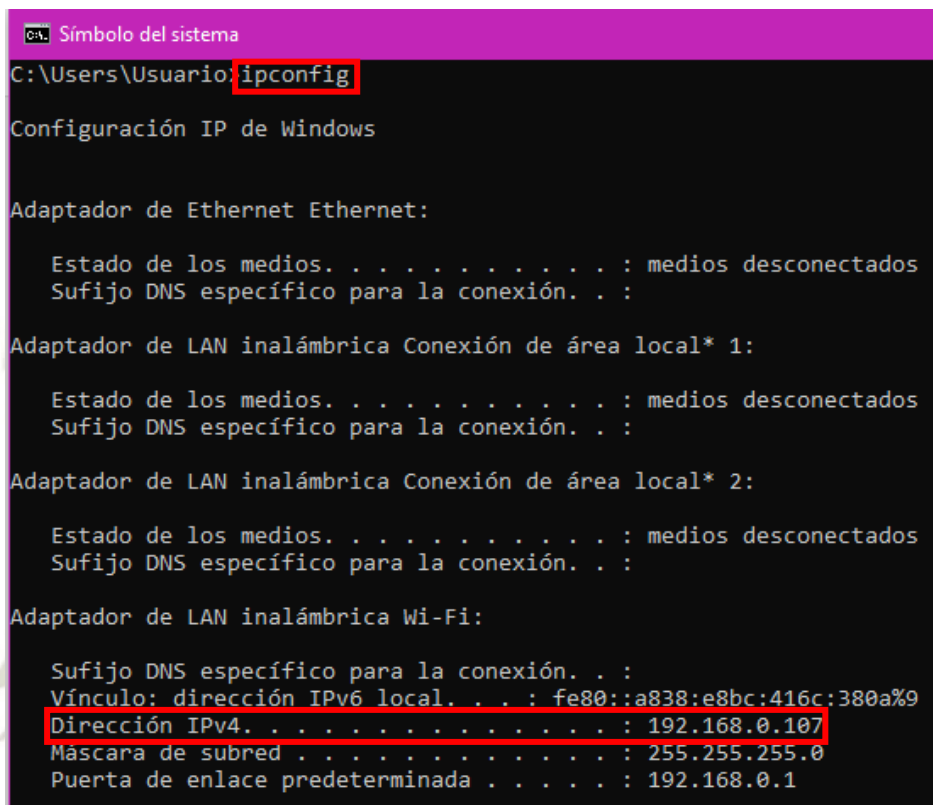
Para la configuración del IP del servidor se necesita tener acceso al modem de WIFI, por ende, entramos a su configuración y le definimos un IP estático ya que ocurrirían problemas de conexión si el IP cambia.

ID	Nombre del Cliente	Dirección MAC	IP Asignado	Tiempo de Arrendamiento
1	LAPTOP-3EUKQMEN	EC:8E:B5:5B:19:8B	192.168.0.104	01:50:46
2	DESKTOP-1017L9T	CC:B0:DA:64:2B:A3	192.168.0.107	Permanente
3	Galaxy-Tab-S6-Lite-d e-Vanessa	EE:43:C9:FD:E7:14	192.168.0.105	01:47:25

Figura 4.17 IP permanente del servidor de datos

Fuente: <http://192.168.0.1/>

Ejecutamos el comando “ipconfig” para asegurar que el IP fue asignado correctamente.



```
C:\Users\Usuario>ipconfig

Configuración IP de Windows

Adaptador de Ethernet Ethernet:

    Estado de los medios. . . . . : medios desconectados
    Sufijo DNS específico para la conexión. . :

Adaptador de LAN inalámbrica Conexión de área local* 1:

    Estado de los medios. . . . . : medios desconectados
    Sufijo DNS específico para la conexión. . :

Adaptador de LAN inalámbrica Conexión de área local* 2:

    Estado de los medios. . . . . : medios desconectados
    Sufijo DNS específico para la conexión. . :

Adaptador de LAN inalámbrica Wi-Fi:

    Sufijo DNS específico para la conexión. . :
    Vínculo: dirección IPv6 local. . . : fe80::a838:e8bc:416c:380a%9
    Dirección IPv4. . . . . : 192.168.0.107
    Máscara de subred . . . . . : 255.255.255.0
    Puerta de enlace predeterminada . . . . : 192.168.0.1
```

Figura 4.18 Comando “ipconfig”

Fuente: CMD

7. NODE RED

Node Red es un editor de flujo online donde se pueden añadir o eliminar nodos y conectarlos entre si permitiendo una comunicación entre ellos. existen 2 tipos de nodos; un nodo de inyección o un nodo de función.

Su plataforma está basada en Node.js y fue creada para simplificar el desarrollo de aplicaciones orientadas al manejo de eventos asíncronos, como las aplicaciones de IoT.

Esta herramienta nos permite crear el diseño para que pueda ser visualizado en diferentes dispositivos.

Se usan los siguientes nodos:

- Nodo mqtt
- Nodo debug
- Nodo function
- Nodo gauge
- Nodo chart

A continuación, se muestra el paso a paso para la instalación de Node Red

Paso 1: Descarga de motor de Node Red

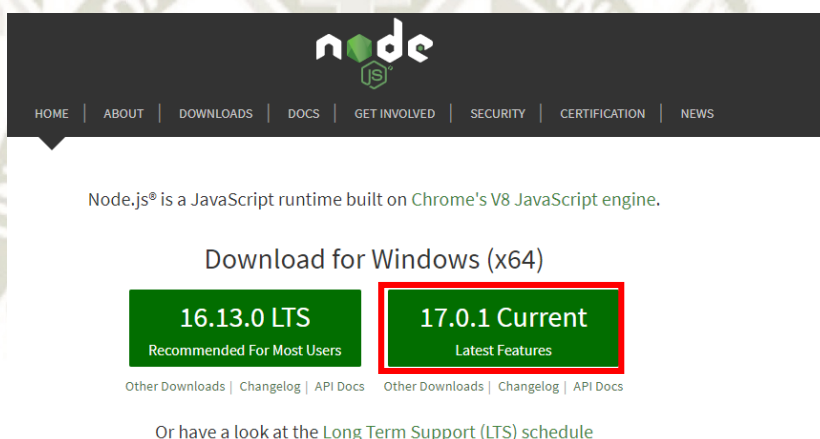


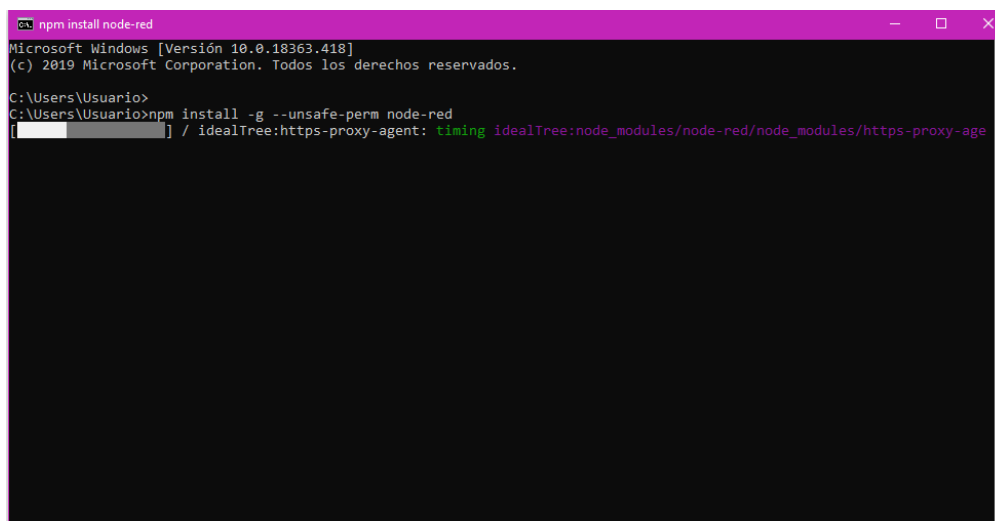
Figura 4.19 Motor de Node Red

Fuente: <https://nodejs.org/es/download/>

Paso 2: Instalación de Node Red

La instalación de Node Red como un módulo global agrega el comando de Node Red a la ruta de nuestro sistema, por ende, se ejecuta el siguiente comando:

```
npm install -g --unsafe-perm node-red
```



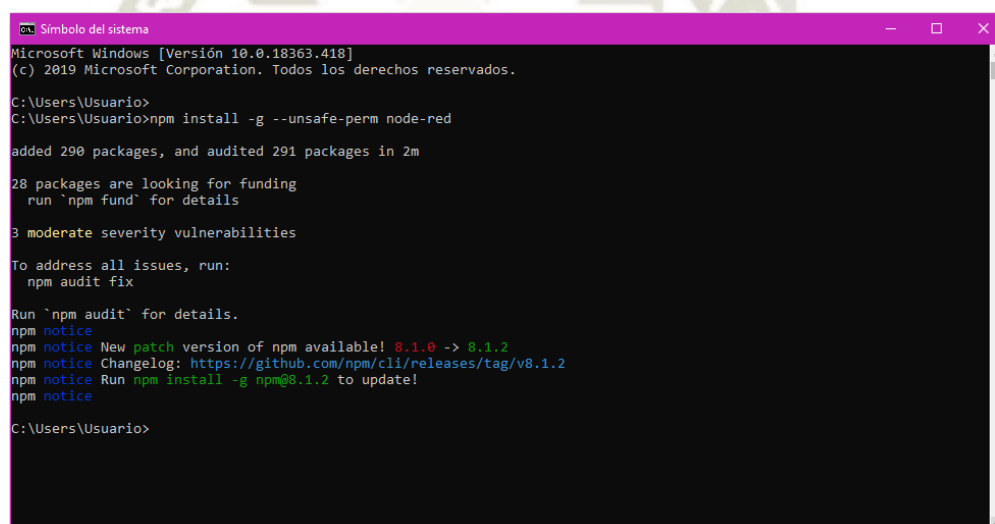
```

C:\Users\Usuario> npm install node-red
Microsoft Windows [Versión 10.0.18363.418]
(c) 2019 Microsoft Corporation. Todos los derechos reservados.

C:\Users\Usuario>
C:\Users\Usuario> npm install -g --unsafe-perm node-red
[ ] / idealTree:https-proxy-agent: timing idealTree:node_modules/node-red/node_modules/https-proxy-age
  
```

Figura 4.20 Ejecución del comando ofrecido por el fabricante

Fuente: Comando CMD



```

C:\Users\Usuario>
C:\Users\Usuario> npm install -g --unsafe-perm node-red

added 290 packages, and audited 291 packages in 2m

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

3 moderate severity vulnerabilities

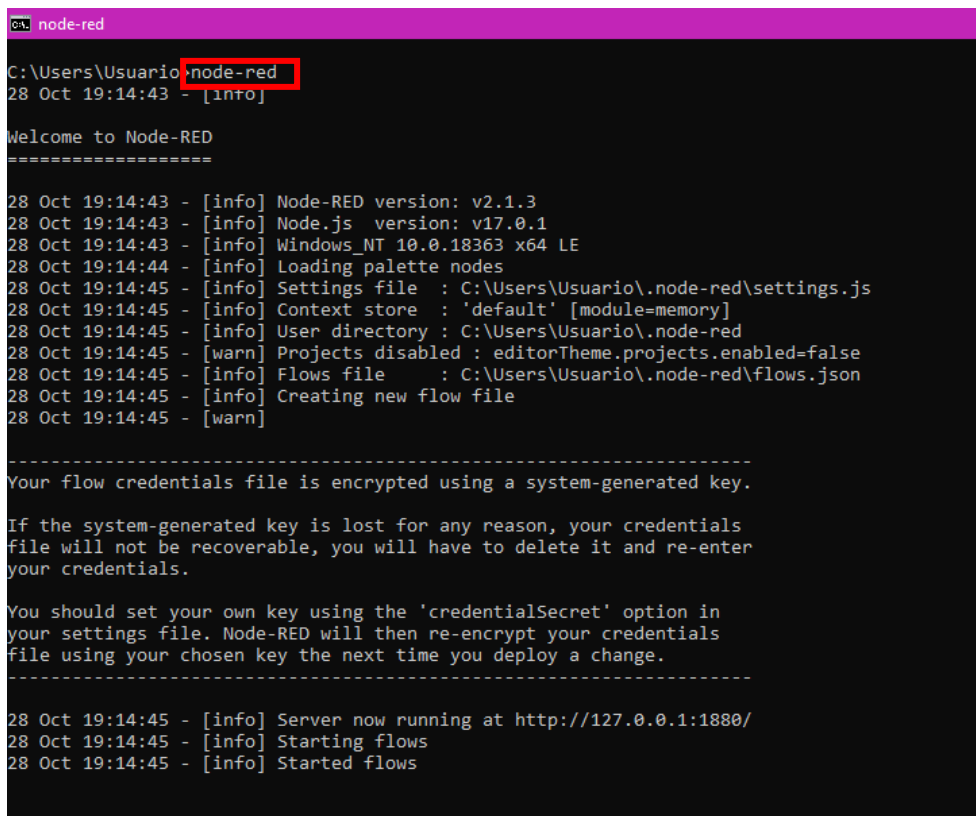
To address all issues, run:
  npm audit fix

Run `npm audit` for details.
npm notice
npm notice New patch version of npm available! 8.1.0 -> 8.1.2
npm notice Changelog: https://github.com/npm/cli/releases/tag/v8.1.2
npm notice Run `npm install -g npm@8.1.2` to update!
npm notice
C:\Users\Usuario>
  
```

Figura 4.21 Verificación de instalación exitosa de Node Red

Fuente: Comando CMD

Paso 3: Ejecutar el comando Node Red



```

C:\Users\Usuario>node-red
28 Oct 19:14:43 - [info]

Welcome to Node-RED
=====

28 Oct 19:14:43 - [info] Node-RED version: v2.1.3
28 Oct 19:14:43 - [info] Node.js version: v17.0.1
28 Oct 19:14:43 - [info] Windows_NT 10.0.18363 x64 LE
28 Oct 19:14:44 - [info] Loading palette nodes
28 Oct 19:14:45 - [info] Settings file : C:\Users\Usuario\.node-red\settings.js
28 Oct 19:14:45 - [info] Context store : 'default' [module=memory]
28 Oct 19:14:45 - [info] User directory : C:\Users\Usuario\.node-red
28 Oct 19:14:45 - [warn] Projects disabled : editorTheme.projects.enabled=false
28 Oct 19:14:45 - [info] Flows file : C:\Users\Usuario\.node-red\flows.json
28 Oct 19:14:45 - [info] Creating new flow file
28 Oct 19:14:45 - [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.
-----

28 Oct 19:14:45 - [info] Server now running at http://127.0.0.1:1880/
28 Oct 19:14:45 - [info] Starting flows
28 Oct 19:14:45 - [info] Started flows
  
```

Figura 4.22 Ejecución del comando Node Red

Fuente: Comando CMD

Paso 4: Escribir en el buscador localhost:1880 para que se abra el entorno de desarrollo de Node Red.

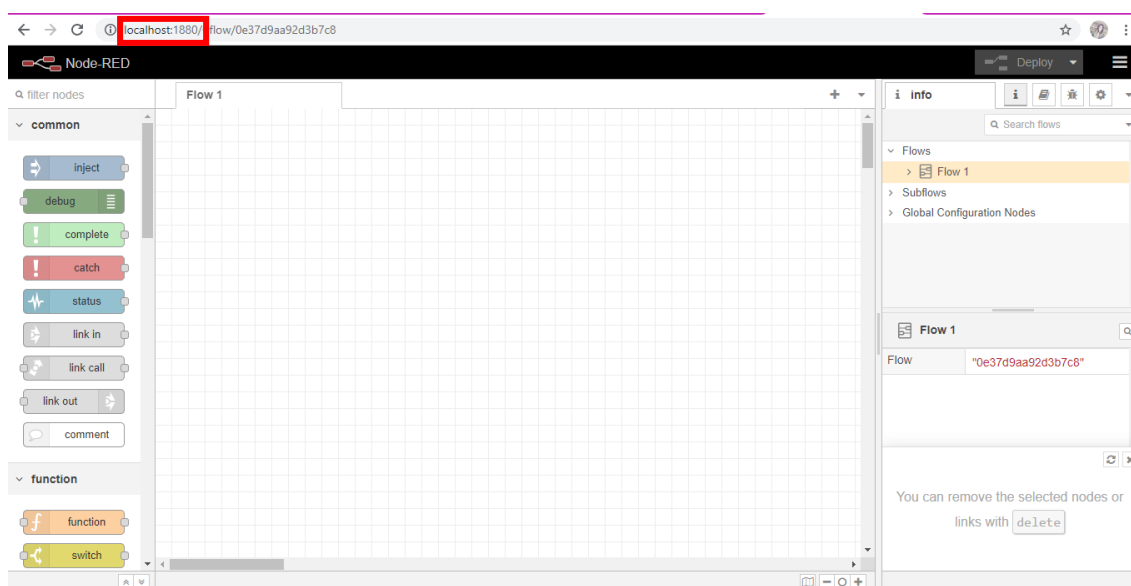


Figura 4.23 Entorno de desarrollo de Node Red
Fuente: Node Red, 2021

Paso 5: Instalación de librerías anexas:

