

*Informe de
Práctica Profesional Supervisada
en IEDS/CNEA*

TEMA

Eficiencia Energética

TÍTULO

ANALIZADOR DE REDES DE BAJO COSTO

AUTORES

Sebastián Busch - Denis R. Heredia

ALUMNOS DE LA CARRERA

Ingeniería Electrónica

**Escuela de Producción, Tecnología y Medio Ambiente
Universidad Nacional de Río Negro (UNRN)**

DOCENTE A CARGO

Javier Areta

SUPERVISORES EN EL IEDS

Mariela Lescano

Carlos González Ferrari

Año 2021

I E D S

**INSTITUTO DE ENERGÍA Y
DESARROLLO SUSTENTABLE**

C N E A

**Comisión Nacional de
Energía Atómica**

CONTENIDO DEL PRESENTE DOCUMENTO

- **INFORME TÉCNICO (30 páginas)**
- **DATASHEET (8 páginas)**
- **MANUAL DEL USUARIO (19 páginas)**

UNIVERSIDAD NACIONAL DE RÍO NEGRO
ESCUELA DE PRODUCCIÓN, TECNOLOGÍA Y MEDIO AMBIENTE



INGENIERÍA ELECTRÓNICA

2021

Práctica Profesional Supervisada: “Analizador de redes de bajo costo”

ALUMNOS: Busch Sebastián; Heredia Denis R.
DOCENTE A CARGO: Areta, Javier.
TUTOR: Carlos, González Ferrari; Mariela, Lescano.
FECHA DE ENTREGA: 2 de noviembre del 2021

Índice

PRÓLOGO	3
INTRODUCCIÓN	4
OBJETIVOS	5
Objetivo general	5
Objetivos específicos	5
Metas deseables	5
CRONOGRAMA DE ACTIVIDADES	6
DESARROLLO DE LA SOLUCIÓN	7
1.	7
2.	7
3.	8
4.	8
5.	9
6.	12
7.	14
8.	15
9.	16
10.	17
11.	21
12.	22
PROBLEMAS ENCONTRADOS DURANTE EL DESARROLLO	21
POSIBLES MEJORAS	22
PRESUPUESTO APROXIMADO	23
COMPARATIVA	24
CONCLUSIÓN	25

Índice de tablas

Tabla 1 cronograma	5
Tabla 2 Comparación de mediciones varias entre analizador fluke vs analizador de desarrollo propio	13
Tabla 3 Comparación de medición trifásica entre analizador fluke vs analizador de desarrollo propio	13
Tabla 4 Comparación del error porcentual en función de la corriente rms	14
Tabla 5 presupuesto	22
Tabla 6 comparativa funcional entre los analizadores	23
Tabla 7 comparativa dimensiones físicas	23

PRÓLOGO

El presente trabajo se realizó bajo el marco de Practica Profesional Supervisada de la Universidad Nacional de Río Negro, con el objetivo de aproximar a los estudiantes a una situación laboral real bajo un ambiente controlado.

El mismo se llevó a cabo bajo la dirección de los tutores Dr. Ing. Carlos González Ferrari y la Ing. Mariela Lescano. El objetivo principal fue el diseño y construcción de un analizador de redes de bajo costo destinado a ser instalado en viviendas.

El analizador de redes debe ser capaz de medir la tensión y la corriente rms como así también el desfase de estas ondas entre sí. Luego, mediante el procesamiento de estas señales, determinar los distintos tipos de potencia que se están consumiendo y finalmente exportar todos estos datos a la nube para su posterior análisis. El objetivo final es caracterizar patrones de consumo energético, detectar problemas y plantear estrategias de mejora para corregirlos.

El destinatario de este proyecto en primera instancia es la Comisión Nacional de Energía Atómica (CNEA). Dicho organismo se encargó de brindar los materiales y recursos necesarios para el proyecto, además de su posterior replicación e instalación.

INTRODUCCIÓN

A partir del voltaje V_{rms} y la corriente rms I_{rms} se definen tres tipos de potencias, siendo estas: Activa, Reactiva y Aparente. La primera de ellas es la que se puede aprovechar directamente, también llamada real o verdadera, es utilizada por elementos de índole resistivos y su unidad es el vatio (W). Esta se calcula como:

$$V_{rms} * I_{rms} * \cos(\varphi)$$

Siendo: φ la diferencia de grados en radianes de la onda de corriente respecto de la de tensión.

La potencia reactiva es generada por las bobinas y condensadores, esta no realiza un trabajo efectivo sino que fluctúa por la red entre el generador y los receptores, su unidad es el voltamperio reactivo (Var) y se calcula como:

$$V_{rms} * I_{rms} * \sin(\varphi)$$

La potencia aparente es la total consumida por la carga, su unidad de medida es el voltamperio (VA) y se calcula directamente como:

$$V_{rms} * I_{rms}$$

Desde ahora definiremos al $\cos(\varphi)$ como Power Factor o Factor de Potencia.

- si el argumento del Ángulo es positivo, podemos afirmar que la carga es de índole capacitiva y pueden existir problemas en las líneas de transmisión; debido a que se incrementa el voltaje recibido respecto al enviado.

-si el argumento es negativo diremos que la carga tiene naturaleza inductiva, se observa un aumento del amperaje produciendo un desperdicio de potencia en las cargas lineales, estas son aquellas que su onda de tensión y corriente están relacionadas por un factor proporcional. En las cargas no lineales, es decir aquellas en las que su impedancia varía con la tensión aplicada, se observará ruido armónico. Además de que esto reducirá la vida útil de los conductores, crea fluctuaciones de tensión e incrementa el costo operacional de la transmisión eléctrica y del sistema de distribución.

-si el argumento es muy pequeño diremos que la carga es puramente resistiva y no tendremos problemas.

Gracias a la medición del desfase de la onda será posible identificar situaciones de consumo energético irregular, caracterizarlas y luego evaluar estrategias de mejora a fin de incrementar la eficiencia energética del sitio de instalación del analizador de redes.

OBJETIVOS

Objetivo general

- Brindar al pasante una experiencia de trabajo real, dónde pueda familiarizarse con los procesos de investigación y el trabajo de laboratorio dentro de un ambiente controlado.
- Diseñar y construir un Analizador de redes de bajo costo.

Objetivos específicos

- Adquirir directamente valores de corriente y tensión eficaz, el corrimiento entre las mismas y las potencias de forma analítica.
- Exportar los datos a la nube vía serial, wifi / Ethernet.
- Crear un prototipo funcional, capaz de operar con corrientes trifásicas y monofásicas.
- Operar con corrientes de hasta 30 A.
- Lograr un margen de error en las mediciones menor al 15%
- Elaborar la documentación específica del proyecto, para su posterior replicación.
- Implementar una interfaz gráfica de usuario para monitorear las mediciones efectuadas en tiempo real.
- Monitorear en tiempo real de manera remota por internet los valores medidos por el analizador de redes.

Metas deseables

- Diseñar y construir una placa PCB.
- Construcción de prototipo final funcional, listo para su instalación.

CRONOGRAMA DE ACTIVIDADES

La normativa de la Universidad Nacional de Río Negro (UNRN) establece que, el Programa de Práctica profesional supervisada (PPS) debe poseer un mínimo de 200 horas de trabajo y un máximo de 400. A continuación, se detallan las horas de trabajo dedicadas a las actividades realizadas.

Horas	Actividades realizadas
10	Adquisición de conceptos básicos y planteo de posibles soluciones.
20	Elección de componentes.
55	Adquisición de variables deseadas de forma directa.
5	Adquisición de variables deseadas de forma indirecta.
60	Almacenamiento y exportación de la información obtenida por el dispositivo.
60	Diseño del primer prototipo funcional.
20	Diseño circuito de protección.
20	Pruebas de campo.
30	Diseño de software.
60	Diseño PCB.
60	Construcción y puesta a punto de un modelo robusto.
400	TOTAL

Tabla 1 Listado de tareas y estimación de tiempos de ejecución

DESARROLLO DE LA SOLUCIÓN

1. Adquisición de conceptos básicos y planteo de posibles soluciones

Se realizó un relevamiento bibliográfico exhaustivo acerca de la temática. Este se centró en adquirir conocimiento acerca de corriente alterna tanto monofásica y trifásica, en cómo se calculan los distintos tipos de potencia y por qué aparecen. Además, comprender cuáles son sus efectos y los valores teóricos que deberían obtenerse al conectar distintas cargas conocidas de prueba.

Una vez comprendidos los conceptos, se estudiaron los distintos métodos que podrían utilizarse para obtener las variables de interés. En este sentido, se optó por medir directamente la corriente, la tensión y el desfase entre sus respectivas ondas, para luego calcular de forma analítica el resto de las variables de interés. Se utilizó Gitlab¹ para tener un seguimiento de los cambios realizados durante el proceso.

2. Elección de los componentes

Se utilizaron sensores invasivos² debido a que estos poseen mejor resolución. Luego de evaluar las distintas opciones disponibles en el mercado, se optó por usar un sensor ZMPT101b (anexo 1) para medir la corriente de cada fase y el sensor ACS712 (anexo 2) para la tensión. Los datos emitidos por éstos, fueron procesados en una placa de desarrollo arduino mega (anexo 3). La cual a su vez se encarga de enviar los datos mediante un ethernet shield (anexo 4), un módulo Esp 12 D1 Mini V2 (anexo 5) o vía serial según sea conveniente. Para Alimentar el circuito se utilizó un transformador step down 220v AC a 12v DC. Además, se utilizó un amplificador operacional LM317 (anexo 6) para crear un voltaje de referencia estable, compuertas XOR CD4030 (anexo 7) y operacionales usados como comparadores LM324 (anexo 8) con el fin de identificar el desfase entre la corriente y la tensión. Mientras que para el circuito de protección se optó por utilizar una térmica trifásica de 32 A. Los valores medidos serán mostrados a través de un oled de 128 x 64 píxeles. (anexo 9)

¹ (2021) Gitlab recuperado de <https://gitlab.com/>

² Que un sensor sea invasivo implica que el mismo se debe conectar directamente en serie (en caso de corriente) para lograr hacer la medición. La diferencia con los sensores no invasivos es que son sin contacto.

3. Adquisición de las variables deseadas de forma directa

Los sensores ZMPT y ACS están directamente conectados a la línea de forma invasiva, es decir debemos cortar los cables de línea y conectarlos a las borneras de los sensores. Su principio de funcionamiento se basa en el efecto hall, y los valores medidos se traducen en una señal analógica alterna de 0 a 4.6 V. La cual es leída posteriormente por los pines 8 y 9 del arduino, y luego multiplicada por el código arduino por el factor $\frac{\sqrt{2}}{2}$ a fin de obtener el valor rms de cada caso. A su vez, esta señal es comparada en un amplificador operacional lm324 con un voltaje de referencia creado por un LM317 de 2.3 V que se regula por medio de un potenciómetro. Se escogió este valor debido a que es la mitad del valor máximo que pueden arrojar los sensores, de forma que al comparar la señal analítica con su ecuador, es posible identificar si el ciclo de la onda es positivo o negativo, creando una señal cuadrada tanto para la corriente como para la tensión. Estas señales son leídas por los pines digitales del arduino y comparadas mediante software si la corriente está atrasada o adelantada respecto de la tensión.

Para cuantificar el corrimiento de las señales se optó por introducir las en una compuerta XOR. De este modo, sólo se obtiene un valor de lógica booleana 1 cuando las señales son distintas, lo cual permite medir este tiempo en la placa arduino, para luego convertirlo a radianes y así obtener los distintos valores del ángulo de desfase. Se optó por llevar adelante este procedimiento mediante hardware ya que por software las lecturas obtenidas presentaban demasiado ruido, generando datos no fiables.

A fin de reducir el efecto de los ruidos en las mediciones efectuadas por la placa arduino, se optó por tomar un conjunto de los valores medidos, almacenarlos en un vector y luego calcular su promedio. Este filtrado por promedio, permitió obtener valores más limpios y fiables.

4. Adquisición de las variables deseadas de forma indirecta

Una vez obtenido los datos primarios de las mediciones (tensión, corriente y ángulo de fase), se calculan las siguientes variables derivadas de las primarias:

$$\text{Potencia Activa} = V_{rms} * I_{rms} * \cos(\phi)$$

$$\text{Potencia reactiva} = V_{rms} * I_{rms} * \sin(\phi)$$

$$\text{Potencia aparente} = V_{rms} * I_{rms}$$

5. Almacenamiento y exportación de la información sensada

Al inicio del programa se estableció un tiempo de búsqueda de una conexión de Ethernet y/o Wifi. Una vez establecida la conexión, cada 15 minutos se suben los datos de potencias, ángulo de desfase y valores rms a: *ubidots*³ y *google sheets*⁴, con el objeto de monitorear y registrar los datos medidos y calculados en tiempo real, respectivamente.

Ubidots es una plataforma gratuita con una modalidad paga por elementos adicionales, fue elegida sobre la competencia por la facilidad que provee para el envío de datos, la integración con celulares, guardado de datos para graficarlos y la posibilidad (aunque paga) de exportar los datos, todo esto lo habilitan con su propia API lo cual facilita aún más la integración al proyecto.

La forma en que la API se conecta a nuestro proyecto es mediante un Token proporcionado por la página *ubidots*. Esto enlaza el envío de datos con el receptor de su página.

Una gran ventaja que brinda *Ubidots* es la posibilidad de agregar variables, crear tablas y personalizar cualquier gráfico. Los datos se pueden graficar para tener una salida más visual e incluso tener eventos predeterminados como por ejemplo la creación y envío de alarmas ante determinadas situaciones de consumo energético. Es posible acceder a la visualización de esta información por medio de un link (que puede hacerse privado). La plataforma permite además el monitoreo y visualización de los datos de manera pública, es decir que no es preciso disponer de una cuenta para ellos. En el caso del presente proyecto, el link de acceso es el siguiente:

["https://stem.ubidots.com/app/dashboards/public/dashboard/Az3jxRt7g43oOD6Ki55_Rl31jpA3W_SPOznDrG4qkctE"](https://stem.ubidots.com/app/dashboards/public/dashboard/Az3jxRt7g43oOD6Ki55_Rl31jpA3W_SPOznDrG4qkctE)

Por su parte, *Google sheet* no posee dificultad en cuanto al manejo por parte del usuario, de forma que los datos serán enviados junto con la fecha y hora. Esto se hace automáticamente con la placa *arduino*, de forma que los datos pueden ser extraídos con facilidad para su análisis. El archivo de *google sheet* es simplemente una hoja tipo *excel* compartida. En la misma se reciben y almacenan los datos.

