



Ingeniería Electrónica

Trabajo final de carrera

Sincrofasores y monitoreo de red eléctrica trifásica

Autor

Terzaghi Benjamin

Tutores

Mg.Ing. Guillermo Friedrich

Ing. Adrian Laiuppa

Bahía Blanca | 27 de Febrero de 2024

Sincrofasores y monitoreo de energía eléctrica trifásica

Informe Proyecto Final

Contenido

AGRADECIMIENTOS	4
Resumen	5
1. Capítulo 1: Introducción	6
Generalidades	6
Objetivo del proyecto	7
Objetivos particulares	7
Justificación	7
Estado del arte	8
Alcances y limitaciones	10
Contribuciones	10
Estructura del proyecto	11
2. Capítulo 2: Sistemas de medición fasorial	13
Introducción	13
Normativa vigente	13
Normativa necesaria para homologar el dispositivo	13
Normativa regulatoria para el funcionamiento del dispositivo	18
Definición de Sincrofasor	21
Conceptos básicos	27
Valor RMS	27
Concepto de fasor	28
Concepto de fase	29
Muestreo digital	29
Teorema del muestreo	30
Medición de frecuencia	30
Cálculo de la frecuencia por cruce por cero	30
Sistema de posicionamiento global	33
Introducción	33
Arquitectura	34
Triangulación	35
Tiempo universal coordinado (UTC)	37
3. Capítulo 3: Hardware	38
Introducción	38
Diagrama de conexionado interno	38

Sincrofasores y monitoreo de energía eléctrica trifásica

Características de la placa STM32F407Vg – Discovery	41
Características de la placa ESP32-NodeMCU	48
Generador de señales UNI-T UTG2025A	51
Módulo GPS	56
Módulo receptor	56
Antena GPS	58
4. Capítulo 4: Firmware y Software	61
Introducción	61
Firmware de la STM32F407VG	61
Firmware de la ESP32	62
PDC (Processing Data Center - Centro de Procesamiento de Datos)	68
5. Capítulo 5: Ensayos	70
5.1 Introducción	70
5.2. Funcionamiento del instrumento	70
5.3 Medición de frecuencia	75
5.4 Valor RMS	76
5.5 Ensayos en redes reales	76
6. Conclusiones	83
7. Apéndice A: Código implementado en STM32F407VG	86
8. Apéndice B: Código implementado en ESP32-NodeMCU	110
Bibliografía	128
Referencias	129

Sincrofasores y monitoreo de energía eléctrica trifásica

AGRADECIMIENTOS

En este momento de reflexión y celebración, quiero expresar mi gratitud a todos aquellos que brindaron su apoyo y alentaron a que pueda seguir hasta lograr el tan ansiado objetivo.

En primer lugar, mis padres, Silvina y Gustavo, que con sus palabras fueron capaces de darme la templanza y tranquilidad para poder resolver cada desafío que se presentó a lo largo de toda mi preparación académica. Mi hermano, Lautaro, que con su actitud positiva y palabras fue capaz de convencerme de que estaba en el camino correcto.

A mis compañeros de clase, con quienes terminé formando una hermosa amistad, por todo el apoyo con palabras y con aportes para que cada uno de los desafíos presentados sean sorteados con éxito.

A mis amigos, quienes fueron capaces de dar los momentos de respiro y reflexión para tener más claridad al momento de sortear los distintos desafíos que se fueron presentando a lo largo de todo este proceso.

A todo mi equipo de trabajo, quienes fueron fundamentales al momento de cumplir con lo propuesto a lo largo de todo el proceso.

Por último, y no menos importante, a los profesores y directivos del departamento de Ingeniería Electrónica de la Universidad Tecnológica Nacional Facultad Regional Bahía Blanca, quienes fueron durante todo este proceso, pilares fundamentales para que pueda procesar y entender todo lo necesario para alcanzar los objetivos.

Este proyecto no es solo el fin de mi etapa académica, sino también el comienzo de un nuevo capítulo en mi vida. A todos los que formaron parte de esta maravillosa etapa de mi vida, gracias de corazón por su apoyo inquebrantable. Este logro es tan suyo como mio, y espero poder seguir compartiendo maravillosos momentos en los años siguientes.

Con eterna gratitud.

Benjamin Terzaghi

Sincrofasores y monitoreo de energía eléctrica trifásica

Resumen

El presente trabajo trata sobre el desarrollo de un sistema diseñado para efectuar mediciones de la red eléctrica, con el objetivo de poder detectar eventos perturbadores en el régimen estacionario, que pueden ser causados por diferentes dispositivos conectados a la red. De esta manera se destaca la importancia de la conversión de las ondas de energía eléctrica en sincrofasores.

Un sincrofasor es un fasor en su forma pura, al cual se le asigna la marca temporal en donde fueron efectuadas las mediciones de la onda temporal, para luego procesarlas y obtener su representación fasorial para obtener mediciones sincronizadas y observar las fases medidas en el mismo instante de tiempo.

El presente proyecto consta de dos partes:

- *PMU (Unidad de Medición Sincrofasorial): Es la encargada de tomar las mediciones de la red eléctrica y calcular los valores RMS y ángulo de cada fase, tanto de tensión como de corriente. Los valores calculados son enviados a una PDC (Centro de Procesamiento de Datos). A cada uno de los datos calculados se le aplica una marca de tiempo dada por el módulo GPS, específicamente el tramo UTC que es la marca global de tiempo.*

La PMU fue desarrollada en base a un microcontrolador STM32F407VG para la captura de muestras con conversores A/D de 12 bits y un módulo basado en un SoC ESP32 como unidad de cálculo y comunicación con el PDC, utilizando además un módulo de posicionamiento global (GPS) para obtener la marca de tiempo UTC para cada una de las mediciones realizadas y así poder sincronizarlas,

- *PDC (Centro de Procesamiento de Datos): Esta parte del proyecto es la que recibe los valores calculados por la PMU y los presenta en dos diagramas fasoriales, uno para la tensión y otro para la corriente, ambos fasores con su marca temporal. Gracias a ésta, la PDC la compara con una onda patrón ideal y de esta comparación surge la detección de las diferentes perturbaciones que se pueden tener en una fase del tendido eléctrico en un determinado instante de tiempo. En este caso, se guarda el dato del fasor (Valor RMS, ángulo de fase, instante de tiempo) para que quede registro de lo sucedido y además se activa una alarma la cual va asociada a una alarma de aviso. El desarrollo de éste fue realizado en Python, con una plantilla de gráficos fasoriales, al cual se le agregó un socket de comunicación UDP para hacer posible la recepción de las mediciones tanto por WiFi o Ethernet cableada de la PMU a la PC donde está la PDC.*

Palabras clave: Sincrofasor - fasor - PMU - PDC

Sincrofasores y monitoreo de energía eléctrica trifásica

1. Capítulo 1: Introducción

Generalidades

Actualmente, y como resultado del aumento del uso de convertidores estáticos, la contaminación en las redes eléctricas se ha incrementado. Por tal motivo los instrumentos convencionales basados en la medición de señales sinusoidales han quedado obsoletos y surge la necesidad de utilizar sistemas de medición que logren sensor con mayor precisión las señales de las redes eléctricas.

Los fasores proveen medición directa de las unidades fundamentales de voltaje y corriente, incluyendo magnitud y ángulo. El ángulo es medido con respecto a un tiempo de referencia. Los datos no son normalmente coincidentes en el tiempo, pero son recolectados sobre un periodo de algunos segundos. Bajo condiciones normales cercanas a las de operación en estado estacionario, estos datos son aproximadamente coincidentes.

Para llevar a cabo la medición de fasores en una red interconectada se requiere de una fuente de sincronización, que puede ser local o global; es decir que el muestreo de datos en dos puntos de la red está sincronizado. Si además se cuenta con el Tiempo Universal Coordinado (UTC, *Universal Time Coordinated*) del proceso, entonces se puede enviar la medición a un lugar remoto acompañándolo del estampado de tiempo. Al alinear el estampado de tiempo de las mediciones obtenidas de diferentes nodos, se obtienen mediciones simultáneas. El sistema de posicionamiento global (GPS, *Global Positioning System*) permite realizar la sincronización de las mediciones.

Dado que existen desfasajes entre las señales de voltaje y de corriente en la transmisión de la energía eléctrica a través de los diferentes elementos de la red eléctrica, se requiere un dispositivo que provea mediciones sincronizadas en dicha red. Las mediciones sincronizadas de fasores abren la posibilidad de sintetizar el proceso de estimación de estado, es decir que permite obtener un diagnóstico rápido y preciso del estado de la red ensayada.

Sincrofasores y monitoreo de energía eléctrica trifásica

Las Unidades de Medición Fasorial (PMU, *Phasor Measurement Units*) permiten obtener mediciones fasoriales de corriente y de voltaje sincronizadamente en diferentes puntos de la red y transmitir la información de los fasores al centro de control, donde pueden ser comparadas, evaluadas y procesadas.

Objetivo del proyecto

Poner en evidencia la importancia de los sincrofasores para el monitoreo de las líneas de una red trifásica mediante el diseño de una unidad de medición sincrofasorial basada en microcontrolador, que sea capaz de proporcionar mediciones fasoriales de corriente y voltaje sincronizadas con el tiempo global.

Objetivos particulares

- ◇ Desarrollar los algoritmos necesarios para la medición de fasores y una comunicación confiable en cuanto a latencia y precisión de los datos, en entornos de programación ESP32-NodeMCU (Arduino IDE).
- ◇ Desarrollar un correcto muestreo de las ondas de tensión y corriente mediante los conversores de la placa STM32F4-Discovery (CubeIDE – CubeMX), y asignar a cada grupo de datos recolectados el estampado de tiempo UTC (Universal Time Cordinated – Tiempo Universal Coordinado) dado por el módulo GPS U-BLOX NEO6M.

Justificación

En todas las aplicaciones del sistema eléctrico, se tiene como primer requerimiento la medición de las variables que se necesitan para realizar el análisis y el control de dicho sistema. Las aplicaciones de monitoreo y protección de sistemas requieren el intercambio de información entre los equipos y las aplicaciones. En cada caso, los procedimientos utilizados para el cálculo, registro y transmisión de fasores de voltaje y corriente se realizan de forma diferente, en cuanto a

Sincrofasores y monitoreo de energía eléctrica trifásica

protocolos de comunicación y algoritmos de cálculo principalmente, con el fin de conseguir el resultado más preciso y poder visualizarlo en el menor tiempo posible.

