

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL



Facultad de Ingeniería en Electricidad y Computación

**IMPLEMENTACIÓN DE UN ANALIZADOR DE REDES TRIFÁSICO
USANDO ARDUINO**

PROYECTO INTEGRADOR

Previo la obtención del Título de:

Ingeniero en Electricidad

Presentado por:

Alejandro José Briones Romero

Stalin Rafael Orrala Mejia

GUAYAQUIL - ECUADOR

Año: 2022

DEDICATORIA

El presente proyecto lo dedico a mi madre y mi padre que me han apoyado económica y emocionalmente durante el transcurso de mi vida estudiantil, también quiero dedicarle este proyecto a mi novia que siempre ha estado ahí para apoyarme en las buenas y en las malas, y por último a mi abuela y a mi tía que me recibieron con las puertas abiertas en su residencia para que pueda continuar con mis estudios.

AGRADECIMIENTOS

Nuestros más sinceros agradecimientos a todos los profesores de ESPOL que han compartido sus conocimientos y experiencias con nosotros para que podamos ser unos excelentes ingenieros, también agradecer a nuestros amigos politécnicos que han estado presente durante toda la carrera de ingeniería eléctrica.

DECLARACIÓN EXPRESA

“Los derechos de titularidad y explotación, me(nos) corresponde conforme al reglamento de propiedad intelectual de la institución; *Alejandro José Briones Romero* y *Stalin Rafael Orrala Mejía* doy(damos) mi(nuestro) consentimiento para que la ESPOl realice la comunicación pública de la obra por cualquier medio con el fin de promover la consulta, difusión y uso público de la producción intelectual”

Alejandro Jose Briones
Romero

Stalin Rafael Orrala
Mejía

EVALUADORES

Ing. Ivan David Endara Velez

PROFESOR DE LA MATERIA

Ing. Johnny W. Rengifo Santana

PROFESOR TUTOR

RESUMEN

En la industria de la electricidad el analizador de redes es un equipo fundamental y básico para poder realizar estudios energeticos sobre cualquier tipo de instalación; lamentablemente estos equipos son muy costosos y pocas son las industrias o empresas que pueden contar con ellos. De aquí nace la idea del proyecto que es implementar un analizador de redes trifásico con arduino, de bajo presupuesto, con la finalidad de brindar una alternativa mas economica, de fácil obtencion y que cumpla con las funciones basicas de uno de alto costo.

Para la implementación del analizador se usaron materiales de facil obtención, como lo fueron sensores de voltaje y corriente, que sirvieron para la fase de adquisicion de las señales, el Arduino MEGA 2560 en conjunto con el modulo ethernet shield W5500 que desempeñaron la fase de procesamiento de datos y comunicación con el ordenador para la correspondiente visualizacion de la interfaz, ademas de esto se siguieron las normas de la IEC e IEEE para las formulas y rangos de las variables eléctricas que participaron.

Los resultados que se obtuvieron gracias a las diferentes pruebas que se hicieron con el prototipo trabajando a la par con un analizador de alta gama (el cual sirvió como patrón de medida para la respectiva comparación) fueron muy positivos, ya que los valores de voltajes, corrientes, potencias, factor de potencia tuvieron una gran similitud, pues estos presentaron bajos error porcentuales.

Se logro obtener un analizador de redes trifásico con arduino económico capaz de cumplir las funciones basicas de cualquier analizador de alto valor monetario.

Palabras claves: Analizador, Arduino, Normativas, Económico.

ABSTRACT

In the electricity industry, the network analyzer is a fundamental and basic equipment to be able to carry out energy studies on any type of installation; Unfortunately, this equipment is very expensive and there are few industries or companies that can count on it. From here comes the idea of the project that is to implement a three-phase network analyzer with arduino, low budget, in order to provide a cheaper alternative, easily obtained and that meets the basic functions of a high-cost one.

For the implementation of the analyzer, materials of easy acquisition were used, such as voltage and current sensors that served for the signal acquisition phase, the Arduino MEGA 2560 in conjunction with the ethernet shield W5500 module that performed the data processing phase. and communication with the computer for the corresponding visualization of the interface, in addition to this, the IEC and IEEE standards were followed for the formulas and ranges of the electrical variables that participated.

The results that were obtained thanks to the different tests that were done with the prototype working alongside a high-end analyzer (which served as a measurement standard for the respective comparison) were very positive, since the voltage values, currents, powers, power factor had a great similarity, since they presented low percentage errors.

It was possible to obtain a three-phase network analyzer with an economical Arduino capable of fulfilling the basic functions of any analyzer with a high monetary value.

Keywords: Analyzer, Arduino, Regulations, Economic.

ÍNDICE GENERAL

| | |
|---|------|
| EVALUADORES..... | 5 |
| RESUMEN..... | I |
| <i>ABSTRACT</i> | II |
| ÍNDICE GENERAL..... | III |
| ABREVIATURAS | VI |
| SIMBOLOGÍA | VII |
| ÍNDICE DE FIGURAS..... | VIII |
| ÍNDICE DE TABLAS | X |
| CAPÍTULO 1 | 11 |
| 1.Introducción | 11 |
| 1.1 Descripción del problema | 12 |
| 1.2 Justificación del problema..... | 12 |
| 1.3 Objetivos..... | 12 |
| 1.3.1 Objetivo General | 12 |
| 1.3.2 Objetivos Específicos | 12 |
| 1.4 Marco teórico | 13 |
| 1.4.1 Arduino..... | 13 |
| 1.4.2 Arduino UNO..... | 13 |
| 1.4.3 Arduino Mega 2560 | 14 |
| 1.4.4 Arduino Mega ADK..... | 15 |
| 1.4.5 Arduino Ethernet | 15 |
| 1.4.6 Arduino IDE | 17 |
| 1.4.7 Transformada rápida de Fourier (FFT)..... | 17 |
| 1.4.8 Calidad de energía..... | 18 |
| 1.4.9 Variaciones de larga duración..... | 18 |

| | | |
|--------------------------------|---|----|
| 1.4.10 | Desbalance de tensión..... | 19 |
| 1.4.11 | Estándares de calidad de energía..... | 19 |
| 1.4.12 | Medición de calidad de energía | 19 |
| CAPÍTULO 2..... | | 21 |
| 2. Metodología | | 21 |
| 2.1 | Fase de Adquisición | 22 |
| 2.1.1 | Medición de voltaje..... | 22 |
| 2.1.2 | Sensor ZMPT101B..... | 22 |
| 2.1.3 | Sensor ACS712..... | 23 |
| 2.2 | Fase de Acoplamiento | 23 |
| 2.3 | Fase de Comunicación | 23 |
| 2.4 | Fase de procesamiento de datos..... | 24 |
| 2.5 | Fase de visualización..... | 24 |
| 2.6 | Proceso de la implementación del analizador de redes con Arduino..... | 24 |
| 2.7 | Normativas..... | 25 |
| 2.7.1 | Norma IEC 61000-4-30 | 25 |
| 2.7.2 | Norma IEC 61000-4-11 | 25 |
| 2.7.3 | Norma IEC 61000-4-7 | 26 |
| 2.7.4 | Norma IEEE 1459 | 26 |
| 2.8 | Captura de corrientes de arranques. | 27 |
| 2.9 | Procesamiento de datos. | 28 |
| 2.10 | Diseño Interfaz Gráfica. | 29 |
| CAPÍTULO 3..... | | 34 |
| 3. Resultados y Análisis | | 34 |
| 3.1 | Instrumento patrón..... | 34 |
| 3.2 | Medida de error | 36 |

| | | |
|---|--|----|
| 3.3 | Medida de Voltajes, Corrientes y Potencias | 36 |
| 3.4 | Contrastación de voltaje de línea..... | 37 |
| 3.5 | Contrastación de corriente de línea. | 39 |
| 3.6 | Contrastación de potencias por fase..... | 39 |
| 3.7 | Contrastación de potencias totales y factor de potencia..... | 39 |
| 3.8 | Corriente de arranque de un motor..... | 43 |
| 3.9 | Corriente de arranque de un transformador..... | 45 |
| 3.10 | Análisis de Costos. | 47 |
| 3.11 | Limitaciones de hardware encontradas. | 47 |
| CAPÍTULO 4 | | 48 |
| 4. Conclusiones y Recomendaciones | | 48 |
| BIBLIOGRAFÍA..... | | 49 |
| APÉNDICES | | 50 |

ABREVIATURAS

ESPOL Escuela Superior Politécnica del Litoral

SRAM Static Random Access Memory

EEPROM Electrically Erasable Programmable Read-Only Memory

IDE Integrated Development Environment

ANSI American National Standards Institute

IEC International Electrotechnical Commission

IEEE Institute of Electrical and Electronics Engineers

DFT Discrete Fourier Transform

FFT Fourier Fast Transform

RMS Root Mean Square

TCP/IP Transmission Control Protocol/Internet Protocol

ADC Analog-to-Digital Converter

CSV Comma Separated Values

SIMBOLOGÍA

| | |
|-------------|---------------------------------------|
| $V_{a,b,c}$ | Voltaje en la Fase a, b o c |
| $I_{a,b,c}$ | Corriente en la Fase a, b o c |
| $P_{a,b,c}$ | Potencia Activa en la Fase a, b o c |
| $Q_{a,b,c}$ | Potencia Reactiva en la Fase a, b o c |
| $S_{a,b,c}$ | Potencia Aparente en la Fase a, b o c |
| P_t | Potencia Activa Total |
| Q_t | Potencia Reactiva Total |
| S_t | Potencia Aparente Total |
| F_p | Factor de Potencia |
| V | Voltio |
| A | Amperio |
| W | Vatio |
| var | Voltio Amperio Reactivo |
| VA | Voltio Amperio |

ÍNDICE DE FIGURAS

| | |
|---|----|
| Figura 1: Placa del Arduino Uno | 14 |
| Figura 2: Placa del Arduino Mega 2560 | 15 |
| Figura 3: Placa del Arduino Mega ADK | 15 |
| Figura 4: Placa del Arduino Ethernet Shield | 16 |
| Figura 5: Entorno del software Arduino IDE | 17 |
| Figura 6: Comparación entre las operaciones de una DFT y una FFT | 18 |
| Figura 7: Esquema general de la medición de la calidad de la energía | 20 |
| Figura 8: Diagrama de bloques del sistema..... | 21 |
| Figura 9: Sensor ZMPT101B | 22 |
| Figura 10: Sensor ACS712 | 23 |
| Figura 11: Arduino Mega 2560 con Shield Ethernet | 24 |
| Figura 12: Proceso de ejecución del analizador de redes con Arduino. | 25 |
| Figura 13: Característica de la captura de la corriente de arranque. | 28 |
| Figura 14 Algoritmo del software desarrollado para el analizador de redes. | 29 |
| Figura 15: Pestaña de configuración del analizador de redes. | 30 |
| Figura 16: Pestaña General del analizador de redes..... | 31 |
| Figura 17: Pestañas de Voltaje y Corriente del Analizador de Redes..... | 31 |
| Figura 18: Diagramas Fasoriales de voltaje y corriente. | 32 |
| Figura 19: Pestaña de Potencia del analizador de redes..... | 32 |
| Figura 20: Pestaña de análisis de corriente de arranque..... | 33 |
| Figura 21: Registrador de energía FLUKE 1732..... | 36 |
| Figura 22: Prueba de registro de datos..... | 37 |
| Figura 23: Gráfica de tendencia y distribución de error del voltaje | 38 |
| Figura 24: Grafica de tendencia y distribución de error de la corriente..... | 40 |
| Figura 25 Grafica de tendencia y distribución de la potencia..... | 41 |
| Figura 26: Grafica de tendencia y distribución de Pt, St y Fpt. | 42 |
| Figura 27: Equipos usados para la medición de corriente de arranque. | 43 |
| Figura 28: Corriente de arranque de la Fase A medida desde el prototipo..... | 44 |
| Figura 29: Corriente de arranque de la Fase A medida desde el Fluke 43B. | 44 |

| | |
|--|----|
| Figura 30: Corriente RMS de cada $\frac{1}{2}$ ciclo de las corrientes medida desde el prototipo. | 44 |
| Figura 31: Equipos usados para la medición de la corriente de arranque del transformador..... | 45 |
| Figura 32: Corriente de arranque del transformador medida desde el prototipo..... | 46 |
| Figura 33: Corriente de arranque del transformador medida desde el Fluke 43B. | 46 |
| Figura 34: Armónicos presentes en la corriente de arranque medida desde el prototipo. | 46 |

ÍNDICE DE TABLAS

| | |
|--|----|
| Tabla 1 Comparación de características básicas de las placas Arduino. | 16 |
| Tabla 2 Magnitudes para sistemas trifásicos con armónicos | 27 |
| Tabla 3: Rangos y precisión del FLUKE 1732 en condiciones de referencia..... | 34 |
| Tabla 4: Precisión de potencias del FLUKE 1732 en condiciones de referencia | 35 |
| Tabla 5: Datos de placa del motor para prueba de corriente de arranque..... | 43 |
| Tabla 6: Datos de placa del transformador para prueba de corriente de arranque.... | 45 |

CAPÍTULO 1

1. INTRODUCCIÓN

La energía eléctrica es una de las más utilizada en las industrias, establecimientos comerciales y en los hogares, por lo que la calidad de la energía eléctrica es un factor muy importante que siempre debe considerarse, ya que debido al constante desarrollo de los países, se ha producido un incremento de uso de esta, por lo cual resulta de suma importancia hacer un seguimiento de la calidad de energía y servicio con la finalidad de evitar gastos innecesarios, incrementar la vida útil de aparatos eléctricos y cumplir con los estándares internacionales y nacionales.

El analizador de redes es una herramienta indispensable que permite conocer los indicadores de la calidad de energía y servicio de una red eléctrica. Mediante un analizador de redes se puede medir y monitorear de los indicadores de calidad energética tales como: Frecuencia de red, Variaciones de Voltaje de alimentación, Cambios rápidos de Voltaje, Caídas y Subidas de Voltaje, Interrupciones de Voltaje, Desbalance de Voltajes y Componentes Armónicas de Voltaje y Corriente [1]. Existen muchas razones para la contaminación de los sistemas de energía eléctrica de corriente alterna (AC), incluidos las causas naturales como son los relámpagos, descargas disruptivas, fallas de equipos y de líneas, y los forzados como son las distorsiones de voltajes. Gran parte de los equipos que se usan hoy en día es susceptible a sufrir de algún daño o interrupciones de servicio durante eventos de baja calidad de energía eléctrica; un ejemplo típico es un apagado o reinicio de la computadora, con el resultado de una pérdida de avance en el trabajo realizado en la computadora. A menudo, esto se debe a una mala calidad de energía eléctrica en el suministro de la energía [2]. Debido a estos problemas, la calidad de energía se ha convertido en un área de estudio importante en la ingeniería eléctrica, ha creado un gran desafío tanto para las empresas eléctricas como para las industrias y fabricas que dependen de una buena calidad de energía eléctrica para optimizar sus sistemas eléctricos, es por eso que un analizador de redes eléctricas es indispensable, ya que puede detectar tempranamente los problemas de calidad de energía eléctrica de una forma segura, rápida y detallada [1].

1.1 Descripción del problema

Dada la importancia de la calidad de energía eléctrica y de servicio en los sectores residenciales, comerciales e industriales, es indispensable tener analizadores de redes trifásicos que puedan medir y monitorear estos indicadores de calidad y de servicio, de tal manera que se puedan cumplir con las normas nacionales e internacionales. Pero la mayoría de estos analizadores de redes que se encuentran disponibles en el mercado ecuatoriano son muy caros debido muchos factores como la importación y los costos de los aranceles, por lo que es necesario contar con una opción más económica que permita obtener un analizador de redes trifásico que cumpla con las funcionalidades básicas de medición y monitoreo de la calidad de energía y servicio eléctrico.

1.2 Justificación del problema

En el mercado ecuatoriano los analizadores de red de trifásicos no son baratos por lo que son pocas empresas o personas naturales las que pueden adquirirlo. Por otro lado, Ecuador no es un país que desarrolla tecnología, por lo que es importante para el país empezar a desarrollar e importar su tecnología de manera que pueda crecer en el mercado tecnológico mundial. En conclusión, el desarrollo de un prototipo de analizador de redes trifásico usando componentes de bajo costo proporcionará una herramienta importante para el análisis de la calidad de energía en sistemas eléctricos y dará un aporte tecnológico al Ecuador.