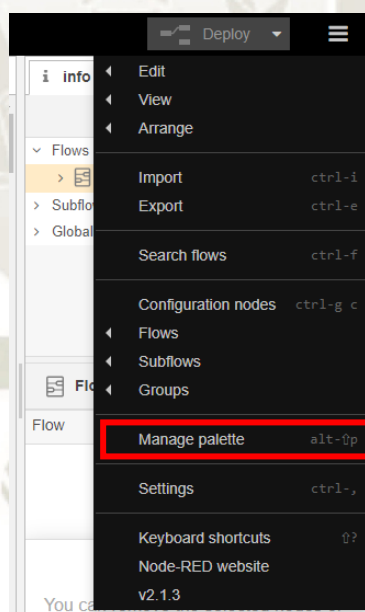


Figura 4.24 Configuración de Node Red
Fuente: Node Red, 2021

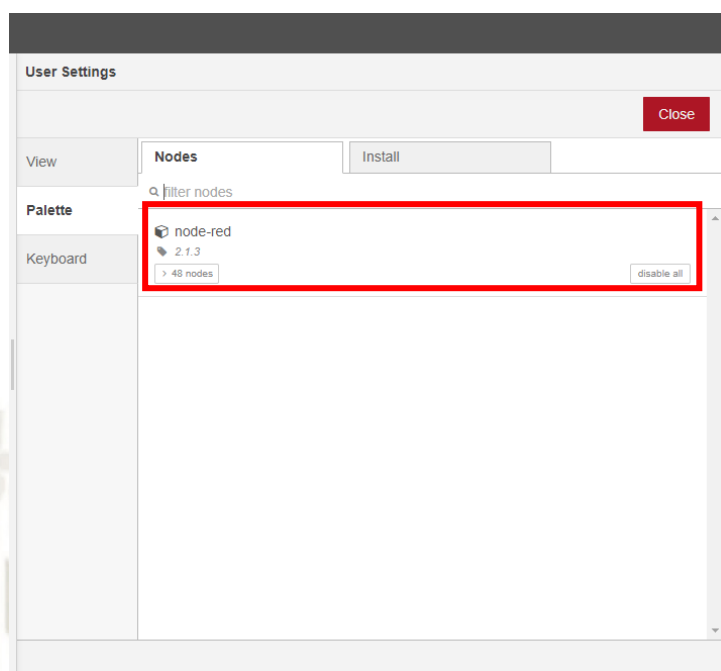


Figura 4.25 Librerías Anexas instaladas

Fuente: Node Red, 2021

Paso 5: Instalación de base de datos

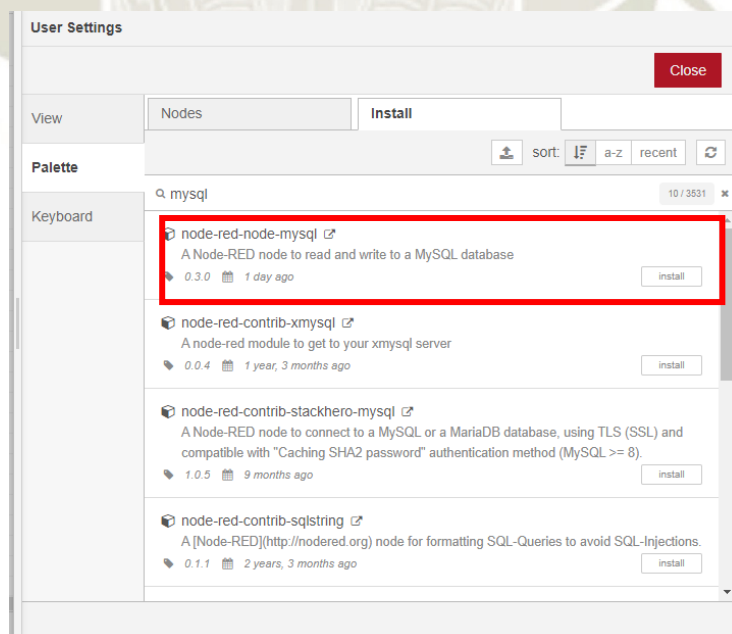


Figura 4.26 Instalación de la base de datos en Node Red

Fuente: Node Red, 2021

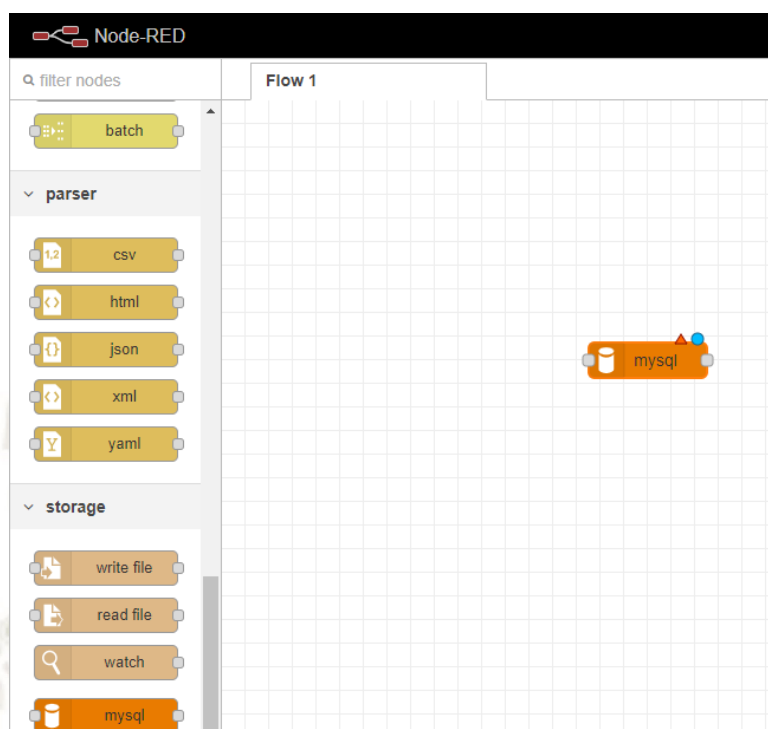


Figura 4.27 Nodo de base de datos en Node Red

Fuente: Node Red, 2021

Paso 6: Instalación de dashboards

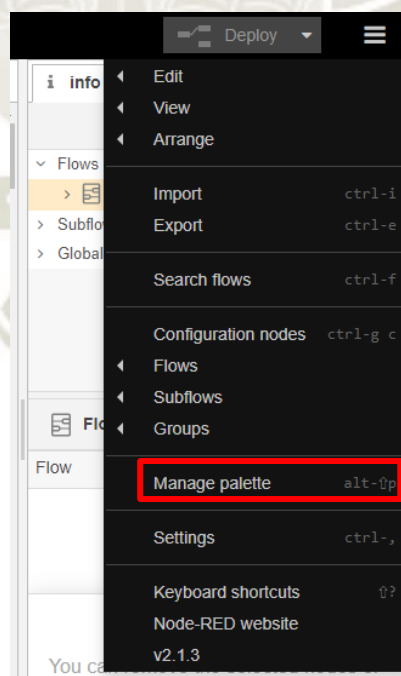


Figura 4.28 Configuración de Node Red

Fuente: Node Red, 2021

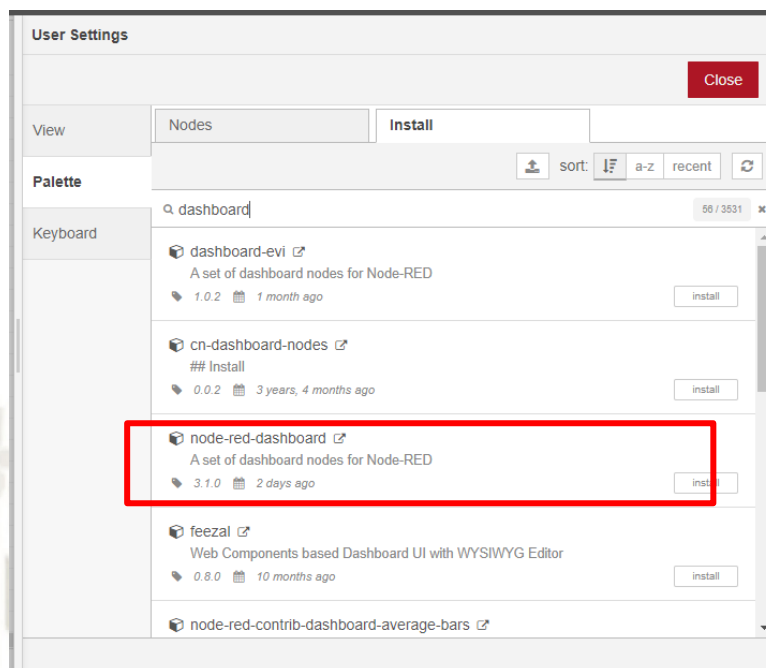


Figura 4.29 Instalación de Dashboards en Node Red

Fuente: Node Red, 2021

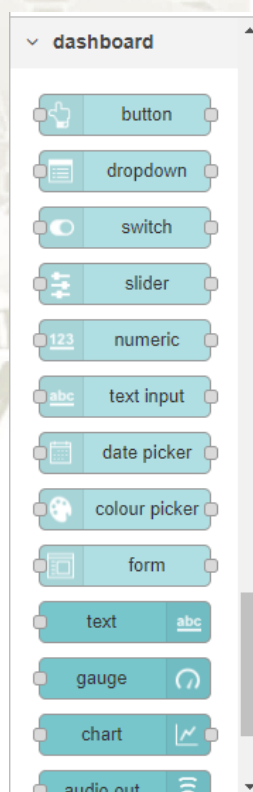


Figura 4.30 Nodos de Dashboards en Node Red

Fuente: Node Red, 2021

7.1 NODOS DE GAUGE Y CHART

- **GAUGE**

Nodo que crea graficas de calibración con un diseño sencillo. Puede ser visualizado en una laptop o celular.

Crea las calibraciones de cada medición hecha por los equipos.

- **CHART**

Nodo que crea las tendencias con respecto al tiempo de los valores generados por los nodos gauge, se generó un nodo para cada valor medido por los equipos.

Adicionalmente se utilizó un nodo de inyección el cual nos sirvió para proporcionar un dato y corroborar que nuestros nodos de visualización funcionan correctamente.

Para visualizar los dashboards se escribe lo siguiente en el navegador web:
<http://localhost:1880/ui>

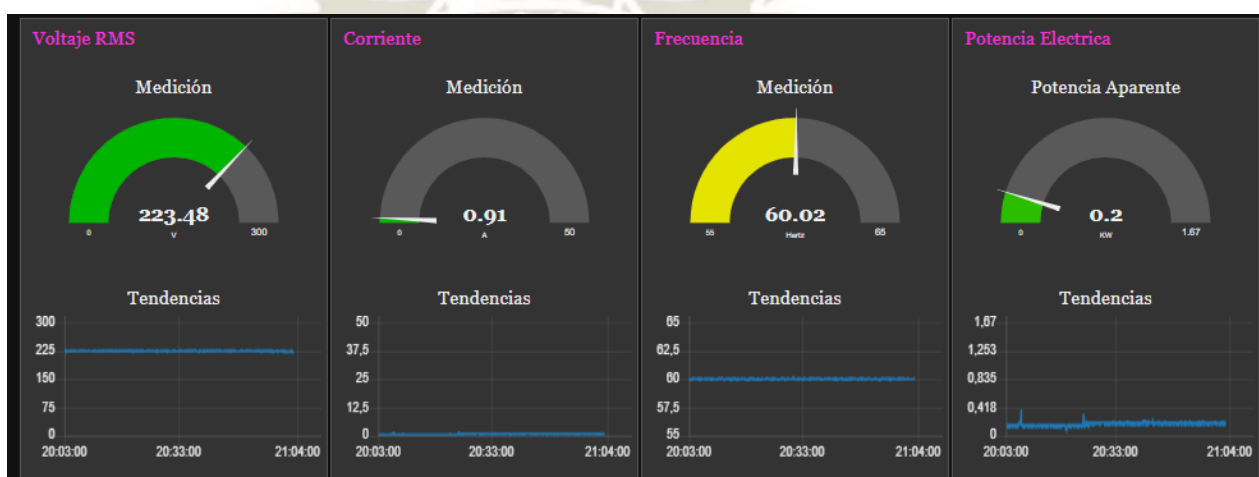


Figura 4.31 Visualización de Dashboards

Fuente: Elaboración propia

7.2 AEDES BROKER

Se utiliza el nodo MQTT encargado de recibir todos los datos del equipo, suscribiéndome a un tópico que será el encargado de publicar el mensaje.

Cada equipo tiene una dirección IP diferente, como:

- 192.168.0.103: IP del módulo ESP32 medidor de energía.
- 192.168.0.112: IP del servidor de Node Red

Para la instalación del bróker Aedes se siguen los siguientes pasos.

Paso 1: Se busca el bróker

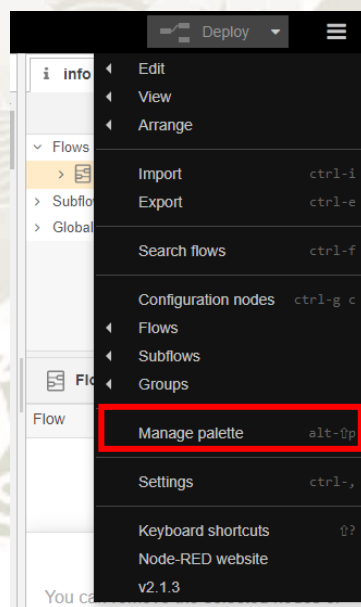


Figura 4.32 Configuración de Node Red

Fuente: Node Red, 2021

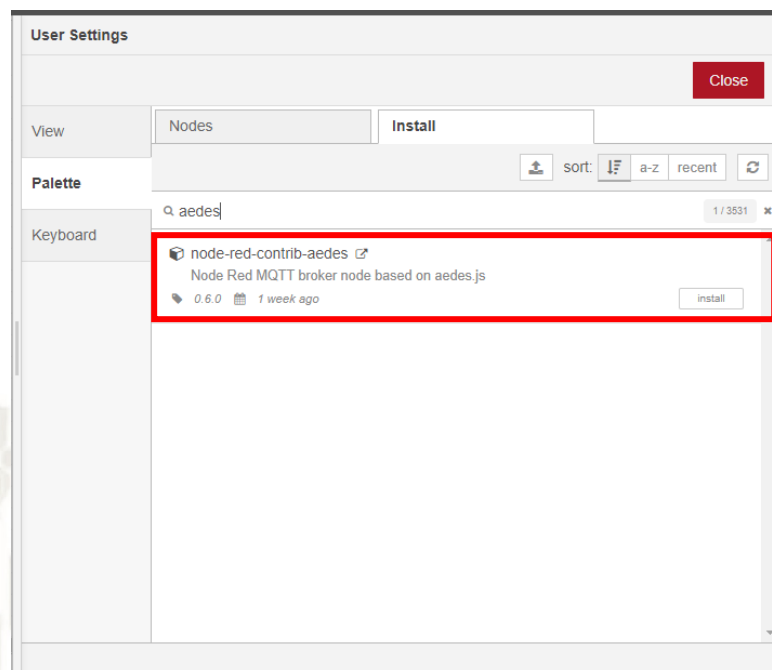


Figura 4.33 Instalación de Bróker Aedes en Node Red

Fuente: Node Red, 2021



Figura 4.34 Nodo de Bróker Aedes en Node Red

Fuente: Node Red, 2021

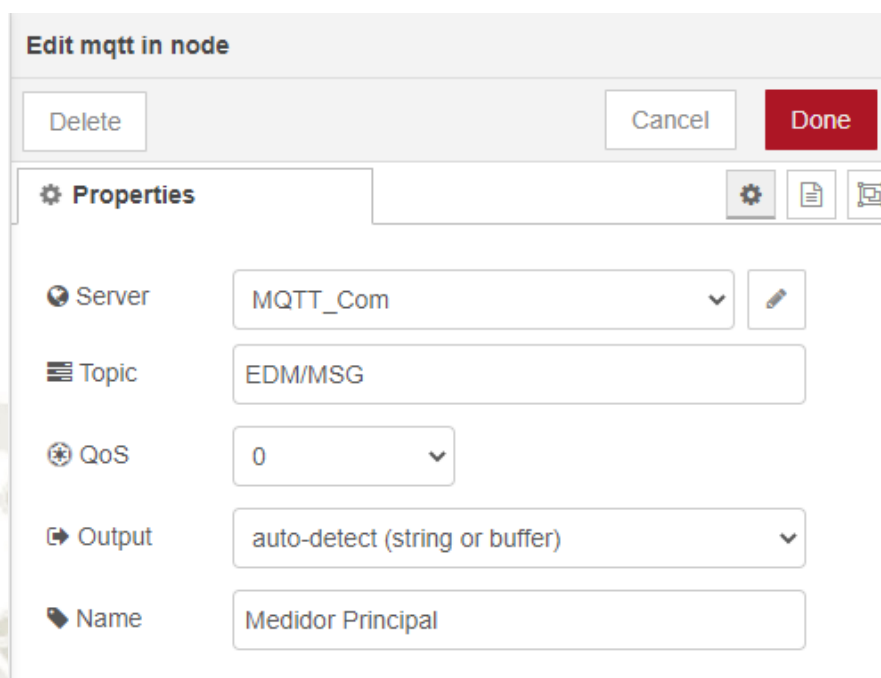


Figura 4.35 Configuración del bróker MQTT Aedes

Fuente: Node Red, 2021

Aedes es un bróker que ya se encuentra integrado en el mismo Node Red, inicialmente se estaba haciendo uso del bróker Home Assistant, pero ocurrió el problema de que al momento de desconectar dicha máquina virtual se perdían los datos, por ende, se procedió a utilizar Aedes.

7.3 BASE DE DATOS XAMPP

XAMPP es un server que trabaja principalmente bajo la base de datos MySQL. Es el encargado de crear, editar, consultar y gestionar bases de datos en MySQL rápido y sencillo.

A continuación, se muestra la instalación paso a paso de XAMPP.