La mayor parte de las herramientas existentes estiman la distribución de fasores a partir de medidas no sincronizadas realizadas en diversos puntos de la red. La medición de fasores sincronizados tiene un gran potencial de aplicación en sistemas eléctricos de potencia, tales como la medición de frecuencia y magnitud de voltaje y corriente, estimación de estado, así como control y monitorización de la red. La medición de los fasores de voltaje en tiempo real mejora la capacidad de respuesta de los operadores y permite el desarrollo de sistemas de prevención de inestabilidades y de respuesta automática más rápida y eficaz.

En el caso de líneas críticas, las mediciones se sincronizan mediante algún mecanismo que brinda esquemas de protección diferencial para la detección de fallas. Así, se considera que la protección diferencial es la forma de protección más fiable. Este método de protección se refiere a un método utilizado en ingeniería eléctrica para proteger equipos y sistemas eléctricos contra fallas eléctricas, especialmente aquellas relacionadas con cortocircuitos. La idea principal es comparar las corrientes de entrada y salida de un equipo o sistema y activar una protección si hay alguna diferencia significativa entre ellas. En el futuro, las unidades PMU podrían ser utilizadas para suministrar protección diferencial.

Entonces, se justifica el siguiente trabajo mediante la utilización de las tecnologías mencionadas para el desarrollo del PMU, sabiendo que los medidores fasoriales sincronizados vía satélite no se fabrican en Argentina.

Estado del arte

Phadke, Thorp y Adamiak (1983) describen los fundamentos teóricos y el proceso computacional para el cálculo de fasores en tiempo real. La Transformada Discreta de Fourier (DFT) se utiliza para la obtención de fasores.

El Tecnológico de Virginia (1988) construyó la primera Unidad de Medición Fasorial (PMU). Dicha

Sincrofasores y monitoreo de energía eléctrica trifásica

unidad se sincroniza mediante el Sistema Satelital de Posicionamiento Global (GPS). Esta PMU empieza a ser comercializada por la compañía Macrodyne.

Phadke (1993) describe la PMU y menciona las diferentes formas de sincronización, destacando la sincronización mediante el Sistema de Posicionamiento Global con una exactitud de 1 ms.

En 1994 se utilizan PMUs sincronizadas vía satélite para la obtención de fasores en la red de la compañía de Potencia Georgia. También, se utilizan para confirmar la ecuación del flujo de potencia en líneas de transmisión.

Yutaka Ota y Hideki Fujita (2000) utilizan unidades de medición fasorial para evaluar la estabilidad de un sistema de potencia. Se utiliza un microcontrolador DSP y la fuente de sincronización es el pulso por segundo proporcionado por un receptor del Sistema de Posicionamiento Global.

Rafael San Vicente (2004) desarrolla en México una PMU sincronizada vía satélite utilizando un microcontrolador DSP de Texas Instruments.

Carlos Cuvas (2005) desarrolla en México otra PMU sincronizada vía satélite, utilizando un microcontrolador DSP de Freescale.

Blanca Verónica Hernández (2009) desarrolla en México otra PMU sincronizada vía satélite, utilizando un microcontrolador de ColdFire.

Abraham Valdiosera Marroquín (2013) desarrolla un medidor inteligente con sistema de comunicación bidireccional, utilizando un microcontrolador ColdFire junto con un receptor GPS para obtener mediciones fasoriales sincronizadas. Incluye una etapa de comunicación inalámbrica mediante Zig-Bee.

David Sebastián Cárdenas (2015) desarrolla un dispositivo de medición inteligente para la detección de robo de energía, cumpliendo con el estándar IEEE 1459 y los requisitos de la clase “M”, basado en el estándar IEEE C37.118. Incluye un receptor GPS.

Actualmente, la compañía SEL (Schweitzer Engineering Laboratories) es una empresa dedicada al desarrollo y fabricación de equipos y soluciones para la protección, automatización y control de sistemas eléctricos. Esta cuenta con dispositivos de medición de fasores sincronizados instalados en el

Sincrofasores y monitoreo de energía eléctrica trifásica

sistema eléctrico de potencia de los Estados Unidos de América, Monterrey y la Ciudad de México. Es deseable idear un sistema similar para el sistema interconectado en Argentina junto con países de gran porte como son Brasil, Chile, Uruguay, etc.

Alcances y limitaciones

La PMU implementada será capaz de realizar mediciones sincrofasoriales de voltaje y de corriente trifásicas con formas de onda sinusoidales a una frecuencia nominal de 50 Hz. El rango de las mediciones será del 20% al 120% de la señal de voltaje permitido en las entradas del microcontrolador y del 20% al 200% de la señal de corriente permitida en las entradas del microcontrolador, señales que son entregadas por el micro controlador generador de señales, con lo que cumple con la condición de estado estacionario del Estándar IEEE C37.118.1, para luego llevar al prototipo a un ensayo real.

Empleando los fasores se mide el desfase entre voltajes y corrientes, implementando también la medición de la frecuencia. La sincronización se realizará mediante la trama de datos que proporciona el módulo del Sistema de Posicionamiento Global. El GPS proporciona el Tiempo Universal Coordinado (UTC) que, junto con las mediciones, se transmite al PDC para poder ser visualizado el estado de la red trifásica.

Contribuciones

- ◇ Desarrollar una Unidad de Medición Sincrofasorial con los micro controladores mencionados, sincronizada vía satélite, que sirve para implementar sistemas de medición y control, estimación de estado, monitoreo o predicción de estabilidad de una red interconectada.
- ◇ Resultados de los ensayos realizados mediante generadores de señales, que sirven como un emulador de la red trifásica.
- ◇ Resultados de los ensayos en laboratorio conectado a la red trifásica, con cargas reales,

Sincrofasores y monitoreo de energía eléctrica trifásica

que permitan contrastarlos con equipos certificados.

Estructura del proyecto

En el presente capítulo (Capítulo 1) se aborda el problema a resolver, mencionando el objetivo, la justificación, los aportes y la estructura de la tesis. También, se incluye el estado del arte de la misma. Se da una breve reseña general de lo que se conoce, de los elementos prioritarios utilizados en la solución del problema; así mismo, se plantea el alcance y las limitaciones del trabajo. Se hace la descripción del problema a resolver, indicando sus antecedentes y la situación actual del tema. Finalmente, se describe la estructura del trabajo por capítulos.

En el Capítulo 2, se presentan las bases teóricas acerca de los conceptos que conciernen a la medición fasorial sincronizada, entre los cuales se encuentran cálculos específicos y el Sistema de Posicionamiento Global.

En el Capítulo 3, se hace la descripción de las características principales y elementos de funcionamiento de los elementos del Hardware utilizado en la implementación del medidor fasorial. Junto a esto, se describe la interacción de los elementos mencionados y la interconexión de los mismos. También se muestran las características de la placa de desarrollo empleada como una imagen de la conexión entre la tarjeta de desarrollo, la PC y el módulo receptor GPS.

En el Capítulo 4 se presentan los elementos de software utilizados, así como el diagrama en bloques de una PMU y el diagrama conceptual a implementar. Se muestran las funciones, los cálculos para los diferentes parámetros, y se describe el ambiente de desarrollo del microcontrolador.

En el Capítulo 5 se realiza el reporte de los resultados de las pruebas realizadas a la PMU, verificando que cumple con los requerimientos necesarios, ante diferentes condiciones en estado estacionario.

En el Capítulo 6, se reportan las conclusiones de la implementación y las recomendaciones para

Sincrofasores y monitoreo de energía eléctrica trifásica

trabajos futuros en esta línea de estudio.

En la parte final del trabajo se incluyen los apéndices. En el apéndice A se tienen los códigos de programación utilizados en el desarrollo de este trabajo en cuanto al muestreo de las líneas de la red trifásica, tanto de tensión como de corriente. Por último, en el Apéndice B se tiene el código de programación para el procesamiento y el envío de las mediciones al PDC.

2. Capítulo 2: Sistemas de medición fasorial

En este capítulo, se abordan y analizan los diferentes conceptos relacionados con la medición fasorial, comenzando con los métodos que se utilizan en sistemas eléctricos de potencia para calcular fasores de corriente y voltaje, y que aportan la magnitud y la fase de una señal. Se describe el uso de cálculos específicos para obtener los fasores y el funcionamiento del Sistema de Posicionamiento Global para la sincronización de medición de dichos fasores.

Introducción

Para monitorear y diagnosticar sistemas eléctricos de potencia, la medición fasorial es muy importante. Los fasores representan la amplitud y la fase de una señal sinusoidal en estado estable (frecuencia, amplitud y fase constante).

Los cálculos de magnitud y fase corresponden a la transformación de coordenadas rectangulares a polares, cuya representación permite la visualización de los desplazamientos de los ángulos de fase.

Con formas de onda en tiempo real, es necesario definir un tiempo de referencia para medir los ángulos de fase en forma sincronizada. El estándar IEEE 1344-1995 define el inicio de un segundo como el tiempo de referencia para establecer el valor del ángulo del fasor.

Las posibilidades de determinar los fasores a lo largo del sistema de distribución o transmisión de energía en un momento dado, abre una puerta a la solución de estos problemas. Por ello, es de suma importancia analizar y estudiar las técnicas de estimación de fasores.

Normativa vigente

Normativa necesaria para homologar el dispositivo

En términos de construcción y funcionamiento, el equipo debe cumplir con varias normas y regulaciones que aseguren su calidad, seguridad y fiabilidad. Algunas de estas normas y regulaciones son las siguientes:

Normas de seguridad eléctrica: El equipo debe cumplir con las normas de seguridad eléctrica aplicables en la región donde se utiliza. Estas normas establecen los requisitos para la seguridad eléctrica de los equipos y sistemas eléctricos, con el objetivo de prevenir accidentes y lesiones.