1.3 Objetivos

1.3.1 Objetivo General

- Implementar un analizador de redes trifásico utilizando Arduino y componentes de bajo costo para estudios de calidad energía eléctrica y de análisis de arranque de máquinas eléctricas.

1.3.2 Objetivos Específicos

- Desarrollar un algoritmo de cálculo, visualización y registro de parámetros eléctricos usados para la evaluación de los principales indicadores calidad de energía en función de limitaciones técnicas y de tiempo del proyecto.

- Realizar el análisis técnico y económico del analizador de redes trifásico implementado con Arduino para determinar las limitaciones técnicas y la factibilidad de la implementación.
- Contrastar el analizador de redes trifásico implementado con un analizador de redes trifásico patrón calibrado para verificar el porcentaje de error de mediciones.

1.4 Marco teórico

1.4.1 Arduino

Placa hardware libre o PCB (“printed circuit board” o placa de circuito impreso) que posee un microcontrolador reprogramable y pines-hembra que posibilita conectar de una manera fácil y manipulable distintos sensores y actuadores. Las PCBs sirven para conectar de manera eléctrica por medio de camino conductores, distintos componentes electrónicos soldados a ella [3]. Entre sus ventajas tenemos las siguientes:

- Es libre y extensible, siendo capaz de ampliar y mejorar tanto en su diseño del hardware como en su entorno de software.
- Facilidad de instalación y ejecución en cualquier plataforma (Windows, Mac OS, Linux).
- Lenguaje de programación sencillo y de fácil manejo.
- Sus placas son económicamente accesibles y reutilizables.

Existen varios tipos de placas de Arduino, las cuales se mencionarán algunas a continuación:

1.4.2 Arduino UNO

Admite desde 7 hasta 12 voltios en continua y además contiene una salida de 3.3 voltios en dc y otra salida de 5 voltios en dc. Presenta 6 entradas analógicas, cada entrada sirve para tener de 0 a 5 voltios, es decir que se puede obtener 1, 2, 3, 4 voltios como lectura o como salida hacia otros periféricos. También dispone de 14 pines de entrada/salida digital, tal y como se muestra en la Figura 1.

Posee un regulador de voltaje, que se encargará de convertir los 7-12 voltios a 5 voltios con los que trabajará la tarjeta, tiene un conector usb del estándar B por el cuál podremos conectar nuestra laptop o computadoras y programarla con el software.

Para la comunicación con la computadora, esta se la hará a través de la ATmega16u2 que es el driver usb de control que cumple la función de comunicador, pues a través de esta llegarán tipos de datos, estos se convertirán y serán interpretados por el microcontrolador ATmega328P de 8 bits.

La tarjeta funciona con un cristal de 16 Mhz es decir que el microprocesador es capaz de realizar aproximadamente 16 millones de instrucciones en cada segundo [3].



Figura 1: Placa del Arduino Uno [4].

1.4.3 Arduino Mega 2560

Esta placa posee un microcontrolador ATmega2560 a 16Mhz. Como se puede apreciar en la Figura 2 dentro de sus características principales tiene 54 pines de entrada/salidas digitales, 16 entradas analógicas y 4 receptores/transmisores. Posee una memoria flash de 256 KB (memoria del programa de los cuales 8 KB son usados por el bootloader), una memoria SRAM integrada de 8KB donde el Arduino crea y manipula las variables cuando se ejecuta y una EEPROM de 4KB memoria no volátil para mantener datos después de un reset o apagado de microcontrolador, las demás características son parecidas a las del Arduino UNO [3].



Figura 2: Placa del Arduino Mega 2560 [4].

1.4.4 Arduino Mega ADK

Es una placa que tiene mucha similitud con la del Mega 2560, siendo la principal diferencia en que el Mega ADK está capacitado para operar como un dispositivo de tipo “host USB”, es decir que a más de actuar como periférico USB también lo puede hacer como host USB incorporando un conector USB de tipo A para cumplir esta función como lo muestra la Figura 3. En otras palabras, este tipo de Arduino cuenta con la capacidad de conectarse con dispositivos que tengan un puerto de USB periférico como celulares, cámaras de fotos o video, mouse, teclados, mandos de consolas, etc. [3].



Figura 3: Placa del Arduino Mega ADK [4].

1.4.5 Arduino Ethernet

Tiene un microcontrolador ATmega328P igual al del Arduino UNO, es decir que tendrá misma cantidad de memoria flash (SRAM y EEPROM), también tiene la misma cantidad de pines tanto digitales como analógicas.

La principal diferencia con el modelo UNO es que la placa Ethernet integra una plaza de tipo RJ-45 como lo vemos en la Figura 4, la cual sirve para conectarse mediante un cable adecuado a una red de tipo Ethernet.

Su función destacada es que posibilita la transferencia de datos entre ella misma con otro dispositivo que esté conectado a la misma red LAN o viceversa [3].

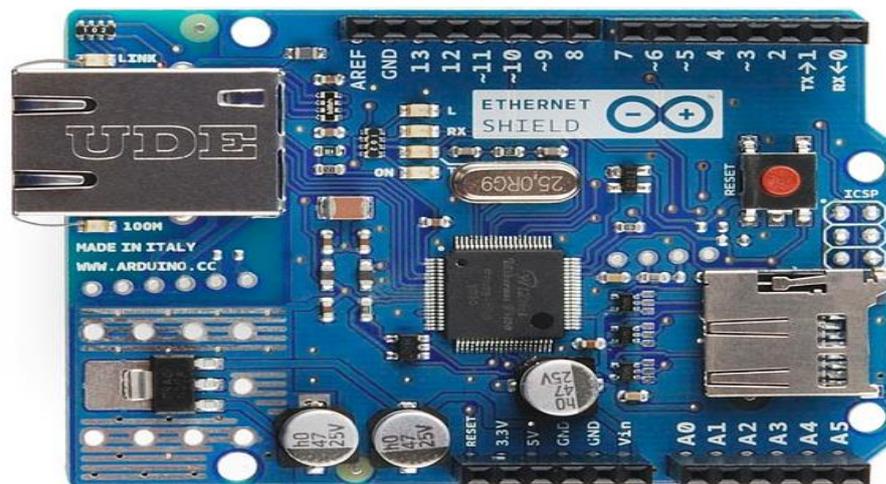


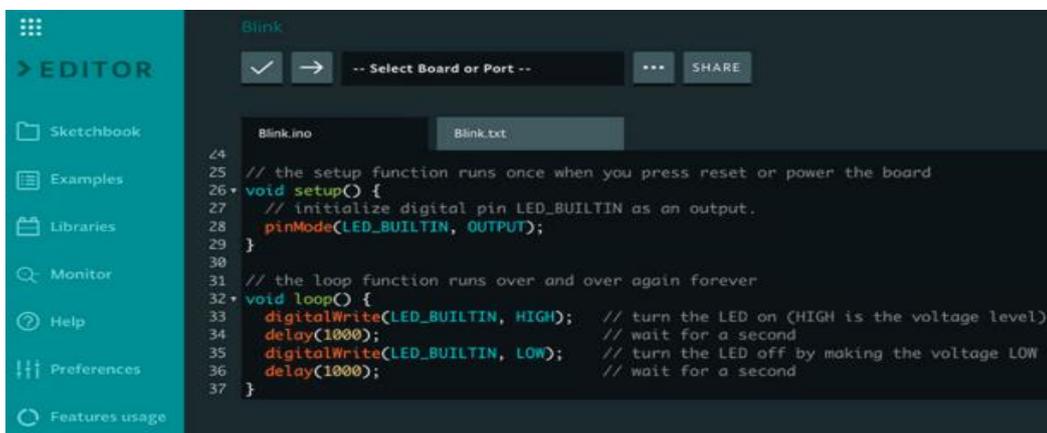
Figura 4: Placa del Arduino Ethernet Shield [4].

Tabla 1 Comparación de características básicas de las placas Arduino.

| Arduino | UNO | Mega 2560 | Mega ADK | Ethernet |
|---------------------------------------|------------|------------|------------|-----------|
| Microcontrolador | ATmega328P | ATmega2560 | ATmega2560 | ATMega328 |
| Voltaje de operación | 5 V | 5 V | 5 V | 5 V |
| Voltaje de alimentación (recomendado) | 7 – 12 V | 7 – 12 V | 7 – 12 V | 7 – 12 V |
| Voltaje de alimentación (límite) | 6 – 20 V | 6 – 20 V | 5.5 – 16 V | 6 – 20 V |
| Corriente por pin | 20 mA | 20 mA | 40 mA | 40 mA |
| Corriente sobre pin 3,3V | 50 mA | 50 mA | 50 mA | 50 mA |
| E/S digitales | 14 | 54 | 54 | 14 |
| Entradas análogas | 6 | 16 | 16 | 6 |
| Memoria Flash | 32 KB | 256 KB | 256KB | 32 KB |
| SRAM | 2KB | 8 KB | 8 KB | 2KB |
| EEPROM | 1 KB | 4 KB | 4 KB | 1 KB |
| Velocidad de reloj | 16 MHz | 16 MHz | 16 MHz | 16 MHz |

1.4.6 Arduino IDE

Es un software de código abierto que se utiliza para escribir y compilar código en el módulo de Arduino. Se encuentra disponible para sistemas operativos como MAC, Windows, Linux; ejecutándose en el entorno Java, el cuál presenta funciones y comandos añadidos para la depuración, edición y compilación del código en el entorno. Por otro lado, su entorno es compatible con los lenguajes de programación C y C++. Está compuesto por partes fundamentales que son el editor y el compilador (Figura 5), los cuales sirven para escribir, editar y cargar el código en el módulo Arduino dado [5]. Es importante mencionar que el programa acepta el uso de librerías para que las tareas sean descritas cómodamente [6].

The image shows the Arduino IDE interface. On the left is a teal sidebar with navigation options: EDITOR, Sketchbook, Examples, Libraries, Monitor, Help, Preferences, and Features usage. The main workspace is dark-themed and displays the 'Blink' sketch. At the top, there are buttons for a checkmark, a right arrow, a dropdown menu labeled '-- Select Board or Port --', and a 'SHARE' button. The sketch code is as follows:

```
24
25 // the setup function runs once when you press reset or power the board
26 void setup() {
27   // initialize digital pin LED_BUILTIN as an output.
28   pinMode(LED_BUILTIN, OUTPUT);
29 }
30
31 // the loop function runs over and over again forever
32 void loop() {
33   digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
34   delay(1000); // wait for a second
35   digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW
36   delay(1000); // wait for a second
37 }
```

Figura 5: Entorno del software Arduino IDE [7].

1.4.7 Transformada rápida de Fourier (FFT)

La Transformada Rápida de Fourier (FFT) es un algoritmo o método matemático para calcular eficientemente la Transformada Discreta de Fourier (DFT) requiriendo un esfuerzo computacional mucho menor que realizándolo por el cálculo directo de la DFT, optimizando tanto el número de operaciones a realizar como el tiempo de cómputo, se muestra en la Figura 6. Por ejemplo, si el cálculo directo de la DFT tarda 1 hora, con la FFT estaría disponible en solamente 35 segundos. Esta herramienta facilita el análisis de espectro de potencia y la simulación de filtros por medio de computadoras digitales [8].

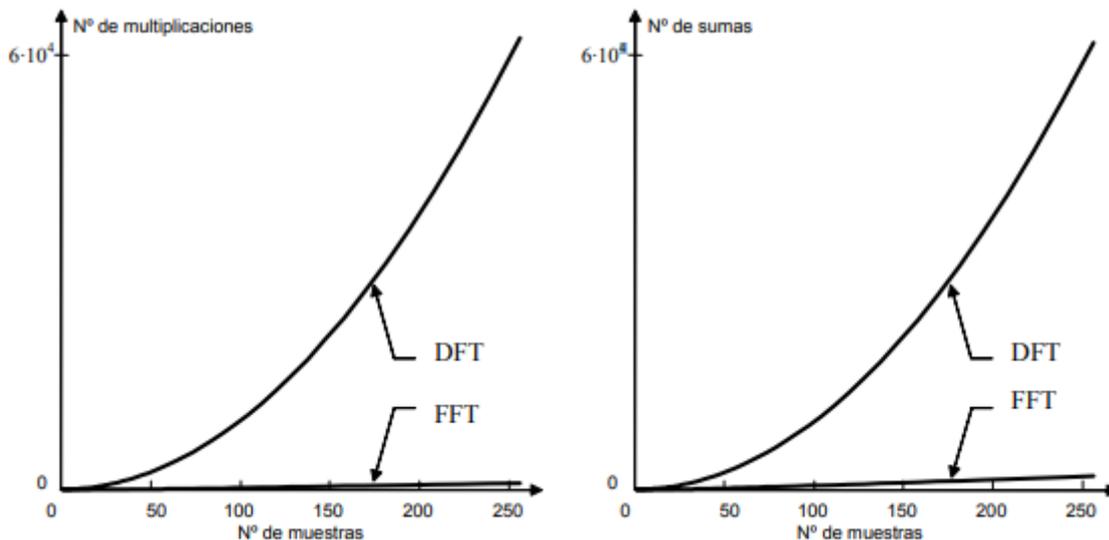


Figura 6: Comparación entre las operaciones de una DFT y una FFT [8].

1.4.8 Calidad de energía

Ausencia de interrupciones, sobretensiones, deformaciones producidas por armónicos en la red y variaciones de voltaje rms suministrado al usuario [9].

Existen muchas maneras de clasificar los problemas de calidad de energía, cada una de estas usa un atributo para organizar el problema. Para algunas regulaciones como la ANSI C84.1 lo más importante es el tiempo del evento, por otro lado tenemos a la IEEE-519 la cual se enfoca en la forma de onda (magnitud y duración) de cada evento y en caso de estándares como la IEC usa el rango de frecuencia del evento para clasificar los problemas [10]. A continuación, se mencionan algunos eventos que pueden causar problemas de calidad de energía:

1.4.9 Variaciones de larga duración

Se define como la desviación del valor eficaz de la tensión con respecto al valor de la tensión nominal con una duración de más de 1 minuto, se tienen los siguientes tipos según la IEEE-1159.

Interrupción sostenida: siendo esta la más grave donde la tensión cae a cero y no regresa automáticamente, pues su duración según la IEC es de más de 3 minutos.

Subtensiones: es equivalente a un sag, es decir una caída de voltaje provocada por una disminución o pérdida del voltaje eficaz (RMS) en el sistema eléctrico.

Sobretensiones: es equivalente a un swell, que no es más que una elevación del valor eficaz del voltaje.

1.4.10 Desbalance de tensión

Cuando se habla de desbalance de tensión nos referimos a un sistema trifásico, el cual en su magnitud por fase o en los ángulos entre fases puede haber una variación y que no todos sean iguales o constantes, representado problemas para el sistema. La forma más fácil de calcular el factor de desviación es dividiendo la desviación máxima del voltaje sobre el voltaje promedio por 100. Los rangos permisibles para este factor oscilan entre el 0.2% - 0.5%.

1.4.11 Estándares de calidad de energía

Las técnicas de medición y los rangos aceptables de los indicadores de calidad de energía están estandarizados de acuerdo con normas nacionales e internacionales. Dentro de las normas internacionales de calidad de la energía eléctrica tenemos la de la Comisión Electrotécnica Internacional (IEC por sus siglas en inglés) con su norma IEC 61000 y la del Instituto de Ingenieros Eléctricos y Electrónicos (IEEE por sus siglas en inglés) con su norma IEEE-519 [11]. La norma IEC consiste en seis partes, cada una de las cuales conta de una o más secciones.

1.4.12 Medición de calidad de energía

Un esquema general para llevar a cabo mediciones de calidad de energía es mostrado en la Figura 7. Los equipos de medición incluyen a los transformadores de instrumentación y los filtros anti-aliasing analógicos y digitales. El filtro anti-aliasing es necesario para eliminar las frecuencias por encima de la frecuencia de Nyquist (mitad de la frecuencia de muestreo) que puedan causar una mala conversión analógica-digital de la señal, de modo que pierda características de la señal analógica original. Luego de esto se procede a muestrear las señales de voltaje y corriente, para después hacer un preprocesamiento de los datos que a menudo es llevado a cabo en computadoras que se encuentran lejos del equipo de monitoreo de calidad de energía [12]. Para muchos equipos de monitoreo de calidad de energía, el primer paso es el cálculo de las características más significativas. Estas pueden ser el valor RMS (Root Mean Square), la frecuencia y el espectro de frecuencia. Por lo general se utilizan valores promedios durante un cierto intervalo de tiempo, por ejemplo un intervalo de 10, 12, 150 o 180 ciclos,

10 minutos y 2 horas. Para después hacer el cálculo y registro de los indicadores de la calidad de energía eléctrica y de los eventos que se presenten durante el intervalo [12].