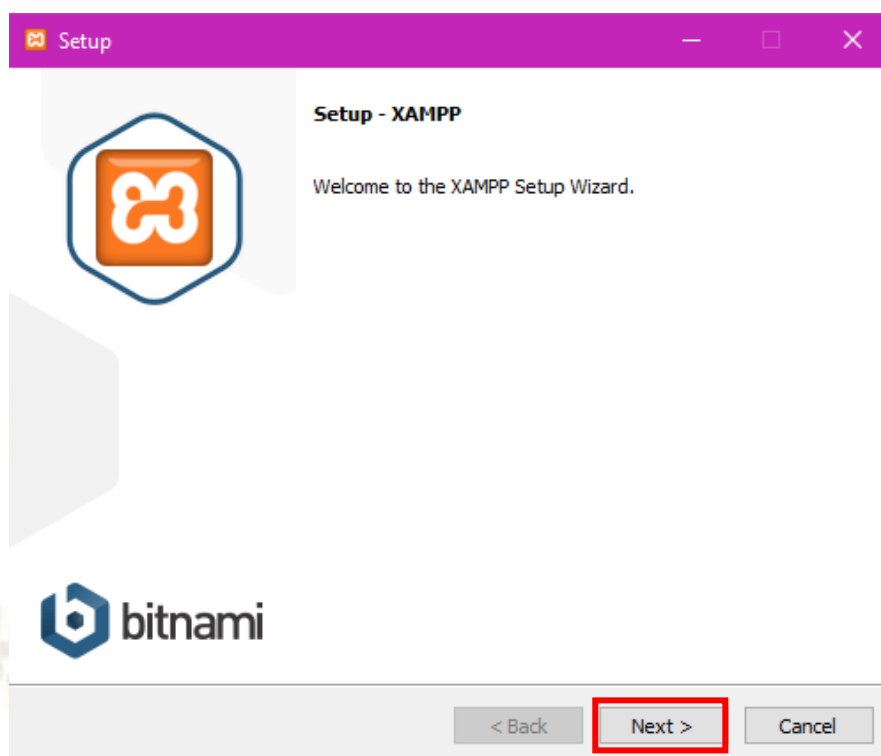


Figura 4.36 Mensaje de bienvenida de XAMPP

Fuente: XAMPP 2021

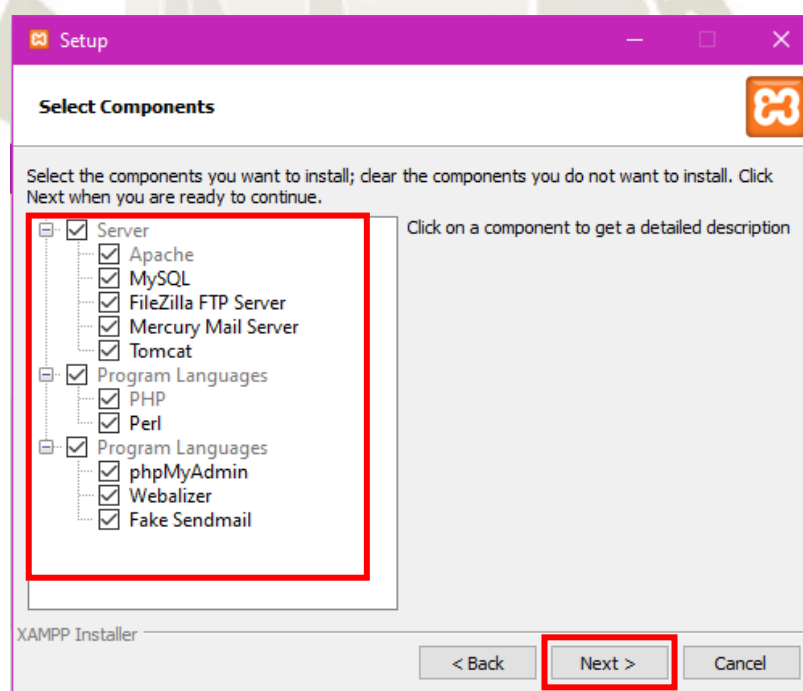


Figura 4.37 Selección de componentes de XAMPP

Fuente: XAMPP 2021

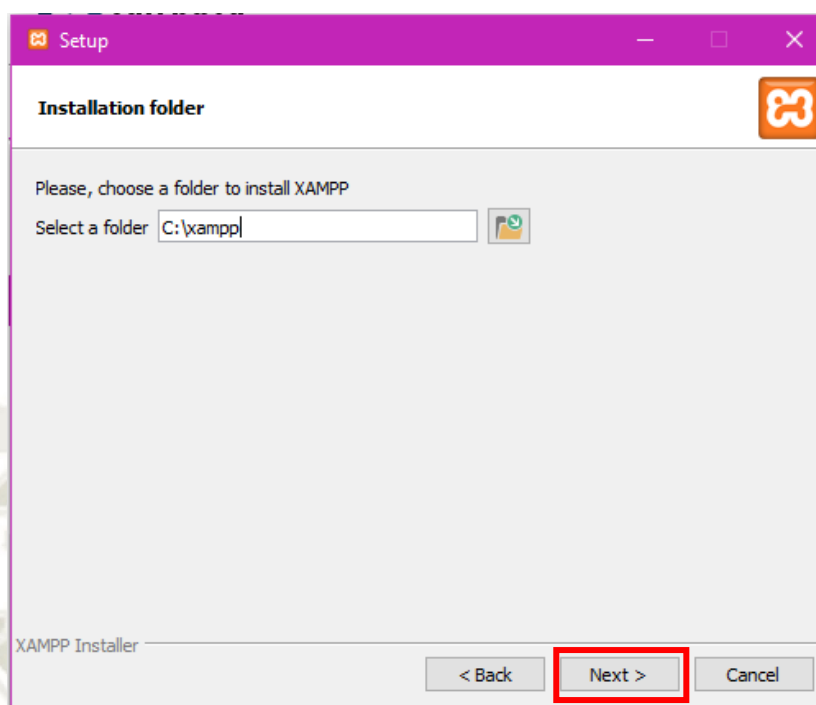


Figura 4.38 Selección de la carpeta donde se instalará XAMPP

Fuente: XAMPP 2021

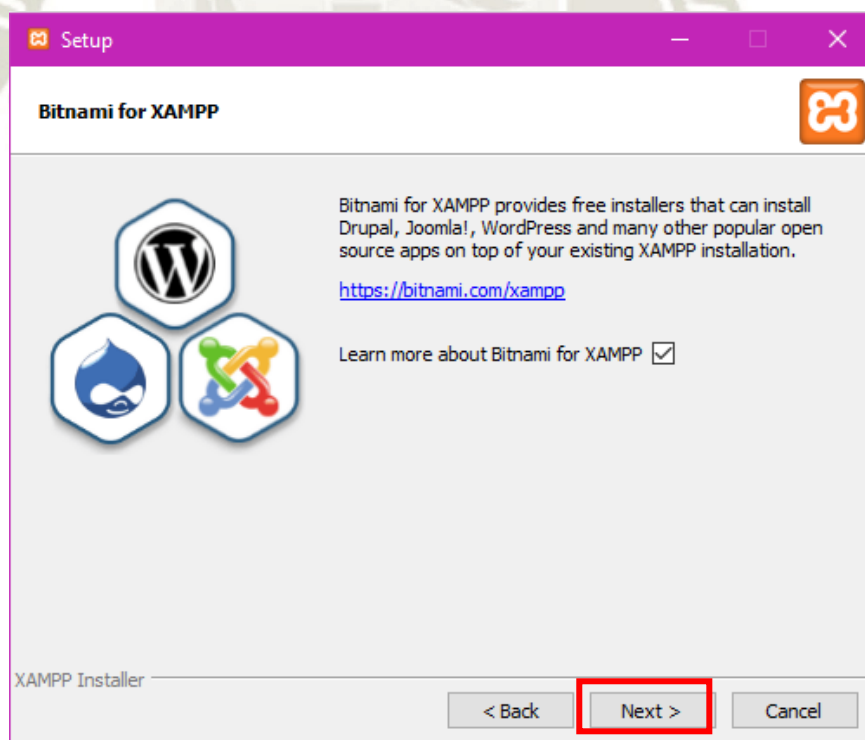


Figura 4.39 Mensaje de Bitnami para XAMPP

Fuente: XAMPP 2021

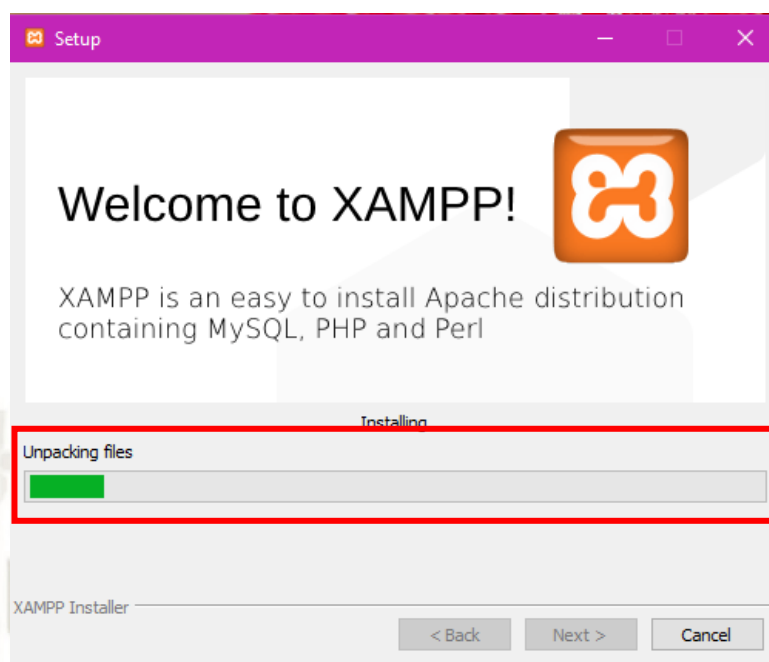


Figura 4.40 Instalación de XAMPP

Fuente: XAMPP 2021

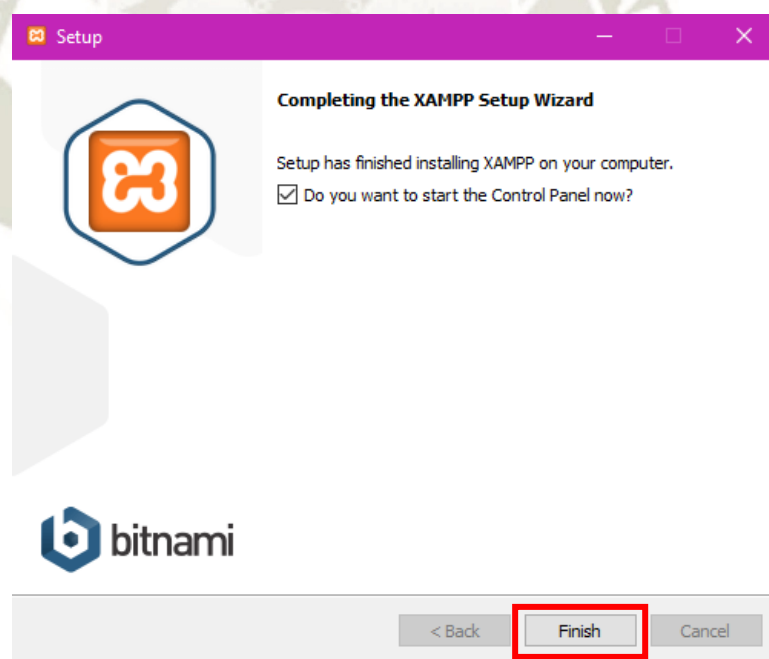


Figura 4.41 Completando el asistente de configuración de XAMPP

Fuente: XAMPP, 2021

Por último, abrimos XAMPP y accionamos el módulo de Apache y MYSQL

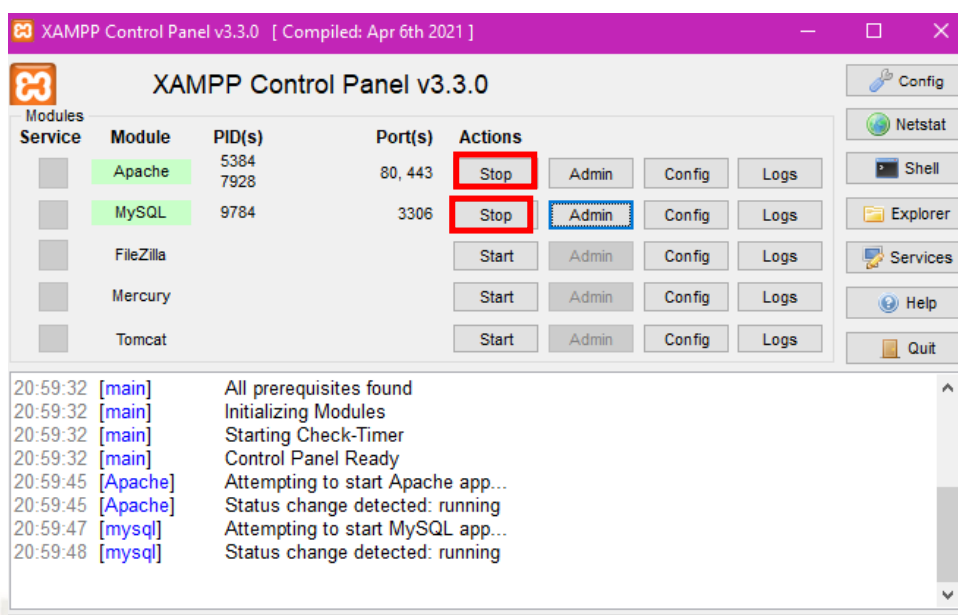


Figura 4.42 Panel de control principal de XAMPP

Fuente: XAMPP 2021

Una vez que la base de datos esta activada accionamos admin en Apache, este nos redireccionará a la página de la figura 4.44 y creamos una cuenta

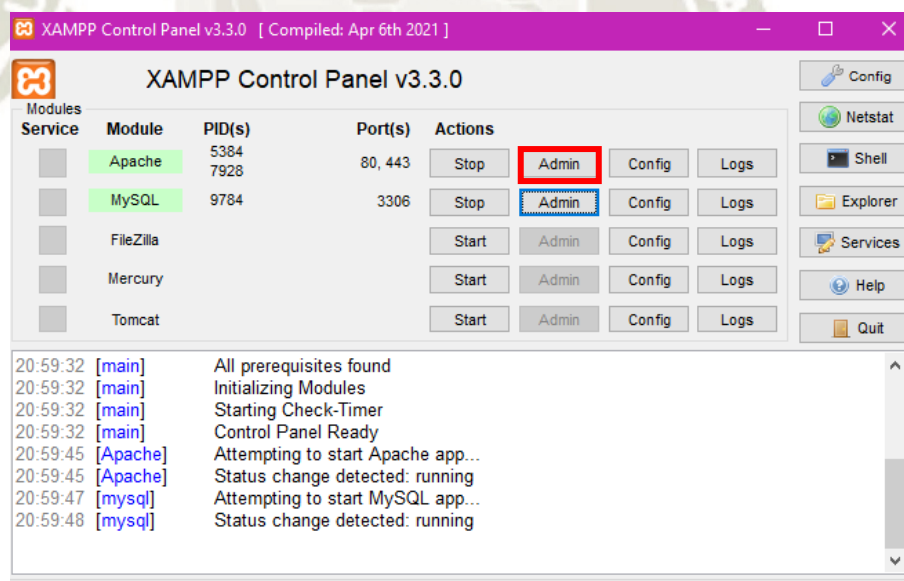


Figura 4.43 Panel de control principal de XAMPP

Fuente: XAMPP 2021

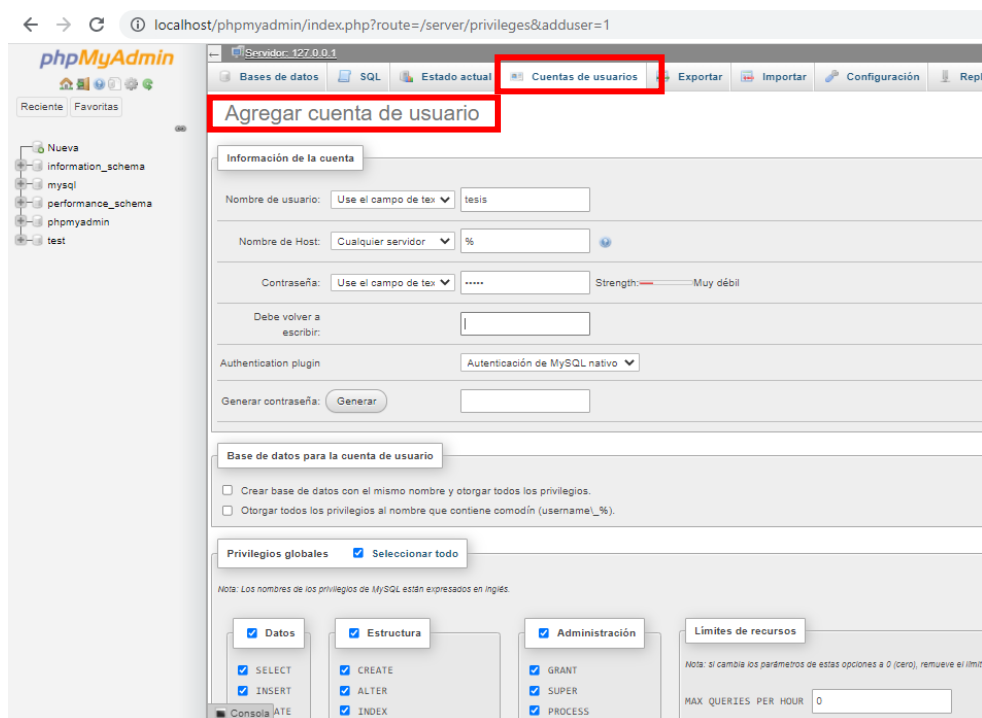


Figura 4.44 Pagina web de XAMPP

Fuente: XAMPP, 2021

Ahora creamos la base de datos:

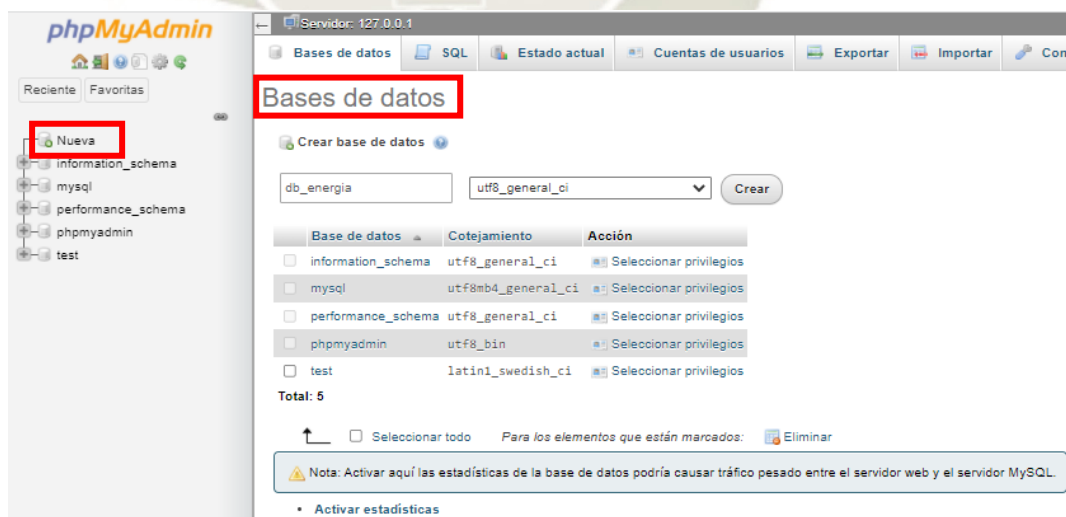


Figura 4.45 Creación de la base de datos en XAMPP

Fuente: XAMPP 2021

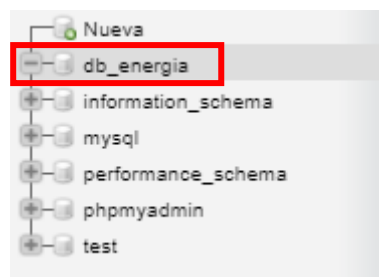


Figura 4.46 Base de datos creada exitosamente

Fuente: XAMPP 2021

Ahora hacemos la conexión entre Node Red y XAMPP.

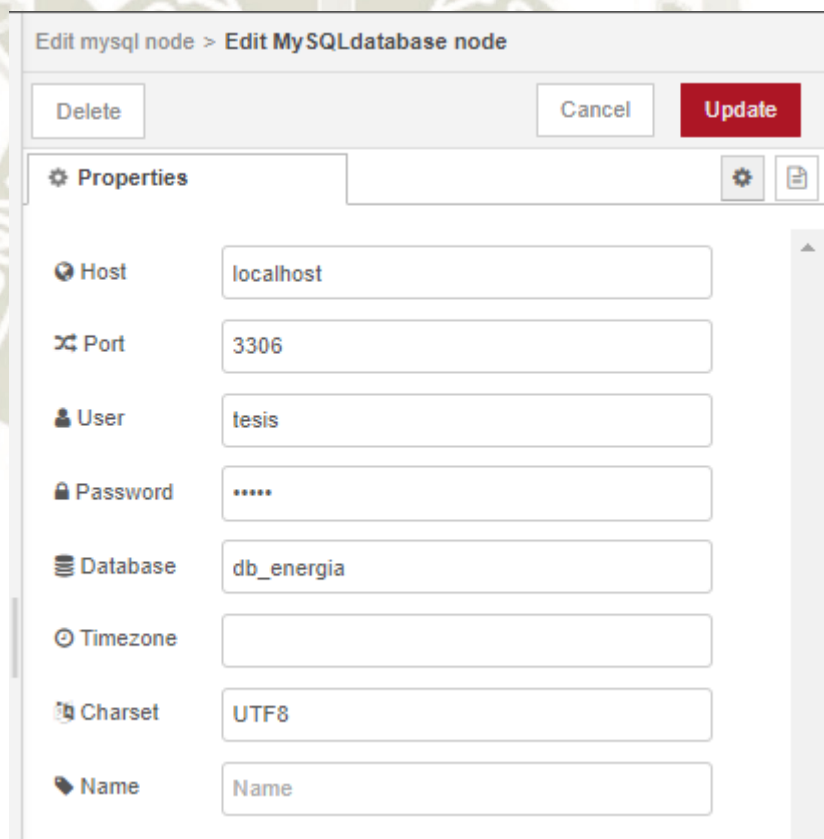


Figura 4.47 Conexión de la base de datos en el entorno de Node Red

Fuente: Node Red, 2021

Por consiguiente, se puede visualizar que en Node Red la base de datos fue conectada exitosamente.

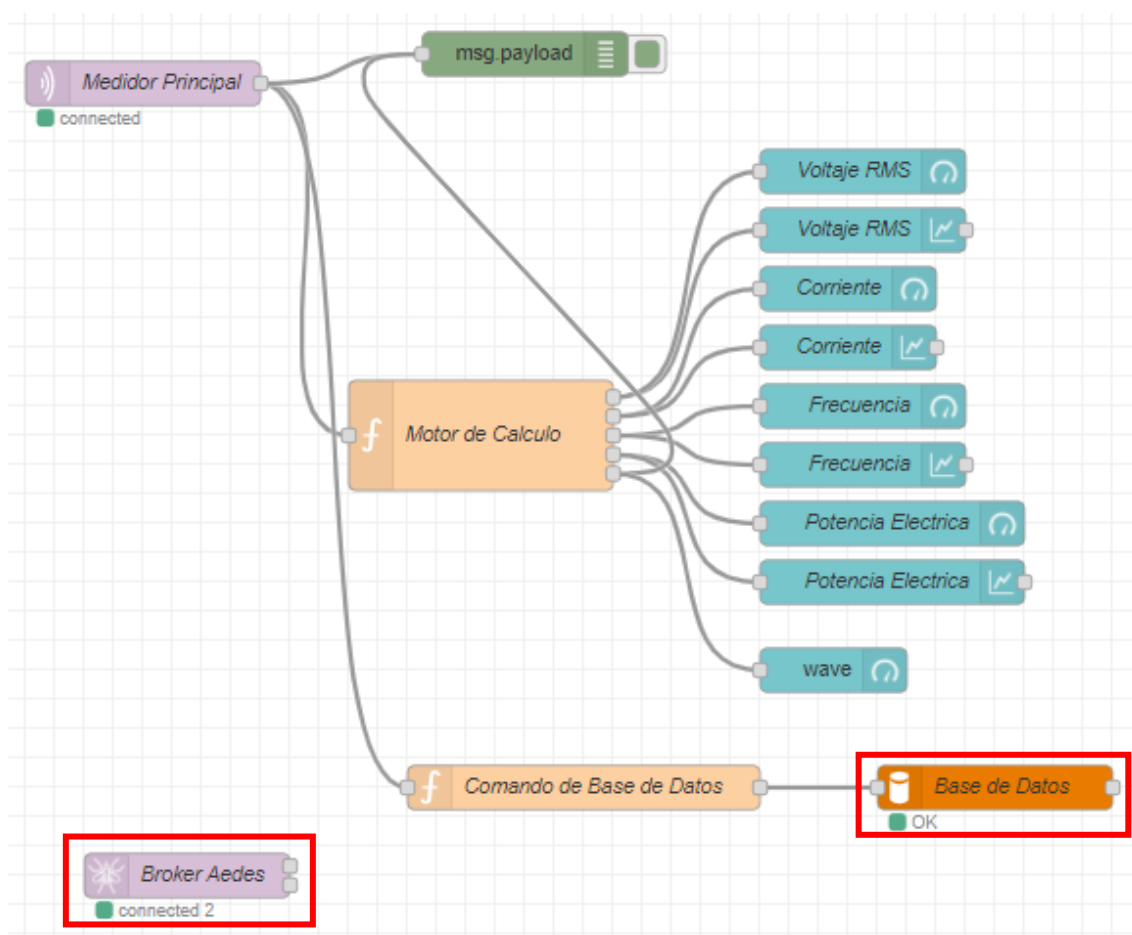
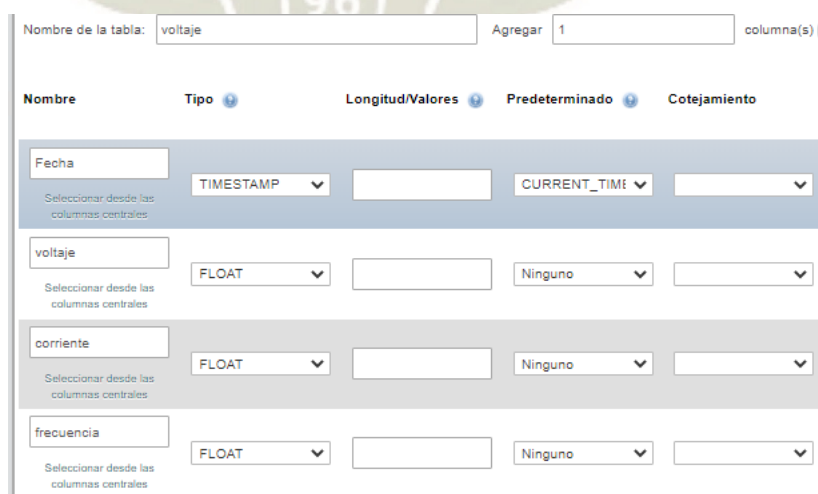


Figura 4.48 Conexión exitosa de la base de datos en Node Red

Fuente: XAMPP 2021

Por último, creamos las tablas de acuerdo a nuestra conveniencia para obtener los datos enviados por el medidor de energía.



Nombre	Tipo	Longitud/Valores	Predeterminado	Cotejamiento
Fecha	TIMESTAMP		CURRENT_TIME	
voltaje	FLOAT		Ninguno	
corriente	FLOAT		Ninguno	
frecuencia	FLOAT		Ninguno	

Figura 4.49 Creación de tablas en la web de XAMPP

Fuente: XAMPP 2021

8. DIAGRAMAS DE FLUJO DEL SISTEMA OPERATIVO

8.1 DIAGRAMA DE FLUJO DEL PROGRAMA PRINCIPAL

En el flujograma 4.1 se muestra el diagrama general indicando las diferentes tareas que van a poder analizar los medidores de energía.

Los equipos presentan las siguientes funciones:

- Comunicación a la red WIFI
- Visualización de datos en la pantalla LCD
- Almacenamiento de datos en la nube
- Actualización de hora y fecha

Al conectar nuestro equipo a la red principal de la vivienda podremos visualizar en una primera pantalla el SSID y RSSI, esperando unos breves segundos se visualiza una segunda pantalla con los datos del IP local, el tópico de publicación y en una tercera pantalla el servidor al cual está conectado; en este caso mi laptop.