Sincrofasores y monitoreo de energía eléctrica trifásica

Las normas de seguridad eléctrica varían según la región y el país, pero en general establecen los requisitos para la seguridad eléctrica de los equipos y sistemas eléctricos, con el objetivo de prevenir accidentes y lesiones. Algunas de las normas de seguridad eléctrica más comunes son las siguientes:

- Norma IEC 60364: Esta norma establece los requisitos para la seguridad eléctrica en las instalaciones eléctricas de baja tensión. Define los requisitos de diseño, instalación y mantenimiento de las instalaciones eléctricas, incluyendo los requisitos para la protección contra contactos directos e indirectos, la selección y protección de los conductores eléctricos, entre otros.
- Norma NFPA 70: La NFPA 70, también conocida como el Código Eléctrico Nacional (NEC) en los Estados Unidos, establece los requisitos para la instalación eléctrica segura en edificios y estructuras. Define los requisitos de diseño, instalación y mantenimiento de las instalaciones eléctricas, incluyendo los requisitos para la protección contra contactos directos e indirectos, la selección y protección de los conductores eléctricos, entre otros.
- Norma IEC 61010: Esta norma establece los requisitos de seguridad para los equipos eléctricos de medida, control y laboratorio. Define los requisitos para la protección contra contactos eléctricos, sobrecargas, cortocircuitos, entre otros.
- Norma UL: La norma UL (Underwriters Laboratories) es una norma de seguridad eléctrica utilizada en los Estados Unidos. Define los requisitos para la seguridad eléctrica de los equipos y sistemas eléctricos, incluyendo los requisitos para la protección contra contactos directos e indirectos, la selección y protección de los conductores eléctricos, entre otros.

Normas de calidad y confiabilidad: El equipo debe cumplir con las normas de calidad y confiabilidad aplicables en la industria eléctrica, como la norma ISO 9001 para sistemas de gestión de calidad y la norma ISO 17025 para laboratorios de ensayo y calibración.

- La norma ISO 9001 es un estándar internacional que establece los requisitos para un sistema de gestión de la calidad (SGC). Esta norma especifica los criterios para un SGC que puede ser utilizado por una organización de cualquier tamaño, tipo o sector, ya sea público o privado.

Los rangos que establece la norma ISO 9001 son los siguientes:

- Requisitos generales: establece los principios generales del SGC y define los términos utilizados en la norma.
- Responsabilidad de la dirección: define los roles y responsabilidades de la alta dirección en la implementación y mantenimiento del SGC.

Sincrofasores y monitoreo de energía eléctrica trifásica

- o Gestión de los recursos: aborda la asignación de recursos y la gestión del personal y la infraestructura necesarios para implementar el SGC.
- o Realización del producto o servicio: define los requisitos relacionados con el proceso de producción o prestación del servicio, desde la planificación hasta la entrega.
- o Medición, análisis y mejora: establece los requisitos para el seguimiento, medición, análisis y mejora continua del SGC.

La norma ISO 9001 establece los requisitos para un SGC que ayuda a las organizaciones a mejorar la eficacia y eficiencia de sus procesos, aumentar la satisfacción del cliente y mejorar la rentabilidad. La norma se aplica a cualquier organización que desee demostrar su capacidad para proporcionar productos o servicios que cumplan con los requisitos del cliente y los legales y reglamentarios aplicables.

- La norma ISO 17025 es un estándar internacional que establece los requisitos generales para la competencia técnica de los laboratorios de ensayo y calibración. Esta norma se aplica a laboratorios que realizan pruebas o calibraciones y que necesitan demostrar su capacidad para producir resultados técnicamente válidos y precisos.

Los rangos que establece la norma ISO 17025 son los siguientes:

- o Requisitos generales: establece los principios generales para la gestión de la calidad y la competencia técnica de los laboratorios.
- o Requisitos de gestión: aborda los requisitos para la gestión del sistema de calidad, incluyendo la documentación, el control de registros y la revisión del sistema.
- o Requisitos técnicos: define los requisitos técnicos para la realización de pruebas y calibraciones, incluyendo la validación de métodos, la trazabilidad y la incertidumbre de medición.
- o Selección y uso de equipos: establece los requisitos para la selección, calibración y mantenimiento de los equipos utilizados en las pruebas y calibraciones.
- o Muestreo: define los requisitos para la selección de muestras y la planificación del muestreo.
- o Manipulación de objetos de ensayo o calibración: define los requisitos para el manejo y almacenamiento de los objetos de ensayo o calibración.

Sincrofasores y monitoreo de energía eléctrica trifásica

- o Aseguramiento de la calidad de los resultados: establece los requisitos para la validación de los resultados, el control de calidad interno y el uso de ensayos inter laboratorios.

La norma ISO 17025 establece los requisitos para la competencia técnica y la calidad de los laboratorios de ensayo y calibración. La norma se aplica a laboratorios de cualquier tamaño y sector que realizan pruebas o calibraciones y que necesitan demostrar su capacidad para producir resultados técnicamente válidos y precisos.

Normas de medio ambiente: El equipo debe cumplir con las normas de medio ambiente aplicables en la región donde se utiliza, como la norma ISO 14001 para sistemas de gestión ambiental. Estas normas establecen los requisitos para minimizar el impacto ambiental de los equipos y sistemas eléctricos.

- La norma ISO 14001 es un estándar internacional que establece los requisitos para un sistema de gestión ambiental (SGA). Esta norma se enfoca en cómo una organización puede gestionar sus actividades, productos y servicios de manera que reduzcan su impacto ambiental y cumplan con los requisitos legales y reglamentarios aplicables.

Los rangos que establece la norma ISO 14001 son los siguientes:

- o Requisitos generales: establece los principios generales del SGA y define los términos utilizados en la norma.
- o Política ambiental: define los requisitos para la definición, implementación y comunicación de la política ambiental de la organización.
- o Planificación: aborda los requisitos para la identificación y evaluación de los aspectos ambientales de las actividades, productos y servicios de la organización, así como la definición de objetivos, metas y planes de acción para reducir su impacto ambiental.
- o Implementación y operación: establece los requisitos para la implementación y operación del SGA, incluyendo la estructura organizativa, la formación del personal, el control operativo y la gestión de emergencias.
- o Verificación y revisión: define los requisitos para la medición, seguimiento y evaluación del desempeño ambiental de la organización, así como la revisión del SGA por la dirección y la realización de auditorías internas.
- o Mejora continua: aborda los requisitos para la mejora continua del SGA, incluyendo la corrección de las no conformidades y la implementación de acciones preventivas.

Sincrofasores y monitoreo de energía eléctrica trifásica

La norma ISO 14001 establece los requisitos para un SGA que ayuda a las organizaciones a reducir su impacto ambiental y cumplir con los requisitos legales y reglamentarios aplicables. La norma se aplica a cualquier organización que desee gestionar sus actividades, productos y servicios de manera que reduzcan su impacto ambiental y mejoren su desempeño ambiental.

Normas de diseño y fabricación: El equipo debe cumplir con las normas de diseño y fabricación aplicables en la industria eléctrica, como la norma IEC 61010 para equipos eléctricos de medida, control y laboratorio. Estas normas establecen los requisitos para el diseño, fabricación y prueba de los equipos eléctricos, con el objetivo de asegurar su calidad, seguridad y fiabilidad

- La norma IEC 61010 es un estándar internacional que establece los requisitos de seguridad para equipos eléctricos utilizados en laboratorios, plantas industriales y otras aplicaciones similares. La norma se aplica a equipos que generan, utilizan o miden energía eléctrica, y tiene como objetivo garantizar la seguridad de las personas y los bienes en estas aplicaciones.

Los rangos que establece la norma IEC 61010 son los siguientes:

- o Requisitos generales: establece los principios generales de seguridad para equipos eléctricos, incluyendo los requisitos para la documentación y la identificación del equipo.
- o Requisitos de protección: define los requisitos para la protección de las personas y los bienes contra los riesgos eléctricos, mecánicos y térmicos.
- o Requisitos de puesta a tierra: establece los requisitos para la puesta a tierra del equipo y los requisitos para los dispositivos de protección contra sobretensiones.
- o Requisitos de la fuente de alimentación: define los requisitos para la fuente de alimentación del equipo, incluyendo la protección contra sobrecarga y cortocircuitos.
- o Requisitos de los sistemas de control: establece los requisitos para los sistemas de control del equipo, incluyendo la protección contra el mal funcionamiento y los requisitos de los dispositivos de seguridad.
- o Requisitos de los equipos de medición: define los requisitos para los equipos de medición utilizados en el equipo, incluyendo los requisitos de precisión y la calibración.

La norma IEC 61010 establece los requisitos de seguridad para equipos eléctricos utilizados en laboratorios, plantas industriales y otras aplicaciones similares. La norma se aplica a equipos

Sincrofasores y monitoreo de energía eléctrica trifásica

que generan, utilizan o miden energía eléctrica, y tiene como objetivo garantizar la seguridad de las personas y los bienes en estas aplicaciones.

Normativa regulatoria para el funcionamiento del dispositivo

- IEEE Std. 60255-118-1-2018

Esta norma se enfoca en la protección de sistemas eléctricos de potencia contra fallas y perturbaciones, y establece los requisitos mínimos que los equipos de protección deben cumplir para garantizar su correcto funcionamiento.

A continuación, se detallan los parámetros que se deben cumplir según esta norma:

Funcionamiento adecuado en todas las condiciones: el equipo debe ser capaz de funcionar de manera adecuada en todas las condiciones a las que pueda estar expuesto, ya sean ambientales o de funcionamiento.

Sensibilidad: el equipo debe ser capaz de detectar y responder a todas las fallas y perturbaciones para las que está diseñado.

Tiempo de operación: el equipo debe ser capaz de operar en un tiempo adecuado para evitar daños al sistema eléctrico y minimizar las interrupciones de suministro eléctrico.

Capacidad de coordinación: el equipo de protección debe estar diseñado para coordinarse con otros equipos de protección dentro del sistema eléctrico, para evitar la activación simultánea de múltiples equipos y minimizar los impactos en el sistema.

Fiabilidad: el equipo debe ser confiable y estar diseñado para minimizar la posibilidad de fallas en su operación.

Seguridad: el equipo debe cumplir con los requisitos de seguridad eléctrica y mecánica para proteger a las personas y al equipo de cualquier riesgo eléctrico o mecánico.

Compatibilidad electromagnética (EMC, Electromagnetic Compatibility): el equipo debe ser diseñado para cumplir con los requisitos de compatibilidad electromagnética, para evitar interferencias electromagnéticas con otros equipos y sistemas.

Estos son algunos de los principales parámetros que se deben cumplir según la norma IEEE Std. 60255-118-1-2018 para garantizar el correcto funcionamiento de los equipos de protección de sistemas eléctricos de potencia. Es importante recordar que la conformidad con esta norma puede variar dependiendo del tipo de equipo de protección, por lo que se deben seguir las especificaciones de cada equipo y la interpretación de la norma realizada por los especialistas.