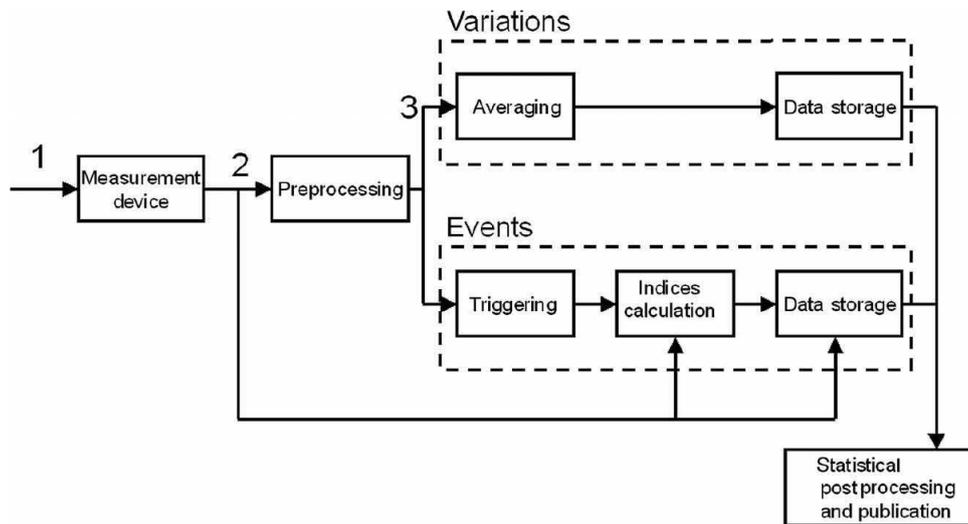


Figura 7: Esquema general de la medición de la calidad de la energía [12].

CAPÍTULO 2

2. METODOLOGÍA

Para la implementación del analizador de redes con Arduino se siguió una serie de procedimientos y fases que posibilita un mejor entendimiento de lo que fue su desarrollo, en la Figura 8 se procederá a la explicar mediante un diagrama de bloques estas fases.

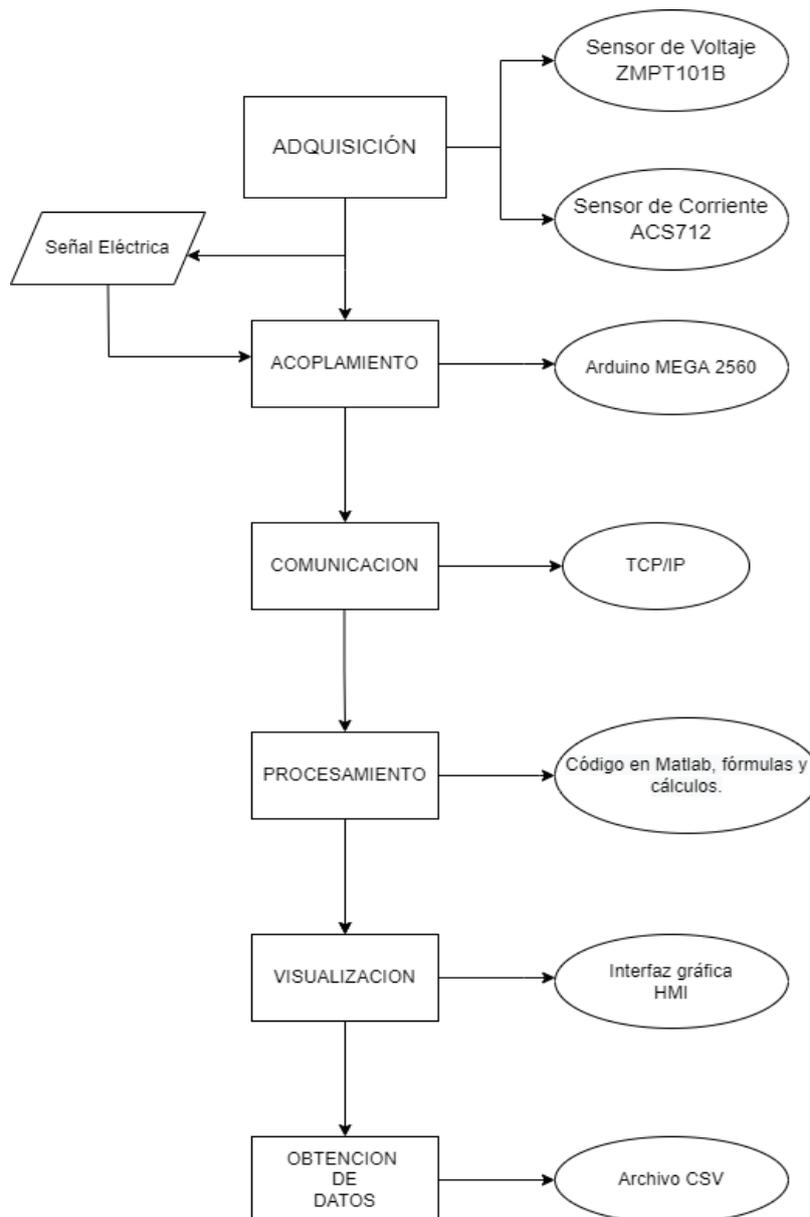


Figura 8: Diagrama de bloques del sistema

2.1 Fase de Adquisición

Aquí se procedió a la obtención e interpretación de las señales de voltaje, corriente, potencia, emitidas por parte de la red eléctrica, motores o transformadores.

2.1.1 Medición de voltaje

Esta fase es de mucha importancia, pues será aquí donde se obtendrán los valores diferenciales de potencial eléctrico (medido en voltios en AC), los cuales servirán para los respectivos cálculos de frecuencia, factores de potencia, etc. Toda esta lectura se llevará a cabo mediante el controlador Arduino.

2.1.2 Sensor ZMPT101B

Este módulo nos permite bajar el voltaje que tenemos en la entrada hasta un valor seguro y utilizable para la entrada del ADC. Tiene una entrada de voltaje alterno y una salida en la cual se encuentra unos pines que irían hacia el convertidor analógico digital.

Es muy importante tener en cuenta que este módulo cuenta en su interior con un transformador de aislamiento que añade protección al circuito, es decir que cualquier inconveniente que presente el voltaje de entrada no se verá afectado el voltaje de salida, dado que estos se encuentran aislados entre sí.

El sensor ZMPT101B tiene la capacidad de medir hasta un voltaje de 250 VCA, cuenta con un ajuste de giro múltiple con potenciómetro para ajustar la salida a DC, ya que internamente reproduce una señal de onda adecuada para la lectura analógica de la tarjeta Arduino. En la Figura 9 se muestra el sensor con las partes que lo conforman.



Figura 9: Sensor ZMPT101B

2.1.3 Sensor ACS712

En el circuito tenemos un hilo conductor por el cual circulará la corriente que deseamos medir, internamente el sensor de efecto hall mide la corriente magnética que se genera cuando pasa la corriente por ese conductor convirtiendo este campo en un voltaje variable. Esto significa que a mayor cantidad de corriente que tengamos, mayor voltaje vamos a tener en un pin. Este módulo consta de una bornera para conectar la línea que vamos a medir y 3 pines, los cuales 2 son para la alimentación y el pin restante es para la salida analógica.

Existen 3 tipos de sensores de corriente, diferenciados por su rango de corriente y su sensibilidad. Para nuestro proyecto haremos uso del sensor ACS712 de rango -30 a 30 A con una sensibilidad de 66 mV/A el cual lo vemos en la Figura 10.



Figura 10: Sensor ACS712

2.2 Fase de Acoplamiento

Su función es convertir la señal física obtenida de la red eléctrica en una señal digital que se puede utilizar para realizar cálculos y convertirla a valores reales. Esto se lo lleva a cabo a través del controlador Arduino Mega 2560, mediante el ADC de 10 bits que lee los valores analógicos en un rango de 0 a 5 V de los sensores de corriente y voltaje.

2.3 Fase de Comunicación

Para la fase de comunicación se empleó el Arduino Ethernet Shield, el Arduino MEGA 2560 y el modelo TCP/IP. La conexión trata de colocar el shield sobre el Arduino y conectar el cable ethernet a un router o modem para empezar la

programación haciendo uso del protocolo TCP/IP a través del cual se logra la comunicación del Arduino hacia la computadora.

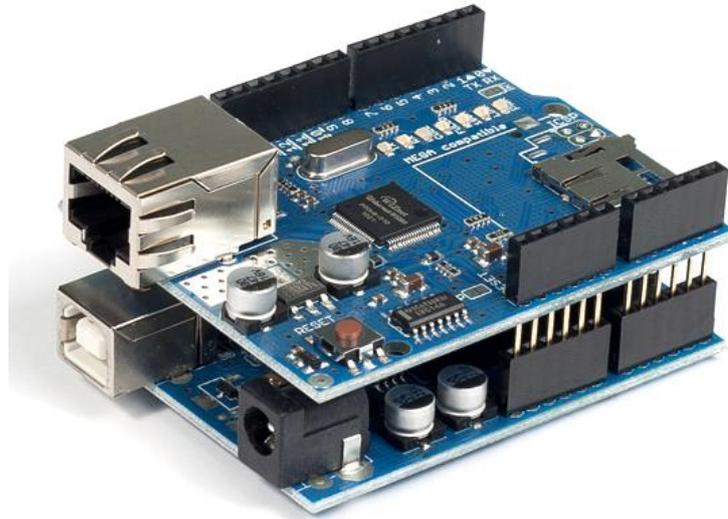


Figura 11: Arduino Mega 2560 con Shield Ethernet

2.4 Fase de procesamiento de datos

En esta fase se efectúa el registro de la información obtenida y reporte de algún evento en caso de darse. Gracias a los valores de voltaje y corriente que se adquirieron previamente, en esta etapa se pasa a calcular valores de potencia, factor de potencia y demás.

2.5 Fase de visualización

A través de una interfaz gráfica interactiva realizada en MATLAB, se procede a mostrar los resultados finales que dejará al usuario observar con facilidad los datos reales tomados de la red eléctrica u otro objeto de estudio (motor, transformador, etc).

2.6 Proceso de la implementación del analizador de redes con Arduino

A continuación, en la Figura 12 se muestra como es el proceso de inicio-fin de la conexión con red, visualización de los resultados y toma de datos, implementando el analizador de redes.

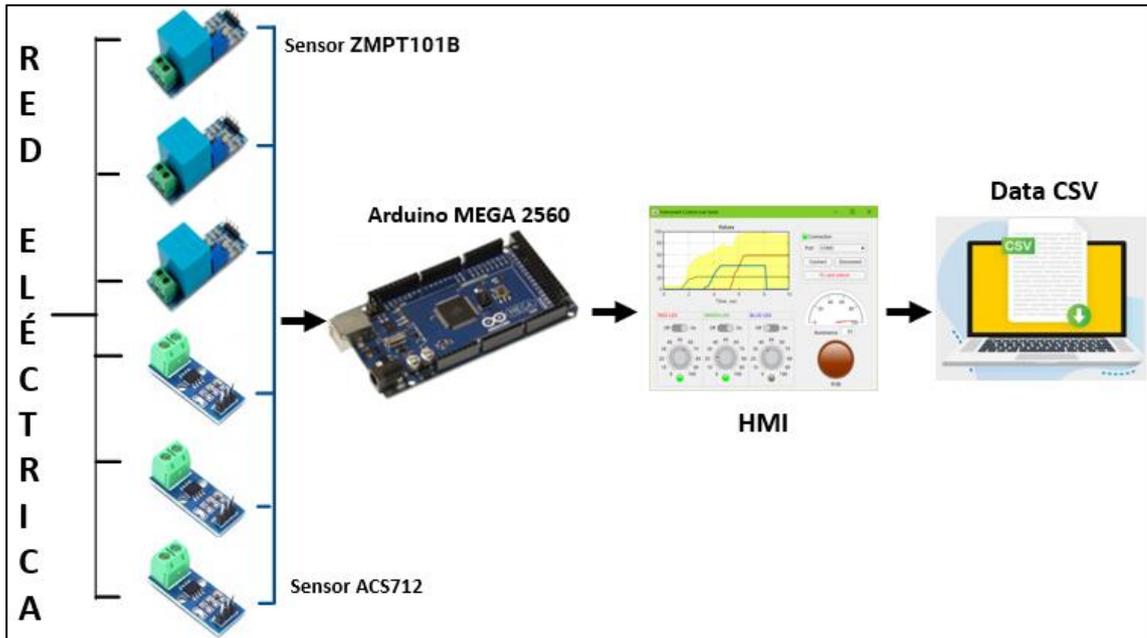


Figura 12: Proceso de ejecución del analizador de redes con Arduino.

2.7 Normativas

2.7.1 Norma IEC 61000-4-30

Se encarga de definir las formas de medición de los datos de calidad de suministro de energía y la manera de analizar los resultados. La norma expresa como medir sin establecer los umbrales, también manifiesta los métodos de detección y evaluación de: los huecos de tensión, sobretensiones temporales y las interrupciones del voltaje de suministro.

2.7.2 Norma IEC 61000-4-11

Define los métodos de prueba de inmunidad y el rango de niveles de prueba preferidos para equipos eléctricos y electrónicos conectados a redes de suministro de energía de bajo voltaje para caídas de voltaje, interrupciones breves y variaciones de voltaje. Esta norma se aplica a equipos eléctricos y electrónicos que tengan una corriente nominal de entrada que no supere los 16 A por fase, para conexión a redes de CA de 50 Hz o 60 Hz. El objetivo de la norma es establecer una referencia común para evaluar la inmunidad de los equipos eléctricos y electrónicos cuando están sujetos a huecos de tensión, interrupciones breves y variaciones de tensión.

2.7.3 Norma IEC 61000-4-7

Se aplica a la instrumentación destinada a medir componentes espectrales en el rango de frecuencia de hasta 9 kHz que se superponen a la fundamental de los sistemas de suministro de energía a 50 Hz y 60 Hz. Esta norma distingue entre armónicos, interarmónicos y otros componentes por encima del rango de frecuencia armónica, hasta 9 kHz.

2.7.4 Norma IEEE 1459

La norma trata los conceptos de potencias eléctricas en sistemas simples, como por ejemplo uno monofásico con onda sinusoidal pura, para posteriormente incluir sistemas más complejos, y finalmente abordar el sistema trifásico no balanceado con ondas no sinusoidales. El estándar emplea nuevas definiciones de potencias aparentes, para los cuales se consigue un sistema trifásico equivalente que obedece las siguientes características:

- Equilibrado en tensiones
- Balanceado en corrientes
- Mismas pérdidas del circuito trifásico verdadero

Con base a esto se obtienen tensión y corriente equivalente para sistemas de cuatro y tres hilos.

- Para un sistema trifásico de cuatro hilos:

$$V_e = \sqrt{\frac{1}{18} [3(V_a^2 + V_b^2 + V_c^2) + V_{ab}^2 + V_{bc}^2 + V_{ca}^2]}$$

$$I_e = \sqrt{\frac{I_a^2 + I_b^2 + I_c^2 + I_n^2}{3}}$$

- Para un sistema trifásico de tres hilos:

$$V_e = \sqrt{\frac{V_{ab}^2 + V_{bc}^2 + V_{ca}^2}{9}}$$

$$I_e = \sqrt{\frac{I_a^2 + I_b^2 + I_c^2}{3}}$$

Considerando lo anterior podemos obtener la potencia aparente efectiva como:

$$S_e = 3V_e I_e$$

A continuación, se muestra una tabla con las cantidades de potencia para un sistema trifásico con formas de ondas no sinusoidales:

Tabla 2 Magnitudes para sistemas trifásicos con armónicos

| Cantidad | Combinado | Fundamental | No Fundamental |
|-------------------------|---------------|-----------------------------------|-------------------------------------|
| Aparente | $S_e (VA)$ | $S_{e1} - S_1^+ - S_{1U}$ (VA) | $S_{eN} - S_{eH} (VA)$ |
| Activa | $P (W)$ | $P_1^+ (W)$ | $P_H (W)$ |
| No Activa | $N (var)$ | $Q_1^+ (var)$ | $D_{eI} - D_{eV} - D_{eH}$ (var) |
| Utilización de la línea | $P_f = P/S_e$ | $P_{F1}^+ = P_1^+/S_1^+$ | - |
| Polución armónica | - | - | S_{eN}/S_{e1} |
| No balance de la carga | - | S_{1U}/S_1^+ | - |

2.8 Captura de corrientes de arranques.

El prototipo implementado tiene la capacidad de capturar corrientes de arranques de máquinas eléctricas, estas corrientes son sobrecorrientes que se producen cuando existe una baja impedancia en la carga. A diferencia de las corrientes de fallas estas sobrecorrientes duran un tiempo definido y no es necesario despejarlas, ya que tienen a estabilizarse tras unos instantes a un valor de corriente inferior.