Debido a la inexistencia de una API para subir los datos a *google sheet* fue necesario desarrollar un script propio que cumpla esta función, lo que hace es acceder cada cierto tiempo a una dirección. La misma, en sus parámetros tiene los datos que son enviados y así el script hace el mapeo entre los parámetros y la colocación de los mismos en el *google sheet*.

³ (2021) *Ubidots* recuperado de <https://ubidots.com/>

⁴ (2021) *GoogleSheet* recuperado de sheets.google.com

A continuación, se muestra un ejemplo de si se realiza un acceso a esta página el script va a entender cómo colocar todo:

```
1 https://script.google.com/macros/s/AKfycbyCbu3wSrjZ3RNvQI2oBuB6n3WgzDr4zHr2Pa0fsj6sjJCyDTQMlbpI9/exec?  
vrms=$vrms$&irms=$irms$  
4 5 6
```

1. es el protocolo de comunicación utilizado, https tiene encriptación por lo que es seguro
2. es el identificador del datasheet a acceder
3. la función a hacer en el datasheet
4. nombre de la variable en el datasheet
5. el valor pasado de la variable a cambiar (por simplicidad el mismo nombre)
6. el "&" quiere decir que termine con esa variable y voy a poner otra

De esta forma cada vez que el arduino haga un POST (acceso a la página) de esta cadena específica de caracteres con el valor deseado a enviar todo se pondrá automáticamente en el datasheet sin que el usuario tenga problemas de ningún tipo ya que todo esto es invisible al usuario final.

Se encontró una dificultad cuando se optó por realizarlo vía ethernet debido a que este utiliza el protocolo http para comunicarse, mientras que google sheet exige https. Ethernet shield utiliza el protocolo http porque es de menor consumo y la cantidad de datos que necesita enviar es menor, esto es debido a que no se encriptan los datos. En cambio, en https los datos se encriptan con el objeto de no ser interceptados a medio camino. Sin embargo, esto dificulta el envío de los datos como consecuencia de que se incrementa su cantidad.

Para solventar esto se utilizó un servicio terciario el cual es una página http que genera una redirección a la página https de google sheet deseada. Esto a su vez trae aparejados 2 problemas: el primero de ellos es que para hacer un uso gratuito de la aplicación, existe un límite de una página por el cual sólo se pueden hacer 100 envíos al día (para el proyecto en cuestión son necesarios 90), y el otro es que, por lo explicado previamente, los datos son más susceptibles de ser interceptados, siendo un punto vulnerable de la solución planteada.

Por todo lo explicado, se optó por subir los datos a dos lugares, ya que cada uno de ellos presenta diferentes limitaciones. En el caso de ubidots sólo se pueden subir 5 datos a la vez cada 10 segundos y estos sólo se pueden visualizar desde la página, ya que no es posible extraerlos con la versión gratuita. Por otra parte google sheet permite almacenar los datos y luego exportarlos para su posterior análisis. Sin embargo, la versión gratuita de google sheet tiene una limitación de 100 subidas por día, para nuestro proyecto es suficiente debido a que sólo necesitamos 90. Cabe destacar que, de ser necesario un número mayor de subidas de información, es posible incrementar a 1000 subidas por día con un costo de 2 dólares mensuales.

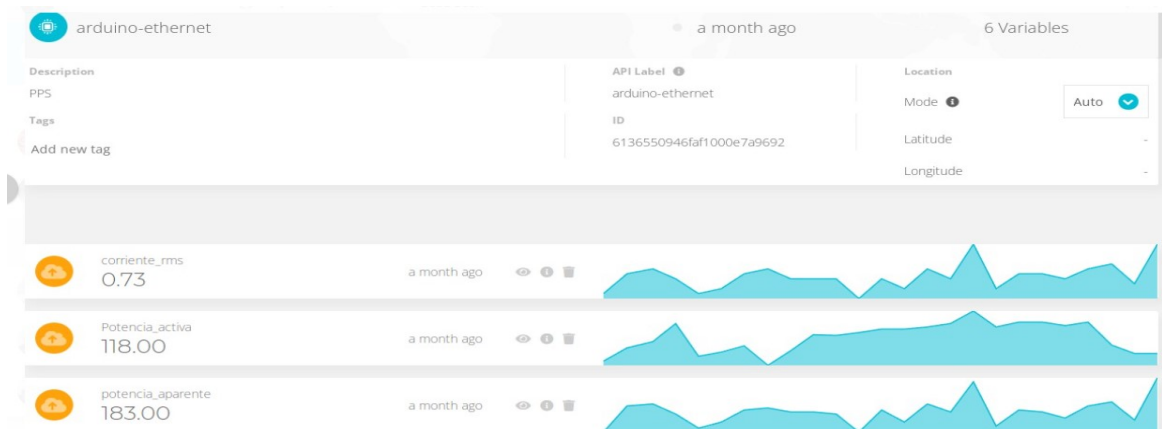


Ilustración 1: Visualización de datos históricos desde ubidots

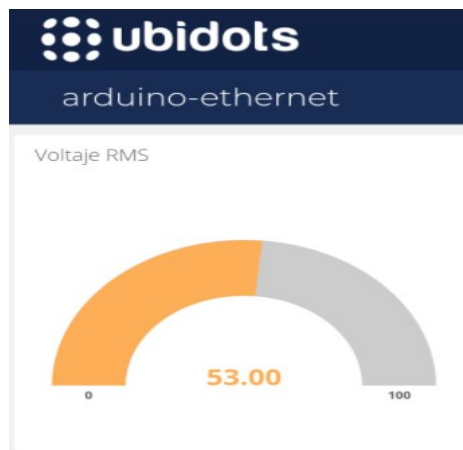


Ilustración 2: Visualización de datos en tiempo real

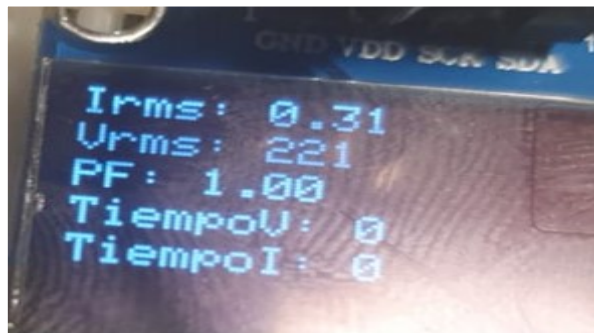
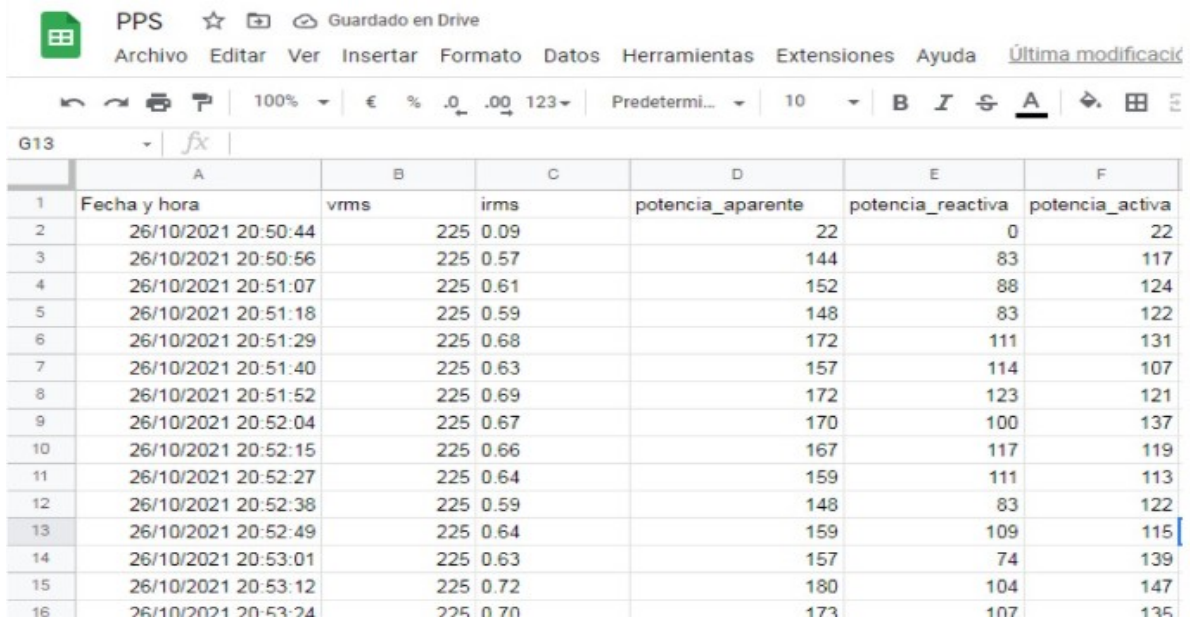


Ilustración 3: datos visto OLED



	A	B	C	D	E	F
1	Fecha y hora	vrms	irms	potencia_aparente	potencia_reactiva	potencia_activa
2	26/10/2021 20:50:44	225	0.09	22	0	22
3	26/10/2021 20:50:56	225	0.57	144	83	117
4	26/10/2021 20:51:07	225	0.61	152	88	124
5	26/10/2021 20:51:18	225	0.59	148	83	122
6	26/10/2021 20:51:29	225	0.68	172	111	131
7	26/10/2021 20:51:40	225	0.63	157	114	107
8	26/10/2021 20:51:52	225	0.69	172	123	121
9	26/10/2021 20:52:04	225	0.67	170	100	137
10	26/10/2021 20:52:15	225	0.66	167	117	119
11	26/10/2021 20:52:27	225	0.64	159	111	113
12	26/10/2021 20:52:38	225	0.59	148	83	122
13	26/10/2021 20:52:49	225	0.64	159	109	115
14	26/10/2021 20:53:01	225	0.63	157	74	139
15	26/10/2021 20:53:12	225	0.72	180	104	147
16	26/10/2021 20:53:24	225	0.70	173	107	135

Ilustración 4: Visualización de datos desde google sheets

Debido a que, sólo se acepta un tipo de conexión, una vez que tenga establecida una conexión WiFi o Ethernet se deberá reiniciar el Analizador de Redes si se quiere cambiar al otro tipo.

La conexión vía Ethernet no supone mayores dificultades para el usuario final, está resuelta en su totalidad por software y para efectuarse el usuario simplemente debe conectar el cable ethernet.

Para la conexión Wifi se utilizó un módulo ESP8266 modelo ESP12F, el mismo cuenta con una antena integrada en la placa tipo patch, esto trae una serie de limitaciones, tanto de rango y de sensibilidad, es por esto que se recomienda utilizar el dispositivo en interiores y lo más cerca del modem posible tratando de evitar obstrucciones que degradan la señal como paredes. Para lograr establecer esta conexión se realizó una interfaz amigable para el usuario, donde éste simplemente debe ingresar la contraseña de la red wifi a la que se desea conectar, en la memoria del módulo se guardan las redes para su futura utilización, una vez conectado, el display notificará que está conectado y se comenzarán a enviar los datos como se explicó anteriormente. Debido a que, requerimos enviar datos cada 15 minutos, se utiliza el modo sleep del ESP12F a fin de optimizar el consumo energético del mismo ya que este representa un porcentaje alto del consumo total del aparato.

6. Diseño del primer prototipo funcional

Para esto se utilizó un protoboard⁵, sin embargo, algunos componentes debieron ser soldados a una placa ranurada debido a los problemas de ruido y falsos contactos que producían los cables.

⁵ Es una placa de pruebas con conexiones eléctricas internas respetando un patrón. Permite la fácil conexión y desconexión de los componentes, haciendo mucho más rápidas las pruebas.

En el circuito se colocó una térmica independiente a la del edificio dónde se hizo la prueba de conexión.

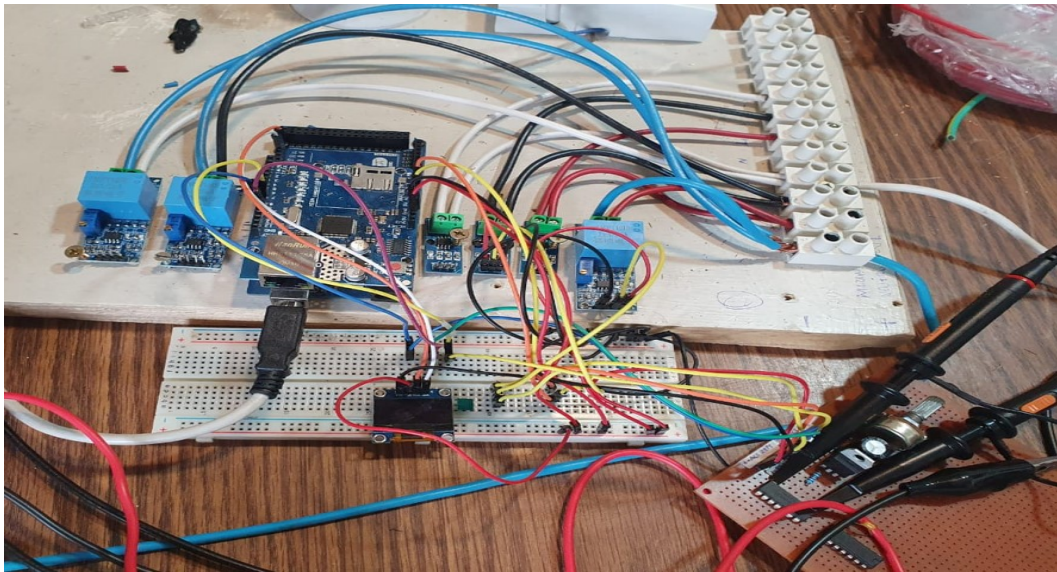


Ilustración 5: Protoboard (véase anexo 10)

Se comprobó el correcto comportamiento de las señales mediante un osciloscopio. Para ilustrarlo, en la siguiente imagen se muestra la onda de tensión analógica respecto de su par cuadrada.