Figura 4.50 Visualización de la primera pantalla

Fuente: Elaboración propia

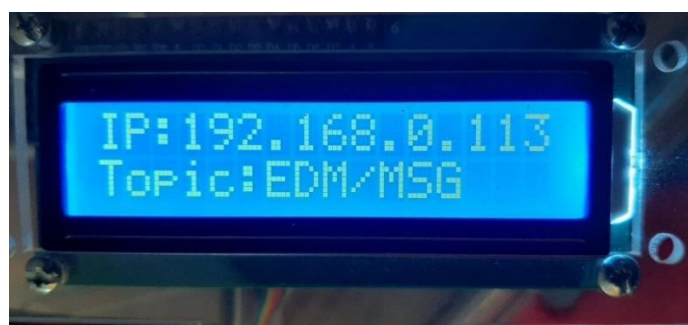


Figura 4.51 Visualización de la segunda pantalla

Fuente: Elaboración propia

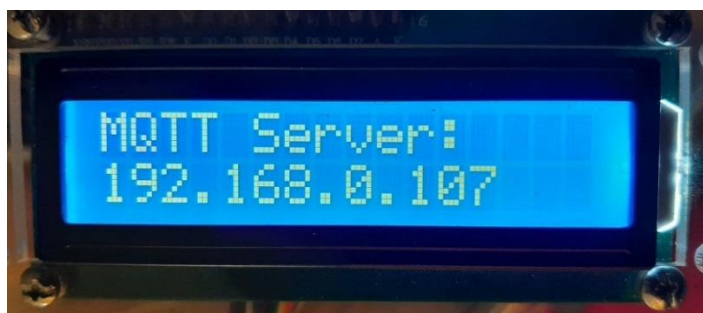
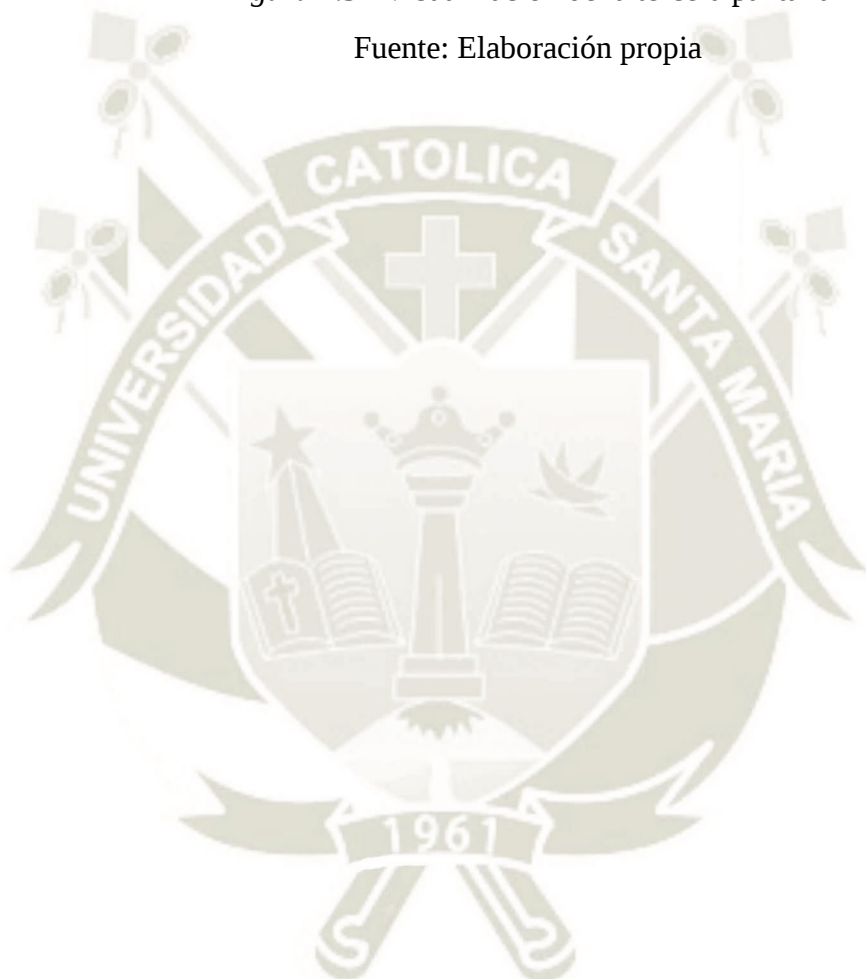
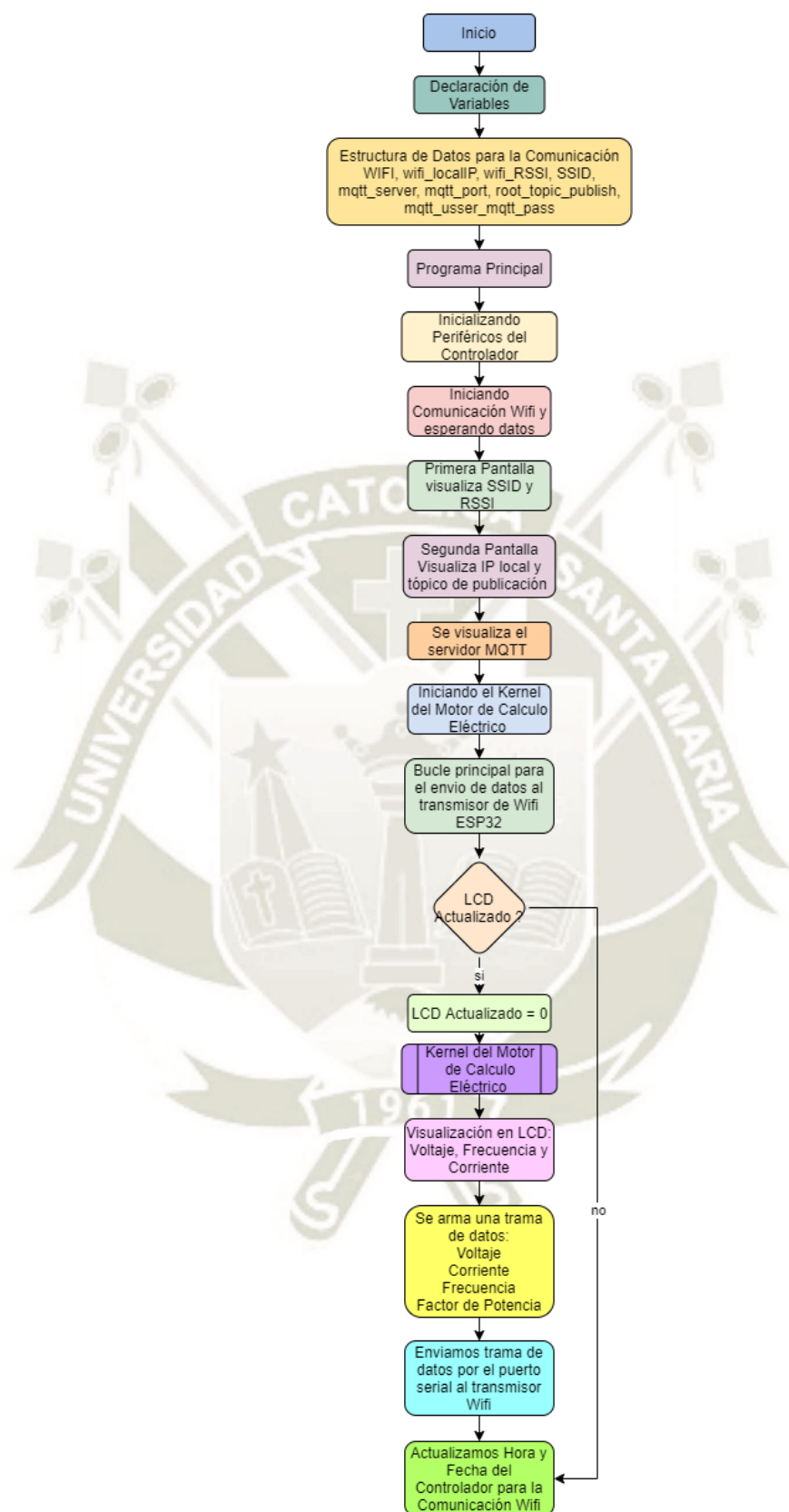


Figura 4.52 Visualización de la tercera pantalla

Fuente: Elaboración propia





Flujograma 4.1 Sistema de control general

Fuente: Elaboración propia

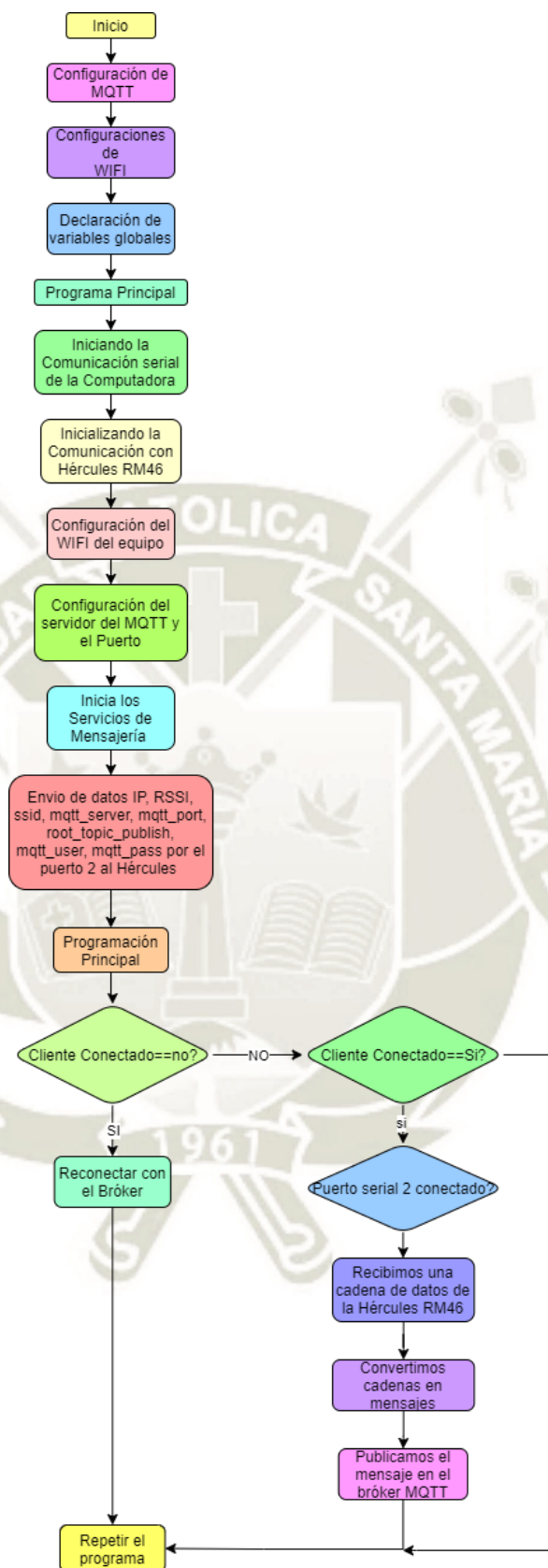
8.2 DIAGRAMA DE FLUJO DEL MÓDULO WIFI ESP32

Tenemos la interfaz del módulo WIFI ESP32, el cual se encarga de la conexión a la red de nuestras viviendas y así nosotros como usuarios podamos ver el consumo de nuestras viviendas por nuestros celulares, computadoras o tablets, teniendo en cuenta que el servidor es una computadora instalada en una estación concentradora de datos.

Como podemos observar en la figura 4.51, el servidor obtiene una IP de la red local del Wifi de las viviendas.

Para establecer una conectividad con la red local el equipo de medición de energía sigue los pasos en el Flujograma 4.2, donde se puede observar el siguiente procedimiento de conectividad con la red local:

- i. El microcontrolador establece en su memoria interna el Tópico y la mensajería del protocolo MQTT.
- ii. Se declaran las variables globales del ESP32 y se inicializa la comunicación serial a 9600 bites por segundo (baudios), con el microcontrolador Hércules RM46, quien se encarga de realizar las mediciones de corriente y voltaje y enviar esta comunicación.
- iii. Se comunica con la red Wifi a través del SSID y la contraseña, para luego obtener una IP automática del MODEM de la vivienda.
- iv. Luego consulta la comunicación con el Bróker de datos AEDES que se encuentra instalada en el servidor, para reportar que empezara a publicar datos.
- v. Espera en un bucle cerrado que el procesador Hércules RM46 envíe los datos de medición de energía, para almacenarlos en una cadena de texto y luego publicarlos en el Bróker AEDES.
- vi. Se repite el proceso desde el punto 4.



Flujograma 4.2 Módulo ESP32

Fuente: Elaboración propia

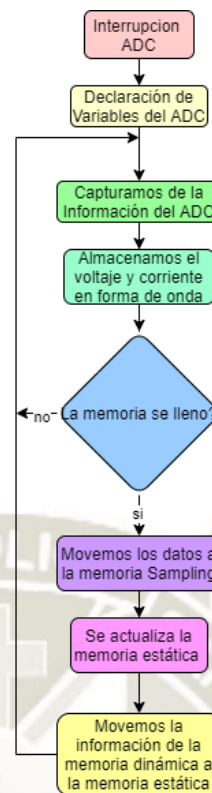
8.3 DIAGRAMA DE FLUJO DEL MOTOR DE CÁLCULO

El siguiente algoritmo se ejecuta en el microcontrolador Hércules RM46 debido a su alta capacidad en memoria RAM, memoria Flash y la velocidad de su procesador de 220 MHz.

El algoritmo ejecuta dos tareas en paralelo:

Tarea 1: Muestreo por interrupciones.

- i. Se genera una interrupción cada 40 microsegundos.
- ii. El procesador al atender la interrupción, realiza una captura de la corriente y el voltaje muestreados por el ADC del microcontrolador y lo almacena en un sector de memoria de un array de datos en punto flotante.
- iii. En conjunto con esta tarea un contador de eventos va incrementando su valor hasta llegar a un límite de 2048 eventos.
- iv. Este contador también funciona como índice para mover los datos dentro del array.
- v. Una vez almacenados los 2048 eventos, el procesador congela la interrupción y procede a mover todo el array de 2048 valores a otro sector de memoria.
- vi. Este sector de memoria, alimenta de información al Kernel del motor de cálculo eléctrico.



Flujograma 4.3 Muestreo por interrupciones

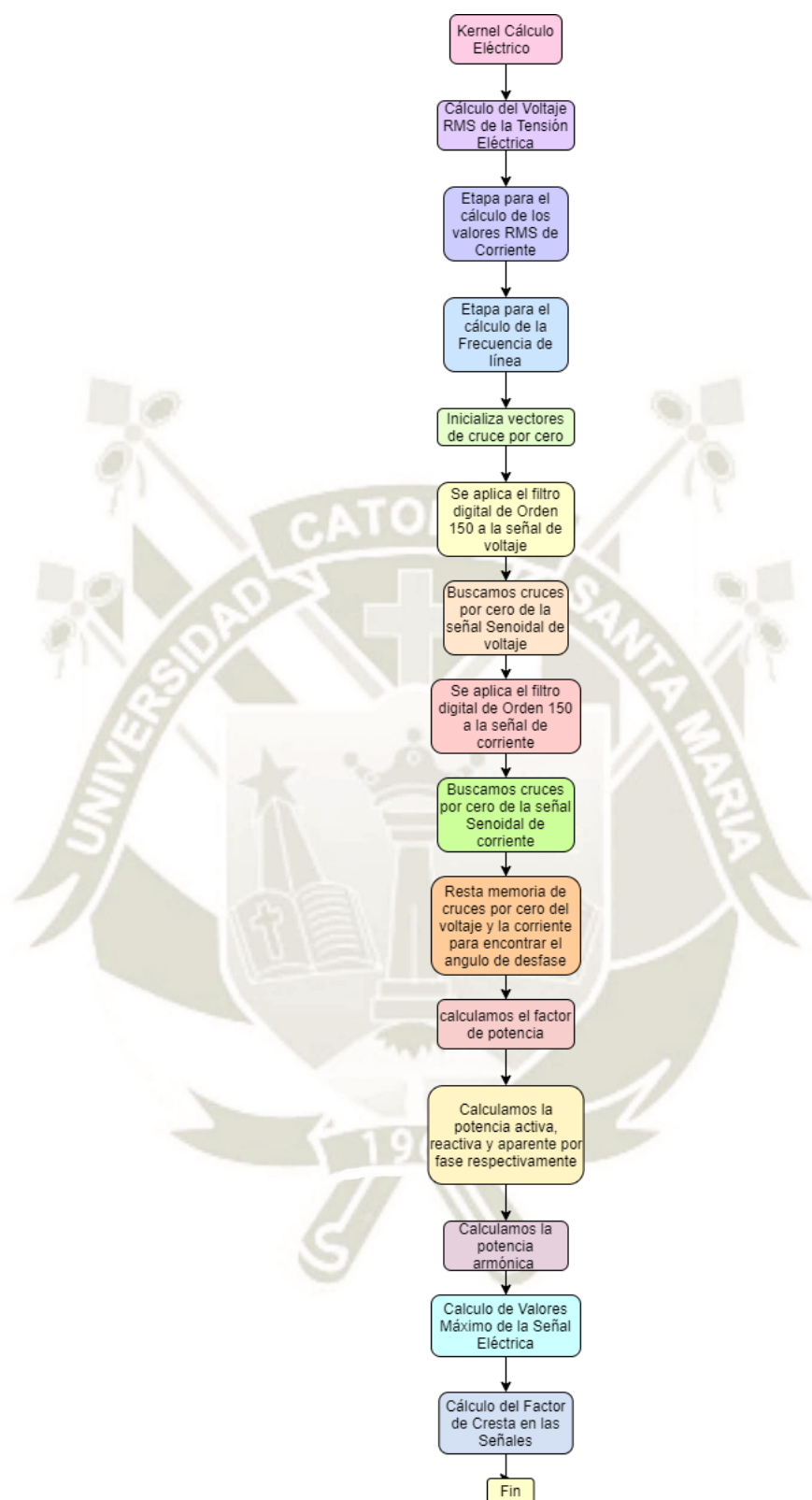
Fuente: Elaboración propia

Tarea 2: Kernel del Motor de Calculo Eléctrico

- i. El Kernel de cálculo eléctrico recibe el array de datos de la corriente y el voltaje y procede.
- ii. La primera tarea que se realiza es ejecutar el algoritmo del RMS para los datos de la corriente y el voltaje.
- iii. Luego se aplica un filtro digital para solo tener las frecuencias fundamentales de la corriente y el voltaje.
- iv. Una vez que se tiene la corriente y el voltaje sin armónicos en la memoria del microcontrolador, se procede a buscar sus cruces por cero.
- v. Los cruces por cero se almacenan en la memoria RAM del microcontrolador para luego calcular un promedio de dimensión entre un cruce por cero y otro.
- vi. Con esta dimensión se estima el ángulo de desfase entre la corriente y el voltaje.
- vii. Este ángulo de desfase sirve para el cálculo del factor de potencia eléctrica.

- viii. El microcontrolador primero calcula la potencia aparente, luego con el factor de potencia, calcula la potencia activa y por último la potencia reactiva.
- ix. Con los datos almacenados en la memoria RAM en forma de estructura de datos, se envían a través de mensajería serial al ESP32.
- x. El proceso se repite desde el paso 1.





Flujograma 4.4 Motor de cálculo

Fuente: Elaboración propia

8.4 DIAGRAMA DE FLUJO DE NODE RED

Node Red se ejecuta en el servidor de datos, cumple la función de reunir la información enviada por el equipo, escribir en la base datos y generar reportes o Dashboard en la WEB.

Procedimiento:

- i. El nodo “Bróker Aedes”, se encarga de establecer la comunicación el ESP32 a través del tópico de datos.
- ii. El nodo Medidor Principal, se encarga de leer la mensajería enviada por el ESP32 al servidor.
- iii. El nodo Motor de cálculo, ejecuta las siguientes líneas de comando.

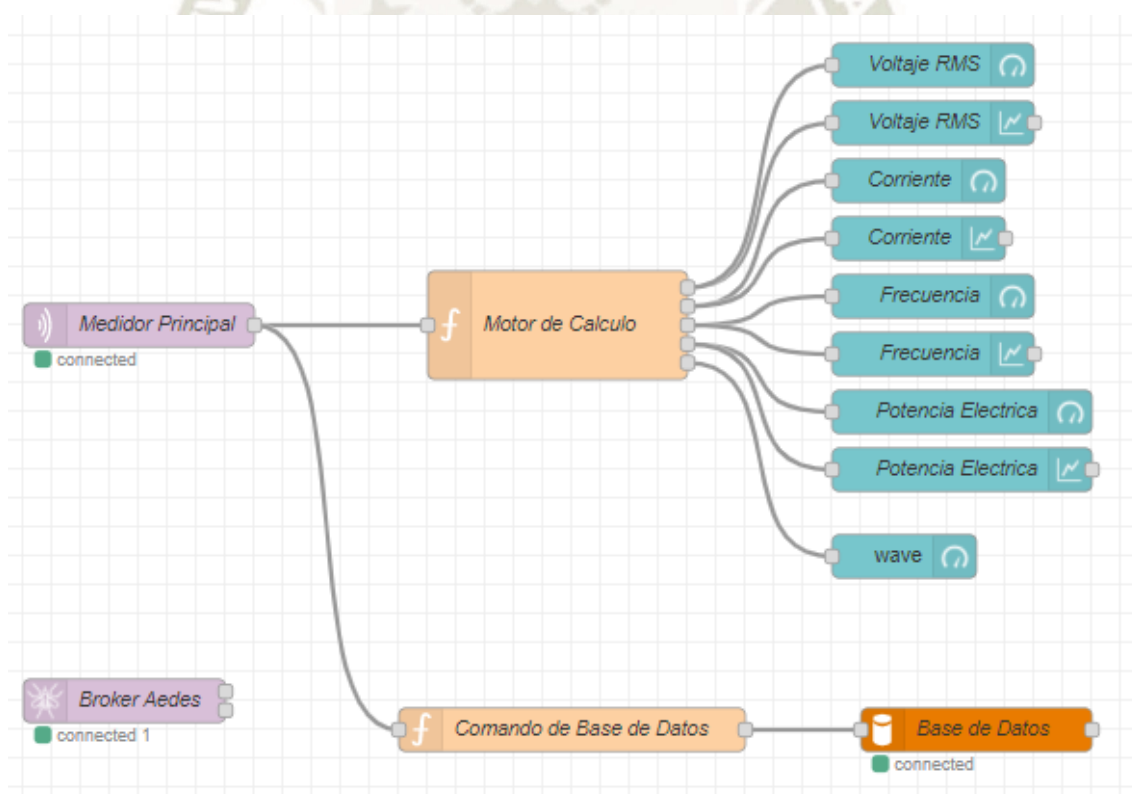
```
var output=msg.payload.split(",");
/*Variables recibidas del equipo transmisor/
var Voltaje=parseFloat(output[0],2);
var Corriente=parseFloat(output[1],2)-0.15;
var Frecuencia=parseFloat(output[2],2);
var Energia_KWH=parseFloat(output[3],2);
/Variables calculadas/
var Potencia_Electrica=(Voltaje * Corriente)/1000; //KW
var msg0={payload:Voltaje};
var msg1={payload:Corriente.toFixed(2)};
var msg2={payload:Frecuencia};
var msg3={payload:Potencia_Electrica.toFixed(2)};
var msg4={payload:Energia_KWH.toFixed(2)};
return [msg0,msg1,msg2,msg3,msg4]
```

- iv. El código anterior se encarga de leer la mensajería **msg.payload.split(",")** y cortar este mensaje en diferentes partes para luego utilizar el **parseFloat**, para convertir el mensaje en un valor decimal.
- v. Para finalizar este bloque, la mensajería se parte en 5 salidas diferentes, lo que permite generar 5 dashboards.

- vi. El nodo comando de base de datos, solo tiene como función recibir la mensajería y agregar la cabecera de los códigos de la base de datos.

```
var output=msg.payload.split(",");
var Voltaje=parseFloat(output[0]);
var Corriente=parseFloat(output[1]);
var Frecuencia=parseFloat(output[2]);
msg.topic= "INSERT INTO voltaje(voltaje,corriente,frecuencia)
VALUES('"+Voltaje+"','"+Corriente+"','"+Frecuencia+"')";
return msg;
```

- vii. La programación de Node Red no funciona en bucle, se ejecuta siempre que se recibe un nuevo mensaje por parte del ESP32.