La corriente de arranque se mide con un modo de disparo único ("Trigger") que registra las tendencias de corriente tras producirse el evento de sobrecorriente. El evento se produce cuando la forma de onda de corriente es superior a los límites predefinidos. La captura comienza cuando el valor Arms de medio ciclo de una de las fases es superior al umbral predefinido, y termina cuando es menor al umbral menos la histéresis definida.

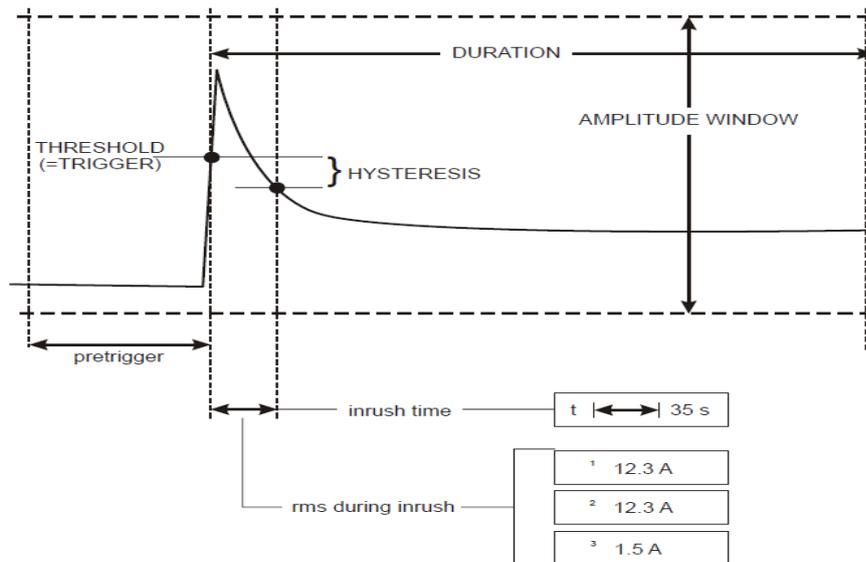


Figura 13: Característica de la captura de la corriente de arranque.

2.9 Procesamiento de datos.

Para el procesamiento de las señales eléctricas de voltaje y corrientes obtenidas mediante el sistema de adquisición desarrollado con la placa Arduino, se elaboró un programa que sigue el algoritmo presentado en la Figura 14, el programa desarrollado primero lee los datos empaquetados en uint_8 de un buffer de 6144 bytes que son enviados vía TCP/IP desde el puerto ethernet de Arduino, luego el programa convierte los bytes recibidos en datos double que puedan ser procesados por las funciones matemáticas de Matlab, después el programa procede a aplicar un filtro FIR digital para luego proceder a actualizar la gráfica de las formas de onda de voltaje o corriente y además calcula los valores RMS mediante la definición. Una vez obtenido los datos convertidos se procede a calcular la transformada rápida de Fourier (FFT) para determinar las magnitudes de las componentes armónicas de los voltajes y las corrientes además de determinar las potencias reactivas y activas correspondientes, una vez calculado todos los parámetros necesarios, se proceden a mostrarlos en la interfaz gráfica del analizador de redes. Luego de terminado el proceso anterior, el programa vuelve a leer los 6144 bytes para repetir el proceso descrito anteriormente. Si el registro esta iniciado el programa hace un proceso en paralelo donde registra los parámetros eléctricos más significativos en un buffer hasta que se haya cumplido el intervalo de tiempo definido por el usuario para luego proceder a determinar los valores máximos, promedios, mediana y mínimos de ese intervalo de tiempo y guardarlos en un CSV.

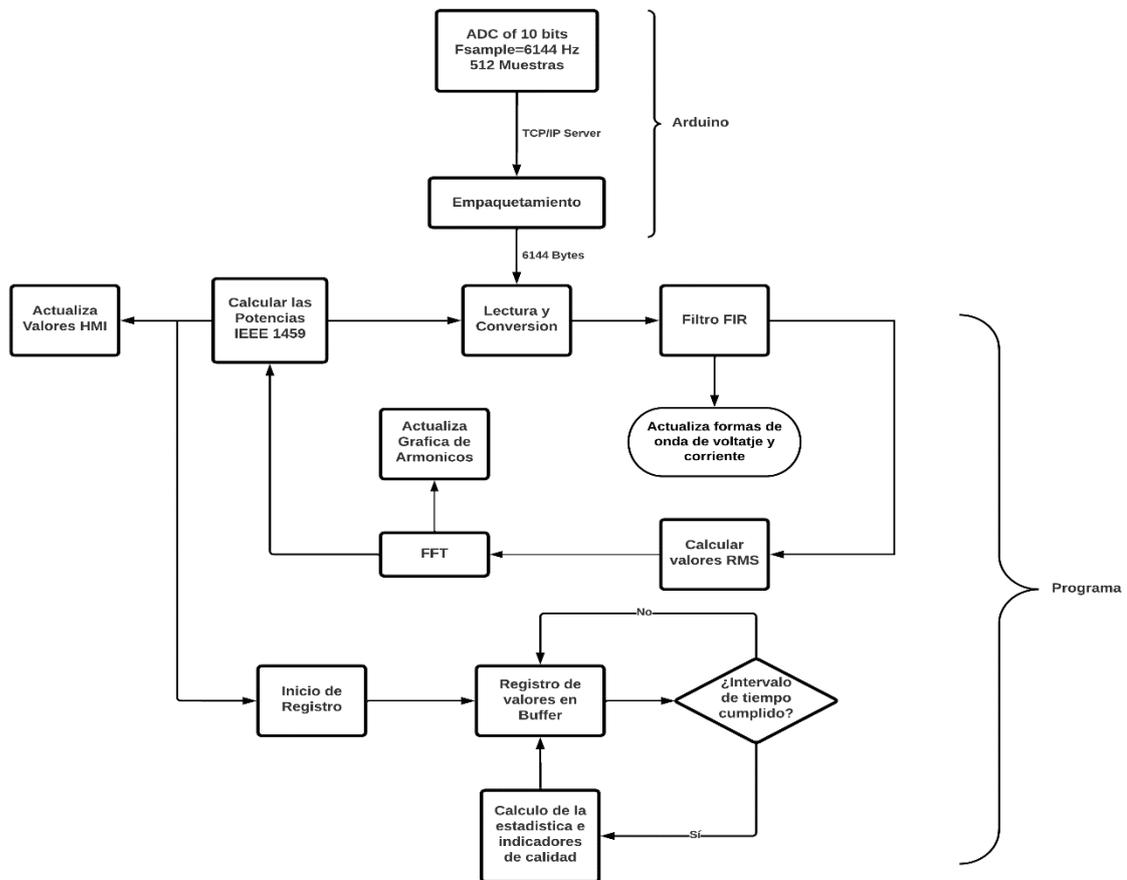


Figura 14 Algoritmo del software desarrollado para el analizador de redes.

2.10 Diseño Interfaz Gráfica.

Mediante la herramienta App Designer del software de programación MATLAB, se diseñó una interfaz gráfica que permita hacer el análisis y tratamiento de las señales eléctricas de voltaje y corriente obtenidas a través del ADC (Analog Digital Converter) del microcontrolador de Atmega de la placa Arduino Mega. El software hace el tratamiento de las señales de voltaje y corriente para la obtención de los parámetros de potencia activa, reactiva y aparente, además del factor de potencia. Por otro lado, el software es capaz de mostrar las formas de onda, la tendencia de las variables de voltaje, corriente y potencia en una ventana de 1 minuto, así como registrar en un archivo csv durante intervalos que van desde 1 segundo hasta 15 minutos.



Figura 15: Pestaña de configuración del analizador de redes.

En la Figura 15 se puede la pestaña de configuración del analizador de redes, donde el usuario podrá realizar las siguientes funciones:

- **Conectividad:** Sirve para establecer la comunicación vía TCP/IP con el servidor de Arduino que tiene una IP y puerto definida de 169.254.108.245 y 100 respectivamente.
- **Datos de la red:** Sirve para definir los parámetros de la red eléctrica como lo es el voltaje de línea a neutro, frecuencia, tipo de red, corriente de carga máxima y la corriente de cortocircuito máxima. Estos parámetros servirán para los respectivos cálculos de los indicadores de calidad de energía eléctrica.
- **Configuración de Registro:** En este apartado el usuario podrá definir el intervalo de tiempo en el cual se va a registrar los parámetros eléctricos calculados por el software.
- **Configuración de Sensores:** En este apartado el usuario podrá insertar los valores de la sensibilidad de los sensores de voltaje y corriente que vaya a utilizar.

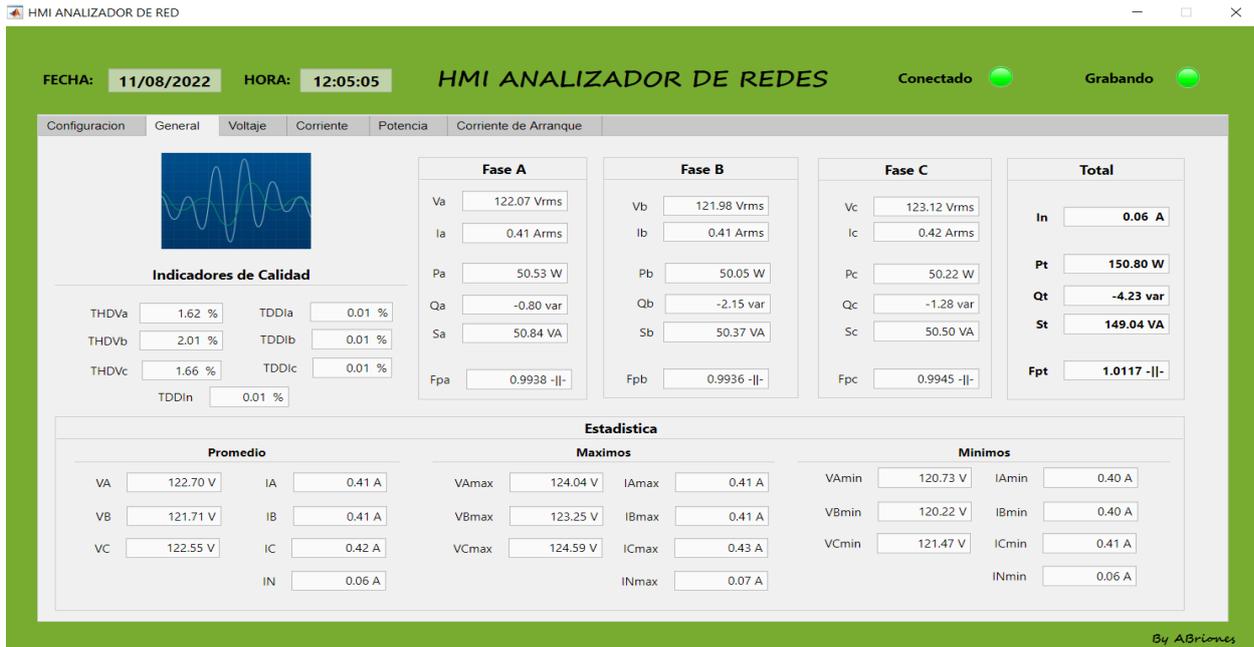


Figura 16: Pestaña General del analizador de redes.

En la Figura 16, se puede observar la vista general del equipo donde se hace un recopilatorio de todas las variables medidas por el equipo. Las variables medidas y presentadas son: Voltaje de línea a neutro, Corriente de Línea, Potencias Activas, Reactivas y Aparente de Fase y Total y el Factor de Potencia de Dispersión de cada fase y el total. También se puede observar en esta pestaña los indicadores de calidad calculados como lo es el THD (Total Harmonic Distortion) de voltaje y el TDD (Total Demand Distortion) de corriente que son los indicadores más representativos de estas variables. Por último en esta vista se puede observar un recopilatorio de la estadística de las variables de voltaje y corriente siempre y cuando haya iniciado el registro desde la pestaña de configuración.



Figura 17: Pestañas de Voltaje y Corriente del Analizador de Redes.

En la Figura 17, se puede observar las vistas de las variables de voltaje y corriente medidas de la red eléctrica, en estos apartados el usuario puede observar las gráficas de forma de onda, el valor RMS (Root Mean Square) y el diagrama de barras de los armónicos presentes en las señales. Además el apartado cuenta un botón para visualizar los fasores de voltaje y corriente. Y por último el usuario podrá ver los valores RMS y los respectivos ángulos de cada fase, así como la estadística y los indicadores de calidad descritos en la vista general.

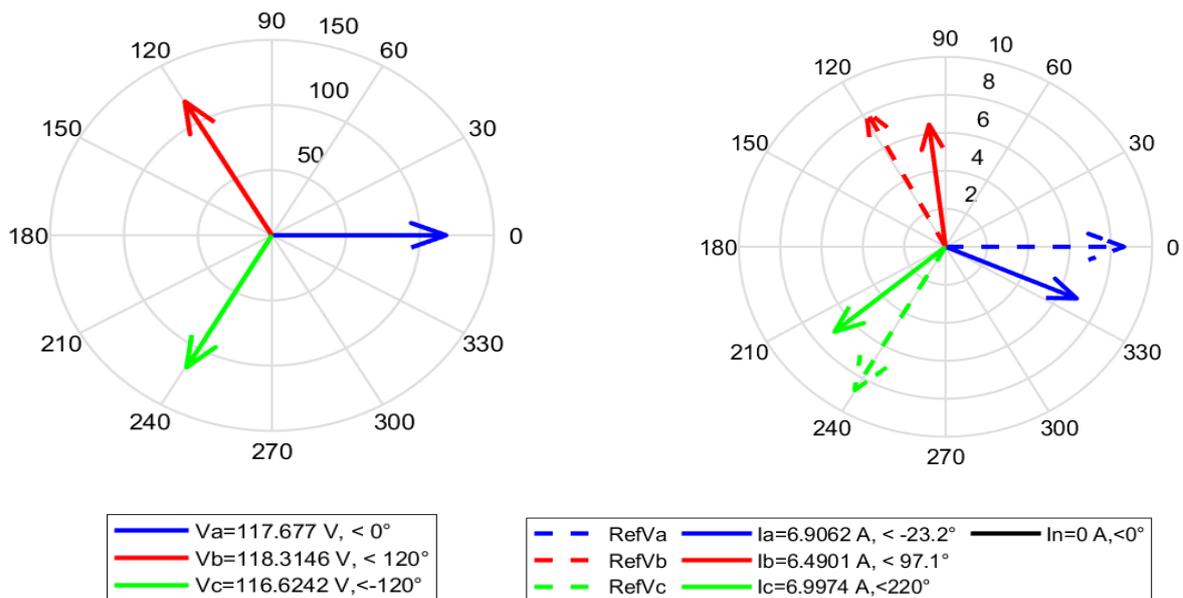


Figura 18: Diagramas Fasoriales de voltaje y corriente.



Figura 19: Pestana de Potencia del analizador de redes.

En la Figura 19, se puede observar la vista de potencia que muestra el software del analizador de redes, donde muestra las gráficas de tendencia de la potencia activa de cada fase y la total, y la gráfica de tendencia del factor de potencia de dispersión de cada fase y la total. Además en este apartado se puede visualizar los valores de la potencia activa, reactiva y aparente. Así como el factor de potencia y la energía activa y aparente consumida por la red eléctrica.