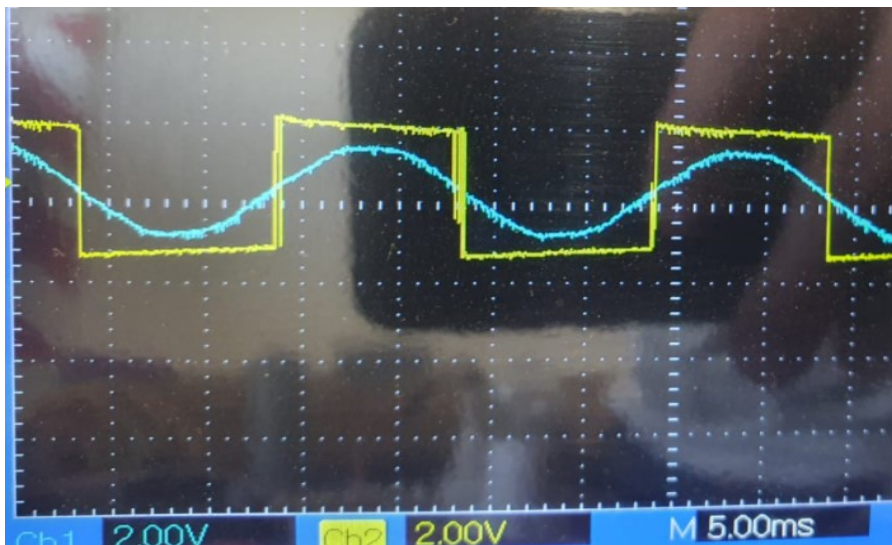


Ilustración 6: Onda de tensión vista en osciloscopio antes y después de ser comparada por el amplificador operacional

También se realizó la calibración de los sensores ZMPT de manera individual mediante osciloscopio y software. Además, se verificó que la tensión de referencia producida por el LM317 sea efectivamente la mitad de la onda.

Por último, se probó un mismo circuito en de forma individual en cada fase, obteniendo resultados similares, de esta forma fue posible asegurar el correcto funcionamiento de cada fase. Finalmente se evaluaron las tres fases en simultáneo para la red trifásica.

Es importante aclarar que, los cables utilizados para transportar corriente deben ser de 4 mm, ya que al tolerar hasta 36A, permiten tener un margen de tolerancia de un 16,7 cuando la corriente máxima en el circuito es de 30A.

7. Diseño del circuito de protección

A fin de proteger los componentes contra corrientes nominales superiores a las que están preparadas, se dispuso de una térmica de 26 A, es redundante ponerla debido a que los lugares donde será instalado el aparato debería contar con una. Sin embargo, optamos por conservarla debido a que permite aislarnos de la tensión de línea, es decir si nuestro aparato presenta un corto no se verá afectada la térmica del lugar.

En un principio se optó por utilizar fusibles cerámicos a modo de protección, esta idea fue descartada debido a las complejidades que le puede generar al usuario para volver a poner el aparato en funcionamiento. Además que, los componentes que pueden sufrir daños no son costosos.

7.1 Circuito de by pass

En caso de que el circuito de protección sea activado, el analizador de redes dejará de funcionar, sin embargo queremos que la vivienda no pierda el suministro eléctrico. Por lo tanto, se diseñó un circuito de by pass. Consiste en un relé normalmente cerrado para cada fase, el mismo estará polarizado por nuestro analizador por lo tanto no circulará corriente. Cuando el circuito de protección es activado, nuestro analizador ya no podrá polarizar los relé y éstos crearán un camino alternativo para el paso de corriente. El neutro no cuenta con un relé, sino que está directamente conectado. Fue necesario utilizar dos espacios con la misma fase "R" debido a que utilizando un solo espacio el circuito seguía energizado a través del transformador step down

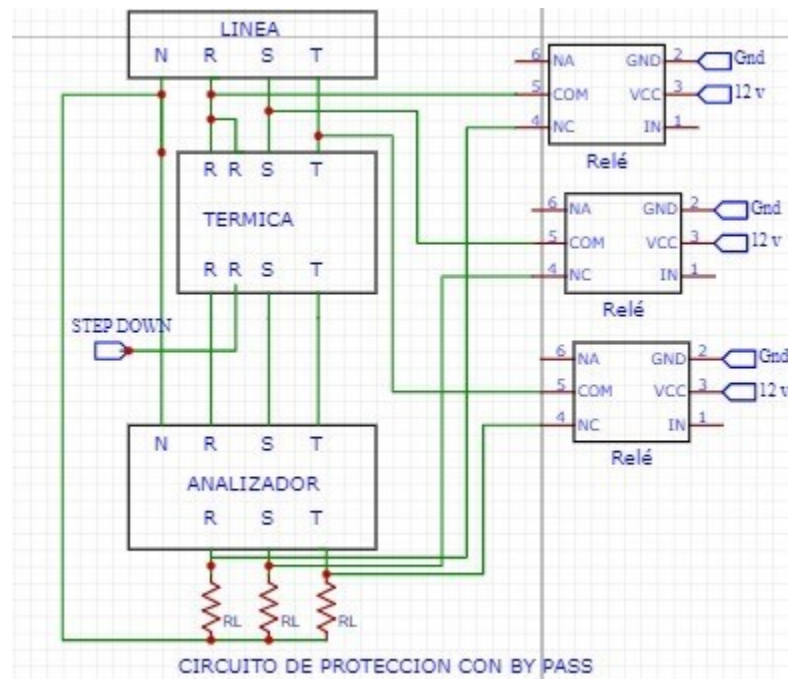


Ilustración 7: Circuito de protección

8. Pruebas de campo

Las mismas se realizaron en el laboratorio de electricidad de la CNEA, espacio brindado por el Ing. Marcelo Farías, quien además supervisó las mismas. En primera instancia se trabajó con corriente monofásica, luego el proyecto se escaló para ser usado en líneas trifásicas. Esto se llevó a cabo probando cada fase por separado y luego las tres fases simultáneamente. Se utilizaron cargas de naturaleza inductiva, capacitiva y resistiva, cuyos valores de potencia eran conocidos. Los resultados obtenidos por el dispositivo desarrollado fueron medidos y comparados con un analizador de redes comercial FLUKE 1735 Power Logger.

Los datos medidos se muestran en la siguiente tabla:

	Analizador comercial	Analizador de bajo costo	Error Relativo	Error Relativo	Error Relativo

	V rms	I rms	Pf	V rms	I rms	pf	Voltaje	Corrient	Phase Margin
Resistencias paralelo de 25.5 ohm total	218 V	8.05 A	1	220	8.53 A	1	1%	6.3%	0%
3 resistencias paralelo (27.5 ohm total) conectadas en serie con 5 capacitores de 16 pf c/u en serie	218 V	4.64 A	0.56	220	5.06 A	0.50	1%	9%	10%
Amoladora de banco	218 V	2.90 A	0.39	220	2.95 A	0.30	1%	1%	23%

Tabla 2 Comparación de mediciones varias entre analizador fluke vs analizador de desarrollo propio

Se comprobó mediante un tester la correcta tensión entre cada fase y neutro (220 V), además de las fases entre sí (380 V). Además, se midió el consumo total del analizador de redes construido, el cual es de 28,2 mA que resulta en un total de 6W cuando no se envía ningún dato. Al momento de enviar datos, el valor aumentará pero realmente al ser tan poco frecuente es despreciable en nuestro caso.

En última instancia se conectó un motor trifásico con un pico de arranque de 100 A obteniendo los siguientes resultados:

	Analizador comercial			Analizador de bajo costo			Error Relativo Voltaje	Error Relativo Corrient	Error Relativo Phase Margin
	V rms	I rms	Pf	V rms	I rms	pf			
Fase 1	221 V	3.84 A	0.13	216	3.95 A	0.09	2%	3%	30%
Fase 2	219 V	3.53 A	0.15	219	3.82 A	0.11	0%	8%	26%
Fase 3	220 V	3.7 A	0.11	221	4.3 A	0.09	0.4%	16%	18%

Tabla 3 Comparación de medición trifásica entre analizador fluke vs analizador de desarrollo propio

9. Cálculo del error relativo en la medición

Por la teoría de propagación de errores, para calcular el error relativo en una multiplicación se deben sumar los errores relativos de cada término, siendo éstos el módulo del error absoluto / valor real. El valor resultante se multiplica por 100 para obtener el valor porcentual.

Como valor real se toman los datos entregados por el analizador de redes comercial, entonces el error absoluto es la diferencia entre éste y el surgido de la medición efectuada con el analizador de bajo costo.

Como se tienen diferentes errores para cada potencia y medición, se calcula un error promedio. El error promedio porcentual fue de un 16% para una cantidad promedio de 3 mediciones.

Corriente	Error
<3 A	23.8%
3A a 6 A	18.8%
>6 A	6.3%

Tabla 4 Comparación del error porcentual en función de la corriente rms

Se observa que el error relativo disminuye al aumentar la corriente, esto es por la resolución de nuestros sensores

El error es aproximado debido a la cantidad de muestras analizadas. Se considera aceptable respecto al alcance del proyecto

10. Desarrollo de software

Se implementó un código original en lenguaje C, funciona en arduino y se encarga de la adquisición, el procesamiento de los datos y el control de la calibración de las señales de tensión y corriente.

Posteriormente, se implementó un programa en C# instalable en windows y portable. Este programa es una interfaz gráfica capaz de conectarse al puerto serie requerido, recibir los datos para ser mostrados de forma gráfica en tiempo real y guardarlos en el disco de la computadora en formato csv, de forma que puedan ser luego analizados con otro programa.

Una manera alternativa de visualizar los datos de cada fase es a través de un oled integrado, el cual también indica si se requiere llevar a cabo una calibración en el dispositivo.

Por tratarse de programas de desarrollo e implementación propia, su código fuente se incluye en el anexo 11.

Seba	🔒 100%
avc653461	🔒 60%
HUAWEI-6kGc	🔒 32%
Speedy-7C9638	🔒 32%
HUAWEI-ufnh	🔒 22%
Malvinas	🔒 22%

Seba
password

save

[Scan](#)

Ilustración 8: Inicio de programa mostrando las redes disponibles

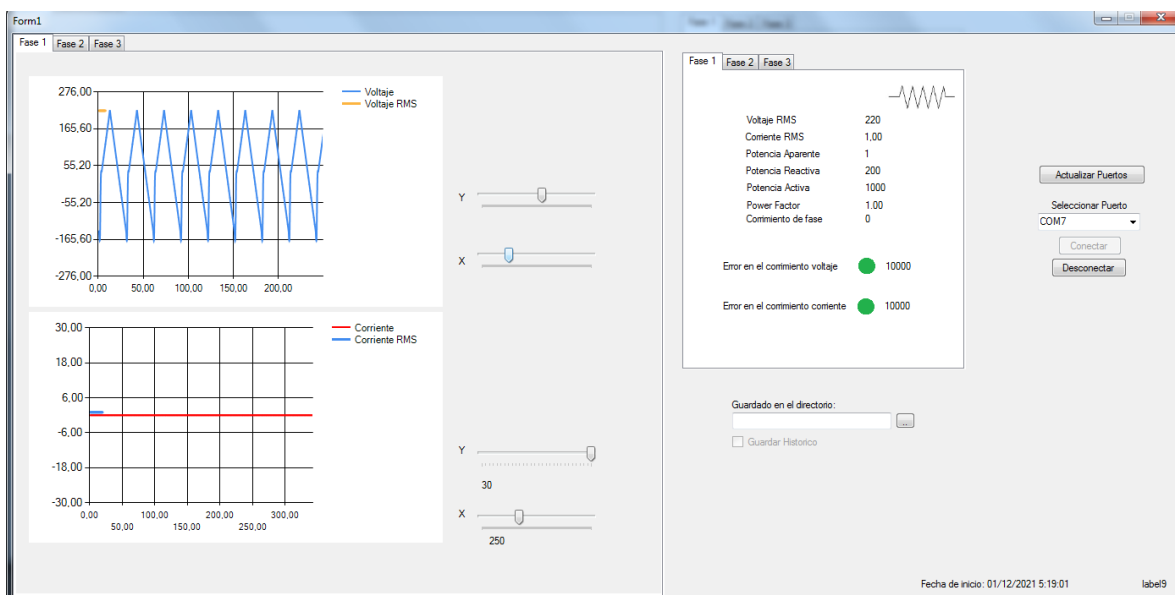


Ilustración 9 Visualización del programa construido en C#

Para facilitar el trabajo en el software se utilizó un conocido gestor de versiones, gitlab. El mismo evita cualquier error que se podría tener en el guardado y además es útil para la revisión del código y trabajo en equipo.

Adicionalmente, se incluye un manual del usuario para la correcta comprensión y operación del programa cuya interfaz gráfica se muestra en la Figura 7.

10.1 Descripción de la Interfaz

Para comenzar, la interfaz gráfica tiene en la parte derecha el puerto al que está conectado el arduino para “escucharlo” y recibir los datos.

La misma se divide en 2 grandes sectores, a la izquierda presenta una parte gráfica que muestra la forma de la onda, mientras que el de la derecha muestra datos fijos referidos a cómo pueden ser la naturaleza de la onda y de las potencias, indica si la calibración es correcta, entre otras cosas que se describen con más detalle en el manual de usuario.

En la parte inferior tiene una sección destinada a guardar los datos en la computadora mientras se adquieren (en principio una vez cada 15 minutos, pero próximamente esto podrá configurarse desde la interfaz gráfica junto con la indicación de qué datos y cada cuánto tiempo serán guardados). Esto se hace para las 3 fases de manera separada, por lo cual la interfaz gráfica se encuentra dividida en pestañas, cada una de las cuales tiene la información de cada fase, permitiendo de esta forma un acceso simple y claro.

10.2 Memoria utilizada por el programa

Se realizaron pruebas durante 90 minutos para comprobar el correcto funcionamiento. A partir del gráfico de memoria podemos ver que el proceso tiene un consumo máximo de 30 MB (la memoria RAM total ronda los 8 GB a 16 GB hoy en día, por lo que sería un consumo que ronda el 3%). Des-aloca correctamente la memoria utilizada para así no llenarse (se puede ver cómo se van colocando los datos en memoria y luego hay una bajada abrupta donde se des-alocan para volverlos a colocar los nuevos).