Flujograma 4.5 Node Red

Fuente: Elaboración propia

CAPÍTULO V
IMPLEMENTACIÓN DEL EQUIPO

En este capítulo se describirá cada parte de la elaboración del equipo con todos los circuitos diseñados para los dos equipos medidos de energía eléctrica basados en IoT para gestión de consumo energético.

1. ETAPA ANÁLOGA

1.1 CIRCUITO DEL SENSOR DE VOLTAJE

Se realizó el circuito en protoboard de acuerdo con el diseño del esquemático de la figura 3.12, haciéndose las pruebas pertinentes para su correcto funcionamiento.

Una vez realizado el circuito en protoboard para detectar posibles fallas y corregirlas anticipadamente se procedió a realizar dicho circuito en 2 placas para los 2 equipos.

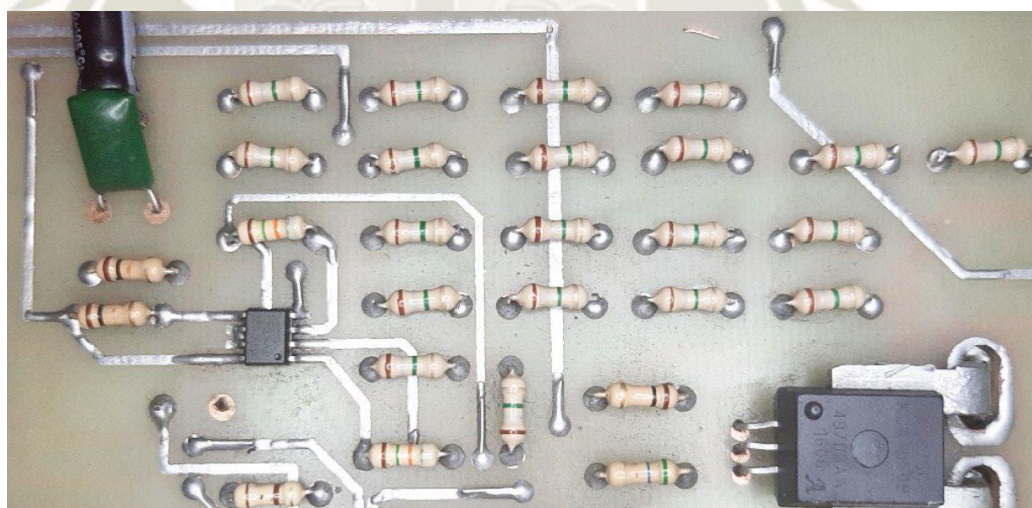


Figura 5.1 Circuito equipo sensor de voltaje armado en placa

Fuente: Elaboración propia

1.2 CIRCUITO DEL SENSOR DE CORRIENTE

Para la etapa del sensado de corriente se utilizó un sensor de corriente ACS 756-050, primeramente, se hizo pruebas con un circuito simple para corroborar que el sensado sea el correcto.

Una vez que las pruebas fueron las que se necesitaban se procedió a realizar dicho circuito en 2 placas para los 2 equipos.

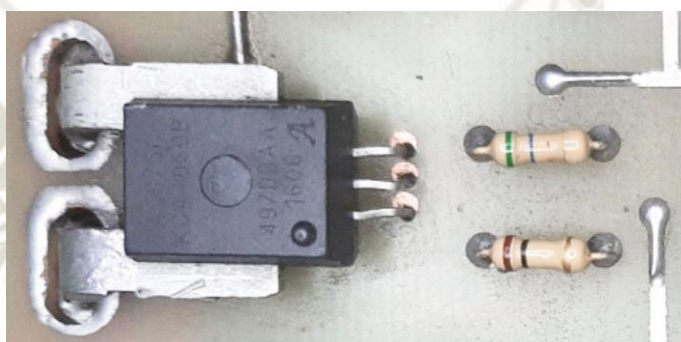


Figura 5.2 Circuito equipo sensor de corriente ACS750-50 armado en placa

Fuente: Elaboración propia

Se realizó la medición de la salida del sensor en un osciloscopio, así como el cálculo del valor RMS.

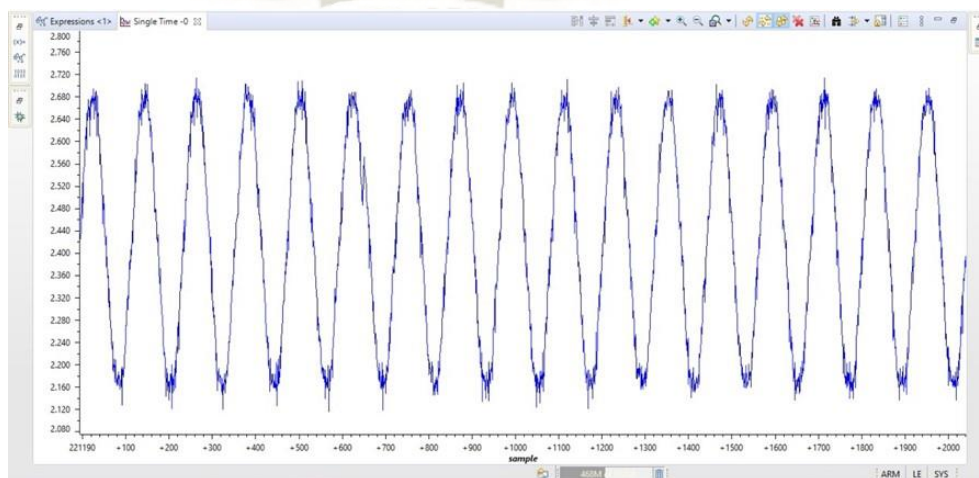


Figura 5.3 Señal de salida del sensor de corriente

Fuente: Elaboración propia

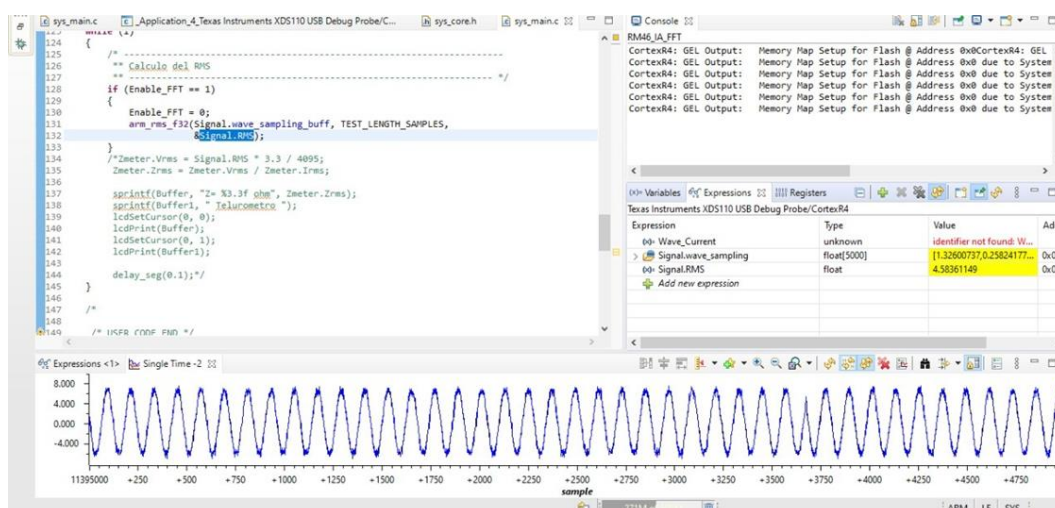


Figura 5.4 Cálculo del valor RMS

Fuente: Elaboración propia

1.3 CIRCUITO DEL SENSOR DE FRECUENCIA

En el circuito del sensor de frecuencia se basa en un comparador y en compuertas lógicas inversoras, también, se utiliza la señal de voltaje como la fuente para la medición de frecuencia.

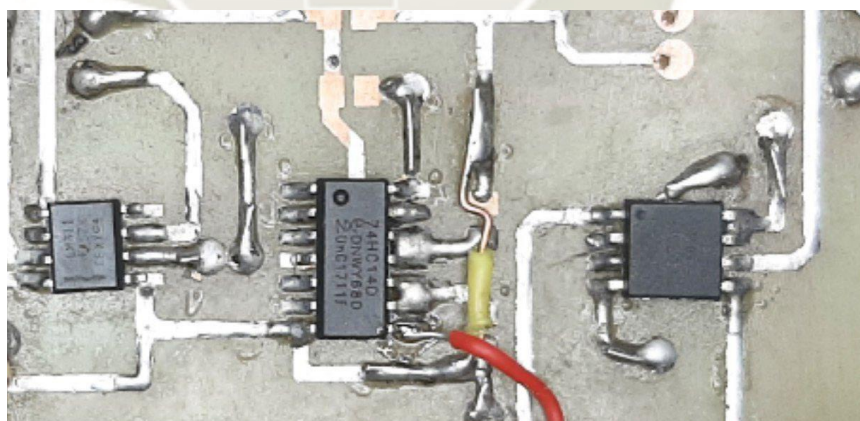


Figura 5.5 Circuito equipo sensor de corriente armado en placa

Fuente: Elaboración propia

2. ETAPA DIGITAL

2.1 CIRCUITO DE RELOJ

El circuito de reloj es el encargado de almacenar los eventos en la nube con hora y fecha actualizada.

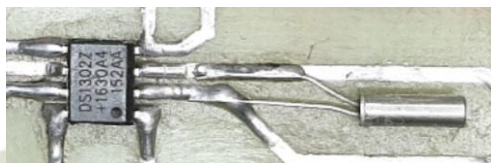


Figura 5.6 Circuito de reloj DS1302

Fuente: Elaboración propia

2.2 CIRCUITO DE LA MEMORIA EEPROM 25LC256

La memoria EEPROM es la encargada guardar los archivos almacenados.



Figura 5.7 Circuito de memoria EEPROM 25LC256

Fuente: Elaboración propia

3. ETAPA DE VISUALIZACIÓN DE DATOS E INTERFAZ

3.1 CIRCUITO PARA BOTONERAS DE CONTROL

La única función de las botoneras es para la reparación del equipo cuando este se descalibre, únicamente pueden ser manipuladas por un personal calificado. Por ende, el usuario no tiene acceso a ellas.

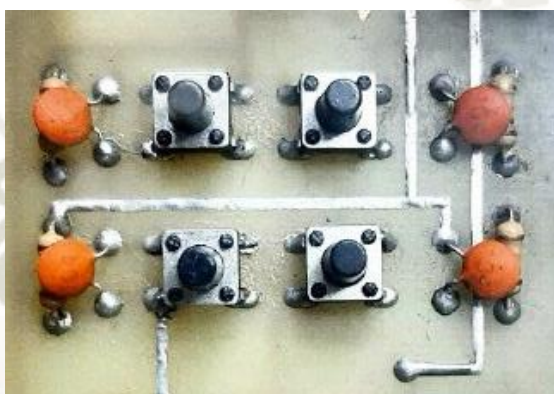


Figura 5.8 Circuito para botoneras de control

Fuente: Elaboración propia

3.2 CIRCUITO PARA LA PANTALLA LCD SERIAL

La pantalla LCD es la encargada de mostrar los valores medidos.



Figura 5.9 Circuito para pantalla LCD

Fuente: Elaboración propia

4. EQUIPOS MEDIDORES DE ENERGÍA ELÉCTRICA



Figura 5.10 Equipo medidor de energía eléctrica

Fuente: Elaboración propia

5. INSTALACIÓN Y CONFIGURACIÓN DEL EQUIPO MEDIDOR DE ENERGÍA

IMPORTANTE: la instalación debe ser realizada por personal con conocimientos en trabajo con líneas eléctricas activas, es decir 220 VAC, con los respectivos Equipos de Protección Personal.

Pasos para la instalación del equipo:

- i. Revisar el área donde se va a instalar el equipo medidor de energía, de preferencia en la llave principal de la vivienda.
- ii. Demarcar el área de trabajo para poder trabajar con equipos energizados.
- iii. Luego realizar una identificación de peligros y evaluación de riesgos y controles.
- iv. Abrir el tablero eléctrico principal de la vivienda.



Figura 5.11 Tablero eléctrico principal de una vivienda

Fuente: Elaboración propia

- v. Identificar la llave principal de energía.

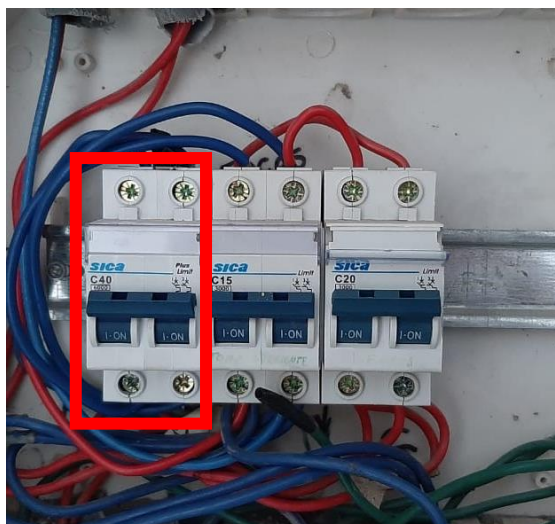


Figura 5.12 Llave principal del tablero

Fuente: Elaboración propia

- vi. Se deben desconectar los cables eléctricos que salen por debajo de la llave principal

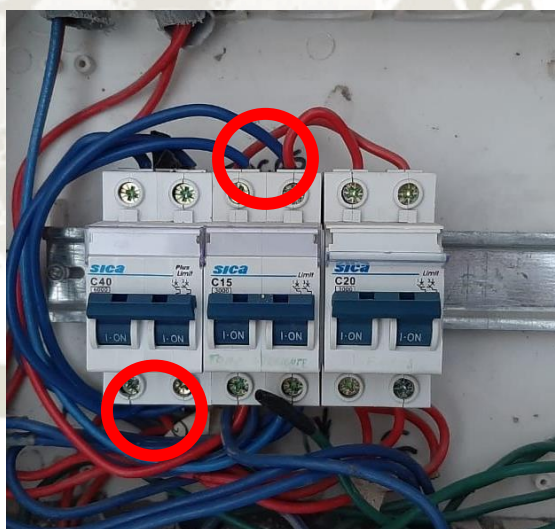


Figura 5.13 Cables a desconectar

Fuente: Elaboración propia

- vii. Conectar un par de cables de la llave principal a la entrada de energía del medidor.
- viii. Los cables que fueron desconectados de la llave principal, se deben conectar a la salida de energía del medidor.

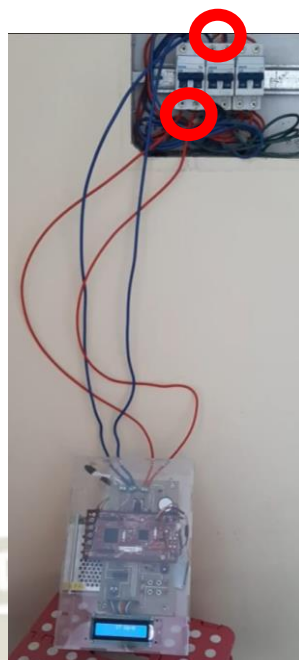


Figura 5.14 Cables del equipo conectados a la llave principal

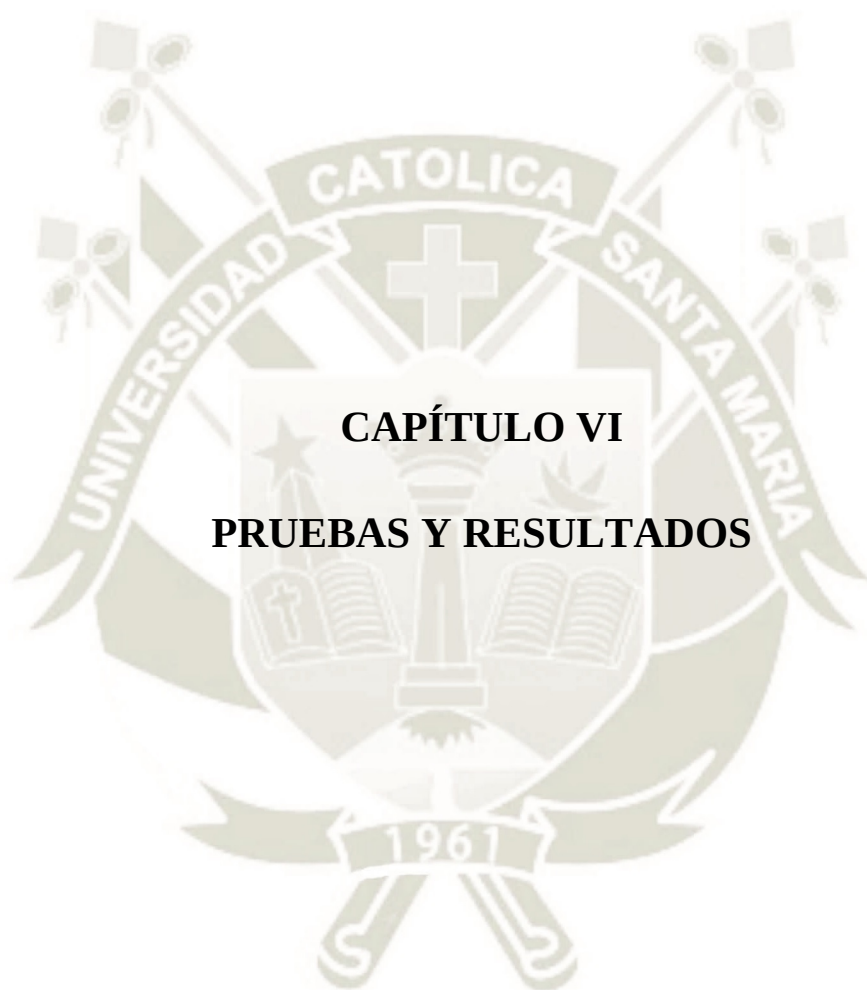
Fuente: Elaboración propia

- ix. Fijar el equipo a la pared cerca del tablero principal de energía.
- x. Para la configuración de la red wifi, se tiene que conectar la laptop con el software de comunicación y enviarle los parámetros de la red wifi local.
- xi. Luego de la configuración de red, se tiene que visualizar en pantalla que se conectó correctamente a la red, esta confirmación se da cuando el equipo muestra en su pantalla LCD la dirección IP.
- xii. La comunicación el servidor de datos, se da cuando se muestra por pantalla el tópico de comunicación MQTT.
- xiii. El equipo inicia la medición y realiza la transmisión de datos cada 5 segundos.



Figura 5.15 Equipo midiendo correctamente

Fuente: Elaboración propia



CAPÍTULO VI

PRUEBAS Y RESULTADOS

Teniendo el equipo ya implementado correctamente se hicieron las pruebas en diferentes viviendas conectándose exitosamente a la red WIFI de cada vivienda.

1. CONEXIÓN DE LOS EQUIPOS A LA RED WIFI DE LAS VIVIENDAS

Para que los equipos midan correctamente la energía de Las viviendas, primero deben de reconocer su propio IP, así como el IP del servidor.



Figura 6.1 Reconocimiento de IP del equipo

Fuente: Elaboración propia



Figura 6.2 Reconocimiento de IP del servidor

Fuente: Elaboración propia

Luego, el equipo debe de conectarse automáticamente a la red wifi de las viviendas, esto gracias al módulo ESP32 que es el encargado de esta tarea.



Figura 6.3 Conexión exitosa a la red WIFI

Fuente: Elaboración propia

2. MEDICIÓN DE VOLTAJE Y CORRIENTE ELÉCTRICA

Para la medición eléctrica se muestra una primera pantalla indicando dicha medición.



Figura 6.4 Título de medición eléctrica

Fuente: Elaboración propia

Luego el equipo empieza con las mediciones de voltaje y corriente requeridas como se muestra a continuación.

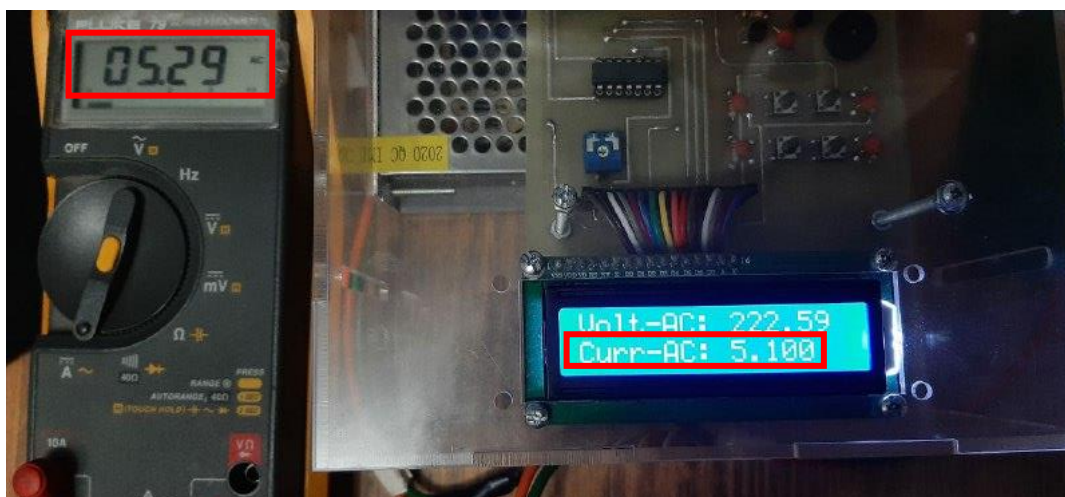


Figura 6.5 Medición de voltaje y corriente AC equipo 1

Fuente: Elaboración propia



Figura 6.6 Medición de voltaje y corriente AC equipo 2

Fuente: Elaboración propia

Como se puede observar en las figuras 6.5 y 6.6, los equipos miden voltaje en AC y corriente en AC, corroborando dicha medición con un multímetro. Para llegar a este resultado, se tuvo que calibrar el sensor de corriente.