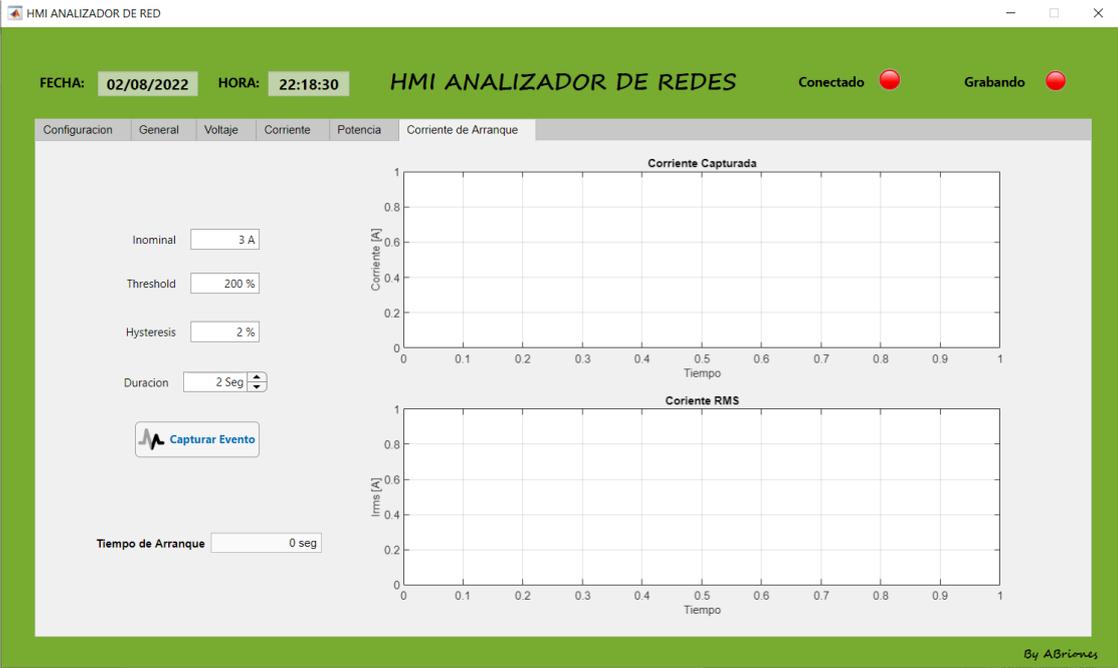


Figura 20: Pestana de análisis de corriente de arranque.

En la Figura 20, se puede observar el apartado del análisis de corriente de arranque para maquinas eléctricas; en esta vista el usuario puede capturar la forma de onda de la corriente de arranque así como la gráfica de valor eficaz de cada medio ciclo de la corriente, para ello el usuario debe ingresar el valor de la corriente nominal estimada de la maquina eléctrica así como el “Threshold” que es el umbral para empezar la captura del evento transitorio, la “Hysteresis” que define en el franco de bajada donde dejara de capturar y por último la duración estimada del evento transitorio.

CAPÍTULO 3

3. RESULTADOS Y ANÁLISIS

3.1 Instrumento patrón

Para realizar la contrastación de valores se usó como patrón de medida el del registrador de energía FLUKE 1732, puesto que es un equipo ya verificado y aceptado conforme a las normas internacionales. Este equipo fue de mucha ayuda para así poder (a más de calibrar de la calibración) ver que tan preciso son los valores que arroja nuestro prototipo de acuerdo con el patrón de medida del FLUKE. A continuación, se muestra en las siguientes tablas la precisión en condiciones de referencia del FLUKE 1732:

Tabla 3: Rangos y precisión del FLUKE 1732 en condiciones de referencia

| Parámetros | | Rango | Resolución Máxima | Precisión intrínseca en condiciones de referencia (% de Lectura + % de Rango) |
|------------------|-----------------|------------------|----------------------|--|
| Voltaje | | 1000 V | 0.1 V | $\pm(0.2\% + 0.01\%)$ |
| Corriente | 1500 A Flexible | 150 A | 0.01 A | $\pm(1\% + 0.02\%)$ |
| | | 1500 A | 0.1 A | $\pm(1\% + 0.02\%)$ |
| | 3000 A Flexible | 300 A | 1 A | $\pm(1\% + 0.03\%)$ |
| | | 3000 A | 10 A | $\pm(1\% + 0.03\%)$ |
| | 6000 A Flexible | 600 A | 1 A | $\pm(1.5\% + 0.03\%)$ |
| | | 6000 A | 10 A | $\pm(1.5\% + 0.03\%)$ |
| | 40 A | 4 A | 1 mA | $\pm(0.7\% + 0.02\%)$ |
| | | 40 A | 10 mA | $\pm(0.7\% + 0.02\%)$ |
| Frecuencia | | 42.5 Hz to 69 Hz | 0.01 Hz | $\pm 0.1\%$ |
| Entrada Auxiliar | | 10 Vdc | 0.1 mV | $\pm(0.2\% + 0.02\%)$ |
| Voltaje Min/Máx | | 1000 V | 0.1 V | $\pm(1\% + 0.1\%)$ |

| | | | |
|-------------------|------------------------|------------------------|-----------------------|
| Corriente Min/Máx | Definida por accesorio | Definida por accesorio | $\pm(5\% + 0.2\%)$ |
| THD en voltaje | 1000% | 0.1 % | $\pm(2.5\% + 0.05\%)$ |
| THD en corriente | 1000% | 0.1 % | $\pm(2.5\% + 0.05\%)$ |

Tabla 4: Precisión de potencias del FLUKE 1732 en condiciones de referencia

| Precisión intrínseca en condiciones de referencia | | | | | |
|--|-------------------------------|-------------------------------------|------------------------|------------------------|------------------|
| (% de Lectura + % de Rango) | | | | | |
| Parámetros | Cantidad de influencia | 1500 A Flexible | 3000 A Flexible | 6000 A Flexible | 40 A |
| Potencia Activa Energía Activa | $FP \geq 0.99$ | 1.2% + 0.005% | 1.2% + 0.0075% | 1.7% + 0.0075% | 1.2% + 0.005% |
| Potencia Aparente Energía Aparente | $0 \leq FP \leq 1$ | 1.2% + 0.005% | 1.2% + 0.0075% | 1.7% + 0.075% | 1.2% + 0.005% |
| Potencia Reactiva Energía Reactiva | $0 \leq FP \leq 1$ | 2.5% de la potencia aparente medida | | | |
| Incertidumbre adicional en % del rango | $V_{P-N} > 250V$ | 0.015% | 0.0225% | 0.0225% | 0.015% |

Otras Condiciones:

- Ambiental: $23^{\circ}\text{C} \pm 5^{\circ}\text{C}$, instrumento en funcionamiento durante al menos 30 minutos, sin campo eléctrico/magnético externo, $HR < 65\%$
- Condiciones de entrada: $\cos\phi/FP = 1$, señal sinusoidal $f = 50\text{ Hz}/60\text{ Hz}$, fuente de alimentación $120\text{ V}/230\text{ V} \pm 10\%$

- Especificaciones de corriente y potencia: voltaje de entrada 1fase: 120 V/230 V o 3 fases *wye/delta*: 230 V/400 V, Corriente de entrada: $I > 10\%$ de Irango
- Conductor primario de pinzas o bobina de Rogowski en posición central
- Coeficiente de temperatura: agregue 0,1 x la precisión especificada por cada °C por encima de 28°C o por debajo de 18°C

3.2 Medida de error

Puesto que se trata de un equipo de medición, es de vital importancia saber la precisión de los valores que se irán midiendo, como lo son: voltajes, corrientes, potencias, energías, etc. Para aquello será muy necesario el cálculo del error de medición con la finalidad de poder observar que tan exacto es el grado de la medida que se está realizando mediante el prototipo.

Se utilizó el registrador de energía FLUKE 1732, que se muestra en la figura, el cual estuvo registrando los valores a la par con el prototipo, ambos trabajando bajo el mismo escenario.



Figura 21: Registrador de energía FLUKE 1732

Para el cálculo del error se tomó como valor verdadero o teórico el valor que nos mostraba el FLUKE 1732, mientras que para el valor medido o experimental se tomó el valor que nos mostraba nuestro prototipo.

3.3 Medida de Voltajes, Corrientes y Potencias

Para el registro de los datos de voltaje, corriente, potencia, factor de potencia, THD, etc., se hizo uso de un banco de resistencias y banco de capacitores tal y como se

muestra en la figura, obteniendo así los valores por parte del fluke y de nuestro prototipo; en un escenario con carga desbalanceada.

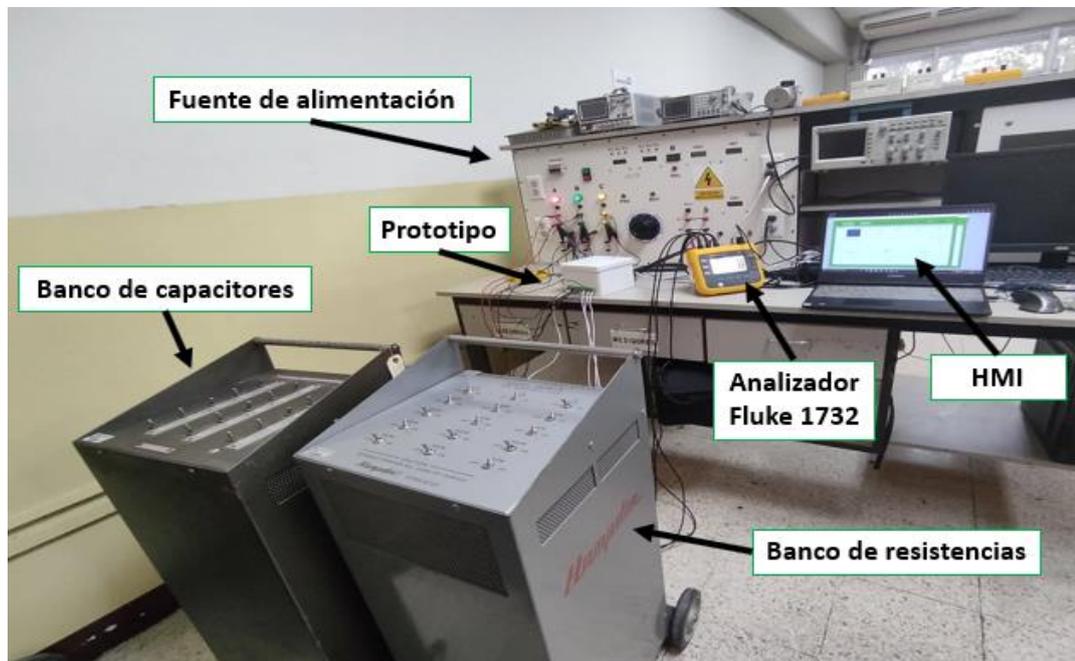


Figura 22: Prueba de registro de datos

3.4 Contrastación de voltaje de línea.

Con el patrón Fluke 1732, se procedió a registrar los valores de voltaje de línea en la carga trifásica, en la Figura 23 se puede observar que el voltaje de la fase A tiene una mejor semejanza con la tendencia registrada por el patrón, esto se pudo comprobar con el análisis de errores donde la distribución normal del error determino que la media aritmética del error fue de -0.08305% y el verdadero error para lo que sería el sensor de la fase A es de 0.2834% de error. Para el sensor de la fase B, se obtuvo una media de -0.8097% de error con un error definido para este sensor de 0.2785% . Por último, para la fase C se obtuvo una media de 0.7312% de error con un error definido para este sensor de 0.4389% .

Con estos resultados se puede concluir que los sensores de voltaje tienen una alta fluctuación en los valores registrados lo que se puede observar en la graficas de tendencia pero mantienen un bajo porcentaje de error de máximo 0.4389% con respecto al valor medido.

Cabe mencionar que la prueba realizada y en el software diseñado se incluyó un filtro digital pasa bajo de tipo FIR, por lo que las fluctuaciones no son producto del ruido electrico presente en la red.

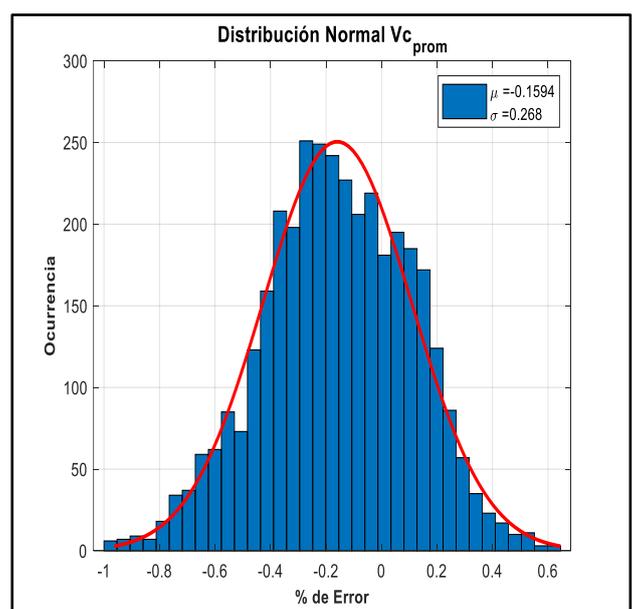
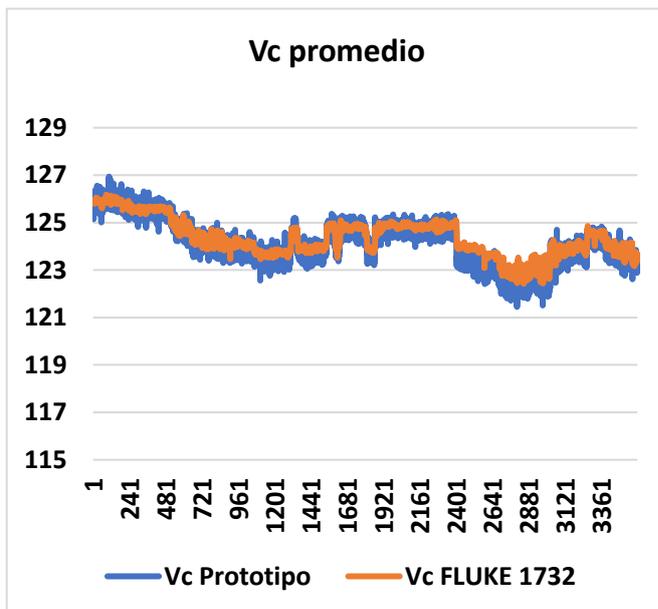
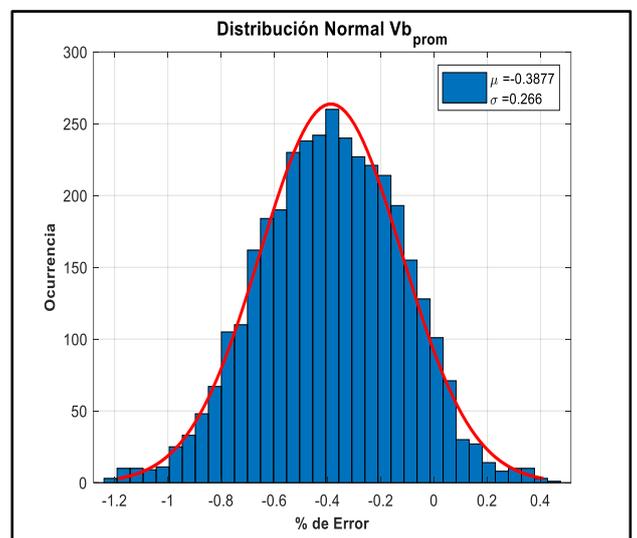
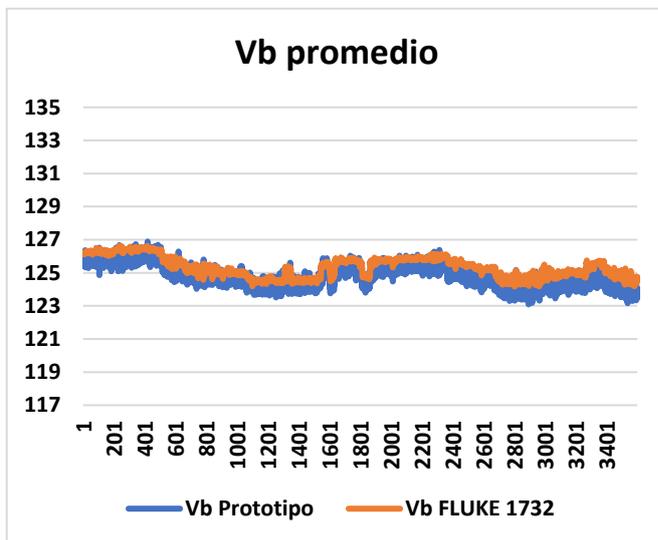
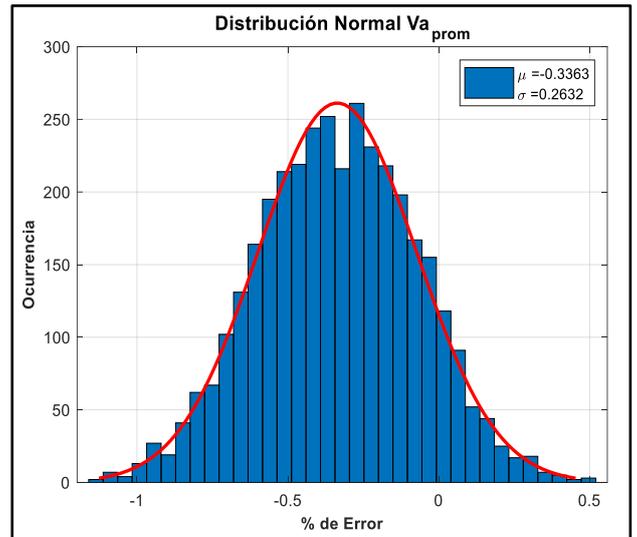
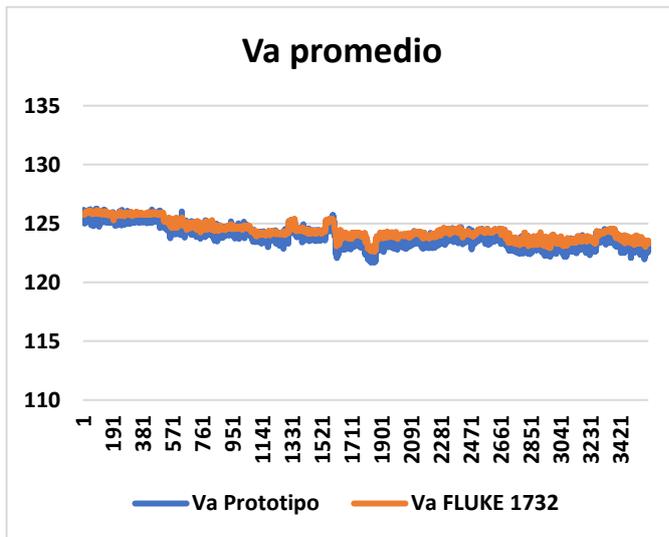