Sincrofasores y monitoreo de energía eléctrica trifásica

- Estándar IEEE C37.118: La medición, protección y sistemas de control basados en el uso de la computadora digital, llegaron a ser características comunes de las subestaciones eléctricas de potencia. Dicho sistema usa el muestreo de datos para calcular diversas cantidades, tales como el fasor de corriente y de voltaje. En muchas funciones de adquisición de datos y sistemas de protección, los fasores y su utilidad se incrementarán en un futuro, debido a la referencia que tienen en base a un tiempo común. Esto se emplea para sincronizar la señal de entrada procesada en varios sitios de medición.

De varios fasores sincronizados se obtiene un grupo de mediciones simultáneas, que proporciona un método mejorado en gran escala para analizar los fenómenos dinámicos del sistema de potencia, a fin de mejorar el monitoreo, protección y control del mismo sistema. En el estándar IEEE C37.118 se proporcionan definiciones claras de varios conceptos, entre los cuales se encuentra el concepto de fasor, sincrofasor, errores, etc. Además, determina los elementos con los que debe contar el sistema de comunicación para que el instrumento envíe los datos de las mediciones que realiza.

Esta norma trata las mediciones fasoriales sincronizadas que se utilizan en los sistemas eléctricos de potencia. Define la medición, proporciona un método de cuantificación de las mediciones y calidad de las pruebas especificadas. Define también, formatos para la transmisión de datos en tiempo real.

Los recientes desarrollos en el campo de la adquisición de datos en sistemas de potencia proveen al usuario la habilidad de calcular cuantitativamente los fasores en tiempo real. Diferentes enfoques de hardware y software están siendo implementados para registrar, transmitir y analizar estas mediciones, tanto en tiempo real como después del evento. El estándar para sincrofasores ayudará a asegurar máximos beneficios para la medición de fasores y permite el intercambio de información entre una amplia variedad de sistemas para los usuarios tanto en tiempo real como en mediciones de fasores fuera de línea.

Para el propósito de este documento, los siguientes términos y definiciones aplican.

- ◇ **Anti-alias:** El proceso de filtrado de una señal para eliminar los componentes de esa señal cuya frecuencia es igual o superior a la mitad de la tasa de Nyquist (velocidad de muestreo). Si no se elimina, estos componentes de la señal aparecen como un

Sincrofasores y monitoreo de energía eléctrica trifásica

- componente de menor frecuencia (un alias).
- ◇ **Tiempo Universal Coordinado (UTC, Universal Time Coordinated):** El tiempo del día al que se encuentra el primer meridiano de la Tierra (0° longitud). Es distribuido por varios medios, incluyendo el sistema GPS.
 - ◇ **IEEE punto flotante:** Una representación de 32 bits de un número real, de conformidad con la norma IEEE Std 754.
 - ◇ **Error de frecuencia (FE, Frequency Error):** La medición del error entre la frecuencia teórica y la frecuencia medida para el instante de tiempo dado.
 - ◇ **Error (RFE, Rate Frequency Error) de Razón de Cambio de Frecuencia (ROCOF):** La medición del error entre la ROCOF teórica y la ROCOF medida para el instante de tiempo dado.
 - ◇ **Frecuencia de Nyquist:** Una frecuencia que es la mitad de la frecuencia de muestreo de un sistema procesado de señales discretas.
 - ◇ **Tasa de Nyquist:** Una tasa que es el doble de la frecuencia más alta en la señal de entrada analógica. La señal analógica debe ser muestreada a una tasa mayor que la tasa de Nyquist para ser representada con precisión en forma digital.
 - ◇ **Fasor:** Un número complejo equivalente de una cantidad de una onda seno simple tal que el módulo es la amplitud de la onda seno y el ángulo complejo (en forma polar) es el ángulo de fase de la onda seno.
 - ◇ **Trama:** En este estándar, una trama de datos es un conjunto de mediciones que corresponden a la misma estampa de tiempo.
 - ◇ **Sincronismo:** El estado donde conectan los sistemas de corriente alterna, máquinas, o una combinación, operan a la misma frecuencia y en donde el ángulo de fase se desplaza entre tensiones constantes o varían de una constante y valor medio estable.
 - ◇ **CRC-CCIT:** Revisión de la Redundancia Cíclica (CRC) de 16 bits calculado usando la generación del polinomio $X^{16} + X^{12} + X^5 + 1$, con semilla 0xFFFF (-1).
 - ◇ **Fasor Sincronizado:** Un fasor calculado a partir de datos de muestras utilizando un tiempo estándar, como señal de referencia para la medición. Fasor sincronizado desde sitios remotos tienen una relación de fase definida común.

Sincrofasores y monitoreo de energía eléctrica trifásica

Definición de Sincrofasor

El concepto de fasor, y su definición clásica, fue introducido en el año 1893 a partir de una señal sinusoidal de frecuencia f_0 , valor eficaz A y fase φ constantes, tal como se muestra en la Ec. 2.1.

$$x(t) = \sqrt{2} * A * \cos(2\pi f_0 t + \varphi) \quad (2.1)$$

Con el objeto de poder representar dicha señal mediante un número complejo compuesto por su valor eficaz y fase, se define el correspondiente fasor según la Ec. 2.2, donde se muestra también la relación existente entre la señal original y el fasor.

$$\bar{X} = Ae^{j\varphi} \rightarrow x(t) = R(\sqrt{2} * \bar{X} * e^{j2\pi f_0 t}) \quad (2.2)$$

La representación fasorial de la señal permitió, históricamente, analizar el sistema de potencia considerando su estado estacionario, con frecuencia constante (50 o 60 Hz) y fase inicial correspondiente a un instante de referencia temporal fijo y único para todo el sistema.

Con el crecimiento de las redes interconectadas y malladas, los ángulos de fase entre puntos de la red resultaron de suma importancia para el análisis de estabilidad de las mismas. En base a dichas necesidades, en la década de 1980 fue introducido el concepto de sincrofasor, posteriormente estandarizado mediante la norma IEEE 1344 para finalmente ser formalizado mediante el estándar IEEE C37.118.1 en su versión actual IEEE 60255-118-1-2018 e IEEE C37.118.2 y sus modificaciones.

El sincrofasor se define en base a la medición de la señal de la Ec. 2.1, pero utilizando como referencia temporal el Tiempo Universal Coordinado (UTC, por sus siglas en inglés). De esta manera, se obtiene una única referencia temporal para todas las señales medidas en diferentes puntos de la red. Puede definirse, entonces, al sincrofasor como la representación de la señal $x(t)$ de la Ec. 2.1 mediante la Ec. 2.2, pero donde la fase φ corresponde al ángulo de fase instantáneo entre la señal $x(t)$ y una señal cosenoidal cuya frecuencia es la nominal del sistema y que se encuentra sincronizada con el tiempo UTC. Por lo tanto, la convención, establecida en la norma, puede observarse en la figura 2.1.

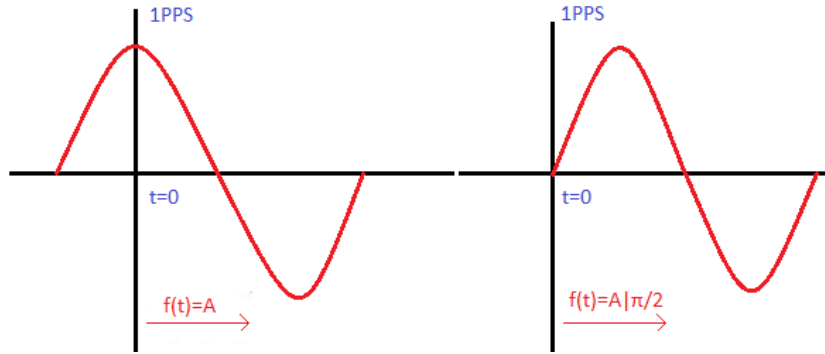


Figura 2.1: Conversión de referencia temporal

Cuando el máximo de la señal medida ocurre en el mismo instante de cambio del segundo del tiempo UTC, la fase del sincrofasor es cero. Y si la señal es cero con pendiente positiva en el instante del cambio de segundo, la fase del sincrofasor resulta -90° .

Entonces, en estado estacionario se puede considerar constante la frecuencia, amplitud y fase de la señal a lo largo de la ventana de medición utilizada. Pero en condiciones más generales, tanto la amplitud como la fase pueden variar debido a oscilaciones de potencia. Por esto es que para analizar el comportamiento del sistema de potencia tanto en condiciones estáticas como dinámicas, se necesita generalizar el modelo de señal tal como se muestra en la Ec. 2.3.

$$x(t) = \sqrt{2} * A(t) * \cos \left(2\pi * \int f(t)dt + \varphi \right) \quad (2.3)$$

Aquí, la amplitud A de la señal podría ser variable y, del mismo modo, la frecuencia de la señal podría variar su valor en el tiempo.

En el caso de considerar una frecuencia variable en el tiempo, es posible definir una función $p(t) = f(t) - f_0$, resultando entonces que $p(t)$ es la desviación instantánea de frecuencia respecto a la nominal. La representación generalizada de la señal resulta, entonces:

$$x(t) = \sqrt{2} * A(t) \cos \left(2\pi \int f_0 + p(t)dt + \varphi \right) = \sqrt{2} * A(t) \cos \left(2\pi f_0 t + 2\pi \int f_0 + p(t)dt + \varphi \right) \quad (2.4)$$

Según el modelo de la Ec. 2.4, la forma más general del sincrofasor se puede analizar según la Ec. 2.5.

$$\bar{X}(t) = A(t)e^{j \left(2\pi \int p(t)dt + \varphi \right)} \quad (2.5)$$

Sincrofasores y monitoreo de energía eléctrica trifásica

Si se considera la amplitud constante y la desviación de frecuencia constante, tal que $p(t) = \Delta f$, el sincrofasor resulta

$$\bar{X}(t) = Ae^{j(2\pi\Delta f t + \varphi)} \quad (2.6)$$

Puede verse que en el caso de analizar una señal con desviación de frecuencia Δf constante medida a intervalos de tiempo constante, el sincrofasor gira uniformemente con una velocidad de giro Δf tal como se muestra en la Figura 2.2.