También se procede a comprobar dichas mediciones en los dashboards creados con sus respectivas tendencias como se muestra a continuación.

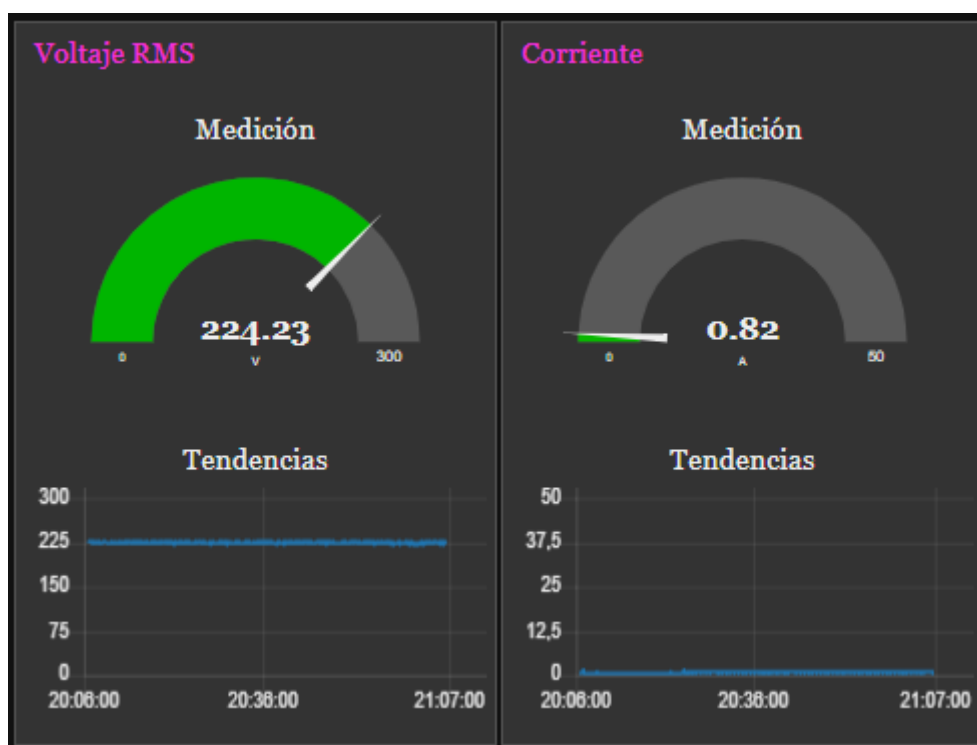


Figura 6.7 Dashboard de voltaje y corriente AC equipo 1

Fuente: Elaboración propia

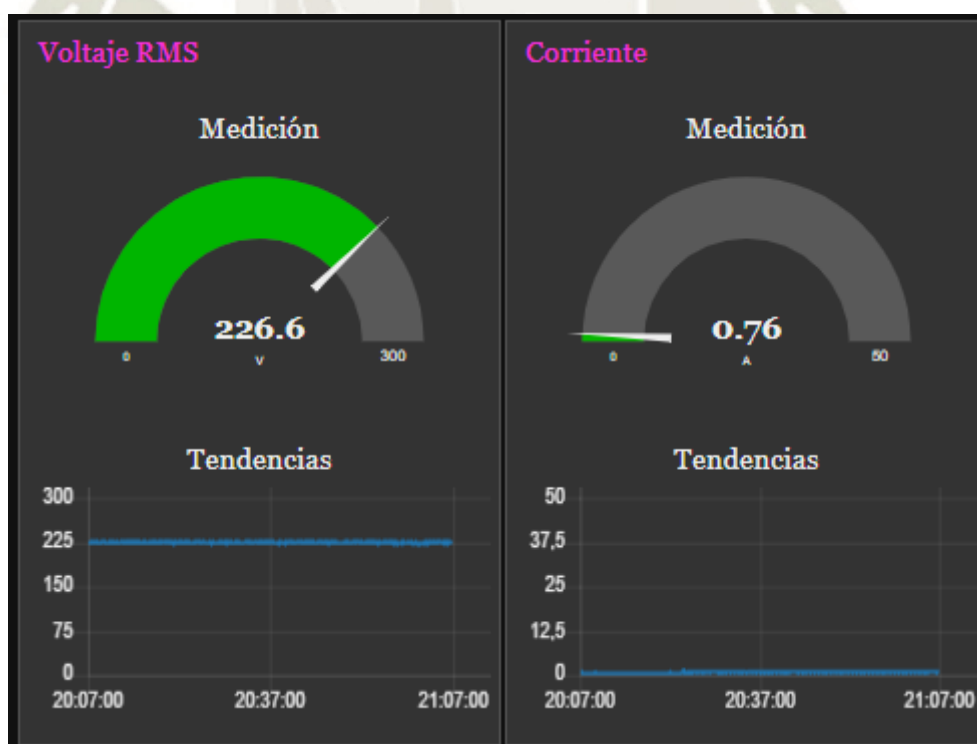


Figura 6.8 Dashboard de voltaje y corriente AC equipo 2

Fuente: Elaboración propia

3. MEDICIÓN DE FRECUENCIA ELÉCTRICA

Para la medición de frecuencia se muestra una pantalla con su respectivo título.



Figura 6.9 Título de medición de frecuencia

Fuente: Elaboración propia

Luego el equipo mide la frecuencia en Hz con su respectiva diferencia, comparando dichos resultados con un multímetro.



Figura 6.10 Medición de frecuencia equipo 1

Fuente: Elaboración propia



Figura 6.11 Medición de frecuencia equipo 2

Fuente: Elaboración propia

Se procede a visualizar los dashboards para ambos equipos con sus respectivas tendencias.



Figura 6.12 Dashboard de frecuencia equipo 1

Fuente: Elaboración propia



Figura 6.13 Dashboard de frecuencia equipo 2

Fuente: Elaboración propia

4. MEDICIÓN DE POTENCIA ELECTRICA (KW)

Los 2 equipos medidores de energía también tienen la capacidad de calcular la potencia activa y la potencia aparente como se muestra a continuación.



Figura 6.14 Título de potencia eléctrica

Fuente: Elaboración propia



Figura 6.15 Cálculo de potencia activa y potencia aparente equipo 1

Fuente: Elaboración propia



Figura 6.16 Cálculo de potencia activa y potencia aparente equipo 2

Fuente: Elaboración propia

A continuación, se muestran los dashboards para la potencia eléctrica en una vivienda.



Figura 6.17 Dashboard de potencia eléctrica

Fuente: Elaboración propia

5. MEDICIÓN DE POTENCIA ELÉCTRICA (KWH)

Por último, el equipo también tiene la capacidad de calcular la energía eléctrica consumida en las viviendas, siendo acumulativa de acuerdo al consumo que se tenga.

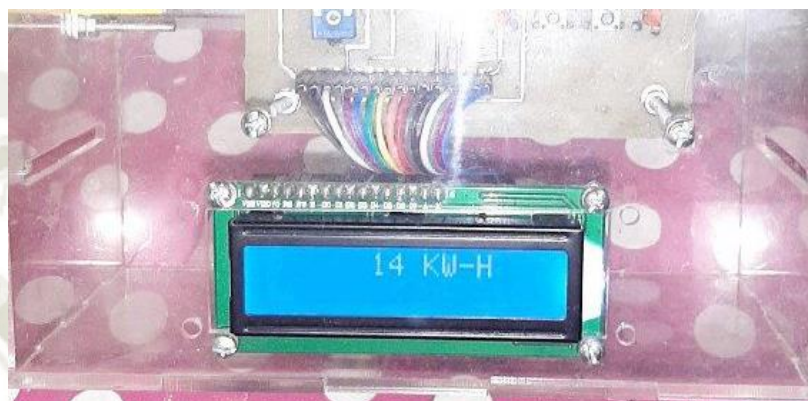


Figura 6.18 Medición de energía eléctrica en una vivienda

Fuente: Elaboración propia

Se muestran los dashboards creados en Node Red para el cálculo de energía eléctrica KWH.



Figura 6.19 Dashboard de medición de energía eléctrica

Fuente: Elaboración propia



Figura 6.20 Dashboard general

Fuente: Elaboración propia

Podemos constatar que la base de datos junto con Node Red están recibiendo los datos correctamente de los equipos medidores de energía.

```

4/11/2021 11:41:53 node: c9f3e34a.8a5f7
msg.payload : number
-0.0010636834279719497

4/11/2021 11:42:48 node: ca2683312ee23c48
EDM/MSG : msg.payload : string[24]
▶ "227.95,1.07,60.08,0.70"

4/11/2021 11:42:48 node: c9f3e34a.8a5f7
msg.payload : number
-0.000674348026816837

4/11/2021 11:43:41 node: ca2683312ee23c48
EDM/MSG : msg.payload : string[24]
▶ "226.41,1.02,60.05,0.95"

4/11/2021 11:43:41 node: c9f3e34a.8a5f7
msg.payload : number
-0.0010902477079592651

4/11/2021 11:44:38 node: ca2683312ee23c48
EDM/MSG : msg.payload : string[24]
▶ "229.45,1.05,60.06,1.00"

4/11/2021 11:44:38 node: c9f3e34a.8a5f7
msg.payload : number
-0.00136663681

4/11/2021 11:45:30 node: ca2683312ee23c48
EDM/MSG : msg.payload : string[24]
▶ "228.39,1.02,60.03,0.37"
    
```

Figura 6.21 Datos recibidos en Node Red

Fuente: Elaboración propia

6. ANÁLISIS ECONÓMICO

El análisis económico de la presente tesis está basado en la inversión que se realizó en la fabricación de los 2 medidores de energía eléctrica basados en IoT para gestión de consumo energético, los componentes adquiridos y la inversión de tiempo en horas.



SENSOR DE VOLTAJE

ítem	DESCRIPCIÓN	VALOR	CANTIDAD	COSTO POR UNIDAD	PRECIO
1	RESISTENCIAS DE CARBÓN A 1/4 W	1.8MOHM	40	S/ 0.1	S/ 4
2	RESISTENCIAS DE CARBÓN A 1/4 W	7.5KOHM	8	S/ 0.1	S/ 0.8
3	OPAMP DE ALTA PRECISIÓN RAIL TO RAIL	MCP6022	2	S/ 5.62	S/ 11.24
4	CONDENSADOR NO POLARIZADO	104	4	S/ 0.3	S/ 1.2

SENSOR DE CORRIENTE

1	SENSOR DE CORRIENTE DE 50 AMPERIOS	ACS756-050	2	S/ 40	S/ 80
2	RESISTENCIAS DE CARBÓN A 1/4 W	5.6K	2	S/ 0.1	S/ 0.2
3	RESISTENCIAS DE CARBÓN A 1/4 W	10K	2	S/ 0.1	S/ 0.2

SENSOR DE FRECUENCIA

1	COMPUERTA SMICHT TRIGGER - SMD	74HC14	6	S/ 1.8	S/ 10.8
2	COMPARADOR CMOS - SMD	LM311	2	S/ 2.56	S/ 5.12
3	RESISTENCIAS DE CARBÓN A 1/4 W	20K	2	S/ 0.1	S/ 0.2
4	RESISTENCIAS DE CARBÓN A 1/4 W	10K	2	S/ 0.1	S/ 0.2
5	CONDENSADOR NO POLARIZADO	104	6	S/ 0.3	S/ 1.8

COMUNICACIÓN WIFI

1	MÓDULO WIFI	ESP-32	2	S/ 40	S/ 80
2	CONECTOR TIPO ESPADÍN HEMBRA		6	S/ 1.5	S/ 9

ALMACENAMIENTO Y RELOJ

1	MODULO EEPROM SPI -SMD	25LC256	2	S/ 5.58	S/ 11.16
2	RELOJ EN TIEMPO REAL - SMD	DS1302	2	S/ 8	S/ 16
3	CRISTAL OSCILADOR DE 32KHZ PARA RELOJ	XTAL 32KHZ	2	S/ 0.5	S/ 1
4	PORTA BATERÍAS PARA BATERÍA CR2032		2	S/ 1	S/ 2
5	PILA TIPO PASTILLA DE 3.0 VOLTIOS	CR2032	2	S/ 5	S/ 10

VISUALIZACIÓN

1	PANTALLA LCD DE 16 CARACTERES X 2 HILOS	LCD16X4	2	S/ 18	S/ 36
---	---	---------	---	-------	-------

2	CONECTOR TIPO ESPADÍN HEMBRA		2	S/ 2	S/ 4
3	TRANSISTOR NPN	2N3904	4	S/ 0.5	S/ 2
4	RESISTENCIAS DE CARBÓN A 1/4 W	2.2K OHM	2	S/ 0.1	S/ 0.2
6	RESISTENCIAS DE CARBÓN A 1/4W	10K	5	S/ 0.3	S/ 1.5
7	CONDENSADOR NO POLARIZADO	104	5	S/ 0.3	S/ 1.5
8	TRIMPOT COLOR AZUL	50K	2	S/ 1	S/ 2
9	CONECTOR TIPO PEINE MACHO		4	S/ 1	S/ 4
10	BORNERAS DE 2 ENTRADAS COLOR VERDE		4	S/ 1	S/ 4
11	PULSADORES DE CABEZA LARGA 4 PATITA PARA SOLDAR EN PLACA		8	S/ 0.5	S/ 4
TOTAL					S/ 309.12

Tabla 6.1 Análisis económico general de los componentes

Fuente: Elaboración propia

ANÁLISIS ECONÓMICO DE LOS EQUIPOS				
ÍTEM	DESCRIPCIÓN	CANTIDAD	COSTO POR UNIDAD	PRECIO
1	CAJA DE ACRÍLICO PARA EL EQUIPO	2	S/ 180	S/ 360
2	PERNOS DE 25 X 14 MM	8	S/ 0.6	S/ 4.8
3	PERNOS DE 10 X 14 MM	8	S/ 0.4	S/ 3.2
4	VOLANDAS	16	S/ 0.1	S/ 1.6
TOTAL				S/ 369.6

Tabla 6.2 Análisis económico de los equipos

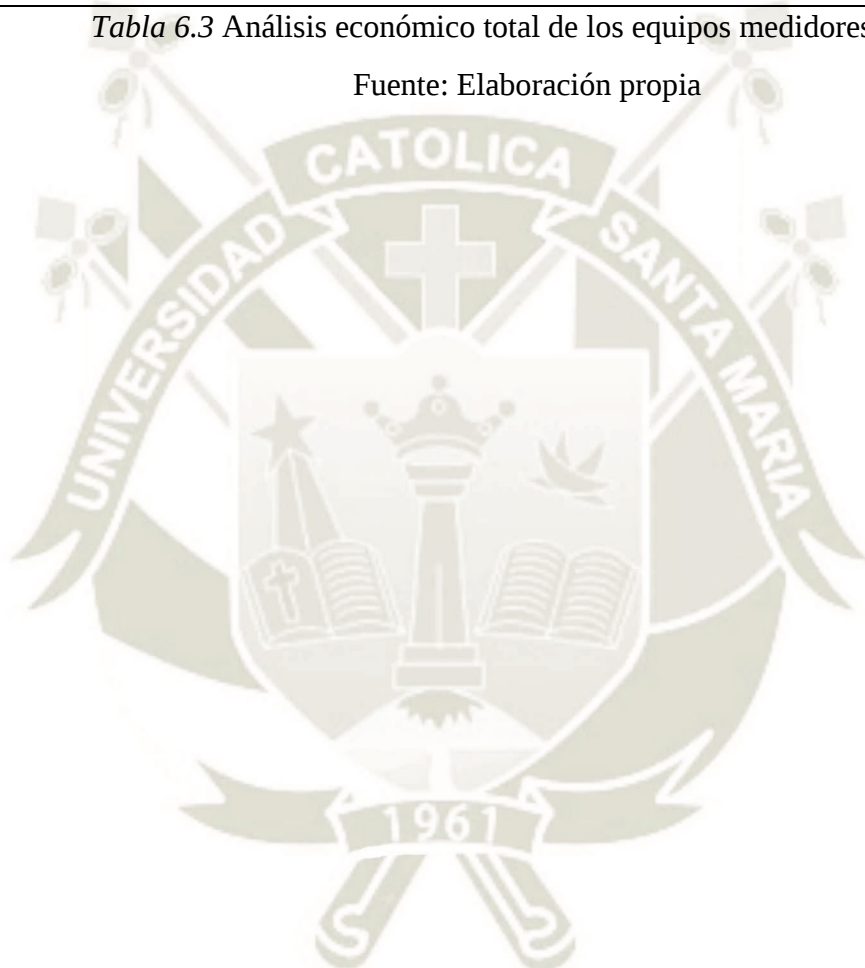
Fuente: Elaboración propia

A estos gastos se les considera las horas de trabajo lo cual implica la implementación de los 2 equipos, el diseño e implementación de las 2 placas, así mismo las pruebas realizadas para su correcto funcionamiento.

DESCRIPCIÓN	COSTO
COMPONENTES ELECTRÓNICOS GENERALES	S/ 309.12
CAJAS DE ACRÍLICO PARA LOS EQUIPOS	S/ 369.6
DISEÑO E IMPLEMENTACIÓN DE LAS PLACAS Y EL EQUIPO	S/ 1800
DESARROLLO DEL PROGRAMA PARA MEDIDORES DE ENERGÍA ELÉCTRICA BASADOS EN IOT	S/ 2000
TOTAL	S/ 4478.72

Tabla 6.3 Análisis económico total de los equipos medidores de energía

Fuente: Elaboración propia



CAPÍTULO VII
CONCLUSIONES Y RECOMENDACIONES

CONCLUSIONES

1. Se implementaron 2 medidores de energía para utilizarlos en viviendas diferentes, para comunicarlos a través de internet con el servidor de datos, ambos medidores publican el mismo tipo de información, pero para la estimación de las especificaciones técnicas del servidor, solo se utilizaron los datos de un equipo.
2. El proyecto se diseñó utilizando diferentes aplicaciones como Code Composer Studio, Halcogen, Multisim, siendo estos útiles para la programación en el procesador utilizado RM46L852 de la familia Texas Instruments ya que es de gama alta, diseñado para aplicaciones industriales, el cual permite una alta precisión en la medición y procesamiento de parámetros eléctricos.
3. Se hizo el uso correcto de los conceptos de Nyquist, Fourier y Radix 4 para diferentes tareas como la medición de voltaje, corriente y frecuencia, así como también el cálculo de potencia.
4. En cuanto a la medición de voltaje, fue necesario diseñar una red de resistencias para su correcto sentido ya que esto no modifica la fase de señal de voltaje y mantienen toda la información de onda requerida.
5. Por otro lado, para la medición de corriente fue necesario utilizar un sensor tipo Hall el cual entrega una señal senoidal con un rango de 0 – 50 A, para que este sensor funcione correctamente se tuvo que hacer pruebas de calibraciones con electrodomésticos caseros para no sufrir incidentes.
6. También se trabajó con Node Red, este es un programador libre, sin licencias que permitió crear los dashboards para que los usuarios puedan visualizar sus consumos en tiempo real mediante sus teléfonos móviles.

7. Por otro lado, para que esto sea posible, se utilizó un software libre que se llama XAMPP el cual permitió crear la base de datos para que los valores medidos sean almacenados en la nube.



RECOMENDACIONES

1. Se recomienda mejorar la protección del equipo ya que para uso común podría ocurrir algún problema con la caja de acrílico, en este caso mejorar dicha protección con NEMA 4X para salpicaduras de agua, viento, lluvia, corrosión, etc.
2. Se recomienda la configuración de forma remota, por si el equipo requiriera de alguna modificación esto se haga a través del mismo protocolo de comunicación ethernet desde una central. Esta configuración requiere de tiempo, y programación más sofisticada.
3. Como se trabaja con datos enviados a la nube, podría existir algún tipo de hackeo por ende se podría utilizar los servidores de Google, Amazon, etc., ya que ellos cuentan con protección antirrobo.
4. Para evitar desconexiones inesperadas de los equipos y así perder mediciones se podría vender o alquilar los servicios a SEAL, y este a su vez instalaría sus propias antenas con su propia red privada y así evitar hackeos o desconexiones.
5. Se recomienda minimizar el tamaño del equipo con montaje SMD, así el equipo se reduciría a la mitad de su tamaño, ocupando menos espacio y siendo más eficiente en cuanto al consumo de energía.
6. Se puede mejorar la visualización de las mediciones en la pantalla LCD utilizando una pantalla más sofisticada.