Figura 23: Gráfica de tendencia y distribución de error del voltaje

3.5 Contrastación de corriente de línea.

Para la corriente de línea se puede observar en la Figura 24, que el prototipo implementado del analizador de redes sigue la misma tendencia de la corriente de línea registrada por el patrón. El porcentaje de error para el sensor de corriente de la fase A se sitúa en 0.3859 % con una media de 0.1035 % de error con respecto al valor medido. El porcentaje de error para el sensor de corriente de la fase B se sitúa 0.2888 % con una media de 0.6814 %. Por último para la fase C, se tuvo un error de 0.4485 % con una media de 0.6596 %. Los resultados muestran que el máximo fue de 0.6596% para el sensor de la fase C, esto es debido a que la fase C fue la que se produjo más cambios de corrientes, sin embargo el porcentaje de error aún se sitúa en un buen valor.

3.6 Contrastación de potencias por fase.

Como la potencia es medida indirectamente, ya que se requiere medir los voltajes y las corrientes de línea; debido a la propagación de errores de las medidas descritas anteriormente, el porcentaje de error será ligeramente mayor, esto se puede evidenciar en la gráfica de la tendencia y distribución del error de la Figura 25, donde el máximo error que se obtuvo fue de 1.07% con respecto a la potencia medida en la fase B, mientras que para la fase A y C se obtuvo un error de 0.78% y 0.52% respectivamente.

3.7 Contrastación de potencias totales y factor de potencia.

Para la potencia total se obtuvo un error de 0.44% con respecto a la potencia total real, para la potencia aparente total se obtuvo un error de 0.259%, mientras que para el factor de potencia distorsionado total se obtuvo un error de 0.09% con respecto a la variable medida del instrumento patrón.

Se puede observar en Figura 26, que el prototipo del analizador de redes trifásico sigue la tendencia de la potencia activa total, potencia aparente total y del factor de potencia total, registra correctamente los cambios de la red eléctrica.

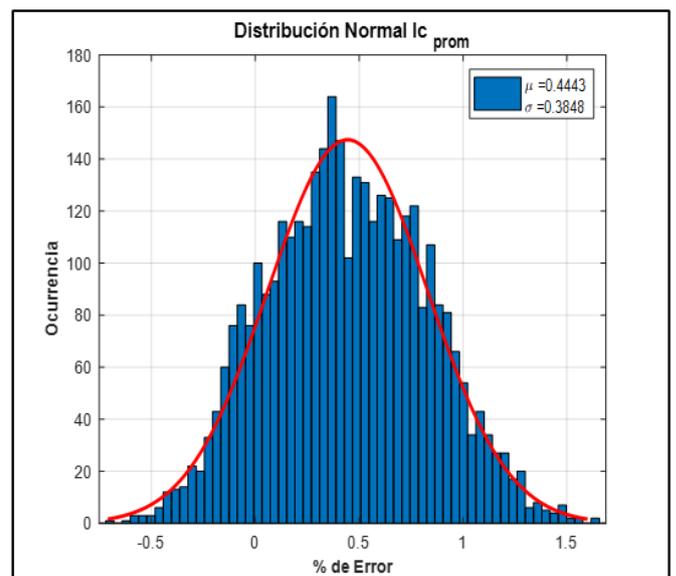
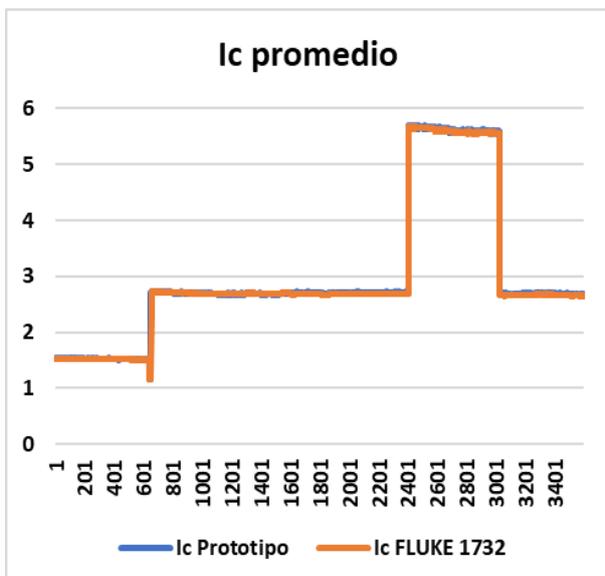
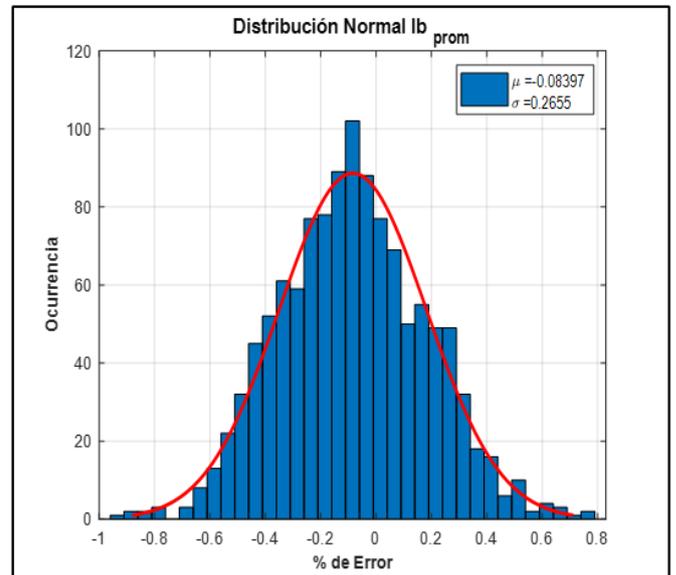
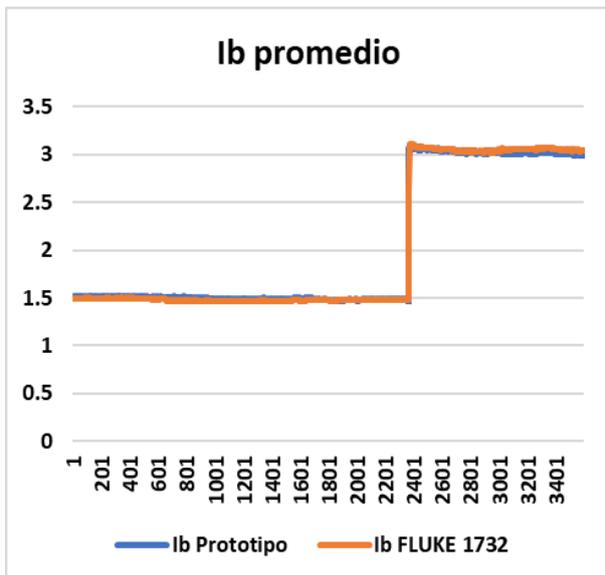
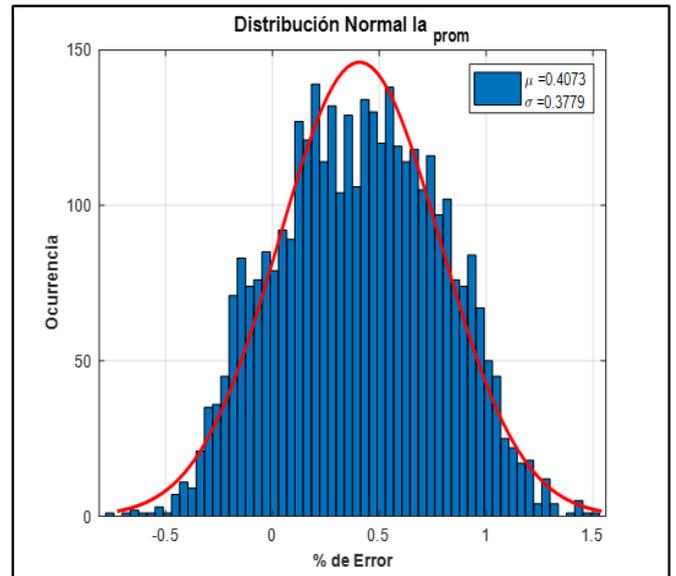
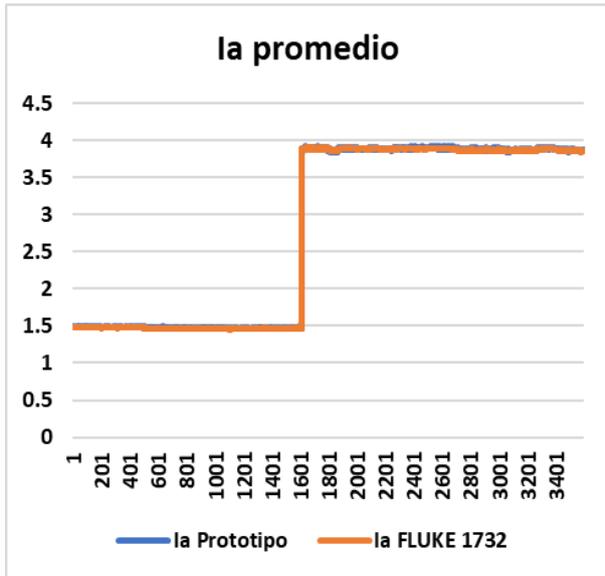


Figura 24: Grafica de tendencia y distribución de error de la corriente.

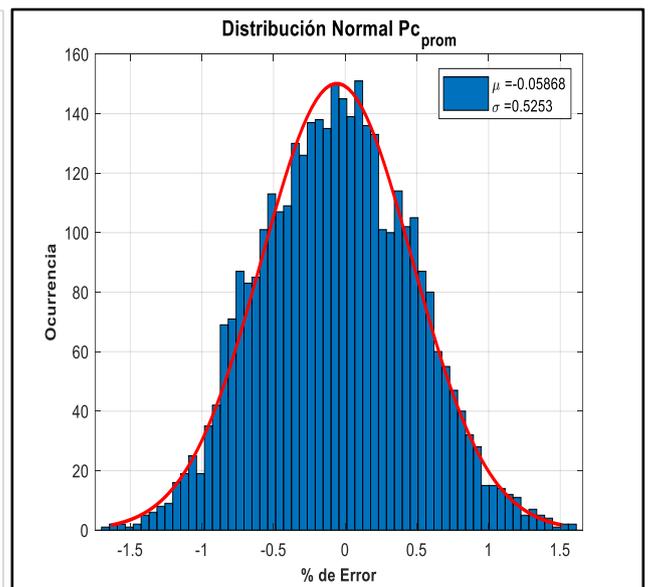
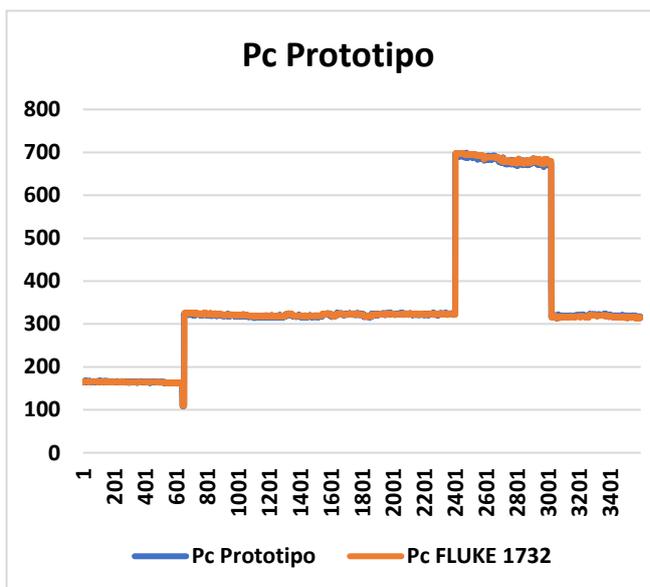
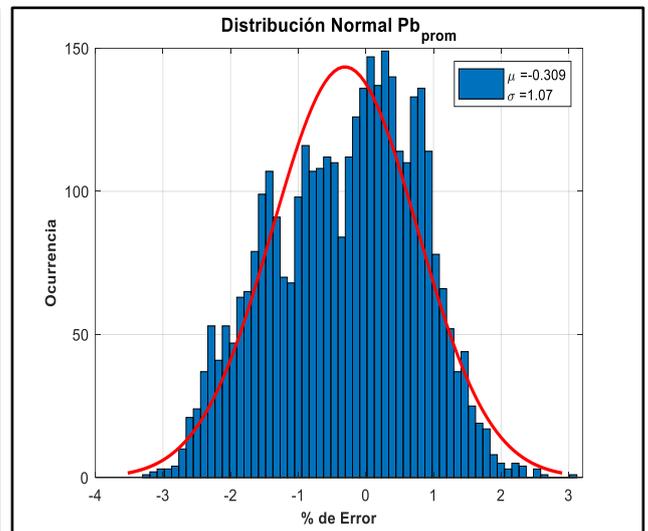
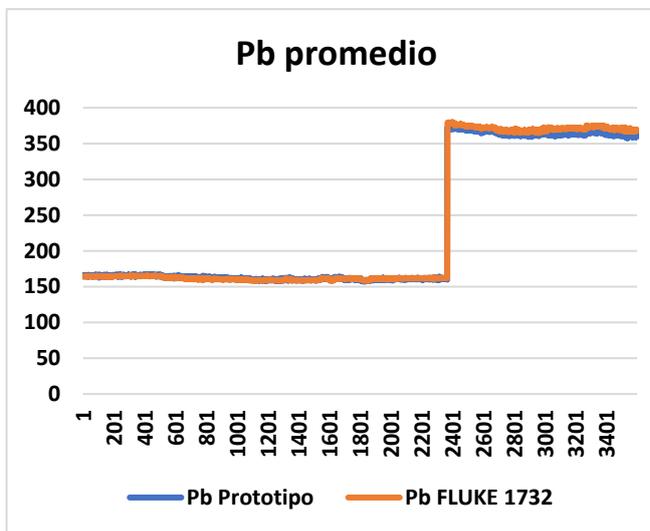
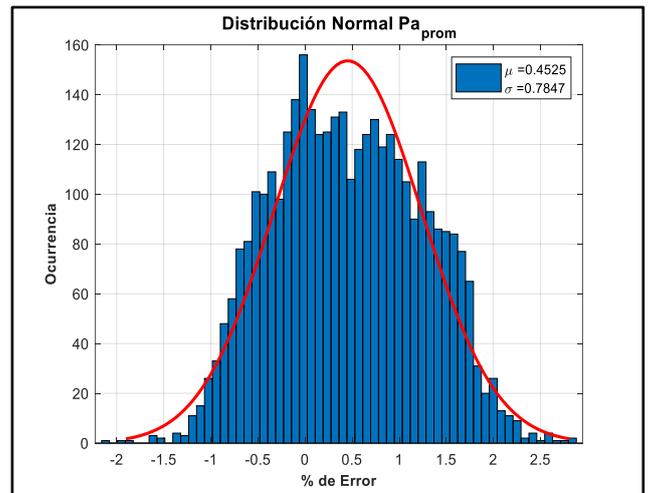
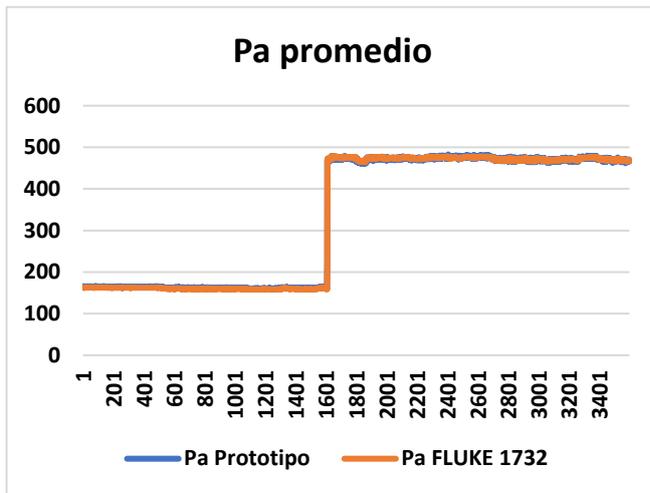


Figura 25 Grafica de tendencia y distribución de la potencia.