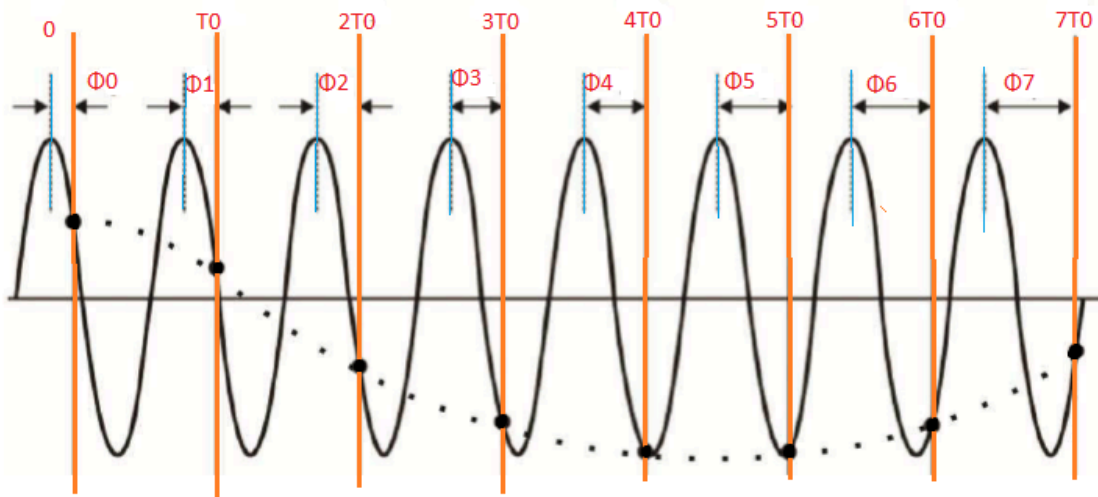


Figura 2.2: Medición periódica de señal con desviación de frecuencia constante
Reproducida con permiso de “<https://www.mdpi.com/1996-1073/12/14/2648>”

Frecuencia y Tasa de Variación de Frecuencia (ROCOF): La norma, que define el concepto y medición de sincrofasores, establece también la medición de la frecuencia y la tasa de variación de frecuencia. Para ello, puede escribirse a la Ec. 2.4 según la Ec. 2.7.

$$x(t) = \sqrt{2}A(t) \cos(\Phi(t))$$

$$\Phi(t) = 2\pi f_0 t + 2\pi \int p(t) dt + \varphi \quad (2.7)$$

A partir de la definición de $\Phi(t)$, se define la frecuencia según la Ec. 2.8.

$$f(t) = \frac{1}{2\pi} * \frac{d\Phi(t)}{dt} = f_0 + p(t) \quad (2.8)$$

Sincrofasores y monitoreo de energía eléctrica trifásica

A su vez, el estándar define también el parámetro Tasa de Variación de Frecuencia (ROCOF, por sus siglas en inglés). En la Ec. 2.9 se puede observar la definición del mismo

$$ROCOF(t) = \frac{df(t)}{dt} = \frac{1}{2\pi} * \frac{d^2\Phi(t)}{dt^2} = \frac{dp(t)}{dt} \quad (2.9)$$

Medición de errores: La norma define lo siguiente

Error vectorial total: En términos generales, para medir la exactitud de la medición, la norma define el Error Vectorial Total (TVE, por sus siglas en inglés) según la Ec. 2.10 donde \hat{X}_r y \hat{X}_i corresponden a la parte real e imaginaria del sincrofasor medido mientras que X_r y X_i corresponden al sincrofasor real.

$$TVE = \sqrt{\frac{(\hat{X}_r - X_r)^2 + (\hat{X}_i - X_i)^2}{X_r^2 + X_i^2}} \quad (2.10)$$

En la Figura 2.3 se observa una representación gráfica del error. Un valor de TVE de 1% puede ser consecuencia tanto de un error de amplitud del 1% con un error de fase cero, como de una medición sin error de amplitud pero con un error de fase de $0,573^\circ$. En términos generales, un valor de TVE de 1% puede deberse a cualquier combinación de valores intermedios de errores de amplitud y fase que determinen que el fasor \hat{X} se ubique sobre una circunferencia de radio 1 centrada en el extremo del fasor X .

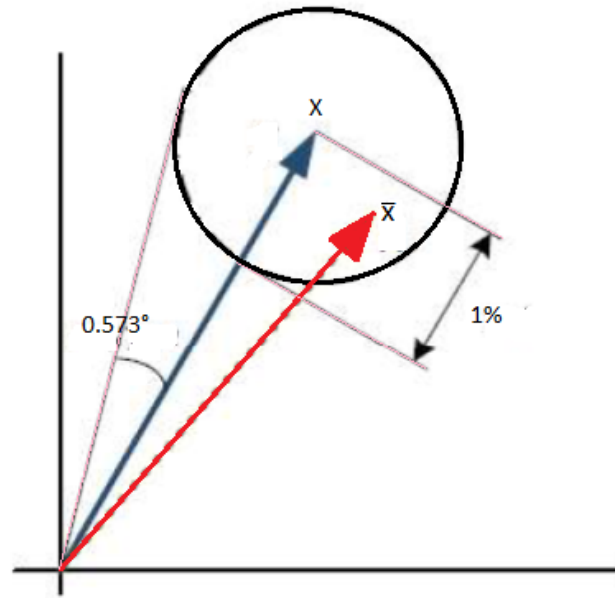


Figura 2.3: Diagrama esquemático – Error Vectorial Total

Se concluye de la Ec. 2.10 y del análisis realizado, que el TVE tiene en cuenta diversos errores, incluyendo los errores de amplitud como de fase. En la Figura 2.4 se puede ver la variación de TVE para los diferentes valores de error de fase, cuando el error de magnitud es constante. Análogamente, en la Figura 2.5, se muestra la variación de TVE para diferentes valores de error de amplitud cuando el error de fase resulta constante.

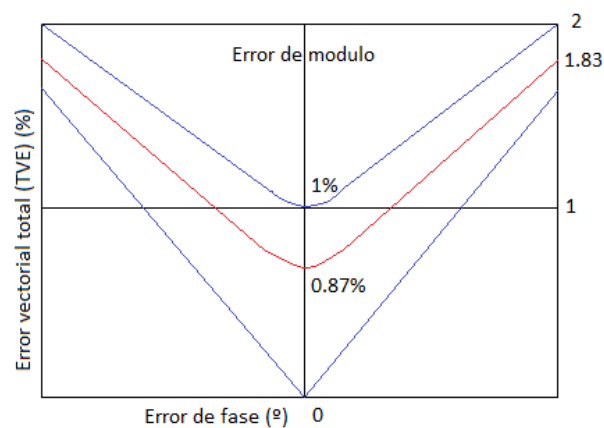


Figura 2.4: TVE en función del error de módulo

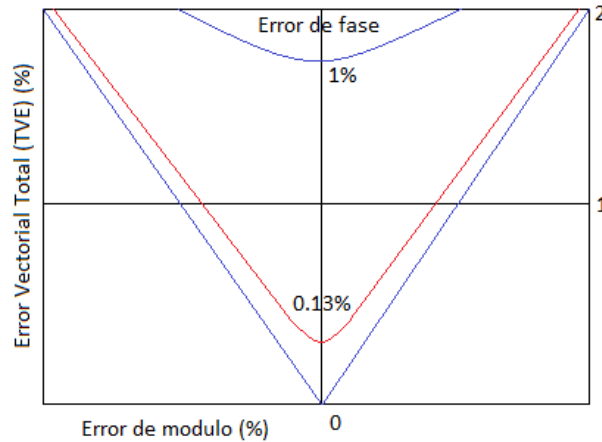


Figura 2.5: TVE en función del error de fase

A su vez, el TVE tiene en cuenta errores de sincronización de la medición con el tiempo UTC. Para una señal de frecuencia 50 Hz (período de 20 ms), un error de fase de 1° corresponde a un error de sincronismo de 55,6 μs. Por lo tanto, para no superar el límite de 0,573° debe tenerse un error de sincronismo menor a ± 31,8 μs.

Error de Frecuencia y ROCOF: Así como la norma establece el TVE para evaluar la calidad de la medición del sincrofasor, también define parámetros que permiten evaluar la medición de frecuencia y ROCOF.

En ambos casos, se define al error como el valor absoluto de la diferencia existente entre el valor medido y el valor real. En las Ecs. 2.11 y 2.12, se observa la expresión del error de medición de frecuencia (FE, por sus siglas en inglés), y del error de medición de ROCOF (RFE, por sus siglas en inglés).

$$FE = |\hat{f}(t) - f(t)| \quad (2.11)$$

$$RFE = \left| \frac{d\hat{f}(t)}{dt} - \frac{df(t)}{dt} \right| \quad (2.12)$$

Conceptos básicos

Para poder implementar cualquier algoritmo de estimación fasorial, es necesario estudiar algunos conceptos básicos en el tema del procesamiento de señales digitales, los cuales se describen a continuación.

Valor RMS

El valor eficaz (RMS, Root Mean Square) o valor cuadrático medio de una onda periódica es igual al valor de la corriente directa que fluye a través de una resistencia de R ohms, ambas disipan la misma potencia promedio. Este concepto es crucial en el procesamiento digital de señales, donde para convertir señales analógicas a formato digital, se busca representar con precisión la información contenida en la onda. La conversión a valores eficaces contribuye a capturar de manera efectiva la potencia real de la señal, facilitando así su procesamiento y análisis.

El cálculo del valor eficaz para una señal periódica se describe con la Ec. 2.13.

$$X_{RMS} = \sqrt{\frac{1}{T} \int_0^T (x(t))^2 dt} \quad (2.13)$$

En donde:

X_{RMS} es el valor eficaz de la señal.

T es el periodo de la señal bajo estudio.

(t) representa la señal analógica.

Para el análisis en tiempo discreto, se utiliza la modificación de la Ec. (2.14), que resulta:

$$X_{RMS} = \sqrt{\frac{1}{N} * \sum_{n=0}^{N-1} x(n)^2} \quad (2.14)$$

En donde:

X_{RMS} es el valor eficaz de la señal.

N es el número de muestras de la ventana de datos.

(n) representa la función periódica en tiempo discreto.