REFERENCIAS

1. Análisis de los Impactos de la Pandemia del COVID-19 sobre el sector energético de América Latina y el Caribe. Mayo 2020. Obtenido de (OLADE – Organización Latinoamericana de Energía) – Ing. Fabio García, Especialista Asociado, el Ing. Adrián Moreno, Consultor y el Dr. Andrés Schuschny, Director de Estudios, Proyectos e Información de la Organización Latinoamericana de Energía (OLADE) Pagina 11.
2. Decreto de urgencias 2020. Obtenido de Diario Oficial de Bicentenario – EL PERUANO <https://busquedas.elperuano.pe/normaslegales/decreto-de-urgencia-que-establece-medidas-complementarias-pa-decreto-de-urgencia-n-035-2020-1865377-1/>
3. ¿Qué es MQTT? Obtenido de Hugo Vargas (2018). Recuperado el 22 de setiembre de 2021, de <https://www.codigoiot.com/base-de-conocimiento/mqtt/>
4. ¿Qué es Node Red? Obtenido de Oscar Gonzales (2017). Recuperado el 23 de setiembre de 2021, de <https://blog.bricogeek.com/noticias/raspberry-pi/tutorial-de-node-red-con-raspberry-pi-y-esp8266/>
5. Huiman Tocto, Nelson Eduardo (17 de noviembre del 2016). Diseño e Implementación de una red de medidores de energía para artefactos domésticos (Pontificia Universidad Católica del Perú), recuperada de <https://tesis.pucp.edu.pe/repositorio/handle/20.500.12404/11748>
6. Valencia Alejo, Martin (22 de abril del 2016). Diseño e Implementación de un analizador de calidad de energía eléctrica (Universidad Católica de Santa María), recuperada de https://alicia.concytec.gob.pe/vufind/Record/UCSM_366e6bb45a0eb6fd434afc15d27fcfad
7. NORMA IEC 62053-21:2020. (2020). Equipos de medida de electricidad. Requisitos particulares. Parte 21: Contadores estáticos de energía activa CA (clases 0,5, 1 y 2). Recuperada de <https://webstore.iec.ch/publication/28660>
8. NORMA IEC 62052-11:2020. (2020). Equipos de medida de electricidad. Requisitos generales, ensayos y condiciones de ensayo. Parte 11. Recuperada de <https://webstore.iec.ch/publication/28212>

9. Ariganello, E. y Barrientos, E. (2010). Redes Cisco. Guía de estudio para profesionales, pág. 773. Recuperado de https://www.academia.edu/40840723/REDES_CISCO_CCNP_a_Fondo_Gu%C3%ADa_de_estudio_para_profesionales
10. Daniel W. Hart (2001). Electrónica de potencia, Cálculo del RMS, primera edición. Recuperado de https://lc.fie.umich.mx/~jorgeahb/Pagina/materias/PIES/kupdf.com_electronica-de-potencia-1ra-edicion-daniel-w-hart.pdf
11. Fernandez Losa, J. (2019). ¿Qué es potencia activa y reactiva? Recuperado el 5 de octubre de 2021, de <https://instrumentacionhoy.blogspot.com/2019/08/potencia-activa-y-reactiva.html>
12. Quiñones, Oswaldo (2019). Internet de las Cosas (pág. 4-5). Recuperado de https://books.google.com.pe/books?id=vnnEDwAAQBAJ&pg=PT18&dq=internet+de+las+cosas,+oswaldo+qui%C3%B1ones&hl=es&sa=X&ved=2ahUKEwj_k_vmF1Nf0AhXEFbkGHUpBBE8Q6AF6BAGKEAI#v=onepage&q=internet%20de%20las%20cosas%2C%20oswaldo%20qui%C3%B1ones&f=false
13. Murky Robot (2020). ¿Qué es MQTT? Recuperado el 10 de octubre de 2021 de <https://www.murkyrobot.com/guias/comunicacion/mqtt>
14. Candela Sola (2007). Fundamentos de Sistemas Operativos. Protocolos de internet, pág. 280. Recuperado de https://books.google.com.pe/books?id=fRK3lbTrNy4C&pg=PA347&dq=Fundamentos+de+Sistemas+Operativos,+Candela+Sola,+protocolos+de+internet&hl=es&sa=X&ved=2ahUKEwiw_6-i1tf0AhWdJLkGHc6tAXQQ6AF6BAGLEAI#v=onepage&q=Fundamentos%20de%20Sistemas%20Operativos%2C%20Candela%20Sola%2C%20protocolos%20de%20internet&f=false
15. Jose Ortega Candela (2020). ¿Qué es bróker AEDES? Recuperado de de Tecnologías para arquitecturas basadas en microservicios (pág. 200)
16. Taiji Hagino, (2021). ¿Qué es Node Red? Recuperado de Practical Node Red Programming (pág. 8)
17. Pablo Sancho. (2020) Representación de nodos en Node Red. Recuperado de <https://www.techedgegroup.com/es/blog/fundamenos-node-red>

18. Premier Global Data Centers (2021) ¿Qué son los puertos TCP/UCP? Recuperado de <https://www.hostdime.com.pe>
19. Pena Mocho, Javier Pardal (2015) ¿Qué es NodeJs? Recuperado de NodeJS para principiantes (pág. 22)
20. Carrion Ramon, Mercedes Gomez (2019) ¿Qué es XAMPP? Recuperado de Usando XAMPP con Bottstrap y WordPress (pág. 4)
21. Learning About Electronics. (2018) ¿Qué es un filtro pasa bajo? Recuperado de <http://www.learningaboutelectronics.com/Articulos/Filtro-paso-bajo.php>
22. Maniqui (2019) ¿Qué es un comparador de voltaje? Recuperado de <https://maniqui.ru/electrnica-de-consumo/electrnica/componentes-4/12522-componentes-de-la-electrnica-cmo-utilizar-un.html>
23. Jose Antonio Lorca (2017) [VIDEO] ¿Qué es un transformador monofásico? Recuperado de https://www.youtube.com/watch?v=UbalPc_5sdk
24. Vicente Garcia. (2000) ¿Qué es una onda senoidal? Recuperado de <https://www.diarioelectronicohoy.com/blog/leccion-9-conocimientos-basicos>
25. SSDIELECT (2021) Comparador LM311. Recuperado de <https://ssdielect.com/cb/comparadores-de-voltaje/77-lm311.html>
26. Hoja de datos On Semiconductor 74HC14 (2021) ¿Qué es un circuito Schmitt Trigger?
27. Garcia Gonzales, Antony. (2014) Protocolo de comunicación SPI. Recuperado de <http://panamahitek.com/como-funciona-el-protocolo-spi/>
28. Texas Instrument – Halcogen, (2021). Software Halcogen.
29. Texas Instrument – CodeComposer, (2021). Software Code Composer Studio.
30. Texas Instruments, Digital Signal Processing Solutions, (1998) Radix4.
31. Operational Amplifiers & Linear Integrated Circuits, Robert F. Coughlin, Frederick F. (pág. 101) Búsqueda por cero.
32. Fullwat profesional solutions (2018) Factor de potencia en cada carga. Recuperado de <http://blog.fullwat.com/que-es-el-factor-de-potencia/>

ANEXOS

ANEXO A: PROGRAMA DE MENÚ GENERAL

```

/* USER CODE BEGIN (0) */

/* USER CODE END */

/* Include Files */

#include "sys_common.h"

/* USER CODE BEGIN (1) */
#include "sys_core.h"
#include "system.h"
#include "sci.h"
#include "het.h"
#include "gio.h"
#include "spi.h"
#include "rti.h"
#include "stdio.h"
#include "adc.h"
#include "etpwm.h"
#include "DS1302.h" //agregando las cabeceras de las funciones
#include "EE25LC256.h"
#include "LCD_3LINES.h"
#include "KERNEL_MCE.h" /*Kernel de Motor de calculo Electrico*/
#include "uartstdio.h"
#include <string.h>
/* USER CODE END */

/** @fn void main(void)

```

```
* @brief Application main function
* @note This function is empty by default.
*
* This function is called after startup.
* The user can use this function to implement the application.
*/
```

```
/* USER CODE BEGIN (2) */
```

```
RTC_DS1302 Calendario;
```

```
bool LCD_Update = 0;
```

```
char Msg_LCD[16] = "";
```

```
char Msg_LCD1[16] = "";
```

```
char Msg_Frecuencia[16] = "";
```

```
char Msg_Frecuencia1[16] = "";
```

```
char Msg_FP[16] = "";
```

```
char Msg_IP[16] = "";
```

```
char Msg_Serial[50] = ""; // Mensajería que se envía por protocolo serial al ESP32
```

```
hetSIGNAL_t Frec_signal;
```

```
float Frec_Line = 0;
```

```
void wait(uint32 time);
```

```
char g_pcCmdBuf[128] = ""; // línea de datos a recibir
```

```
/* -----
```

```
** Estructura de datos para la comunicación Wifi
```

```
** ----- */
```

```
typedef struct
```

```
{
```

```
    char Wifi_LocalIP[20]; // Vector mejora el resultado del cálculo
```

```

char Wifi_RSSI[4];
char SSID[50];
char mqtt_server[50];
char mqtt_port[50];
char root_topic_publish[50];
char mqtt_user[50];
char mqtt_pass[50];
} Wifi_Data;

Wifi_Data Wifi_Client;

void GetServerConfig();
void Set_LCD(int value);
float valor1, valor2;
float DiffFrec;
/* USER CODE END */

//programa principal
int main(void)
{
    /* USER CODE BEGIN (3) */

    //iNICIALIZANDO PERIFERICOS DEL CONTROLADOR
    gioInit();    //General input out
    hetInit();    // high end timer
    rtcInit();    // reloj en tiempo real
    spiInit();
    adcInit();    //Inicializo las configuraciones del ADC
    etpwmInit();  //Inicializo las configuraciones del PWM
    rtiInit();
    lcd_init();   // inicializamos el LCD
    sciInit(); /* initialize sci/sci-lin */
    /* even parity , 2 stop bits */

```

```

/* Enable RTI Compare 0 interrupt notification */
rtiEnableNotification(rtiNOTIFICATION_COMPARE0);
rtiEnableNotification(rtiNOTIFICATION_COMPARE1);

adcEnableNotification(adcREG1, adcGROUP0); //Habilito las interrupciones por
ADC y PWM Trigger
adcStartConversion(adcREG1, adcGROUP0); //Inicializo la conversión del ADC

_enable_IRQ();
_enable_interrupt_(); //Habilito las interrupciones

/* Start RTI Counter Block 0 */
rtiStartCounter(rtiCOUNTER_BLOCK0);

Pin_t LCD_Pin = { gpioPORTB, 3 };
SetPinDirection(&LCD_Pin, pinWrite);

Set_LCD(1); //enciende la pantalla LCD
/*
while (1)
{
Set_LCD(0);
delay_ms(500); //RETADOR DE 1 SEGUNDO
Set_LCD(1);
delay_ms(500); //RETADOR DE 1 SEGUNDO
}*/

// rtc_set_datetime(31, 10, 21, 0, 10,41); //Comando para la configuración de la hora
en el RTC

//goto Initprogram;
/* -----
** Iniciando Comunicacion Wifi y esperando datos
** ----- */

lcd_gotoxy(1, 1); // envias coordenadas

```

```

lcd_text("Iniciando Wifi");           //envias texto
lcd_gotoxy(1, 2);                     // envias coordenadas
lcd_text("Espere ...");               //envias texto

GetServerConfig();

lcd_clear();       //limpiar pantalla LCD
/* -----
** Primera pantalla
** ----- */

sprintf(Msg_LCD, "SSID: %s", Wifi_Client.SSID);
sprintf(Msg_LCD1, "RSSI: %s dBm", Wifi_Client.Wifi_RSSI);

lcd_gotoxy(1, 1);           // envias coordenadas
lcd_text(Msg_LCD);          //envias texto
lcd_gotoxy(1, 2);           // envias coordenadas
lcd_text(Msg_LCD1);         //envias texto

delay_ms(5000); //RETADOR DE 1 SEGUNDO
lcd_clear();       //limpiar pantalla LCD
/* -----
** SEGUNDA PANTALLA
** VISUALIZANDO IP LOCAL, TOPICO DE PUBLICACION
** ----- */

sprintf(Msg_LCD, "IP:%s", Wifi_Client.Wifi_LocalIP);
sprintf(Msg_LCD1, "Topic:%s", Wifi_Client.root_topic_publish);

lcd_gotoxy(1, 1);           // envias coordenadas
lcd_text(Msg_LCD);          //envias texto
lcd_gotoxy(1, 2);           // envias coordenadas
lcd_text(Msg_LCD1);         //envias texto

```

```

delay_ms(5000); //RETARDO DE 1 SEGUNDO
lcd_clear();    //limpiar pantalla LCD

/* -----
** SE VISUALIZA MQTT SERVIDOR
** ----- */

sprintf(Msg_LCD, "%s", Wifi_Client.mqtt_server);
lcd_gotoxy(1, 1);    // envias coordenadas
lcd_text("MQTT Server:");    //envias texto
lcd_gotoxy(1, 2);    // envias coordenadas
lcd_text(Msg_LCD);    //envias texto

delay_ms(5000); //RETARDO DE 1 SEGUNDO
//Initprogram:

lcd_clear();    //limpiar pantalla LCD
/* -----
** Iniciando el MOr de calculo electrico del microcontrolador
** ----- */
// saltaraqu:
Kernel_Init();

/* -----
** Bucle principal para el envio de datos al transmisor de Wifi
** ESP32
** ----- */
while (1)
{
    if (LCD_Update == 1)
    {
        LCD_Update = 0;
    }
}

```

```
//Kernel_Get_Voltaje_Corriente_Potencia_RMS();
Kernel_Calculo_Electrico();
```

```
/* if (Wave_Info.AMP_RMS <= 2.0)
    Wave_Info.AMP_RMS = 0;*/
```

```
lcd_gotoxy(5, 1);           // envias coordenadas
lcd_text("Medicion");      // envias texto
lcd_gotoxy(5, 2);           // envias coordenadas
lcd_text("Electrica");      // envias texto
delay_ms(1000); //RETARDO DE 1 SEGUNDO
lcd_clear();                // limpiar pantalla LCD
```

```
sprintf(Msg_LCD, "Volt-AC: %3.2f ", Wave_Info.VOLT_RMS_D);
sprintf(Msg_LCD1, "Curr-AC: %2.3f ", Wave_Info.AMP_RMS);
```

```
lcd_gotoxy(1, 1);           // envias coordenadas
lcd_text(Msg_LCD);          // envias texto
lcd_gotoxy(1, 2);           // envias coordenadas
lcd_text(Msg_LCD1);         // envias texto
delay_ms(5000); //RETARDO DE 1 SEGUNDO
lcd_clear();                // limpiar pantalla LCD
```

```
lcd_gotoxy(3, 1);           // envias coordenadas
lcd_text("Medicion de");      // envias texto
lcd_gotoxy(4, 2);           // envias coordenadas
lcd_text("Frecuencia");      // envias texto
delay_ms(1000); //RETARDO DE 1 SEGUNDO
lcd_clear();                // limpiar pantalla LCD
```

```
DiffFrec = (60 - Wave_Info.FREC) / 60 * 100;
```

```

sprintf(Msg_Frecuencia, "Hz: %3.3f ", Wave_Info.FREC);
sprintf(Msg_Frecuencia1, "Diff: %3.3f %% ", DiffFrec);
lcd_gotoxy(1, 1);           // envias coordenadas
lcd_text(Msg_Frecuencia);   // envias texto
lcd_gotoxy(1, 2);           // envias coordenadas
lcd_text(Msg_Frecuencia1);  // envias texto
delay_ms(5000); //RETARDO DE 1 SEGUNDO
lcd_clear();                // limpiar pantalla LCD

lcd_gotoxy(5, 1);           // envias coordenadas
lcd_text("Potencia");      // envias texto
lcd_gotoxy(4, 2);           // envias coordenadas
lcd_text("Electrica");    // envias texto
delay_ms(1000); //RETARDO DE 1 SEGUNDO
lcd_clear();                // limpiar pantalla LCD

if (Wave_Info.POT_KW < 0)
    Wave_Info.POT_KW = (-1) * Wave_Info.POT_KW;

if (Wave_Info.POT_KVA < 0)
    Wave_Info.POT_KVA = (-1) * Wave_Info.POT_KVA;

sprintf(Msg_LCD, "KW: %3.3f ", Wave_Info.POT_KW);
sprintf(Msg_LCD1, "KVA: %3.3f", Wave_Info.POT_KVA);
lcd_gotoxy(1, 1);           // envias coordenadas
lcd_text(Msg_LCD);          // envias texto
lcd_gotoxy(1, 2);           // envias coordenadas
lcd_text(Msg_LCD1);         // envias texto
delay_ms(5000); //RETARDO DE 1 SEGUNDO
lcd_clear();                // limpiar pantalla LCD

lcd_gotoxy(4, 1);           // envias coordenadas
lcd_text("Factor de ");    // envias texto

```

```

lcd_gotoxy(5, 2);          // envias coordenadas
lcd_text("Potencia");      // envias texto
delay_ms(1000); //RETARDO DE 1 SEGUNDO
lcd_clear();              // limpiar pantalla LCD

sprintf(Msg_FP, "PF: %1.4f ", Wave_Info.PF);
lcd_gotoxy(1, 1);          // envias coordenadas
lcd_text(Msg_FP);          // envias texto
delay_ms(5000); //RETARDO DE 1 SEGUNDO
lcd_clear();              // limpiar pantalla LCD

/*    lcd_gotoxy(3, 1);      // envias coordenadas
    lcd_text("Hora y Fecha"); // envias texto
    delay_ms(1000); //RETARDO DE 1 SEGUNDO
    lcd_clear();            // limpiar pantalla LCD

    Calendario = rtc_read();

    lcd_gotoxy(4, 1);        // envias coordenadas
    lcd_text(Calendario.MSG_Fecha); // envias texto
    lcd_gotoxy(4, 2);        // envias coordenadas
    lcd_text(Calendario.MSG_Hora); // envias texto
    delay_ms(5000); //RETARDO DE 1 SEGUNDO*/
/* -----
** SEGUNDA PANTALLA
** VISUALIZANDO IP LOCAL, TOPICO DE PUBLICACION
** ----- */

lcd_clear();              // limpiar pantalla LCD

// sprintf(Msg_IP, "%s", Wifi_Client.Wifi_LocalIP);
// sprintf(Msg_LCD1, "Topic:%s", Wifi_Client.root_topic_publish);

lcd_gotoxy(1, 1);          // envias coordenadas

```

```

    lcd_text("Direccion IP:");           //envias texto
    lcd_gotoxy(1, 2);           // envias coordenadas
    lcd_text(Wifi_Client.Wifi_LocalIP);      //envias texto

    delay_ms(5000); //RETARDO DE 1 SEGUNDO
    lcd_clear();      //limpiar pantalla LCD
    /*
    * Envio de datos a modulo WIFI para transmision de datos a la nube
    */
    sprintf(Msg_Serial, "%3.2f,%3.2f,%3.2f,%3.2f\n",
        Wave_Info.VOLT_RMS_D, Wave_Info.AMP_RMS, Frec_Line,
        Wave_Info.PF);

    UARTprintf(Msg_Serial);

}

}

/* USER CODE END */

return 0;
}

/* USER CODE BEGIN (4) */

void GetServerConfig()
{

    UARTgets(g_pcCmdBuf, sizeof(g_pcCmdBuf));

    char *token = strtok(g_pcCmdBuf, ",");
    memcpy(Wifi_Client.Wifi_LocalIP, token, strlen(token) + 1);

```

```

token = strtok(NULL, ",");
memcpy(Wifi_Client.Wifi_RSSI, token, strlen(token) + 1);
token = strtok(NULL, ",");
memcpy(Wifi_Client.SSID, token, strlen(token) + 1);
token = strtok(NULL, ",");
memcpy(Wifi_Client.mqtt_server, token, strlen(token) + 1);
token = strtok(NULL, ",");
memcpy(Wifi_Client.mqtt_port, token, strlen(token) + 1);
token = strtok(NULL, ",");
memcpy(Wifi_Client.root_topic_publish, token, strlen(token) + 1);
token = strtok(NULL, ",");
memcpy(Wifi_Client.mqtt_user, token, strlen(token) + 1);
token = strtok(NULL, ",");
memcpy(Wifi_Client.mqtt_pass, token, strlen(token) + 1);
}

void rtiNotification(uint32 notification)
{
    /* enter user code between the USER CODE BEGIN and USER CODE END. */
    if (notification == rtiNOTIFICATION_COMPARE0) //Temporizador 250ms
    {
        LCD_Update = 1;
        capGetSignal(hetRAM1, 7, &Frec_signal);
        Frec_Line = 1e6 / Frec_signal.period;
    }
}

{
    if (value)
        gpioSetBit( gpioPORTB, 2, 1);
    else
        gpioSetBit( gpioPORTB, 2, 0);
}

```

```
}

void Set_LCD(int value)
{
    if (value)
        gpioSetBit( gpioPORTB, 3, 1);
    else
        gpioSetBit( gpioPORTB, 3, 0);
}

// INTERRUPCION DEL ADC

void adcNotification(adcBASE_t *adc, uint32 group)
{
    //DECLARAMOS LAS VARIABLES DEL ADC
    adcData_t data[4];

    //CAPTURAMOS LA INFORMACION DEL ADC
    adcGetData(adcREG1, adcGROUP0, data);

    //ALMACENAMOS EL VOLTAJE Y LA CORRIENTE EN FORMA DE ONDA
    // EN LA MEMORIA DEL CONTROLADOR
    //MEMORIA DINAMICA
    // SE ACTUALIZA CADA 40 microsegundos -->periodo de muestreo

    // valor1 = data[2].value;

    Voltaje.wave_sampling_buff[ik] = data[1].value - data[0].value;
    Corriente.wave_sampling_buff[ik] = data[2].value - 1800; //Valores normales sin
carga de corriente

    ik++;
    //LA MEMORIA SE LLENO??
    if (ik > TEST_LENGTH_SAMPLES)
```

```
{
//SI LA MEMORIA SE LLENO//
//MOVEMOS LOS DATOS A LA MEMORIA SAMPLING
// la mmemoria estatica se actualiza
// Tt=TEST_LENGTH_SAMPLES*40us
    adcDisableNotification(adcREG1, adcGROUP0);
    ik = 0;
    arm_copy_f32(Voltaje.wave_sampling_buff, Voltaje.wave_sampling,
    TEST_LENGTH_SAMPLES); //MOVEMOS LA INFORMACION DINAMICA
A UNA MEMORIA ESTATICA
    arm_copy_f32(Corriente.wave_sampling_buff, Corriente.wave_sampling,
    TEST_LENGTH_SAMPLES);
    adcEnableNotification(adcREG1, adcGROUP0);
}
}

void wait(uint32 time)
{
    time--;
}

/* USER CODE END */
```