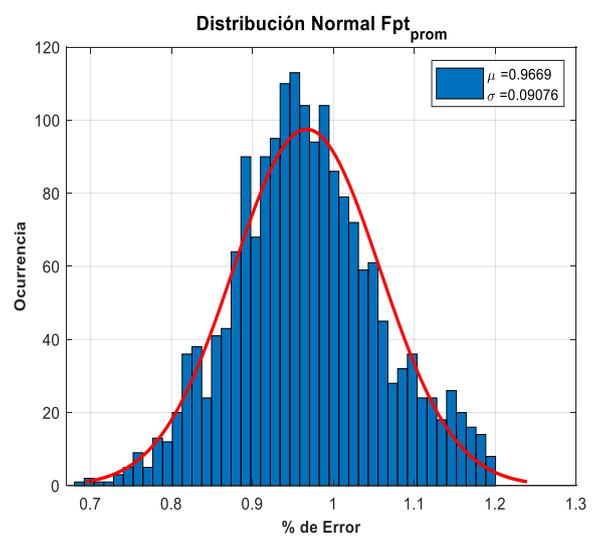
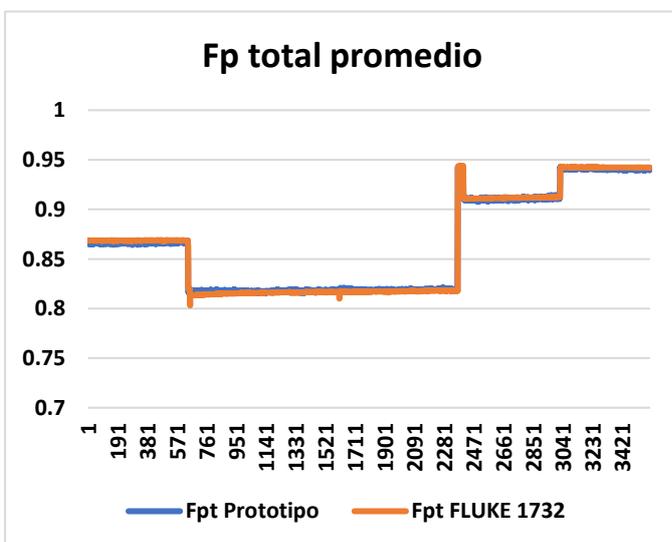
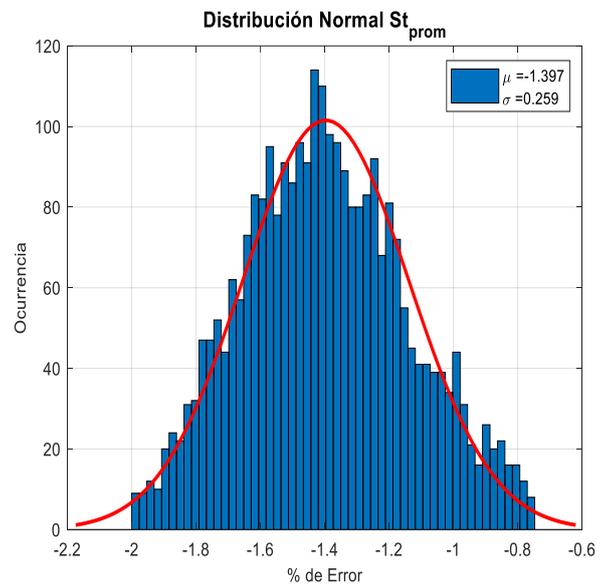
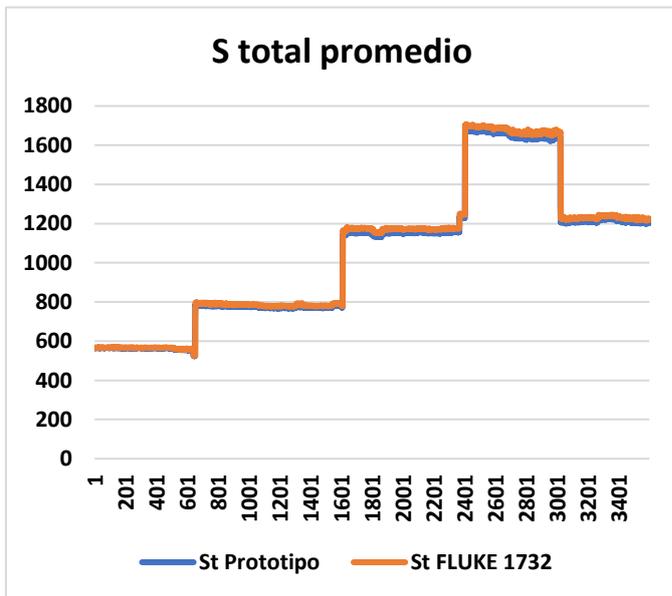
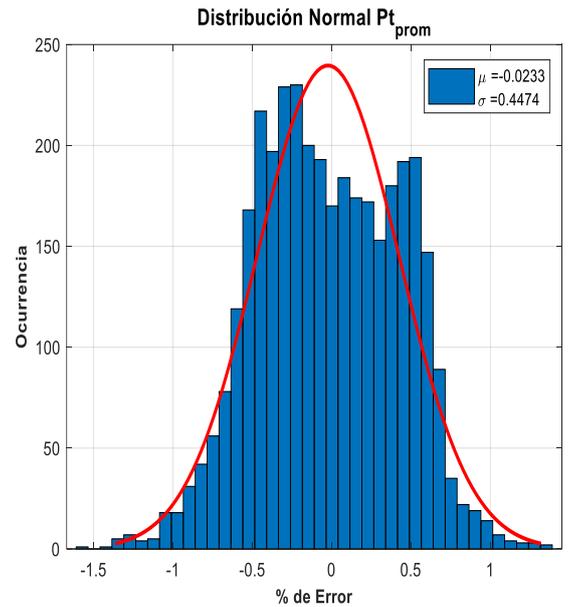
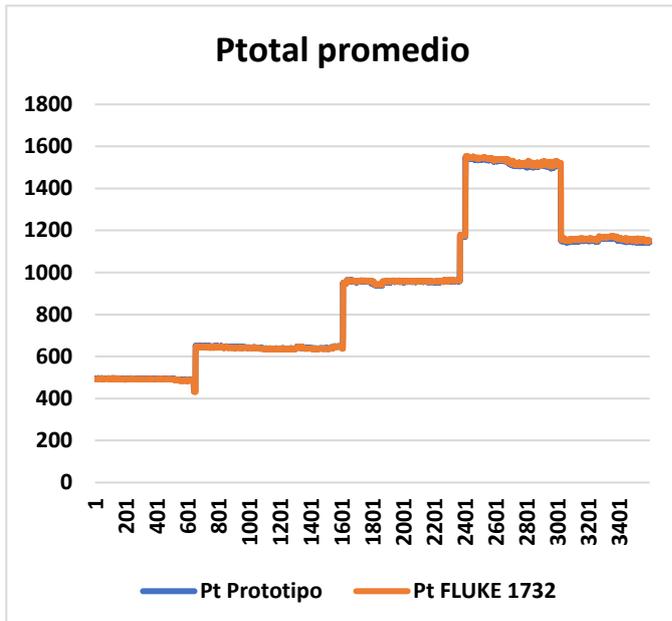


Figura 26: Grafica de tendencia y distribución de Pt, St y Fpt.

3.8 Corriente de arranque de un motor.



Figura 27: Equipos usados para la medición de corriente de arranque.

Tabla 5: Datos de placa del motor para prueba de corriente de arranque.

| Datos de placa del Motor Kato | |
|-------------------------------|------------|
| Parámetro | Valor |
| Voltaje | 220/440 V |
| Amperaje | 15.8/7.9 A |
| Potencia | 5 HP |
| Frecuencia | 60 Hz |
| Rpm | 1762 |

Para la medición y contratación de la corriente de arranque se utilizó un equipo patrón para capturar el evento de la corriente de arranque, el equipo utilizado fue el Fluke 43B. Mediante el modo de funcionamiento de corriente de arranque para el prototipo y el patrón se procedió a hacer un arranque directo de un motor de 5 HP donde los datos de placa están detallados en la **Tabla 5**. Una vez capturada la forma de onda de la corriente se procedió a hacer una comparación de las mediciones. En la Figura 28 y Figura 29 se puede observar que el pico de la corriente medida desde el prototipo fue de 93.44 A, mientras que desde el equipo patrón fue de 94 A dando un error de 0.59%, mientras que el tiempo de duración del arranque medido desde el prototipo fue de 0.70 s, y del equipo patrón fue de 0.78 s.

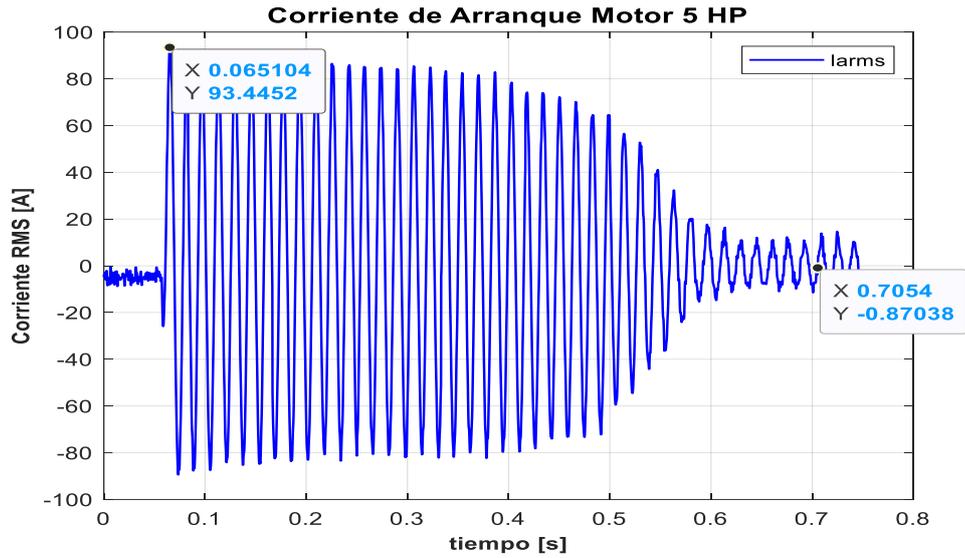


Figura 28: Corriente de arranque de la Fase A medida desde el prototipo.

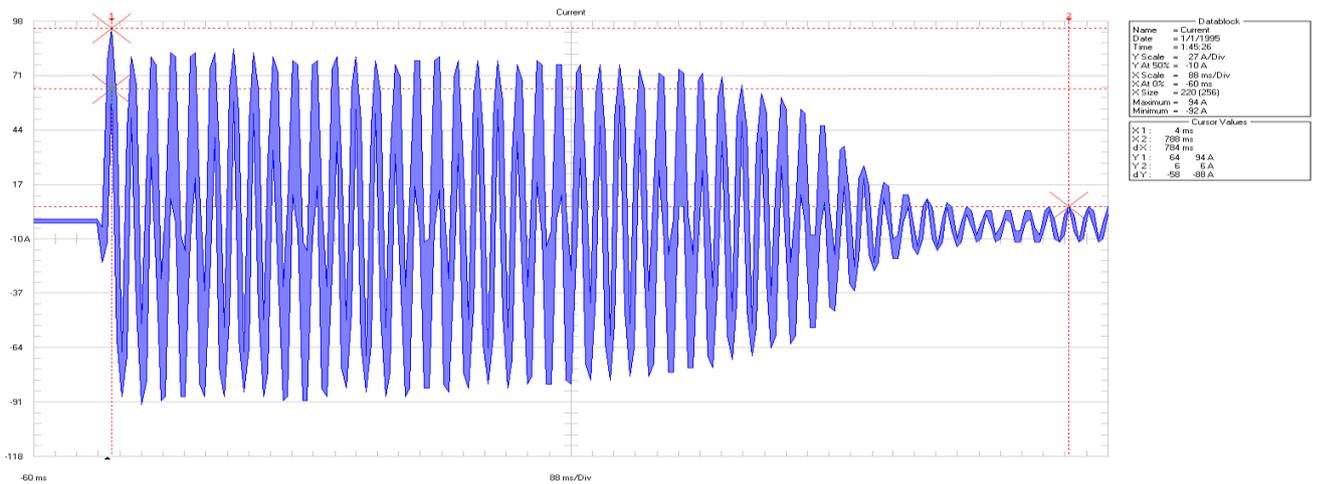


Figura 29: Corriente de arranque de la Fase A medida desde el Fluke 43B.

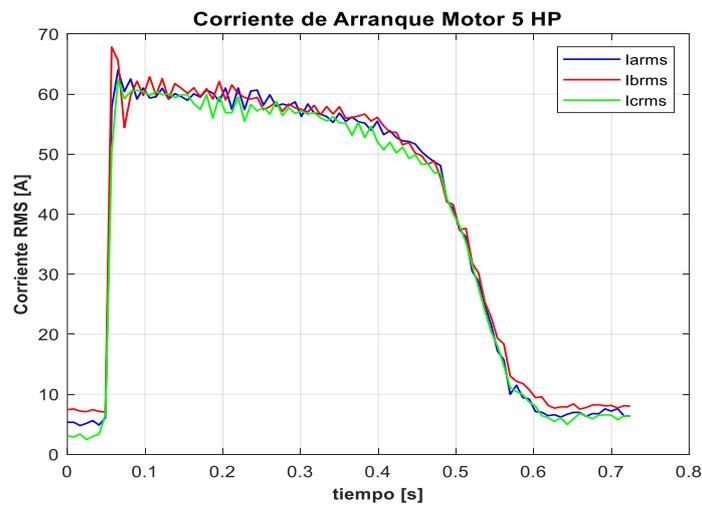


Figura 30: Corriente RMS de cada 1/2 ciclo de las corrientes medida desde el prototipo.