Concepto de fasor

Para un mejor análisis de la corriente alterna (CA), representada por la figura 2.6, siendo su expresión $(\omega t + \varphi)$, dicha expresión puede ser transformada a un sistema de coordenadas polares, tal y como se muestra en la misma figura.

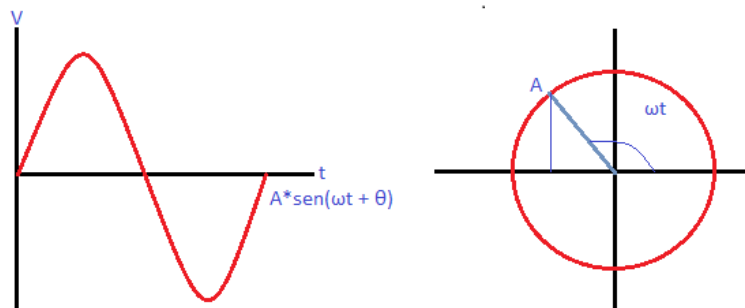


Figura 2.6.: Representación de onda senoidal en coordenadas polares

El radio gira con frecuencia ω describiendo un círculo. La longitud del radio es la amplitud de la onda, en este caso del voltaje. La componente vertical tiene el valor $(\omega t + \varphi)$ en la curva sinusoidal de AC. Este radio giratorio es fasor.

Sincrofasores y monitoreo de energía eléctrica trifásica

Concepto de fase

Fase es una medida de la diferencia de tiempo entre dos ondas sinusoidales. Aunque la fase es una diferencia verdadera de tiempo, siempre es medida en términos de ángulo, en grados o radianes. Eso es una normalización del tiempo que requiere un ciclo de la onda sin considerar su verdadero periodo de tiempo.

La diferencia en fase entre dos formas de onda se conoce con el nombre de defasamiento. Un defasamiento de 360° es un atraso de un ciclo o de un periodo de la onda, lo que realmente no es ningún desplazamiento. Un defasamiento de 90° es un desplazamiento de un cuarto del periodo de la onda. Existen fenómenos llamados defasamiento en atraso y defasamiento en adelante, eso quiere decir que una forma de onda puede ser retrasada respecto a otra o una forma de onda puede ser adelantada respecto a otra.

Para medir la fase, se puede utilizar una señal externa, como el caso de un pulso por segundo proporcionado por un módulo receptor GPS.

Muestreo digital

El muestreo digital es un proceso de adquisición de datos, a intervalos regulares de tiempo, consistente en la obtención del valor que toma la señal original en un momento dado.

El parámetro fundamental del muestreo digital es el intervalo de muestreo ΔS , o su equivalente la frecuencia de muestreo $\frac{1}{\Delta S}$ Hz. Cuanto menor sea ΔS , mayor es el número de valores que se obtienen de la señal, y viceversa.

El resultado de dicho muestreo es la obtención de una serie discreta ordenada $\{x_r\} =$

$\{x_0, x_1, x_2, \dots, x_r, \dots\}$, en la que el índice r indica la posición de orden temporal del valor de x_r . Así, el valor de la señal original, en el tiempo $t = \Delta r$, (t) , se representa por. A la señal continua de origen se le llama serie temporal continua, mientras que a la serie obtenida por el muestreo se le llama serie temporal discreta.

Teorema del muestreo

Según el teorema de muestreo de Nyquist-Shannon, para poder reconstruir con exactitud a forma de onda, es necesario que la frecuencia de muestreo (f_{Δ}) sea como mínimo el doble de la máxima frecuencia (f_{max}) a muestrear.

El teorema de Nyquist indica el valor mínimo necesario para reconstruir la señal original, sin embargo, mientras más muestras se tengan, se podrá reconstruir mejor la señal.

Entonces, para que el muestreo sea correcto, se debe escoger la frecuencia de muestreo de tal forma que:

$$f_s < 2f_{max} \quad (2.15)$$

Normalmente, se suele filtrar la señal de entrada para eliminar las frecuencias que no pueden ser detectadas mediante el proceso del muestreo, para evitar ver frecuencias no existentes (alias).

También, es conveniente que la frecuencia de muestreo sea muy superior al doble de la frecuencia máxima de la señal bajo estudio, puesto que ningún filtro pasa-bajos puede eliminar completamente las frecuencias superiores a la frecuencia de corte. En la práctica, se suele usar la frecuencia de muestreo de entre 5 y 10 veces la frecuencia de corte del filtro pasa-bajos. En este trabajo es 70 veces la frecuencia de la nominal.

Medición de frecuencia

Un contador de frecuencia permite medir y visualizar la frecuencia de una señal. Para realizar el cálculo de la frecuencia del sistema, se tienen métodos diferentes; sin embargo se utilizó el método de cruce por cero, que se explica en la sección siguiente.

Cálculo de la frecuencia por cruce por cero

El algoritmo del cálculo de la frecuencia, mediante el método de cruce por cero, radica en la

Sincrofasores y monitoreo de energía eléctrica trifásica

detección, de un cambio de signo de la señal de entrada con pendiente negativa, como se muestra en la figura 2.7; en ese momento, se cuenta el número entero de muestras hasta el siguiente cruce por cero.

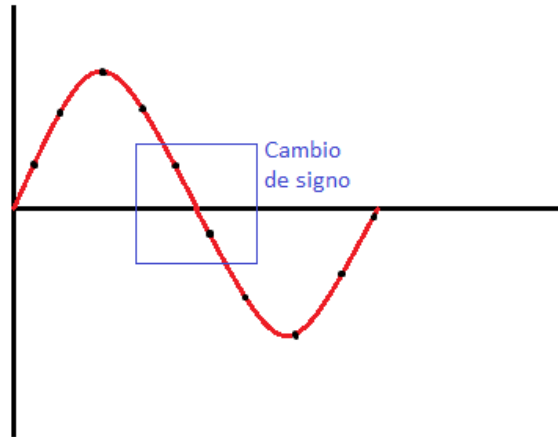


Figura 2.7: Cambio de signo evaluado

El número de muestras de un ciclo no es un número entero, por lo que es necesario calcular las fracciones de muestra obtenidas antes y después del cruce por cero, para obtener las fracciones de muestras dentro del ciclo. Esto se calcula sumando las muestras enteras y las fraccionarias, se obtiene el número exacto de muestras de cada ciclo.

Para calcular el valor de la fracción de la muestra anterior y posterior al cruce, se emplea la ecuación de la recta, como se muestra en la figura 2.8.

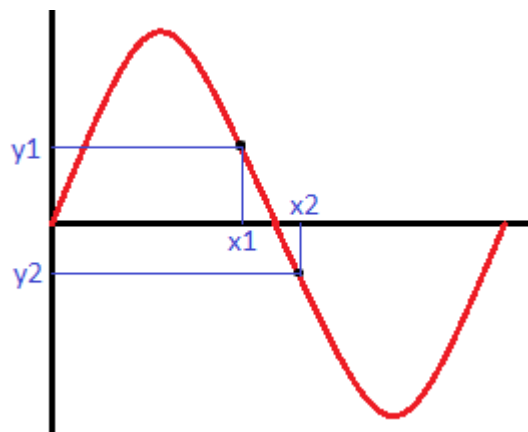


Figura 2.8: Cambio de signo, pendiente descendente

El intervalo que se desea conocer es el de x_1 a x , por lo que se utiliza la ecuación de la pendiente

Sincrofasores y monitoreo de energía eléctrica trifásica

de una recta, descrita por las ecuaciones 2.17 y 2.18:

$$m = \frac{y_2 - y_1}{x_2 - x_1}$$

$$y = mx + b \rightarrow x = \frac{y}{m} - b \quad (2.16)$$

En donde:

m es la pendiente

y_1 es la muestra anterior.

y_2 es la muestra actual.

x_1 Es el tiempo de muestreo anterior.

x_2 es el tiempo de muestreo actual.

El intervalo entre x_1 y x_2 corresponde a un periodo de muestreo, es decir:

$$x_2 - x_1 = 1$$

Analíticamente, si se necesita determinar el valor de x en el momento en que $y = 0$ y $b = y_1$, y sabiendo que $x_2 - x_1 = 1$, se hace uso de las ecuaciones 2.17 y 2.18, teniendo entonces:

$$y = mx + b \rightarrow 0 = mx + y_1 \quad (2.17)$$

Despejando a x resulta:

$$x = -\frac{y_1}{m}$$
$$x = -\frac{y_1}{\frac{y_2 - y_1}{x_2 - x_1}}$$

Sincrofasores y monitoreo de energía eléctrica trifásica

$$\Delta f_a = x = \frac{\Delta y_1}{y_2 + y_1} \quad (2.18)$$

En donde Δf_a es la fracción de muestra anterior al cambio de signo.

Por lo tanto, para calcular la fracción de muestra posterior al cambio de signo, es necesario determinar la parte complementaria del total de la muestra a la que pertenece la fracción de muestra anterior, ya calculada. Es decir, la suma de la fracción de muestra anterior y posterior es igual a la unidad. Así:

$$\Delta f_p = 1 - \Delta f_a \quad (2.19)$$

En donde Δf_p es la fracción de muestra posterior al cambio de signo. Cabe mencionar que, tras realizar la programación de este algoritmo, se obtuvo que el primer resultado arrojado por éste, es erróneo ya que no se tiene conocimiento del instante exacto del ciclo en que comienza a ejecutarse. Después de ello, el resultado brinda valores satisfactorios, mostrando el valor de la frecuencia fundamental, sin identificar la presencia de armónicos en la señal de entrada.

Sistema de posicionamiento global

El Sistema de Posicionamiento Global (GPS, *Global Positioning System*) es un sistema de navegación satelital diseñado por el Departamento de Defensa de Estados Unidos con fines militares de proporcionar estimaciones precisas de posición, velocidad y tiempo.

Introducción

El GPS surgió debido a la necesidad de las fuerzas armadas de tener un sistema de navegación preciso y que funcionara en aplicaciones diversas. El desarrollo de la tecnología GPS descansa en progresos en ciencias físicas, en la electrónica, en ciencias de materiales y en muchas otras, pero fue el desarrollo de dispositivos extremadamente precisos para medir el tiempo – relojes atómicos, junto con el progreso en la tecnología espacial, que en realidad hicieron posible el GPS.