En cuanto al uso del CPU es relativo al procesador que se esté utilizando. Esto fue testado con una máquina que tiene un procesador AMD FX-6300 (del año 2012) y los picos máximos rondan el uso del 10% del procesador cuando recibe todos los datos y los procesa, lo cual resulta reducido en cuanto a la potencia de cómputo que tiene el procesador.

Se considera prestar atención a estas mediciones de memoria y procesador importantes ya que si es un programa que se piensa dejar durante semanas en un lugar cualquier falla por memoria va a causar una sobrecarga del programa terminando en un “crasheo” del mismo.

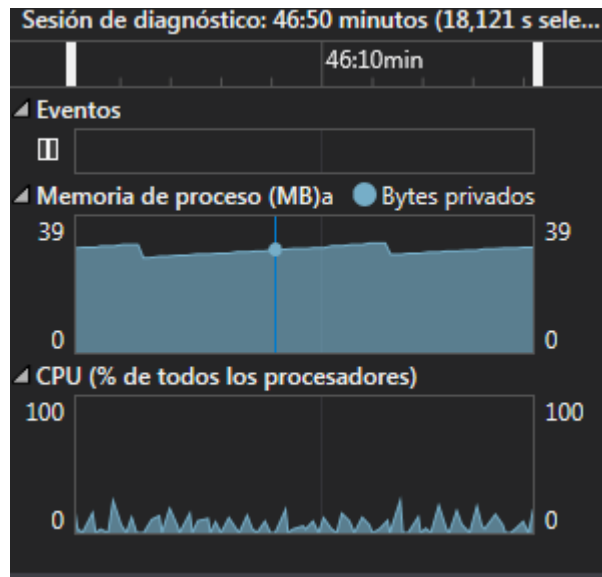


Ilustración 10 Visualización memoria consumida en un tiempo prolongado

Nombre de imagen	Nombr...	CPU	Memoria (espacio d...	Descripción
PPS.exe *32	Seba	10	20.628 KB	PPS

Ilustración 11 porcentaje del consumo de procesador y memoria

10.3 Control de Versiones

Se optó por utilizar un controlador de versiones muy conocido, Gitlab⁶, el cual fue de mucha ayuda para mantener el orden y volver a cambios anteriores en todas las etapas de desarrollo del software, la funcionalidad que tiene es poder tener todos los archivos, tanto código como esquemáticos del circuito en un solo lugar y realizar las actualizaciones con facilidad desde cualquier computadora. Esto es especialmente útil si se necesita moverse a otro lugar ya que los códigos están en la nube y son accesibles desde cualquier dispositivo.

⁶ (2021) Gitlab recuperado de gitlab.com

Name	Last commit
📁 Arduino	Arreglado para el nuevo programa con grafi...
📁 Ejecutable ClickOnce	Arduino medicion de corriente y voltaje junt...
📁 Interfaz usuario	Arreglado para el nuevo programa con grafi...
📄 Esquematico PPS.pdf	Initial commit
📄 ReadMe.md	Update Readme.md
📄 cd	Agrego varios codigos de arduino
📄 git	Agrego varios codigos de arduino

Ilustración 12 Gitlab

En la ilustración 12 se puede ver que en el gitlab está la última versión de los programas de hechos: arduino, la interfaz de usuario y el esquemático.

Algo muy importante de esto es la capacidad de volver a versiones anteriores, esto es, si hubo algún error en el último código que se subió se puede volver fácilmente al anterior que funcionaba correctamente, es un servicio pensado para trabajar con grupos y que todas las personas puedan aportar sin el riesgo de que deje de funcionar permanentemente.

11. Diseño PCB

El circuito PCB de doble faz se diseñó en EasyEDA⁷ (véase anexo 12), luego se imprimió en papel fotográfico de 90g, se pegó sobre una placa limpia de cobre mediante transferencia térmica. Posteriormente, se sumergió la placa en agua durante 15 minutos por lado para eliminar el papel. Finalmente, se atacó la placa con una disolución de cloruro férrico para eliminar todo resto de cobre que no esté bajo tinta, es decir que no forme parte de nuestro circuito.

La placa tiene una dimensión de 15 cm x 15 cm y una altura de 1.6 mm. Para las pistas se utilizó un espesor de 0.74 mm. Para transportar la corriente se usaron cables, sin embargo si se desea reemplazarlos por una pista, esta debería tener un espesor de 5 mm. Los pads son de 2 mm de ancho.

⁷ EasyEDA recuperado de <https://easyeda.com/>

parcialmente fuera para ganar comodidad en las interconexiones, éstas se realizaron con cables de 4 mm.

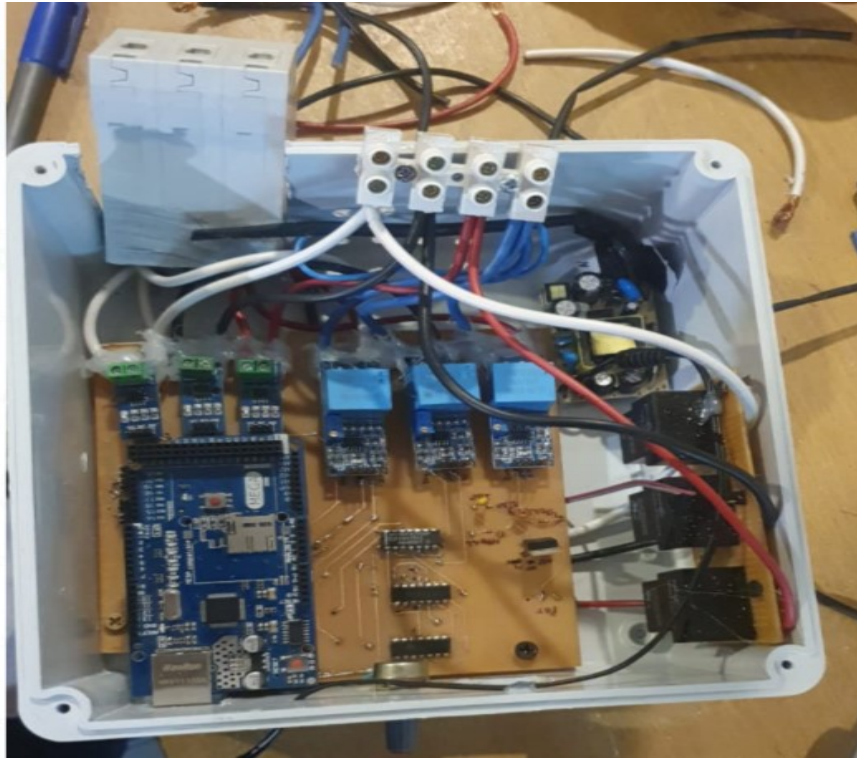


Ilustración 15 Interior del prototipo

PROBLEMAS ENCONTRADOS DURANTE EL DESARROLLO

Algunos componentes son muy sensibles, se tornó difícil hacer pruebas en un protoboard debido a que son muy propensos a tener errores de hardware, por lo tanto, se debió soldar todo a una placa perforada.

Debido a la lógica digital no es posible contar con valores en tiempo real ya que cualquier ruido puede perturbar las mediciones. Para mitigar este inconveniente, y aprovechando en que no es necesaria una respuesta rápida, se usaron promedios de las mediciones.

El sensor de tensión aporta un corrimiento de fase constante debido a la bobina interna que posee, el corrimiento es de -36° y fue corregido mediante software.

El tiempo de respuesta del circuito de protección no es el más rápido, se propuso dejarlo en esta instancia debido a que mejorarlo conlleva mucho tiempo y los materiales que está protegiendo no son excesivamente costosos.

Cuando se visualizó la onda de tensión rms en nuestro software se observó un offset aleatorio, luego de realizar mediciones con un osciloscopio se concluyó que este se produce por software, por lo tanto, fue corregido por programa desarrollado, centrando la onda.

Para alimentar el Arduino Mega mediante un Jack se necesitan 7V mientras que para hacerlo vía USB son necesarios 5V. Esto es porque internamente el arduino aumenta la tensión cuando es alimentado por USB. El resto de los componentes necesitan ser alimentados con 5V, para simplificar el proyecto se utilizó un transformador step down de 220 V AC a 7 V DC , alimenta a la placa arduino, y desde ésta se realiza la alimentación del resto de la lógica a través de su salida de 5V.

Se debe tener en cuenta la ubicación de las vías en el diseño del pcb, ya que las mismas quedan tapadas cuando se sueldan los componentes y dificulta corregir errores.

Para la construcción de un modelo robusto se debe pensar mejor la ubicación de los componentes y su cableado, debido a que esto dificulta el ensamblaje.

La fase de donde se alimentaba el circuito de bypass generaba un camino alterno indeseado manteniendo el circuito energizado en caso de que el sistema de protección sea activado, para esto se debió dividirla en dos, es decir usar dos conexiones de la térmica para la misma fase

POSIBLES MEJORAS

- Sensar la corriente que circula por el neutro: de esta forma podríamos tener un control si existen fugas.
- Utilizar un transformador step down a la entrada de la línea para reducir la tensión y así poder utilizar otro tipo de sensores.
- Utilizar una pantalla más grande para visualizar los datos (la actual es de 128x64 pixeles).
- Corregir mediante hardware los grados aportados por los sensores ZMPT.
- Utilizar Filtros para mejorar la inmunidad al ruido.
- Utilizar únicamente los núcleos de arduino, Ethernet y demás componentes con el fin de reducir espacio físico y mejorar la estética. Además de programarlos con un fin exclusivo, haciéndolos más eficientes.
- Capacidad de auto calibración vía software.

- Utilización de sensores no invasivos.
- Lograr una mayor resolución en corrientes bajas.
- Llevar a cabo la optimización de la energía consumida vía software.
- Realizar un servidor propio mediante el cual se puedan recibir (con una API) los datos que envíe el sensor. Si esto se realiza y se programa correctamente al arduino desaparecerían las limitaciones que las plataformas anteriores plantean y agregaría más libertad al proyecto.
- Agregar la posibilidad de sacar datos vía tarjeta SD.
- Buscar otra manera de conectar con ethernet que sea más actual por encriptación de https.
- Mejorar la antena de WiFi para mayor rango y puede que incluso se pueda usar afuera.
- Rediseñar el pcb y utilizar componentes de montaje superficial a fin de ahorrar espacio.
- El error es muy sensible a la calibración del potenciómetro.

PRESUPUESTO APROXIMADO

Componente	cantidad	Precio total por componente (\$)	Precio por componente en dólares (USD) ⁸
Arduino mega	1	2000	15
Ethernet shield	1	1000	8
Esp 12 D1 Mini V2	1	800	1.5
Acs 712	3	1000	4.5
Zmpt101b	3	6000	3
XOR	1	150	0.3

⁸ El costo en dólares está recuperado de paginas extranjeras por lo que es posible que hayan impuesto que no se tengan en cuenta, aunque los analizadores de redes comerciales normalmente también están valuados en dólares o con impuestos.

Lm324	2	300	1.2
Térmica trifásica	1	1000	13
Fusibles 30 A	3	600	1.5
Varistor 250 V	3	300	3
Relé 30 A	3	1800	9
Transformador step down	1	500	1.25
oled	1	1200	1.66
Potenciómetro	1	150	0.31
Lm317	1	200	0.9
Caja cobertora	1	1300	8
Materiales para fabricar pcb	—	700	
Total		19000	72.12

Tabla 5 Presupuesto

COMPARATIVA

	Analizador Comercial	Analizador de bajo costo
Voltaje	0.5%	1% - 5%
Corriente (> 6A)	0.5%	6% - 12%
Corrimiento de fase [PF] (>6A)	1%	5-20%
Facilidad de instalación	No invasivo, fácil para el personal	Invasivo, más laborioso de instalar
Conexión a internet	No posee	Se conecta vía Wifi, Ethernet y guarda los datos en servidores externos para subirlos a la

		nube.
Alimentación	Con batería, la misma se carga con un transformador que debe ser conectado a la línea	Toma la energía de la conexión de línea por lo que no necesita batería
Precio	2700 dólares	100 dólares

Tabla 6 comparativa funcional entre los analizadores

	Analizador Comercial	Analizador de bajo costo
Alto	110 mm	110 mm
ancho	240 mm	210 mm
largo	180 mm	230 mm
Peso	1.7 kg	1.1 kg

Tabla 7 comparativa dimensiones físicas



Ilustración 16 Analizador de redes comercial FLUKE



Ilustración 17 Analizador de redes de bajo costo

CONCLUSIÓN

Fue una experiencia sumamente gratificante desde lo académico y personal en un ambiente muy agradable con personas siempre dispuestas a ayudar en todo lo que podían y más, donde se aprendieron nuevos temas y tuvimos la posibilidad de manipular nuevos equipos que normalmente no tendríamos acceso con tanta facilidad. Queda demostrado que es posible realizar un analizador de redes, con una buena relación entre su exactitud y precio. El mismo no es difícil de replicar por lo tanto se puede extender a diversos hogares para cumplir su fin. Sus componentes son fácilmente reemplazables en caso de falla y no conllevan mantenimiento.

DATASHEET

Analizador de redes de bajo costo



El analizador de redes está diseñado para trabajar con corrientes monofásicas como trifásicas, los datos obtenidos son: Irms, Vrms, Power Factor, potencia activa, potencia reactiva y potencia aparente. Los mismos pueden ser visualizados mediante el propio oled del analizador, extraídos por el puerto serial o subidos a ubidots y google sheets mediante una conexión wifi o ethernet.