ANEXO B: PROGRAMA DE MOTOR ELÉCTRICO

```
#ifndef INCLUDE_KERNEL_MCE_H_
#define INCLUDE_KERNEL_MCE_H_

#include "arm_math.h"
#include "type_defs.h"
#include "system.h"
#include "het.h"
#include "adc.h"
// #include "KERNEL_GPU.h" /*Kernel de Unidad de Procesamiento Grafico*/
#include "EE25LC256.h" //Habilitado para la modificacion de los factares
/* -----
** Definicion de tamaño de Muestras y longitud
** ----- */

#define TEST_LENGTH_SAMPLES 2048
#define LENGTH_SAMPLES 8192
#define LENGTH_FFT 1024 //longitud de
vectores para magnitud y fase de Fourier
#define ADC_CHANEL_PER_MODULE 4

#define FFT_MAX_HIGH 298
#define FFT_X_POINT 100
#define FFT_X_HARM 6
#define FFT_y_POINT 410
/* -----
** Macro Definida para el Filtro Digital pasabajo
** h = fir1(150,[100/25e3],'low') Matlab
** ----- */

#define BLOCK_SIZE 32
#define NUM_TAPS 151
```

```

/* -----
** Factores de Conversion
** ----- */

// #define FACTOR_Q17_TO_F32 1e4 // Factor de la
conversion Alterna
#define RESOLT_ADC 3.3/4095 // Resolucion
de converion del ADC
#define CURRNT_CONVERSION 587.5 // Ganancia de los
amplificadores diferenciales
#define OFFSET_CURRENT 2.373
#define FACTOR_RESIS 1/0.645161290

#define VOLT_CONVERSION 1210.2272 // Ganancia de los
amplificadores diferenciales
#define FREC_MAX 500 // Division entre
1000

#define PH_TO_LINE 1/sqrt(3) // Factor de
Conversion entre Voltaje de Linea a Fase

#define CURRNT_RESOLT 1 // Resolucion
de la Corriente en la pinza Amperimetrica
#define OFFSET_TRIGGER 200 //
OFFSET para el ajuste del trigger en pantalla
#define TSAMPLING 40e-6 // Periodo de
Muestreo de la Señal Electrica
#define PSHVALUE 360 // Angulo de
Fase de 360°, una vuelta completa
#define MPI 3.141592653/180 // Factor de
conversion de grados a radianes
#define PI 3.141592653 // Valor Almacena de PI

```

```
#define SEACH_L_ZERO      8                // Numero de
veces en las que se busca el cero en un vector

#define UNIT_K            1/1000          // Division
entre 1000

#define FACTOR_MIN_HOR    0.0166667      // Convierte
de minutos a horas

#define SAMPLING_PERID    400             //
Capura de 10 segundos de frecuencia, para luego realizar cambio.

/* -----
** Variables de Conctrol Fluctuaciones y Memoria SD
** ----- */

#define FACTOR_MULT       10              // Factor que
indica numero de decimales almacenado en la memoria SD 10 = 1 decimal, 100 = 2
decimales

#define FACTOR_DIV        10              // Factor que indica numero de decimales
almacenado en la memoria SD 10 = 1 decimal, 100 = 2 decimales

#define LIMIT_LCD         800             // Limite de la
pantalla LCD

#define FLUCT_SCALE       2               // Escala
limite de cada TREND

/* -----
** Matriz de datos del Motor de Calculo
** ----- */

typedef struct
{
    float32_t VOLT_PROM;                // Voltaje promedio en la muestra de señal
    float32_t VOLT_RMS_D;                // Valor RMS DE 10/12 ciclos de la
señal
    float32_t VOLT_FUND;                // Valor RMS de 10/12 ciclos de la fundamental
```

```
//      float32_t VOLT_RMS_12;                                // Valor RMS de
100/120 ciclos
float32_t VOLT_PK;                                           // Voltaje Pico
float32_t VOLT_CF;                                           // Factor de Cresta
float32_t VOLT_OVER;                                         // Overdeviation
float32_t VOLT_UNDER;                                        // Underdeviation
float32_t AMP_PROM;
float32_t AMP_RMS;
//float32_t AMP_RMS_12;
float32_t AMP_FUND;
float32_t AMP_PK;
float32_t AMP_CF;
float32_t VOLT_PSH;
float32_t AMP_PSH;
float32_t POT_KW;
float32_t POT_KVA;
float32_t POT_KVAR;

float32_t POT_KVA_HARM;

float32_t POT_KW_FUND;
float32_t POT_KVA_FUND;
float32_t POT_KVAR_FUND;

float32_t PF;

float32_t POT_KW_HORA;
float32_t POT_KVA_HORA;
float32_t POT_KVAR_HORA;
float32_t FREC;
} MotorCalculo;

/* -----
```

**** Matriz de formas de Onda**

**** ----- */**

typedef struct

```
{
    float32_t wave_sampling[TEST_LENGTH_SAMPLES];// Vector mejora el resultado
    del calculo
    float32_t wave_sampling_buff[TEST_LENGTH_SAMPLES];
    float32_t valor_MAX, valor_MIN, valor_MEAN;
    float32_t fft_MAG[LENGTH_FFT];
    float32_t fft_PHASE[LENGTH_FFT];
    uint16 Wave_ZeroUP[20]; //, Wave_ZeroDWN[10];
    float32_t THD;
    uint32_t max_marks1, min_marks2;
    float32_t Harmonic_Porcent, Harmonic_Amplitud, Harmonic_Angulo;
    // uint16 XY_plot[800], XY_buff[800];
} Harmonics_data;
```

typedef struct

```
{
    int Value_inf;
    int Value_Sup;
    int Trigger_Sup, Trigger_Inf;
    int Time;
    uint16 Value_diff;
    uint16 LENGHT_DATA;
    uint16 lcd_offset;
    uint16 lcd_cst;
    uint16 lcd_ampl;
    uint16 Value_trigger;
    uint16 Sencibilidad;
    uint8 r, g, b;
} Trigger_funcion;
```

```

/* -----
** Variables Dependientes de Calculos anteriores
** ----- */

double T_prom;                // La informacion se carga desde el programa
principal
float FREC_LINEA;
hetSIGNAL_t Duty_Period1;
/* -----
* Variables del ADC
* ----- */
uint32 ik;
Harmonics_data Voltaje, Corriente, Potencia;
MotorCalculo Wave_Info;
Trigger_funcion Wave_signal;
/* -----
** Proformas de funciones
** ----- */

void Kernel_Calculo_Electrico();
uint8 Kernel_Cruce_Cero_Voltaje();
float Kernel_Get_Frecuencia();
float Kernel_Get_Voltaje_RMS();

void Kernel_Get_Voltaje_Corriente_Potencia_RMS();
void Kernel_Calculo_DC();
void Kernel_Calculo_Coeficientes();
void Write_Kernel_Factor(float VALUE_CHA, float VALUE_CHB, float
VALUE_CHC,
                        float VALUE_CHN, bool FACTORY_RST);
void Kernel_Init();
void Kernel_FFT_Init();
void Kernel_FFT_Calculo();

```

```
void InitSamplingADC();  
void StopSamplingADC();  
void Kernel_WaveProm();  
uint16 Seaching_Trigger(float32_t *pSrc, Trigger_funcion *Signal_Analizer);  
uint8 Zero_Seach(float32_t *pSrc, uint16 *pDstUP,  
    Trigger_funcion *Signal_Analizer);  
void Seach_Harmonic(float32_t *pSrc, float32_t *pDst, uint32 Index);  
void arm_cmplx_phase_f32(float32_t *pSrc, float32_t *pDst, uint32_t numSamples);  
void arm_atan_f32(float32_t iny, float32_t inx, float32_t *pOut);  
void Kernel_Calculo_Phazor();  
void fft_mag_phase(Harmonics_data *pSrc);  
#endif /* INCLUDE_KERNEL_MCE_H_ */
```

ANEXO C: PROGRAMA DE RELOJ EN TIEMPO REAL

```
#include "het.h"
#include "gio.h"
#include "DS1302.h"
#include "stdio.h"

/* -----
* GIOA_6 --- CLOCK
* NHET1_4 --- I/O
* NHET1_9 --- RST
* ----- */

#define RTC_SCLK 6
#define RTC_IO 4
#define RTC_RST 9

RTC_DS1302 rtc_read_date()
{
    RTC_DS1302 Reloj;
    Reloj.day = bcdToDec(read_ds1302(0x87));
    Reloj.mth = bcdToDec(read_ds1302(0x89));
    Reloj.year = 2000 + bcdToDec(read_ds1302(0x8d));
    printf(Reloj.MSG_Fecha, "%02u/%02u/%04u ", Reloj.day, Reloj.mth,
        Reloj.year);
    return Reloj;
}

RTC_DS1302 rtc_read_time()
{
    RTC_DS1302 Reloj;
    Reloj.hr = bcdToDec(read_ds1302(0x85));
    Reloj.min = bcdToDec(read_ds1302(0x83));
    Reloj.sec = bcdToDec(read_ds1302(0x81));
}
```

```

sprintf(Reloj.MSG_Hora, "%02u:%02u:%02u\t", Reloj.hr, Reloj.min, Reloj.sec);
return Reloj;
}

RTC_DS1302 rtc_read()
{
    RTC_DS1302 Reloj;
    Reloj.day = bcdToDec(read_ds1302(0x87));
    Reloj.mth = bcdToDec(read_ds1302(0x89));
    Reloj.year = 2000 + bcdToDec(read_ds1302(0x8d));
    Reloj.hr = bcdToDec(read_ds1302(0x85));
    Reloj.min = bcdToDec(read_ds1302(0x83));
    Reloj.sec = bcdToDec(read_ds1302(0x81));
    sprintf(Reloj.MSG_Fecha, "%02u/%02u/%04u", Reloj.day, Reloj.mth,
        Reloj.year);
    sprintf(Reloj.MSG_Hora, "%02u:%02u:%02u", Reloj.hr, Reloj.min, Reloj.sec);
    return Reloj;
}

void write_ds1302_uint8_t(uint8_t cmd)
{
    uint8_t i;
    Pin_t rtc_io = { hetPORT1, RTC_IO };
    SetPinDirection(&rtc_io, pinWrite);
    // This assumes that shiftOut is 'slow' enough for the DS1302 to read the
    // bits. The datasheet specifies that SCLK must be in its high and low states
    // for at least 0.25us at 5V or 1us at 2V. Experimentally, a 16MHz Arduino
    // seems to spend ~4us high and ~12us low when shifting.
    for (i = 0; i <= 7; ++i)
    {
        // output_bit(RTC_IO, shift_right(&cmd, 1, 0));

        output_bit(RTC_IO, (((cmd >> i) & 0x1) != 0));
    }
}

```

```
        output_high(RTC_SCLK);
        delay_us(5);
        output_low(RTC_SCLK);
        delay_us(5);
    }
}

void write_ds1302(uint8_t cmd, uint8_t data)
{
    output_high(RTC_RST);
    write_ds1302_uint8_t(cmd);
    write_ds1302_uint8_t(data);
    output_low(RTC_RST);
}

uint8_t read_ds1302(uint8_t cmd)
{
    uint8_t input_value = 0, i;
    uint8_t bit = 0;

    output_high(RTC_RST);
    write_ds1302_uint8_t(cmd);

    Pin_t rtc_io = { hetPORT1, RTC_IO };
    SetPinDirection(&rtc_io, pinRead);

    for (i = 0; i <= 7; ++i)
    {

        //shift_right(&data, 1, input(RTC_IO));
        bit = gioGetBit(hetPORT1, RTC_IO);
    }
}
```

```

input_value |= (bit << i); // Bits are read LSB first.

// See the note in writeOut() about timing. digitalWrite() is slow enough to
// not require extra delays for tCH and tCL.

output_high(RTC_SCLK);
delay_us(20);
output_low(RTC_SCLK);
delay_us(20);
}
output_low(RTC_RST);

return (input_value);
}

void rtcInit()
{
    Pin_t rtc_clk = { gioPORTA, RTC_SCLK };
    Pin_t rtc_rst = { hetPORT1, RTC_RST };

    SetPinDirection(&rtc_clk, pinWrite);
    SetPinDirection(&rtc_rst, pinWrite);

    uint8_t x;
    output_low(RTC_RST);
    delay_us(20);
    output_low(RTC_SCLK);
    write_ds1302(0x8e, 0);
    write_ds1302(0x90, 0xa4);
    x = read_ds1302(0x81);
    if ((x & 0x80) != 0)
        write_ds1302(0x80, 0);
}

```

```
void rtc_set_datetime(uint8_t day, uint8_t mth, uint8_t year, uint8_t dow,
                    uint8_t hr, uint8_t min)
{

    write_ds1302(0x86, decToBcd(day));
    write_ds1302(0x88, decToBcd(mth));
    write_ds1302(0x8c, decToBcd(year));
    write_ds1302(0x8a, decToBcd(dow));

    write_ds1302(0x84, decToBcd(hr));
    write_ds1302(0x82, decToBcd(min));
    write_ds1302(0x80, decToBcd(0));
}

// Returns the decoded decimal value from a binary-coded decimal (BCD) byte.
// Assumes 'bcd' is coded with 4-bits per digit, with the tens place digit in
// the upper 4 MSBs.
uint8_t bcdToDec(const uint8_t bcd)
{
    return (10 * ((bcd & 0xF0) >> 4) + (bcd & 0x0F));
}

// Returns the binary-coded decimal of 'dec'. Inverse of bcdToDec.
uint8_t decToBcd(const uint8_t dec)
{
    const uint8_t tens = dec / 10;
    const uint8_t ones = dec % 10;
    return (tens << 4) | ones;
}

void output_bit(int rtc_pin, int value)
{
    gpioSetBit(hetPORT1, rtc_pin, value);
}
```

```
}

void output_high(int rtc_pin)
{
    if (rtc_pin == 6)
        gpioSetBit(gioPORTA, rtc_pin, 1);
    else
        gpioSetBit(hetPORT1, rtc_pin, 1);
}

void output_low(int rtc_pin)
{
    if (rtc_pin == 6)
        gpioSetBit(gioPORTA, rtc_pin, 0);
    else
        gpioSetBit(hetPORT1, rtc_pin, 0);
}

void SetPinDirection(const Pin_t *pin, t_pinDir dir)
{
    uint32 uDir = pin->port->DIR;
    if (dir == pinWrite)
    {
        uDir |= (1U << pin->bit);
    }
    else
    {
        uDir &= ~(1U << pin->bit);
    }
    gpioSetDirection(pin->port, uDir);
}

void delay_us(int Microseconds_time)
```

```
{
    float delauseconds = (float) Microseconds_time * 1e-6;
    delay_seg(delauseconds);
}

void delay_ms(int Microseconds_time)
{
    float delauseconds = (float) Microseconds_time * 1e-3;
    delay_seg(delauseconds);
}

void delay_seg(float delay)
{
    int long K_temp = (int long) (1.36e7 * delay);
    //    delay_debug=K_temp;
    int long ii;
    for (ii = 0; ii <= 2 * K_temp; ii++) //10000 algoritmos que se desarrollan en 734us
    {
    }
}
```

ANEXO D: PROGRAMA DE MÓDULO WIFI ESP32

```
#include <Arduino.h>

#include <WiFi.h>

#include <PubSubClient.h>

//*****

//**** MQTT CONFIG ****

//*****

const char *mqtt_server = "192.168.0.106"; //Servidor MQTT del Home Assistant
const int mqtt_port = 1883;           // puerto de enlace para las comunicaciones
const char *mqtt_user = "admin";
const char *mqtt_pass = "admin";
const char *root_topic_subscribe = "sub"; //donde tu lees
const char *root_topic_publish = "pub";  //donde tu escribes en el Home Assistant

//*****

//**** WIFICONFIG ****

//*****

const char* ssid = "Tesis";           // Clave y contraseña del Wifi
const char* password = "vanessa*10";
```

```
//*****

//**** GLOBALES ****

//*****

WiFiClient espClient;

PubSubClient client(espClient);

char msg[25];

float count=0;

//*****

/** FUNCIONES **

//*****

void callback(char* topic, byte* payload, unsigned int length);

void reconnect();

void setup_wifi();

float Voltaje, Corriente,Frecuencia;

void setup() {

    Serial.begin(115200);

    setup_wifi();

    client.setServer(mqtt_server, mqtt_port);
```

```

client.setCallback(callback);

}

void loop() {

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

    client.loop();

    if (client.connected()){
        // String str = "La cuenta es -> " + String(count);

        Voltaje+=count;
        Corriente+=count;
        Frecuencia+=count;

        String str = String(Voltaje)+"," +String(Corriente)+"," +String(Frecuencia);

        str.toCharArray(msg,25);

        client.publish("EDM/HOLA", msg);

        count=count+0.1;
    }
}

```

```

delay(1000);

}

}

//*****
//*  CONEXION WIFI  *
//*****

void setup_wifi(){
    delay(10);

    // Nos conectamos a nuestra red Wifi

    Serial.println();

    Serial.print("Conectando a ssid: ");
    Serial.println(ssid);

    WiFi.begin(ssid, password);

    while (WiFi.status() != WL_CONNECTED) {

        delay(500);

        Serial.print(".");

    }

```

```

Serial.println("");

Serial.println("Conectado a red WiFi!");

Serial.println("Dirección IP: ");

Serial.println(WiFi.localIP());
}

//*****
/*  CONEXION MQTT  *
//*****

void reconnect() {

while (!client.connected()) {

    Serial.print("Intentando conexión Mqtt...");

    // Creamos un cliente ID

    String clientId = "ESP32_VANE";

    clientId += String(random(0xffff), HEX);

    // Intentamos conectar

    if (client.connect(clientId.c_str(),mqtt_user,mqtt_pass)) {

        Serial.println("Conectado!");

        // Nos suscribimos

```

```

if(client.subscribe(root_topic_subscribe)){

    Serial.println("Suscripcion ok");

}else{

    Serial.println("fallo Suscripciión");

}

} else {

    Serial.print("falló :( con error -> ");

    Serial.print(client.state());

    Serial.println(" Intentamos de nuevo en 5 segundos");

    delay(5000);

}

}

}

}

//*****

//*   CALLBACK   *

//*****

void callback(char* topic, byte* payload, unsigned int length){

    String incoming = "";

    Serial.print("Mensaje recibido desde -> ");

    Serial.print(topic);

    Serial.println("");
    
```

```
for (int i = 0; i < length; i++) {  
  
    incoming += (char)payload[i];  
  
}  
  
incoming.trim();  
  
Serial.println("Mensaje -> " + incoming);  
  
}
```



ANEXO E: PROGRAMA DE MOTOR DE CÁLCULO EN NODE RED

```

var output=msg.payload.split(",");

var Voltaje=parseFloat(output[0],2);
var Corriente=parseFloat(output[1],2)-0.15;
var Frecuencia=parseFloat(output[2],2);
var FactorPotencia=parseFloat(output[3],2)*(-1);
var PotenciaAparente=(Voltaje * Corriente)/1000;
var Angulo=Math.acos(FactorPotencia)*180/3.141592;
if (Angulo>90){
    Angulo=Angulo-90;
    Angulo=Angulo*(-1);
}

var PotenciaElectrica= PotenciaAparente * FactorPotencia;
var PotenciaReactiva= PotenciaAparente*Math.sin(Angulo);

var msg0={payload:Voltaje};
var msg1={payload:Corriente.toFixed(2)};
var msg2={payload:Frecuencia};
var msg3={payload:PotenciaElectrica.toFixed(4)};
var msg4={payload:PotenciaAparente.toFixed(4)};
    
```

```
var msg5={payload:FactorPotencia.toFixed(4)};
```

```
var msg6={payload:Angulo.toFixed(4)};
```

```
var msg7={payload:PotenciaReactiva.toFixed(4)};
```

```
return [msg0,msg1,msg2,msg3,msg4,msg5,msg6,msg7]
```



ANEXO F: PROGRAMA DE BASE DE DATOS EN NODE RED

```

var output=msg.payload.split(",");

/**Variables recibidas del equipo transmisor*/

var Voltaje=parseFloat(output[0],2);

var Corriente=parseFloat(output[1],2)-0.15;

var Frecuencia=parseFloat(output[2],2);

var Energia_KWH=parseFloat(output[3],2);


/**Variables calculadas*/

var Potencia_Electrica=(Voltaje * Corriente)/1000; //KW


var msg0={payload:Voltaje};

var msg1={payload:Corriente.toFixed(2)};

var msg2={payload:Frecuencia};

var msg3={payload:Potencia_Electrica.toFixed(2)};

var msg4={payload:Energia_KWH.toFixed(2)};


return [msg0,msg1,msg2,msg3,msg4]

```

**ANEXO G: NORMA IEC 62053-21:2020 (Equipos de medida de electricidad.
Requisitos particulares. Parte 21: Contadores estáticos de energía activa CA
(clases 0,5, 1 y 2)**


Webstore
 International Electrotechnical Commission

[HOME](#)
[SIGN IN](#)
[HELP](#)
[CART](#)

IEC 62053-21:2020

Electricity metering equipment - Particular requirements - Part 21: Static meters for AC active energy (classes 0,5, 1 and 2)

TC 13 | Additional information

Abstract

PREVIEW

IEC 62053-21:2020 applies only to static watt-hour meters of accuracy classes 0,5, 1 and 2 for the measurement of alternating current electrical active energy in 50 Hz or 60 Hz networks and it applies to their type tests only.

This document applies to electricity metering equipment designed to:

- measure and control electrical energy on electrical networks (mains) with voltage up to 1 000 V AC;
- have all functional elements, including add-on modules, enclosed in, or forming a single meter case with exception of indicating displays;

Show more »

Additional information

Details

History

Publication type	International Standard
Publication date	2020-06-17
Edition	2.0
Available language(s)	English/French
TC/SC	TC 13 - Electrical energy measurement and control
ICS	17.220.20 - Measurement of electrical and magnetic quantities
Stability date	2025
Pages	33
File size	1517 KB

**ANEXO H: NORMA IEC 62052-11:2020 (Equipos de medida de electricidad.
Requisitos generales, ensayos y condiciones de ensayo. Parte 11)**



IEC 62052-11:2020

Electricity metering equipment - General requirements, tests and test conditions - Part 11: Metering equipment

TC 13 | Additional information

Abstract

PREVIEW

IEC 62052-11:2020 specifies requirements and associated tests, with their appropriate conditions for type testing of AC and DC electricity meters. This document details functional, mechanical, electrical and marking requirements, test methods, and test conditions, including immunity to external influences covering electromagnetic and climatic environments.

This document applies to electricity metering equipment designed to:

- measure and control electrical energy on electrical networks (mains) with voltage up to 1 000 V AC, or 1 500 V DC;

[Show more »](#)

Additional information

Details

History

Publication type	International Standard
Publication date	2020-06-29
Edition	2.0
Available language(s)	English, English/French
TC/SC	TC 13 - Electrical energy measurement and control 
ICS	17.220.20 - Measurement of electrical and magnetic quantities
Stability date 	2025
Pages	240
File size	6325 KB

ANEXO I: DISEÑO DE CIRCUITO Y PLACA IMPRESA