En el ámbito civil y alegando razones de seguridad, sólo se permite el uso de un subconjunto

Sincrofasores y monitoreo de energía eléctrica trifásica

degradado de señales GPS. Sin embargo, la comunidad civil ha encontrado alternativas para obtener una excelente precisión en la localización mediante las denominadas técnicas diferenciales. Gracias a ellas, las aplicaciones civiles han experimentado un gran crecimiento y actualmente existen más de 70 fabricantes de receptores GPS.

El GPS ha sido adoptado en aplicaciones de transportación terrestre, aviación civil, comercio marítimo, topografía, construcción, minería, agricultura, ciencias geológicas, energía eléctrica, telecomunicaciones, y actividades recreativas a la intemperie.

Arquitectura

La arquitectura del Sistema de Posicionamiento Global está conformada de tres partes, como se muestra en la figura 2.9:

- ◇ El sector espacial, que consta de 24 satélites con una órbita de 26560 km de radio y un periodo de 12 h.
- ◇ El segmento control, constituido por estaciones monitoras, encargadas de mantener en órbita los satélites y supervisar su correcto funcionamiento.
- ◇ El segmento usuario, formado por las antenas y los receptores pasivos situados en tierra.

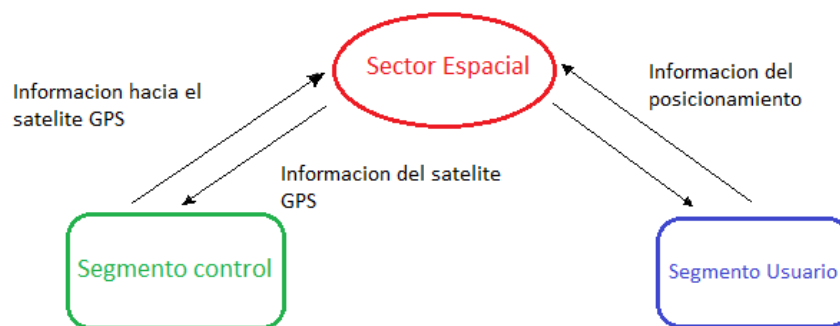


Figura 2.9: Arquitectura del Sistema de posicionamiento Global

Los satélites se mueven en el espacio a una velocidad de 4 km/s y transmiten anuncios de su

Sincrofasores y monitoreo de energía eléctrica trifásica

posición y los receptores determinan las coordenadas del usuario por triangulación geométrica de estas posiciones. La exactitud de su posición depende de plataformas espaciales súper estables, relojes atómicos que sincronizan la constelación y el monitoreo de los satélites desde 5 puntos distribuidos alrededor del globo terráqueo.

La precisión final alcanzable depende de las efemérides determinadas por el segmento control. La precisión nominal de estas efemérides, permite la determinación de la posición de un satélite con error menor de 1 metro en sentido radial, 7 metros a lo largo de la trayectoria y 3 metros transversalmente.

El segmento usuario, como se mencionó anteriormente, comprende los instrumentos utilizados para hallar coordenadas de un punto, hacer navegación o adquirir tiempo con precisión de oscilador atómico, usando las señales radiodifundidas desde los satélites, además de proporcionar un pulso por segundo que está alineado con el tiempo universal coordinado. El equipo esencial está formado por una antena y receptor unidos directamente o mediante un cable.

Es de importancia hacer mención que actualmente, el sector de la energía utiliza un estándar de tiempo, el cual se conoce como estándar IRIG-B, para los registradores de fallas con estampa de tiempo y para otros sistemas de supervisión de eventos en las subestaciones.

A pesar de todo, con los receptores acorde al estándar IRIG-B, la precisión de sincronización es del orden de un milisegundo (1 ms), lo cual no es suficiente para obtener mediciones precisas del sistema de distribución.

Triangulación

La idea principal detrás del GPS es utilizar los satélites en el espacio como puntos de referencia para ubicaciones en la Tierra. Esto se logra utilizando la distancia del receptor GPS hacia tres satélites, lo que permite triangular dicha posición en cualquier parte de la Tierra.

Sincrofasores y monitoreo de energía eléctrica trifásica

El proceso de triangulación se realiza, de manera general, como sigue. El receptor determina la distancia al primer satélite, midiendo el tiempo que tarda una señal emitida por el satélite en llegar a dicho receptor GPS. Esta distancia limita la posición del receptor a la superficie de una esfera que tiene como centro dicho satélite y cuyo radio es de la distancia calculada. Después, se obtiene la distancia del receptor al segundo satélite, lo que indica que el receptor se encuentra sobre las dos esferas de las distancias calculadas. Es decir, el receptor se encuentra en algún punto de la circunferencia que resulta de la intersección de las dos esferas.

Así, si se mide la distancia a un tercer satélite, se limita la posición del receptor GPS mucho más, a los dos puntos en los cuales la tercera esfera corta la circunferencia que resulta de la intersección de las dos primeras esferas. Al medir la distancia a tres satélites, se limita el posicionamiento a sólo dos puntos posibles, como se muestra en la figura 2.10.

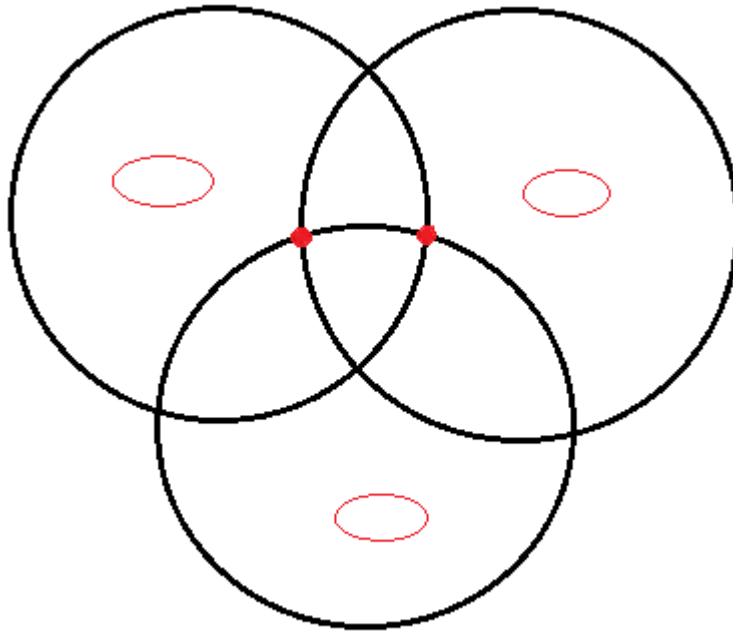


Figura 2.10: Triangulación Satelital.

La decisión de cuál de estos puntos es la posición verdadera, se hace la medición a un cuarto satélite. Normalmente uno de los dos puntos posibles resulta ser demasiado improbable por tener una ubicación lejana de la superficie terrestre, descartándolo sin necesidad de mediciones posteriores.

Sincrofasores y monitoreo de energía eléctrica trifásica

Tiempo universal coordinado (UTC)

El Tiempo Universal Coordinado (UTC), también denominado tiempo civil, sirve como la referencia principal para calcular las distintas zonas horarias en todo el mundo.

El acrónimo UTC no constituye simplemente una abreviatura; más bien, representa una variante del Tiempo Universal (UT) con la adición de la letra "C" (coordinado) para señalar su naturaleza más coordinada en comparación con el UT estándar. Este sistema reemplazó al Greenwich Mean Time (GMT), anteriormente basado en las mediciones realizadas desde el observatorio de Greenwich. La nueva designación se ideó con el propósito de eliminar cualquier asociación con una ubicación específica y, al mismo tiempo, establecer la medición del tiempo en estándares atómicos en lugar de celestes.

El UTC, definido por relojes atómicos, se ajusta para permanecer sincronizado con los ciclos naturales de día y noche. La adición o sustracción de segundos al final de junio y diciembre es determinada por el Servicio Internacional de Rotación de la Tierra, basándose en sus mediciones de la rotación terrestre.

Este sistema horario es ampliamente adoptado en numerosos estándares de Internet. El Protocolo de Tiempo de la Red ha sido desarrollado específicamente para distribuir eficientemente la hora UTC a través de la red.

Introducción

En el presente capítulo se presenta el hardware que se utilizó para la implementación de la Unidad de Medición Fasorial sincronizada vía satélite. Los principales elementos utilizados son listados a continuación:

- ◇ Tarjeta de desarrollo STM32F407VG.
- ◇ Tarjeta de desarrollo ESP32-NodeMCU.
- ◇ Receptor GPS marca U-BLOX NEO6M.
- ◇ Generadores UNI-T UTG2025A para la simulación de la onda trifásica.
- ◇ Analizador de redes Meatrol ME440

Diagrama de conexionado interno

Este diagrama presenta la descripción gráfica de la operación de la PMU para tener un panorama amplio de lo que se está implementando. Así también, el diagrama muestra la interacción que tienen los elementos del hardware con las rutinas que se ejecutan por el firmware.

El diagrama de conexionado interno de la Unidad de Medición Sincrofasorial se presenta en la figura 3.1.

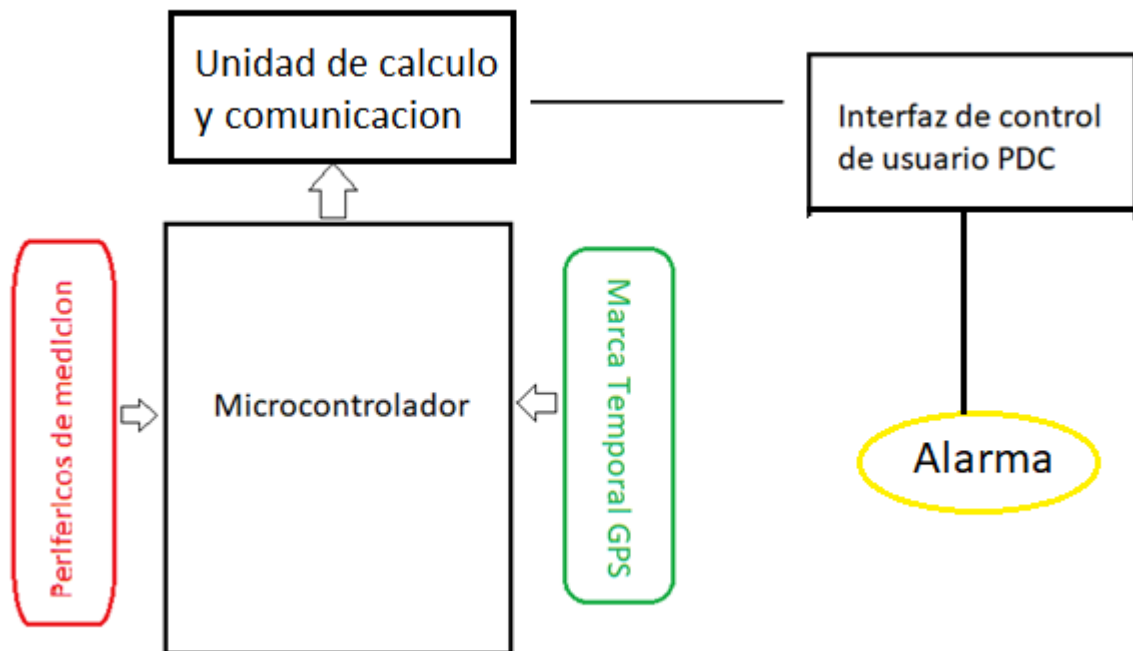


Figura 3.1: Esquema de conexionado

La interacción general es de la siguiente manera: las señales (trifásicas) de corriente y voltaje son reducidas a niveles adecuados para ser procesados por los Convertidores Analógico-Digital del microcontrolador STM32F407VG. Se realiza la programación del ADC para que tome 70 muestras de cada ciclo de 50 Hz, almacenando datos de dos ciclos.