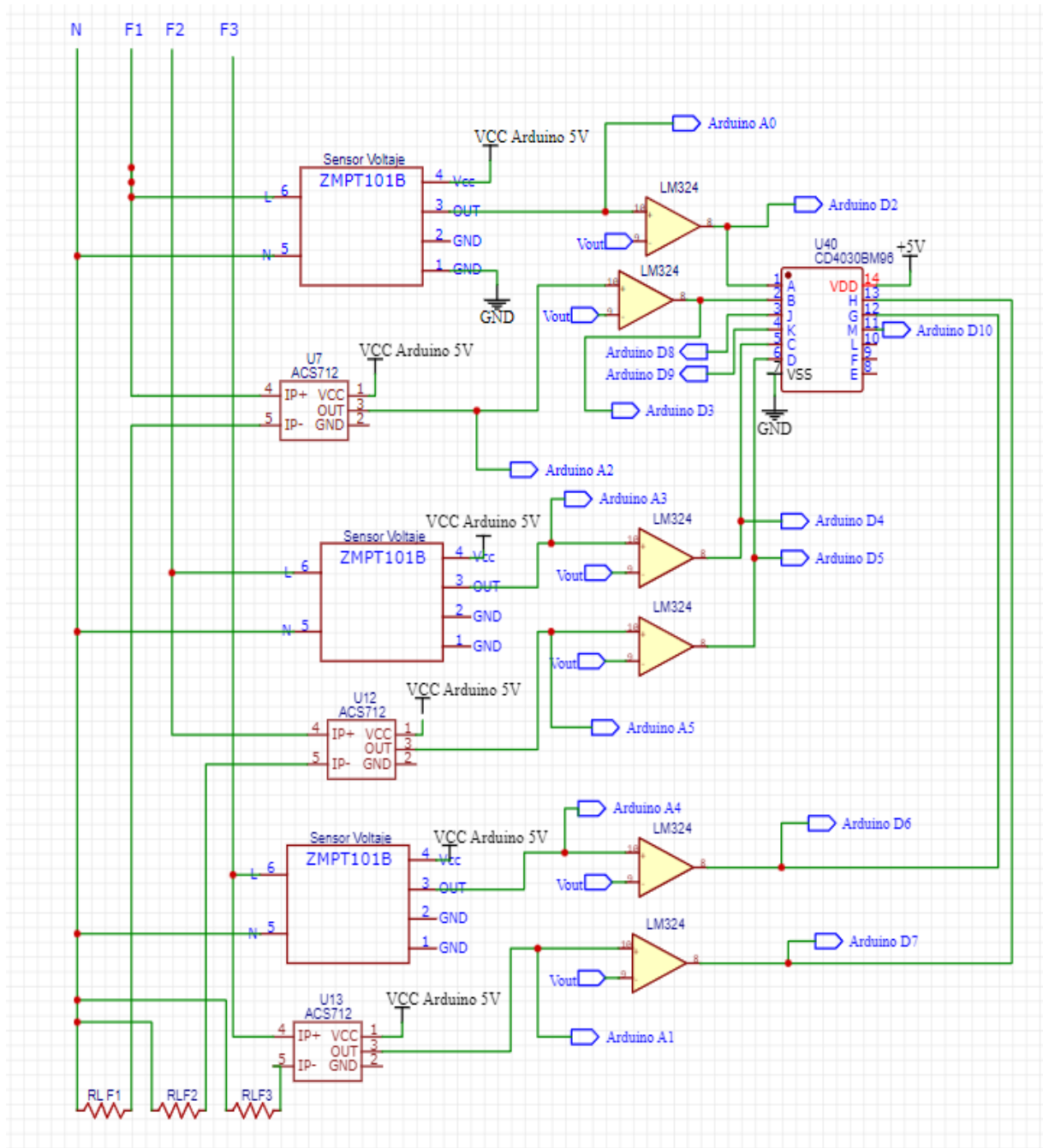
Aplicaciones:

Medición del consumo eléctrico en hogares, talleres y pequeñas industrias con el fin de analizar sus datos y optimizar la eficiencia energética.

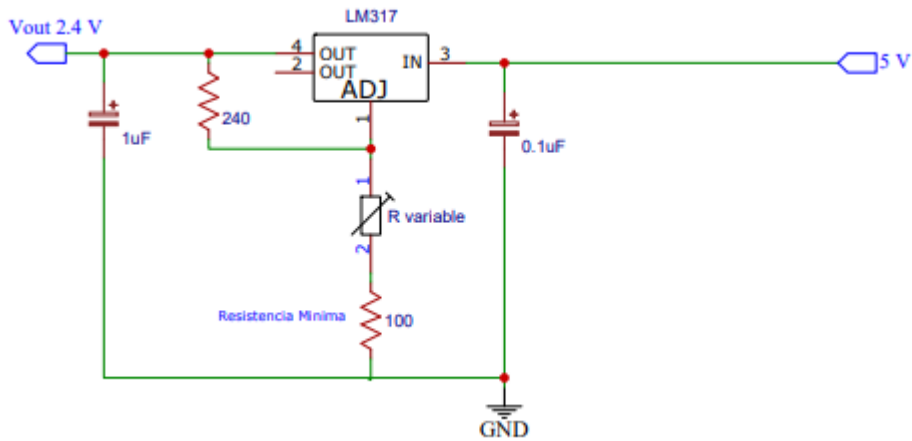
ÍNDICE

Esquemático global	2
Esquemático regulador de voltaje estable	4
Esquemático conexión OLED	4
Esquemático conexión módulo wifi	4
Características Eléctricas	4
Error relativo en la medición	5
Temperatura de operación	5
Dimensiones físicas	5
Calibración	6
Alcance de wifi	6
Circuito de protección	6

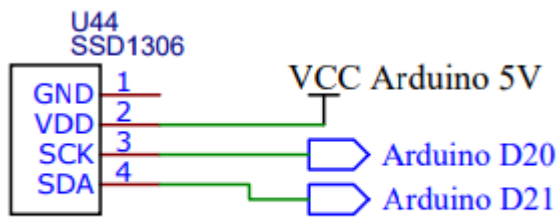
Esquemático global



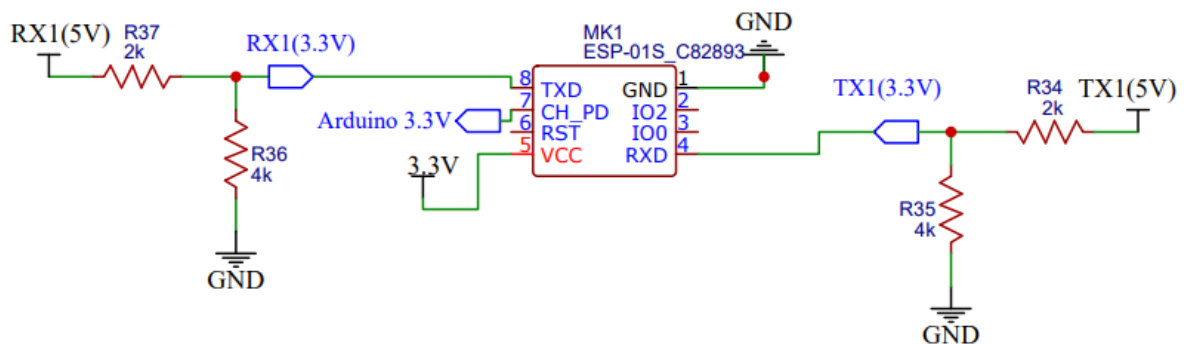
Esquemático regulador de voltaje estable



Esquemático conexión OLED



Esquemático conexión módulo wifi



Características Eléctricas

Corriente mínima	2 A
Corriente máxima	30 A
Tensión mínima	50 V
Tensión máxima	1000 V
Consumo en funcionamiento	430 mA

Error relativo en la medición

Corriente	Error
<3 A	23.8%
3A a 6 A	18.8%
>6 A	6.3%

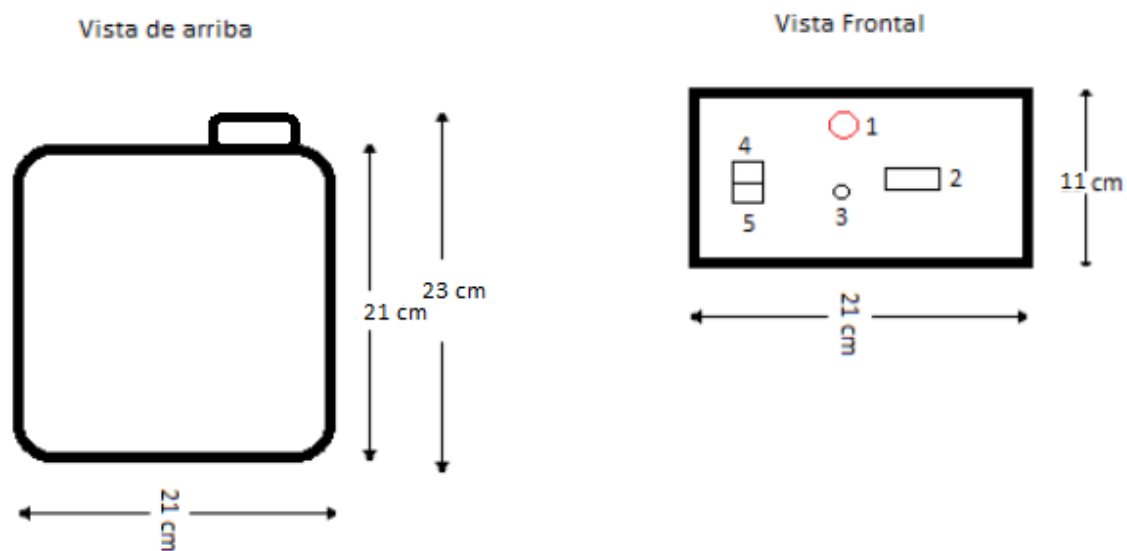
Temperatura de operación

El rango de temperatura está acotado por el lm324, siendo este de 0°C a 70°C, es por esto que no se recomienda su instalación en exteriores.

	Mínimo	Máximo	Unidad
Temperatura XOR	-55	125	°C
Temperatura LM317A	-40	125	°C
Temperatura LM324	0	70	°C
Temperatura Arduino	-40	85	°C
Temperatura ESP12F	-20	85	°C

Dimensiones físicas

Ancho	21 cm
Largo	23 cm
Alto	11 cm
Peso	1.1 kg



- 1: Botón de reset
- 2: OLED
- 3: Potenciómetro de calibración
- 4: Conexión Ethernet
- 5: Conexión USB

Calibración

En caso que el aparato requiera calibración, emitirá un aviso en la pantalla OLED. La misma podrá ser realizada por cualquier usuario moviendo la perilla del potenciómetro hasta que desaparezca el aviso. Es importante la correcta calibración, debido a que está directamente relacionada con el error que obtendrá la medición final del corrimiento de fase.

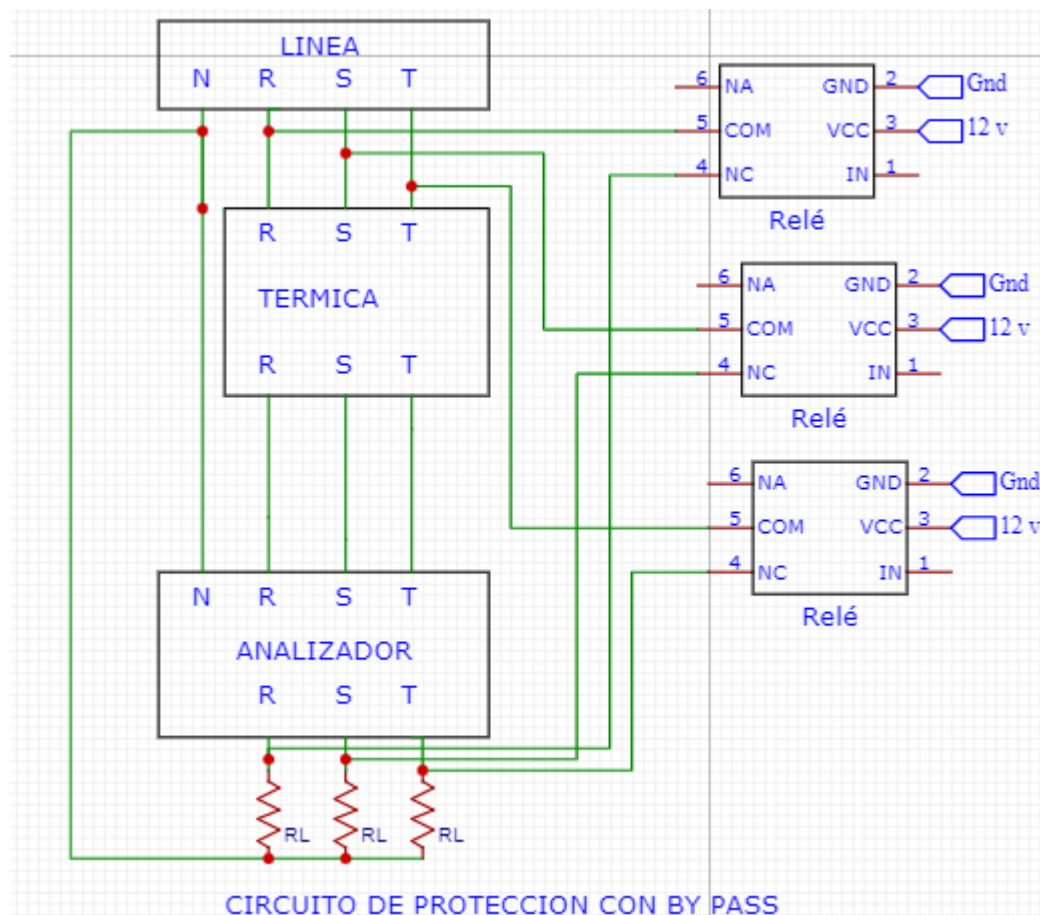
Alcance de wifi

Si la línea de visión es limpia podemos obtener un alcance de hasta 150 metros, sin embargo, cualquier obstáculo reducirá mucho el alcance debido a la frecuencia de operación. Como el aparato está diseñado para medir consumos hogareños es muy posible que nos encontremos con objetos en la línea de visión, es por esto que se recomienda su instalación lo más cercana al modem wifi.

Rango de Frecuencias	2412 - 2484 MHz
Potencia de Transmisión	802.11b: 16 +- 2dBm 802.11g 14-+ 2dBm 802.11n 13+-2dBm
Sensibilidad	-88 dBm

Circuito de protección

En caso de existir una corriente de fuga mayor a 30 mA la térmica se disparará, en este caso se deberá identificar y solucionar el problema antes de volver a re-establecerla. La misma se encuentra señalizada en el analizador. Para evitar perder el suministro eléctrico, en el lugar de la instalación, se activará automáticamente un circuito de BY PASS.