3.9 Corriente de arranque de un transformador.

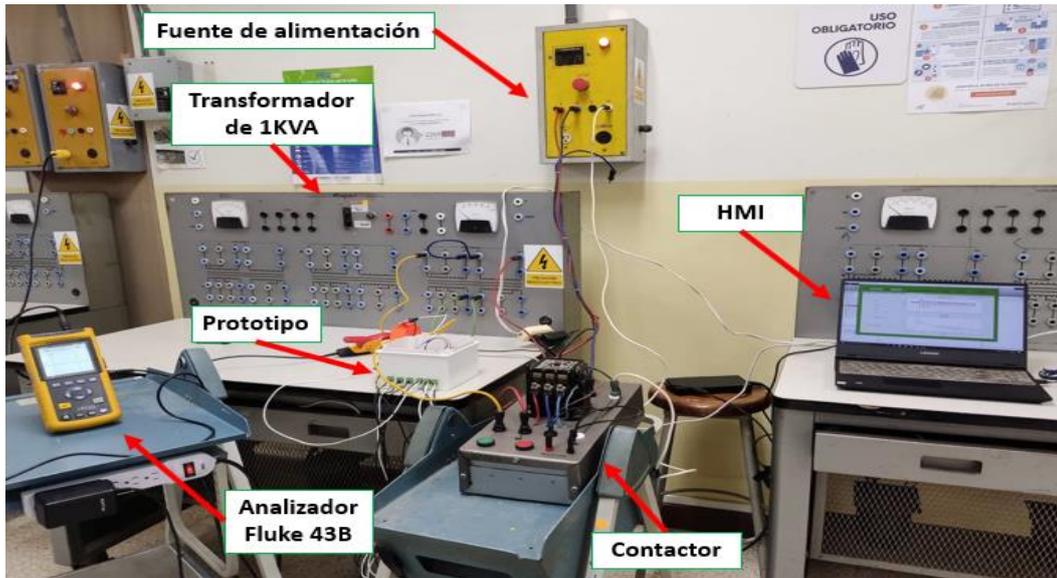


Figura 31: Equipos usados para la medición de la corriente de arranque del transformador.

Tabla 6: Datos de placa del transformador para prueba de corriente de arranque

| Datos de placa del Transformador | |
|----------------------------------|-----------|
| Parámetro | Valor |
| Voltaje | 240/120 V |
| Potencia | 1 KVA |
| Frecuencia | 60 Hz |

Para la medición y constatación de la corriente de arranque de un transformador, se utilizó el transformador de 1kVA del banco de transformadores de prueba del laboratorio de máquinas eléctricas de la ESPOL. El transformador que se utilizó consta de las siguientes características descritas en la Tabla 6. Una vez conectado el prototipo y el equipo patrón se procedió a conectar a transformador al voltaje nominal para que los equipos puedan capturar y registrar la corriente de arranque, las cuales se muestran en la Figura 32 y Figura 33, donde se puede observar que el pico de corriente registrada en el prototipo fue de 35 A, mientras que el del equipo patrón fue de 40 A. Esta diferencia de 5 A es debida a que en esta prueba no se utilizó un TC, por lo que el sensor de corriente del Arduino se saturó, mientras que el tiempo de arranque si fue estimado correctamente siendo de aproximadamente 0.25 s.

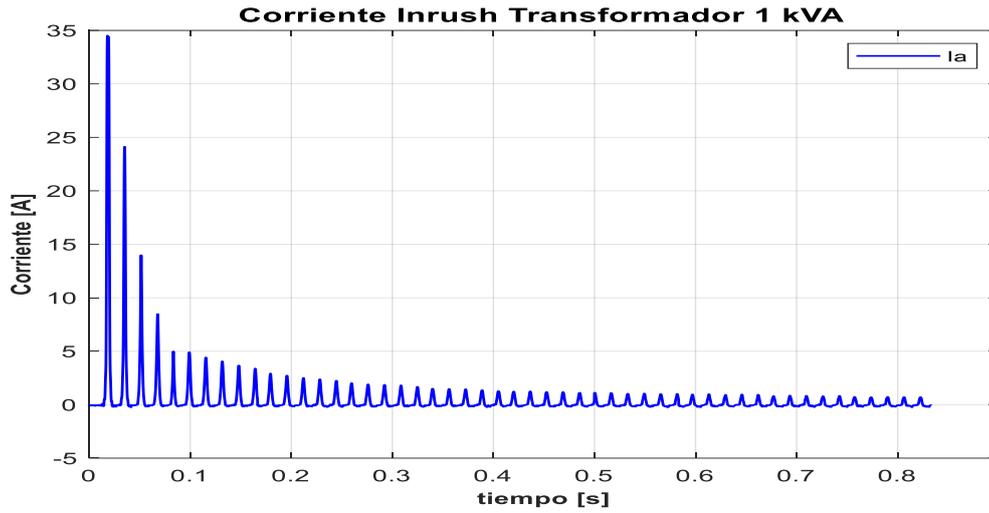


Figura 32: Corriente de arranque del transformador medida desde el prototipo.

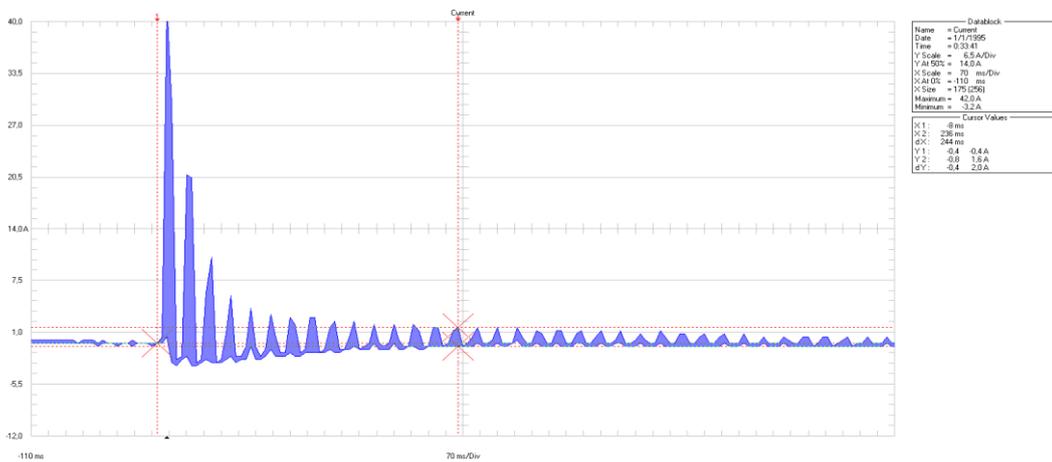


Figura 33: Corriente de arranque del transformador medida desde el Fluke 43B.

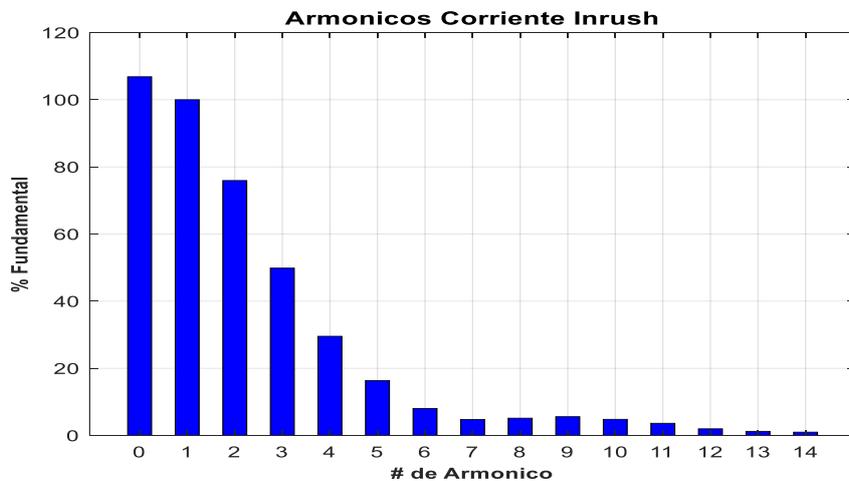


Figura 34: Armónicos presentes en la corriente de arranque medida desde el prototipo.

3.10 Análisis de Costos.

El costo del presupuesto se expone en la tabla, en la cual podemos observar el detalle de la cantidad y valor de cada material utilizado.

Tabla 7: Costos de los elementos usados para la implementación.

| Material | Cantidad | Costo Unitario | Costo |
|--|-----------------|-----------------------|--------------|
| Arduino Mega 2560 | 1 | \$ 25,00 | \$ 25,00 |
| Sensor de voltaje ZMPT10 | 3 | \$ 7,50 | \$ 22,5 |
| Sensor de corriente ACS712 | 3 | \$ 4,70 | \$ 14,1 |
| Arduino Ethernet W5500 | 1 | \$ 15,00 | \$ 15,00 |
| Conectores tipo lagarto | 12 | \$ 0,15 | \$ 1,80 |
| Cable Ethernet 1 metro | 1 | \$ 1,00 | \$ 1,00 |
| Fuente de alimentación 9V | 1 | \$ 5,00 | \$5,00 |
| Caja Paso | 1 | \$ 6,00 | \$ 6,00 |
| Software Analizador de Redes Trifásico | 1 | \$ 200,00 | \$200,00 |
| Precio Final | | | \$ 290,4 |

Un analizador de redes de alta gamma como el analizador Fluke 1732 está costando entre \$3000 y \$4000, por lo que se estima el analizador implementado representa un ahorro del 90% por lo menos, debido a los elementos de bajo costo y que se encuentran disponibles dentro del mercado ecuatoriano, por lo que hace que la reparación sea muy fácil y rápida.

3.11 Limitaciones de hardware encontradas.

Durante las pruebas realizadas, se encontró que existe un tiempo muerto donde la señal analógica no es muestreada debido a que existe un cuello de botella en la comunicación ethernet del shield de Arduino, debido a que la programación de los microcontroladores es secuencial, por lo que al momento de ejecutar la línea donde se envían las señales muestreadas, este se demora aproximadamente 16.6 ms es decir un ciclo de la señal de 60 Hz. Esto es debido a que el chip W5500 de ethernet para la placa Arduino Mega, junto con la librería Ethernet2.h solo es capaz de llegar a una velocidad de comunicación de 300 kbps.

CAPÍTULO 4

4. CONCLUSIONES Y RECOMENDACIONES

El analizador de redes trifásico implementado usando la placa Arduino Mega y el shield de Ethernet, además de los sensores de corriente de efecto Hall ACS712 y los sensores de voltaje ZMPT101B en conjunto con el software del HMI para la visualización y registro de datos que fue desarrollado en la herramienta App Designer de MATLAB, presento un bajo porcentaje de error para las mediciones de voltaje, corriente, potencia activa, potencia reactiva, potencia aparente y factor de potencia al hacer una constatación con un analizador de redes de la marca FLUKE modelo 1732. Esto implica que es factible hacer un analizador de redes trifásico con elementos de bajo costo y que se puede encontrar fácilmente en el mercado ecuatoriano, lo que además permite que su reparación sea fácil, a diferencia de un analizador de alta gamma donde la mayoría de estos equipos son importados por lo que hace que su reparación no sea fácil.

El algoritmo del software basado en la norma IEEE 1459 calculó correctamente las variables eléctricas medidas además de mostrar mucha información importante como lo es la forma de onda, diagrama de barras de armónicos presentes en la red, diagrama fasorial del voltaje y la corriente, así como registrar en varios intervalos de tiempo que se pueden configurar.

Para un futuro proyecto se puede mejorar la conectividad del analizador debido a que el shield de Ethernet limita la velocidad de comunicación a 300 kbps, lo que hace que se pierda un ciclo de las señales de voltaje y corriente debido al cuello de botella presente en la comunicación. Además, se puede mejorar la precisión de los armónicos y de la tendencia del voltaje con un mejor sensor de voltaje o cambiando la placa Arduino por un mejor sistema de adquisición que tengo un ADC de alta precisión.

BIBLIOGRAFÍA

- [1] B. S. K. A.-H. Bhim Singh, POWER QUALITY PROBLEMS AND MITIGATION TECHNIQUES, WILEY, 2022.
- [2] T. Kusko, Power Quality in Electrical Systems, McGraw-Hill, 2007.
- [3] Ó. Torrente Artero, ARDUINO Curso práctico de formación, México: Alfaomega, 2013.
- [4] M. electronics, «ARDUINO.cl,» Ingeniería MCI Ltda, [En línea]. Available: <https://arduino.cl/>.
- [5] M. Fezari, «ResearchGate,» Octubre 2018. [En línea]. Available: <https://www.researchgate.net>.
- [6] J. Ikiss, «Sistema de adquisición de datos con ESP32,» Catalunya, 2020.
- [7] C. Peña, Arduino IDE: configuración y uso, Buenos Aires: Plando S.A., 2020.
- [8] C. J. Carrillo González, Fundamentos del Análisis de Fourier, Vigo, 2003.
- [9] E. Harper, El ABC de la calidad de energía eléctrica, México: Limusa, 2004.
- [10] M. Mohammad A.S. y F. Ewald F., Power Quality in Power Systems and Electrical Machines, California: Elsevier, 2008.
- [11] Ö. YAZLIK, «DESIGN AND DEVELOPMENT OF A SIMPLE POWER QUALITY MONITOR,» 2014.
- [12] G. Bollen, Signal Processing of Power Quality Disturbances, Wiley, 2006.

APÉNDICES

APÉNDICE A

```
// LIBRERIAS USADAS
```

```
#include <SPI.h>
```

```
#include <Ethernet3.h> // antes V2.0.0
```

```
#include "avdweb_AnalogReadFast.h"
```

```
////////////////////////////////////
```

```
//////// VARIABLES PARA EL SISTEMA DE ADQUISICION////////
```

```
int Contador = 0; // primer contador muestreo.
```

```
unsigned long previousTime = micros ();
```

```
#define FrecSample 6144 // Hz FrecSample=Muestras* entero (10,12,,...) multiplo
```

```
#define TimeSample 1000000/6144 //uS 1/FrecSample FrecSample< a 10000 Hz que  
es el limite del ADC arduino
```

```
#define Muestras 512 // Numero de muestras, debe ser potencia de (2) - 1 por el index  
0 del array
```

```
unsigned int va[Muestras];
```

```
unsigned int vb[Muestras];
```

```
unsigned int vc[Muestras];
```

```
unsigned int ia[Muestras];
```

```
unsigned int ib[Muestras];
```

```
unsigned int ic[Muestras];// 768
```

```
////////////////////////////////////
```

```
//// MODULO ETHERNET////////
```

```
byte mac[] = {0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED}; // mac predeterminado
```

```
IPAddress ip(169, 254, 108, 245); // IP DEL SERVER ARDUINO
```

```
EthernetServer miservidor(100); // arduino como servidor PORT 100
```

```
EthernetClient micliente;
```

```
////////////////////////////////////
```

```
void setup() {
```

```
  // put your setup code here, to run once:
```

```
  Ethernet.init(2); // 2 socket
```

```
  Ethernet.begin(mac, ip); // CONFIGURACION ETHERNET
```

```
  analogReference(EXTERNAL);
```

```
  pinMode(A0, INPUT);
```

```
  pinMode(A1, INPUT);
```

```
  pinMode(A2, INPUT);
```

```
  pinMode(A3, INPUT);
```

```
  pinMode(A4, INPUT);
```

```
  pinMode(A4, INPUT);
```

```
}
```

```
void loop() {
```

```
  // put your main code here, to run repeatedly:
```

```
// unsigned long t1 = micros();
```

```
////////// Paso 1 , Muestreo de señales//////////
```

```
while (Contador < Muestras) {
```

```
unsigned long currentTime = micros();// Tiempo actual para contar el periodo de
muestreo
```

```
if (currentTime - previousTime >= TimeSample) { // Periodo de muestreo
```

```
va[Contador] = analogReadFast(A0);
```

```
ia[Contador] = analogReadFast(A3);
```

```
vb[Contador] = analogReadFast(A1);
```

```
ib[Contador] = analogReadFast(A4);
```

```
vc[Contador] = analogReadFast(A2);
```

```
ic[Contador] = analogReadFast(A5);
```

```
previousTime = currentTime;
```

```
Contador = Contador + 1;
```

```
}
```

```
}
```

```
Contador = 0; // Se reinicia el contador
```

```
///// ENVIO DE DATOS VIA TCP/IP /////
```

```
micliente = miservidor.available(); \
```

```
micliente.flush();
```

```
if (micliente) {
```

```
if (micliente.connected()) {
```

```
if (micliente.available()) {
```

```
micliente.write((const uint8_t *) & va, sizeof(va)); // 512 bytes, 1 int = 2bytes 256
intarray=512 bytes
```

```
micliente.write((const uint8_t *) & vb, sizeof(vb));
```

```
micliente.write((const uint8_t *) & vc, sizeof(vc));
```

```
micliente.write((const uint8_t *) & ia, sizeof(ia));
```

```
micliente.write((const uint8_t *) & ib, sizeof(ib));
```

```
micliente.write((const uint8_t *) & ic, sizeof(ic));
```

```
}
```

```
}
```

```
}
```

```
}
```