Manual del usuario analizador de redes de bajo costo



Ilustración 1: analizador visto de frente

INDICE

Manual Técnico Hardware	2
instalación	2
Conexiones externas	2
Conexiones internas	3
Visualización de datos en oled	6
Manual Técnico Software	6
Requerimientos del sistema operativo:	7
Inicio del programa sin conectar	7
Parte gráfica	8
Parte Numérica	9
Guardado de datos	10
Conexión e inicio del programa	10
Inicio del programa	11
Ejemplo de distintas vistas	12
Gráfico del programa comunicado	14
Desconexión	15
Comunicación vía Ethernet	16
Comunicación vía Wifi	16
Errores Conocidos	17

Manual Técnico Hardware

instalación

Conexiones externas



Ilustración 2 analizador parte posterior

Para instalar de forma correcta el dispositivo se deben seguir los siguientes pasos

En primer lugar, se debe conectar el neutro el neutro de la instalación que se desea medir, éste ingresa por el orificio N de la entrada y se conecta internamente a la bornera dispuesta arriba del mismo. Luego, se debe conectar la o las fases correspondientes que ingresaran por los orificios de entrada L1, L2 y L3 y se conectaran internamente a la bornera superior de la térmica. La fase L1 debe ser conectada en dos lugares diferentes de la térmica para asegurar el correcto funcionamiento del circuito de protección, en la imagen se realizó el prototipo con una térmica de tres entradas y luego se agregó una entrada externa para simular una térmica de cuatro entradas. Para un correcto funcionamiento se deberá usar una de cuatro entradas, donde la fase L1 se conecte en dos lugares diferentes.

Por último, se deben conectar las cargas de la casa, los cables de las mismas ingresaran por los orificios de salida N, L1, L2 y L3 donde se conectan inmediatamente a la bornera que está detrás de estos.

Conexiones internas

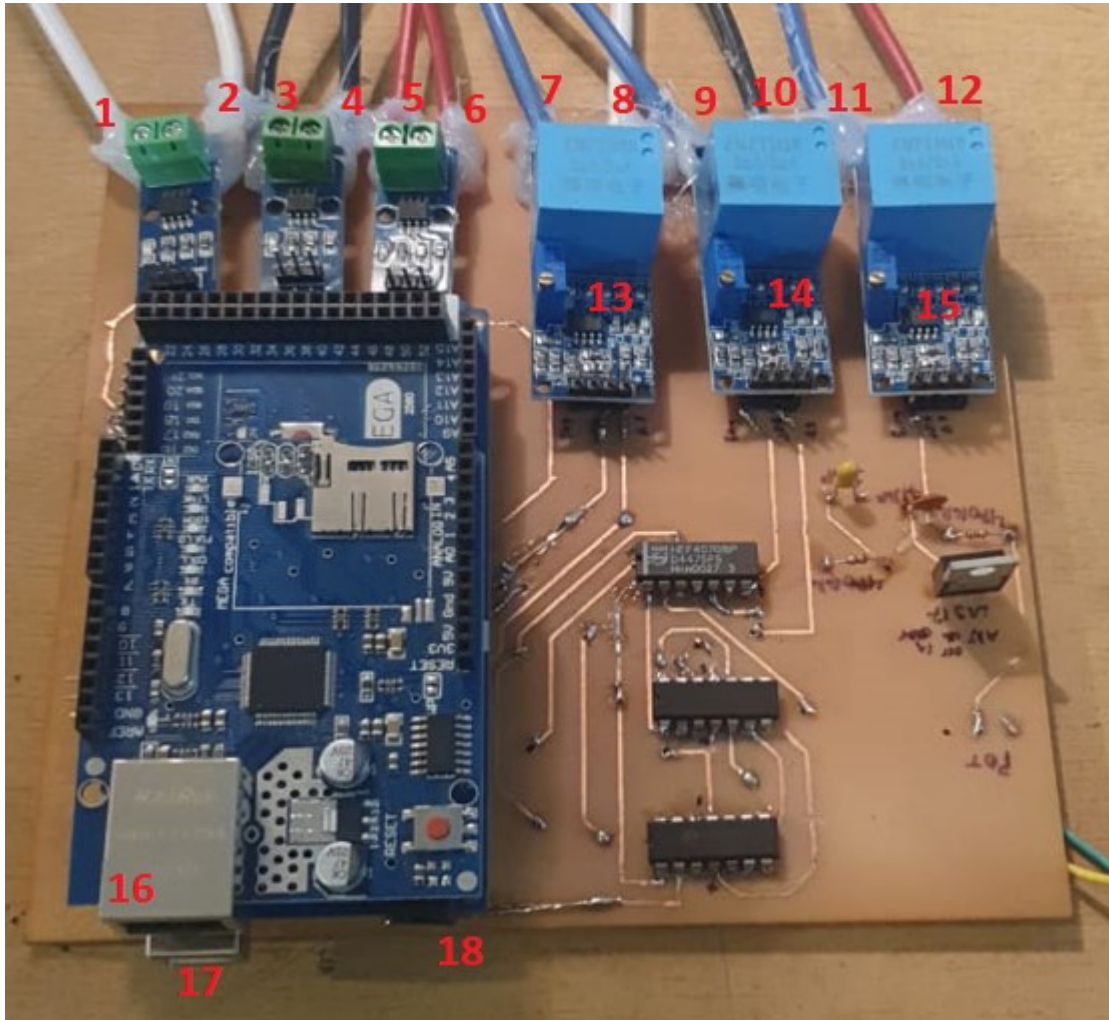


Ilustración 3 referencia de conexiones

1. cable entrada ACS para fase 1
2. cable salida ACS para fase 1
3. cable entrada ACS para fase 2
4. cable salida ACS para fase 1
5. cable entrada ACS para fase 3
6. cable salida ACS para fase 1
7. cable neutro ZMPT fase 1
8. cable fase 1 ZMPT
9. cable neutro ZMPT fase 2
10. cable fase 2 ZMPT
11. cable neutro ZMPT fase 3
12. cable fase 3 ZMPT

13. potenciómetro ZMPT fase 1
14. potenciómetro ZMPT fase 2
15. potenciómetro ZMPT fase 3
16. entrada cable ethernet
17. entrada cable USB
18. entrada alimentación JACK

Primero se deben conectar los sensores acs (cables 7,9,11) al neutro de línea, luego las fases de línea que se desean medir a la entrada de la térmica y desde la salida de la misma a los sensores acs de forma individual (cables 1,3,5), también desde la salida de la térmica debemos conectar la fase de los sensores zmpt (cables 8,10,12) cada sensor debe estar conectado a una fase diferente.

La carga que se desea medir debe estar conectada a salida de fase de los sensores ACS (cables 2,4,6) y al neutro respectivamente.

Una vez realizada esta instalación debemos asegurarnos que los sensores zmpt estén bien calibrados, nos daremos cuenta de esto mediante un aviso en el display OLED.

Para ajustar los mismos se debe mover el potenciómetro de cada uno hasta que el aviso de calibración desaparezca. (referencia 13,14,15). El aparato ya está listo para ser utilizado.

El mismo aparato se alimenta de la línea mediante la conexión jack (referencia 18), sin embargo, también puede hacerlo mediante un USB (referencia 17) por donde también se podrán exportar datos. La conexión vía ethernet se realiza conectado el cable en la referencia 16.

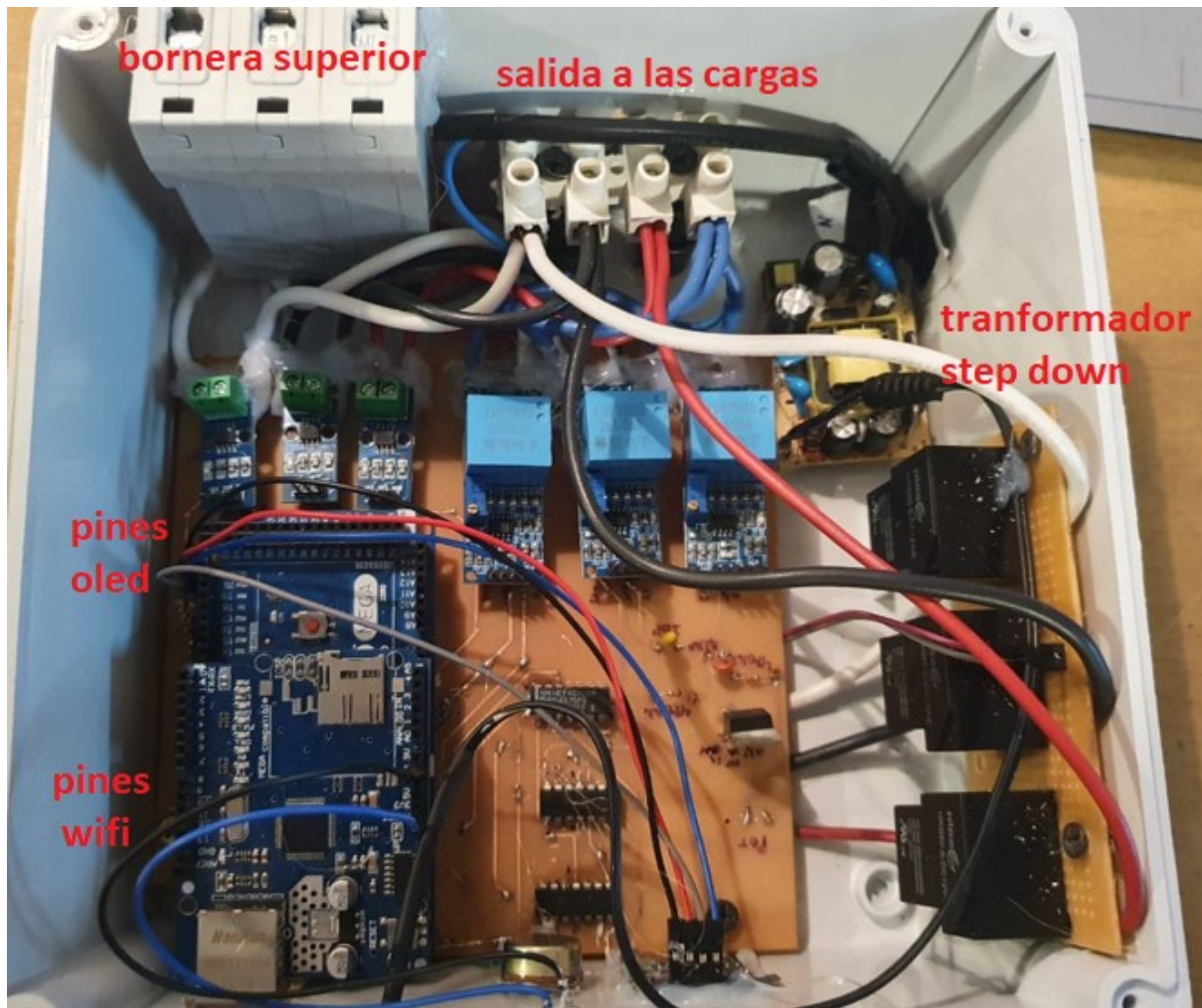


Ilustración 4 interior del prototipo

El transformador Step down se debe conectar a neutro y la primera fase, la salida de este es la alimentación JACK (18), además también se utiliza para alimentar los relés.

CUIDADO: antes de operar con el mismo asegurarse que la bobina está descargada En caso de falla.

Los relés deben conectarse de forma individual a cada fase, el común a la bornera superior y el pin normalmente cerrado a la salida de las cargas.

El botón reset a los pines “reset” y “ground” del arduino. Para el OLED se dispusieron 4 pines a la izquierda del arduino, en orden superior a inferior estos corresponden a GROUND, VCC, SCK y SDA, mientras que los pines del módulo wifi se disponen como GND, VCC, RX, TX bajo el mismo criterio.

Si por algún motivo el circuito entra en corto se activará la térmica de este al mismo tiempo que un circuito de BY PASS de modo que seguiremos teniendo energía en las cargas. Se debe detectar y corregir el problema, luego volver la térmica a su posición original para seguir operando, al restablecer la térmica el circuito de BY PASS dejará de funcionar automáticamente.

Si se activa la térmica de la vivienda, es posible que el corto esté en la carga y no en el analizador de redes.

Visualización de datos en oled

Debido a la cantidad de datos que se requieren visualizar se optó por individualizar para cada fase a fin de ganar claridad, los mismos se mostrarán en el OLED donde cambiará la pantalla automáticamente exhibiendo los de la fase siguiente.

Manual Técnico Software

El programa está diseñado para ser de fácil utilización, al ser una primera versión es posible que contenga algunos errores (bugs) que deberán ser corregidos en versiones futuras.

Las ilustraciones expuestas están hechas con un arduino simulando recepción de los datos con el fin de mostrar todas las características, las magnitudes exhibidas pueden no ser reales.

Requerimientos del sistema operativo:

- Compatible con windows 7 en adelante (testado en windows 7,8 y10), no fue realizada versión compatible con Linux ni IOS.
- Consumo máximo de memoria RAM registrado: 36 MB
- Almacenamiento: 230KB el instalador

- Testeado con resultados óptimos en procesador posterior a 2010

El programa fue probado por 5 horas para obtener estos datos, sin mayores inconvenientes una vez que está bien conectado.

Inicio del programa sin conectar

Al iniciar el programa se va a ver una pantalla casi en blanco con estas características

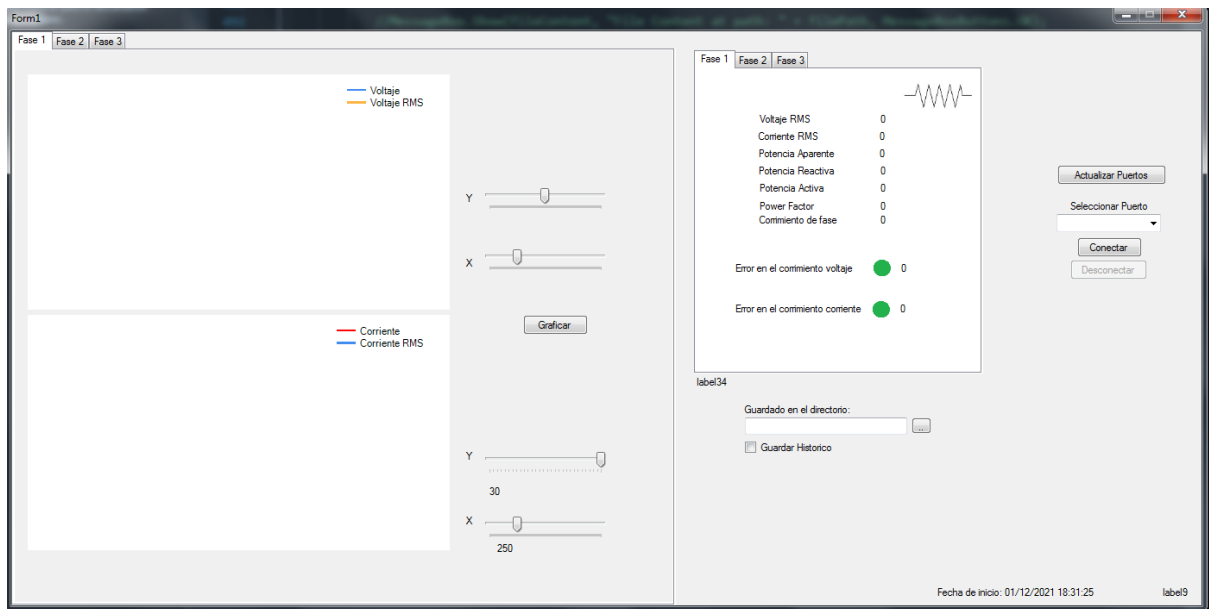


Ilustración 5 inicio del programa

Parte gráfica

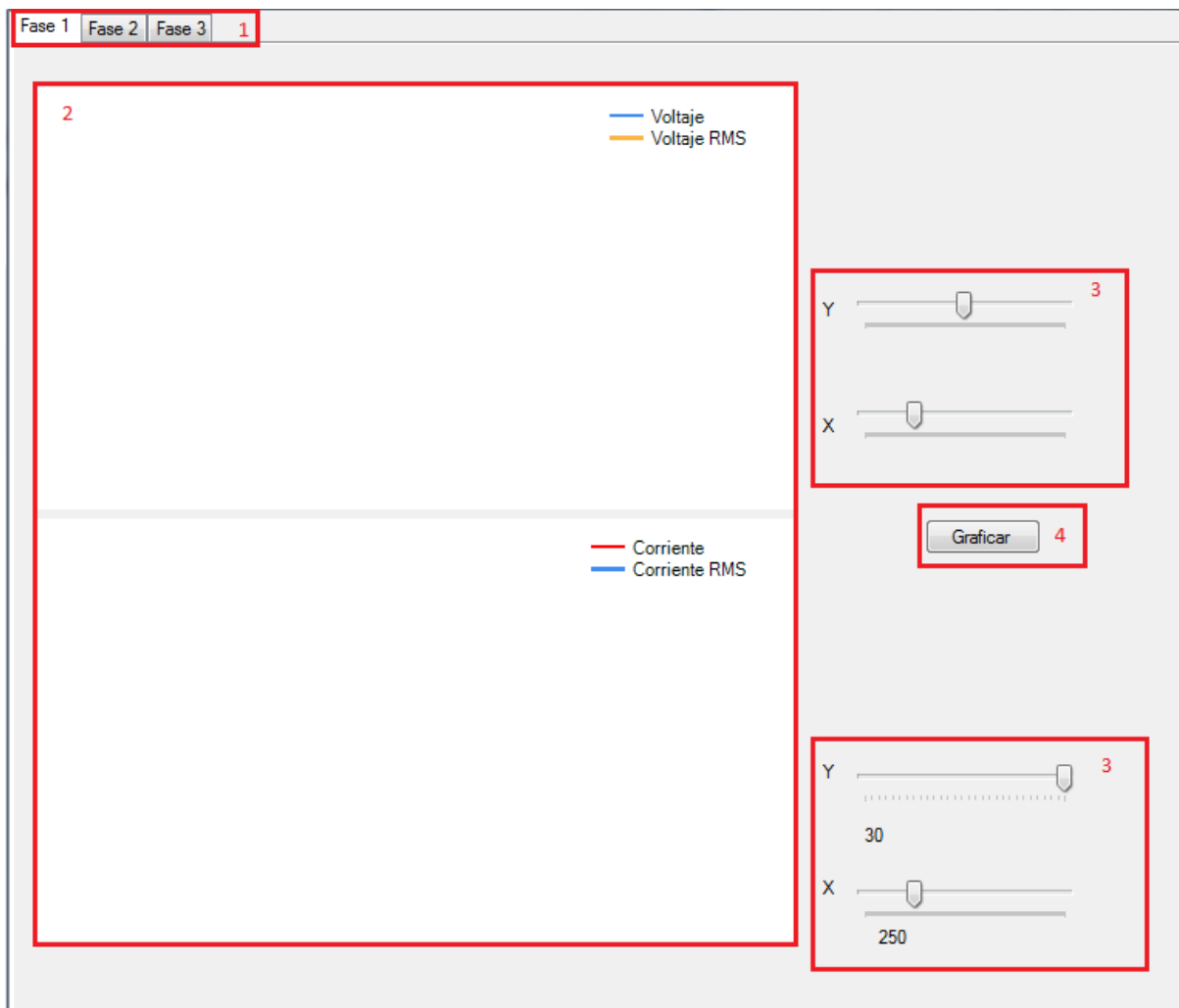


Ilustración 6 partes del software

1. Están separadas las partes gráficas en las 3 fases posibles, las pestañas marcadas habilitan la fácil navegación para visualizar los datos.
2. Es la parte gráfica en sí, donde se mostrará la onda lo mejor interpolada posible, esto también cambia con las distintas pestañas.
3. Cambia la escala del gráfico en cuestión, solo funciona cuando el programa ya está iniciado y recibiendo datos.
4. El botón graficar habilita con los datos adquiridos para que el gráfico comience a funcionar.

Parte Numérica

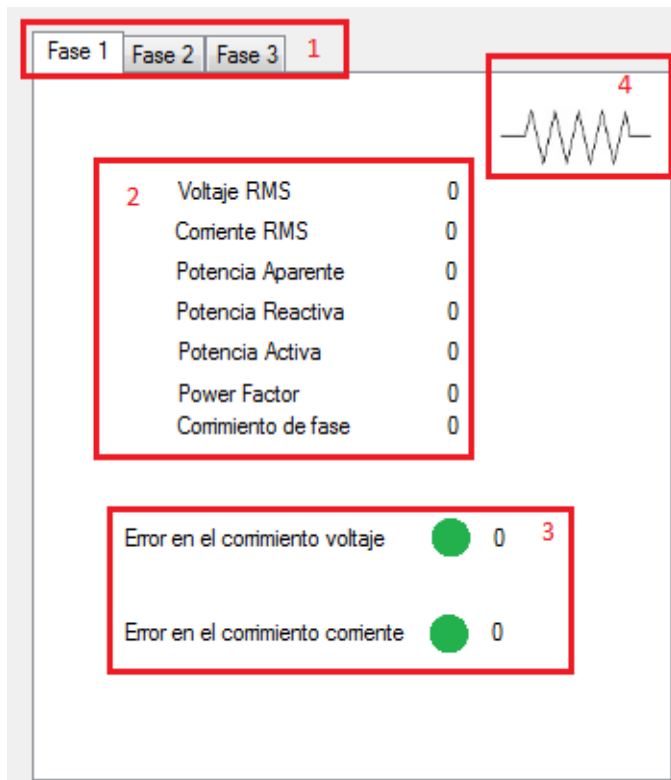


Ilustración 7 visualización de los datos

1. Fases discriminadas y fácilmente navegables.
2. valores numéricos instantáneos, como el voltaje RMS, Potencia y corrimiento de fase.
3. Errores de calibración exhibidos de manera intuitiva, no solo numéricamente ya que los círculos de color verde cambiaran a rojo en caso de que no esté calibrado, por defecto están en verde hasta conectar el programa.
4. Indica la naturaleza de la carga, puede variar entre inductivo, capacitivo y resistivo. La cantidad de corrimiento ya sea adelanto o retraso estará en el corrimiento de fase.

Guardado de datos

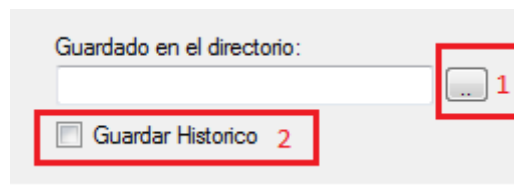


Ilustración 8 guardado de datos

1. Botón utilizado para buscar en el disco de la computadora el lugar donde se guardarán los datos, el programa lo guardará en formato separado por comas (csv) los datos de importancia, estos son corriente rms, voltaje rms, power factor, potencia activa, capacitiva y reactiva para cada fase en un formato separado por comas.
2. Para poder habilitar el guardado antes de conectar el programa se debe hacer click sobre la caja al lado de “guardar histórico”, de esta manera el programa comenzará el guardado en forma local cada 15 minutos de los datos obtenidos.

Conexión e inicio del programa



Ilustración 9 conexión a puertos

1. Actualizar puertos obtiene todos los puertos conectados y disponibles de la computadora, ya sean virtuales o físicos, para poder localizar los puertos luego es obligatorio hacer click en este botón primero.
2. Una vez que los puertos hayan sido actualizados deberían aparecer en el desplegable debajo de seleccionar puerto.
3. Cuando se eligió un puerto válido el botón de conectar comenzará a recibir datos seriales del mismo.
4. El botón de desconectar se habilitará una vez que el botón de conectar haya sido presionado y es para dejar de recibir datos vía puerto serie, por cuestiones prácticas también se desconecta la conexión serie cuando el programa se cierra.

Inicio del programa

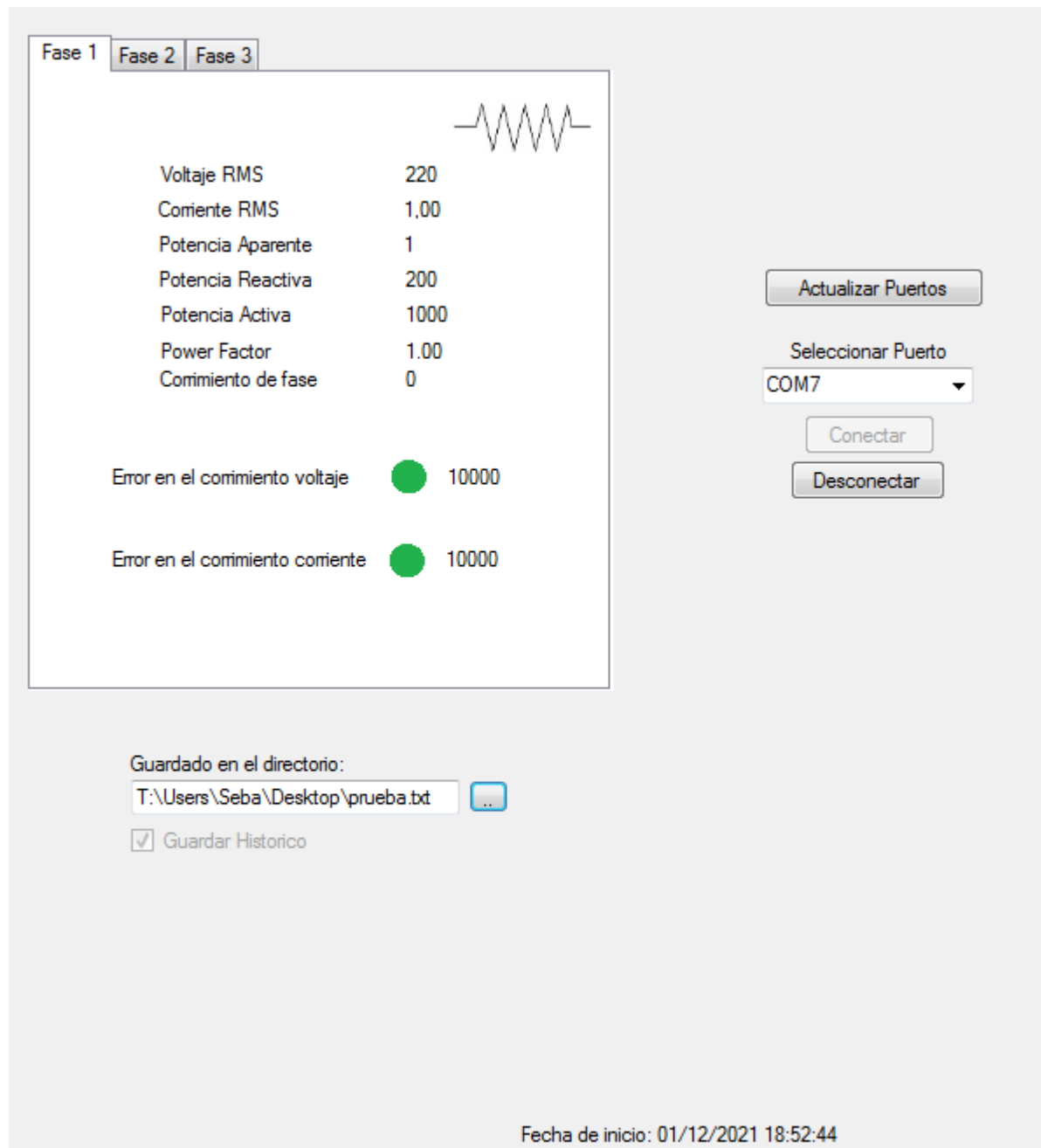


Ilustración 10 programa iniciado correctamente

Ejemplo de cómo se verá el programa una vez seleccionadas las opciones anteriores correctamente, los datos recibidos vía puerto serie son mostrados. También se muestra la fecha de inicio y la hora para tener un registro por si el usuario no lo toma.

Ejemplo de distintas vistas

Ejemplos de cómo se deberían ver las distintas cargas y los errores de calibración.

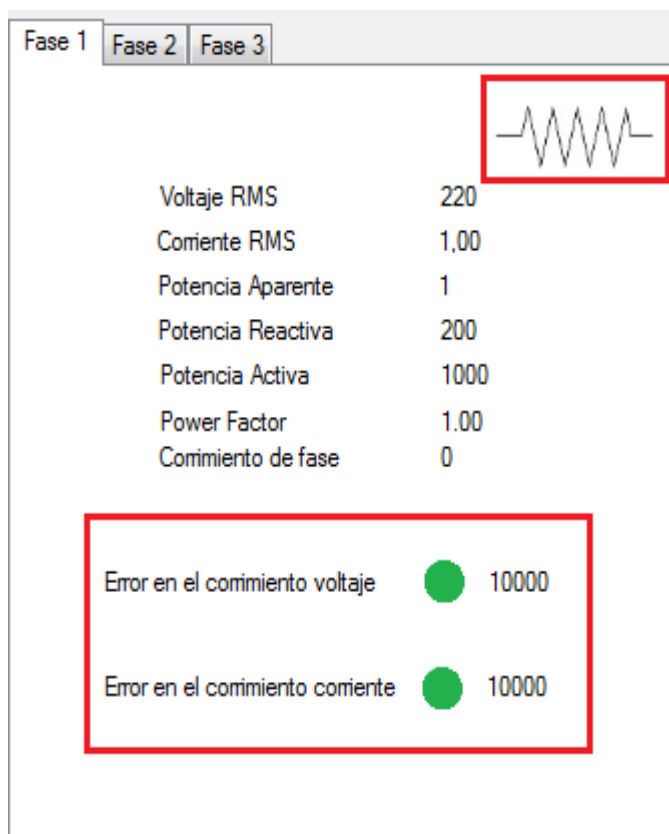


Ilustración 11 ejemplo 1 de posible vista

La fase 1 tiene naturaleza resistiva, por lo tanto, su corrimiento de fase es 0 y está bien calibrada, el objetivo de la calibración es que el número al lado del círculo verde esté lo más cerca posible de 10.000.

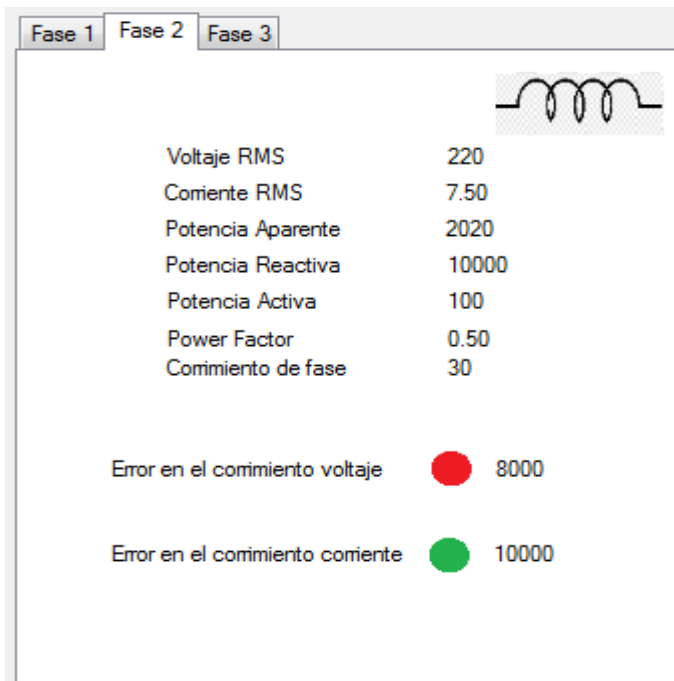


Ilustración 12 ejemplo 2 de posible vista

Existe un error de calibración, denotado con un círculo rojo, tiene una tolerancia de ± 1000 unidades que equivale a un 5% de error.

También se puede apreciar que la naturaleza es inductiva, aunque cuando esté descalibrado es aconsejable no hacer caso al resultado e intentar calibrar ya que el error puede modificar la naturaleza de la carga exhibida.

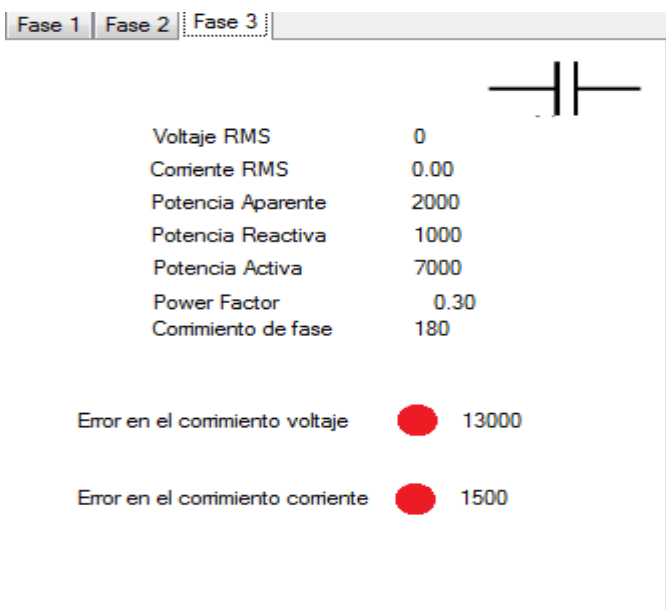


Ilustración 13 ejemplo 3 de posible vista

En la tercera fase están los 2 valores descalibrados y es de naturaleza capacitiva.

Gráfico del programa comunicado

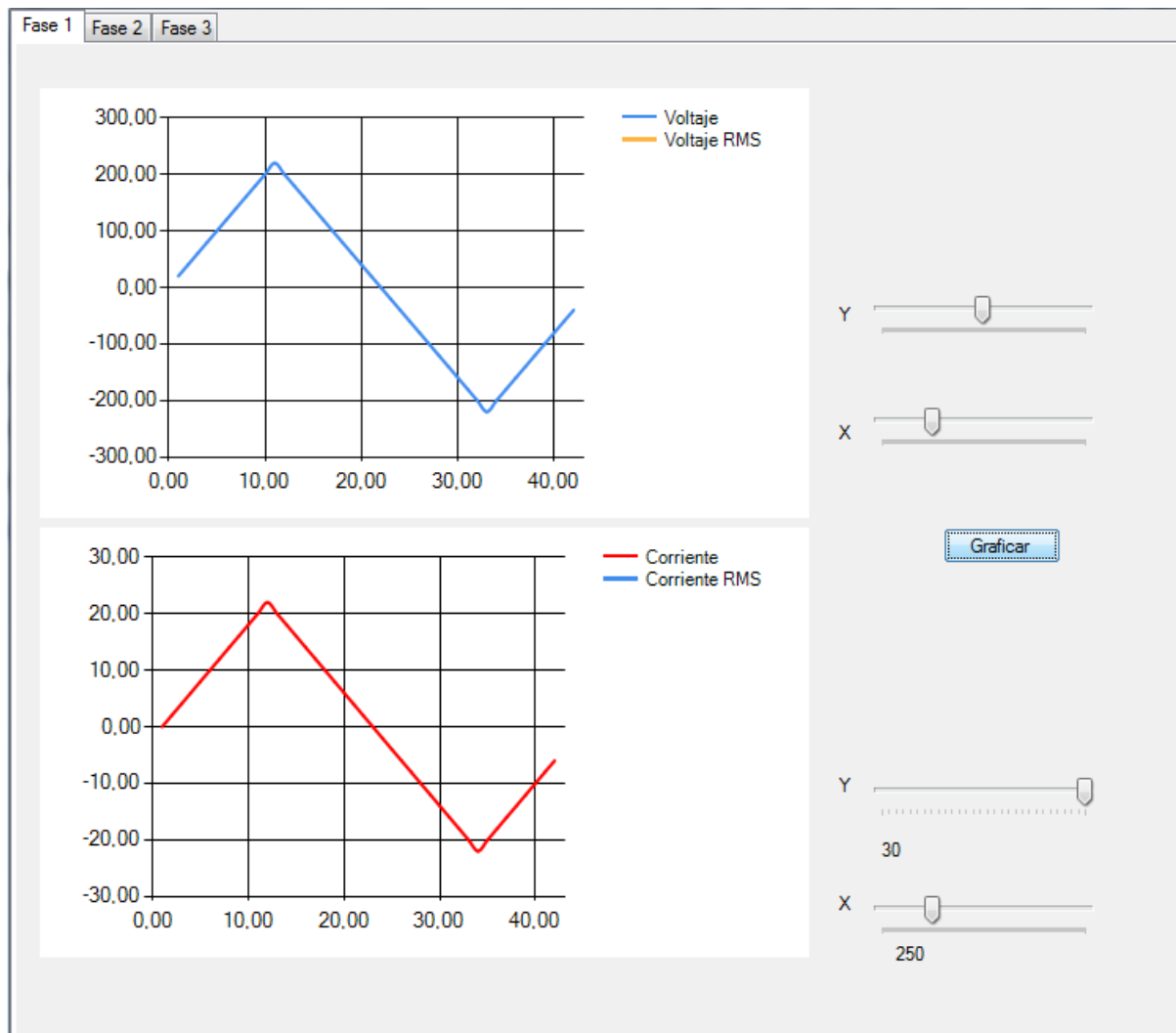


Ilustración 14 visualización de datos gráficamente

El eje Y es la magnitud medida y el eje X son las muestras tomadas, las barras se pueden mover libremente para modificar con qué cercanía se ven los gráficos, funcionan como deslizables.

Comunicación vía Ethernet

Es de fácil utilización ya que fue completamente resuelto de manera interna, lo único que se debe hacer es conectar el cable de ethernet a la entrada del Analizador de redes para el envío de datos cada 15 minutos.

Se recomienda reiniciar el Analizador de Redes con el botón “reset” a fin de evitar posibles fallos del programa.

Comunicación vía Wifi

Cuando se desea establecer una comunicación vía Wifi se debe buscar en las redes disponibles la llamada “PPS”, si se pregunta una contraseña es “12345”.

Una vez conectado a la red debe abrir cualquier explorador (recomendado Chrome) y buscar la dirección de IP proporcionada por el dispositivo.

Cuando visualizan todas las redes Wifi disponibles, debe elegir la que se quiera conectar y proporcionar la contraseña de la red de ser necesario, de no encontrar su red debe hacer click en “Scan” para buscar de nuevo todas las redes.

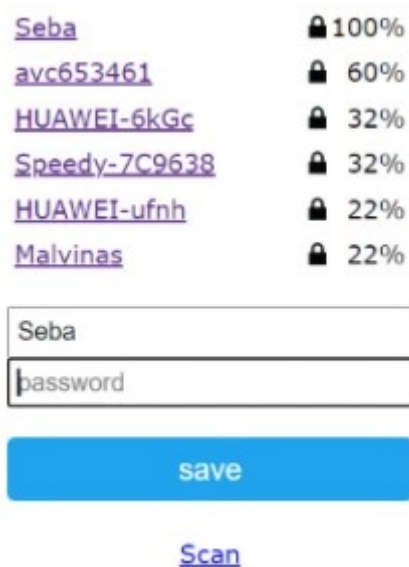


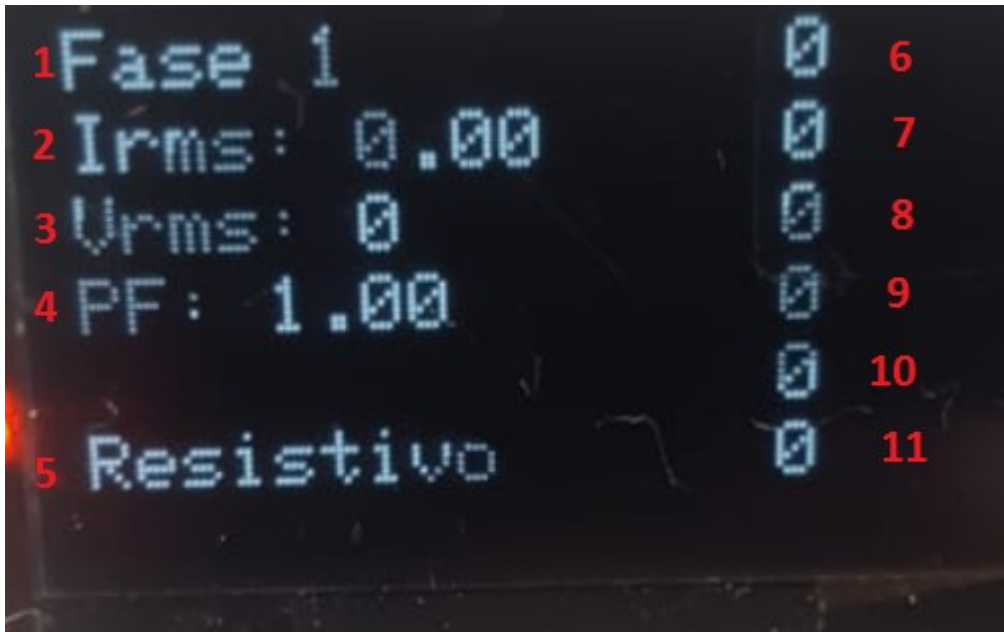
Ilustración 16 conexiones a wifi

Una vez completado este paso se guarda en la memoria del dispositivo la red y se comienza el envío de datos cada 15 minutos, además se alerta al usuario en la pantalla OLED que se encuentra conectado.

Se recomienda reiniciar el Analizador de Redes con el botón “reset” a fin de evitar posibles fallos del programa.

Lectura de la pantalla

La correcta lectura de la pantalla se realiza de la siguiente manera



1. Número de fase que se lee actualmente
2. Corriente RMS que se lee actualmente
3. Voltaje RMS que se lee actualmente
4. Power Factor
5. Indica si es Resistivo, Capacitivo o Inductivo aclarando si la corriente está en fase, adelantada, atrasada respecto al voltaje
6. Indica si hay error en la calibración de la onda de voltaje de la fase 1
7. Indica si hay error en la calibración de la onda de corriente de la fase 1
8. Indica si hay error en la calibración de la onda de voltaje de la fase 2
9. Indica si hay error en la calibración de la onda de corriente de la fase 2
10. Indica si hay error en la calibración de la onda del voltaje de la fase 3
11. Indica si hay error en la calibración de la onda de corriente de la fase 3

Los valores del punto 6 al 11 deben estar en 10.000 para estar correctamente calibrados y conseguir resultados óptimos, con diferencias mayores a 1000 respecto a este valor aparecen los números del costado de la pantalla, de lo contrario no y se sabe que esta bien calibrado el aparato cuando estos números no están.

Errores Conocidos

Errores Importantes:

- El corrimiento de fase hay errores de lectura a veces o aparece negativo en los casos resistivos puros.

Errores leves:

- Cuando el programa se desconecta con el botón “Desconectar” se traba y hay que finalizarlo, de cualquier forma, los datos se guardan por lo que solo es un error que hace un poco incómodo de usar el programa a veces.