De manera paralela, el receptor GPS envía mediante comunicación serial tramas de datos, las cuales son procesadas en el microcontrolador STM32F407VG para obtener la hora UTC sincronizada con el sector espacial.

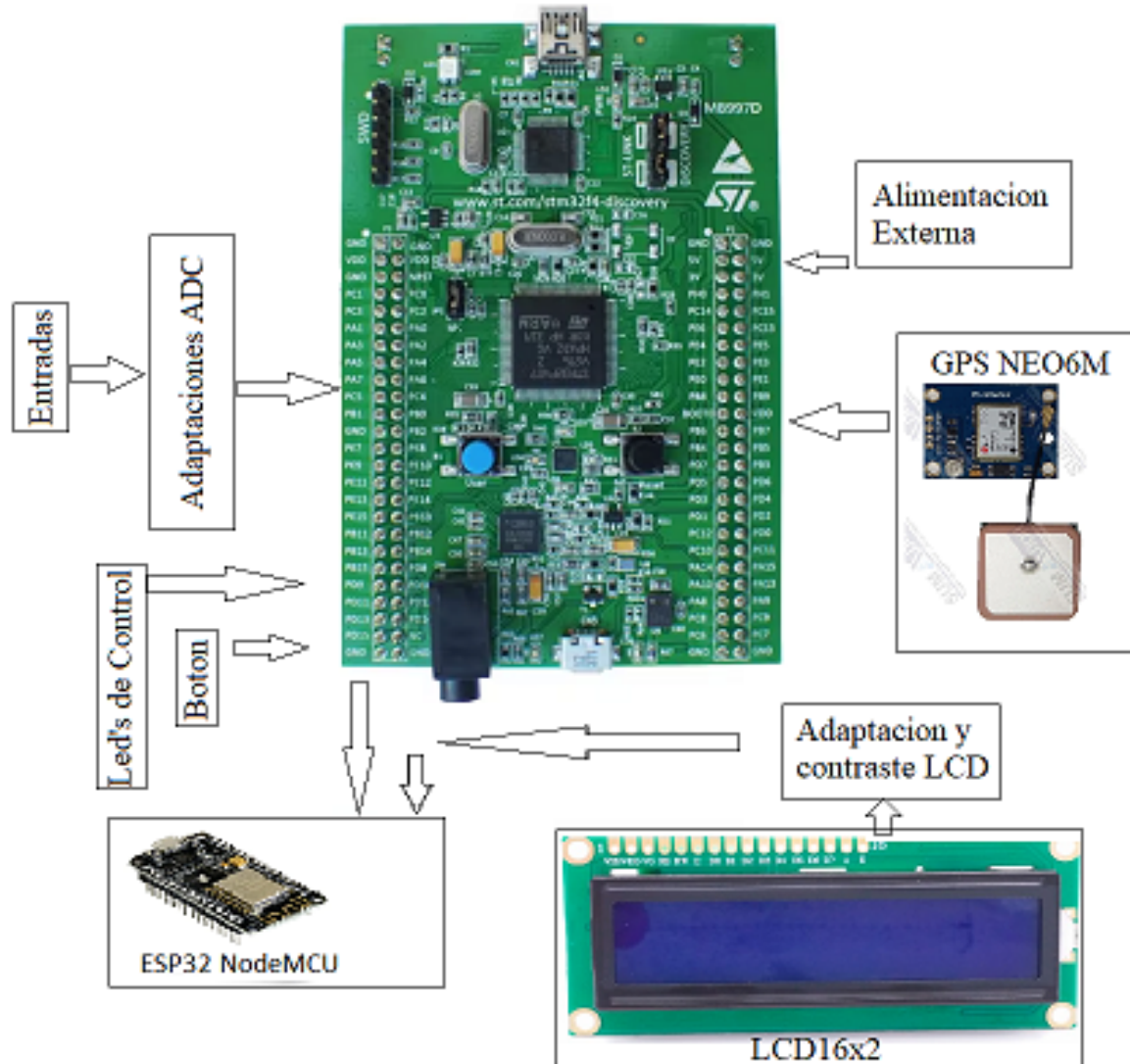


Figura 3.2: Diagrama de bloques de conexionado interno

Una vez entregado los datos de la trama, 22 ms después el puerto serial del GPS envía datos relacionados con la ubicación del receptor GPS y el Tiempo Universal Coordinado de la trama recibida. Dichos datos son recibidos por el puerto serial del microcontrolador, para ser almacenados en un arreglo que permita obtener el valor del UTC.

Las muestras tomadas por los ADC y el valor del UTC son enviados por la interfaz serial a la ESP para que ésta procese las cadenas y obtenga, de forma sincronizada, los parámetros de amplitud y diferencia de ángulo entre los fasores .

Sincrofasores y monitoreo de energía eléctrica trifásica

Características de la placa STM32F407Vg – Discovery

El monitoreo de ondas de energía trifásica implica la medición y análisis de las tensiones y corrientes en los tres conductores de un sistema trifásico. Esto es esencial para asegurarse de que el suministro de energía eléctrica sea estable, eficiente y seguro en aplicaciones como la distribución de energía eléctrica, la automatización industrial, los sistemas de control de motores, la generación de energía y más.

Para llevar a cabo el monitoreo de energía trifásica, se requiere la conversión precisa de las señales analógicas de las tensiones y corrientes en señales digitales que pueden ser procesadas por un microcontrolador como el STM32F407VG. Los ADC desempeñan un papel crucial en este proceso al convertir estas señales analógicas en datos digitales.

En un sistema trifásico, se deben medir al menos tres tensiones y tres corrientes (una por fase). Los ADC en la STM32F407VG tienen múltiples canales de entrada, y se pueden configurar para medir las tensiones y corrientes en cada fase. Los canales de entrada se seleccionan de acuerdo con la conexión física de los sensores de corriente y los divisores de voltaje utilizados para medir las tensiones.

La elección de la resolución de los ADC y el rango de entrada depende de la precisión requerida en las mediciones. Para el monitoreo de energía, es común utilizar resoluciones de 12 o 16 bits para obtener mediciones precisas. Además, el rango de entrada de los ADC se configura para que pueda manejar las amplitudes esperadas de las señales de tensión y corriente.

La velocidad de muestreo de los ADC debe ser lo suficientemente rápida como para capturar con precisión las variaciones en las señales de tensión y corriente. Esta velocidad debe ser ajustada de acuerdo con la frecuencia de la red eléctrica que se está monitoreando. Por ejemplo, en sistemas de 50 Hz, una velocidad de muestreo de al menos 100 muestras por segundo por canal es típica para evitar la pérdida de información en la forma de onda.

Para garantizar mediciones precisas, es común aplicar filtros analógicos antes de la entrada a los ADC para eliminar el ruido no deseado y prevenir el fenómeno de aliasing. Estos filtros pueden ser pasivos o activos, dependiendo de los requisitos del sistema.

Una vez configurados los ADC para el monitoreo de energía trifásica, se inicia el proceso de adquisición de datos. A continuación, se describe cómo se adquieren y procesan las señales de tensión y corriente en cada fase:

Sincrofasores y monitoreo de energía eléctrica trifásica

- Tensión en Fases:

Para medir la tensión en cada fase, se conectan los divisores de tensión adecuados a los canales de entrada de los ADC. Estos divisores de tensión reducen la tensión de fase a un nivel que puede ser manejado por los ADC. Luego, los ADC realizan conversiones de tensión a datos digitales.

- Corriente en Fases

Para medir la corriente en cada fase, se utilizan sensores de corriente, como sensores de efecto Hall o transformadores de corriente (TI). La señal de corriente generada por estos sensores se convierte en una señal de tensión utilizando resistencias de carga y luego se mide con los ADC.

- Cálculos de Potencia y Análisis

Una vez que se han convertido todas las señales analógicas en datos digitales, se pueden realizar cálculos para determinar la potencia activa, reactiva y aparente en cada fase, así como el factor de potencia. Estos cálculos son esenciales para evaluar la eficiencia del sistema de energía y detectar problemas como sobrecargas o desequilibrios de fases.

- Visualización y Almacenamiento de Datos

Los resultados de las mediciones y los cálculos pueden ser visualizados en tiempo real a través de una interfaz gráfica o almacenados para análisis posterior. El microcontrolador STM32F407VG tiene suficiente capacidad de procesamiento y memoria para llevar a cabo estas tareas.

- Gestión de Interrupciones y Eventos

En aplicaciones de monitoreo de energía, es importante gestionar interrupciones y eventos basados en las mediciones de los ADC. Por ejemplo, se pueden configurar interrupciones para detectar sobrecargas o fluctuaciones de tensión anormales y tomar medidas preventivas o alertar al sistema de control.

La figura 3.3, mostrada a continuación, es la conexión física de la PMU realizada para este proyecto. Se observa que la alimentación del módulo PMU está conectada a los bornes de la tarjeta que suministran 5V, además se tiene la conexión del GPS a la terminal Serial de la placa STM32F407VG, el terminal Tx del módulo GPS al pin UART_Rx de la tarjeta para el envío y recepción del UTC.

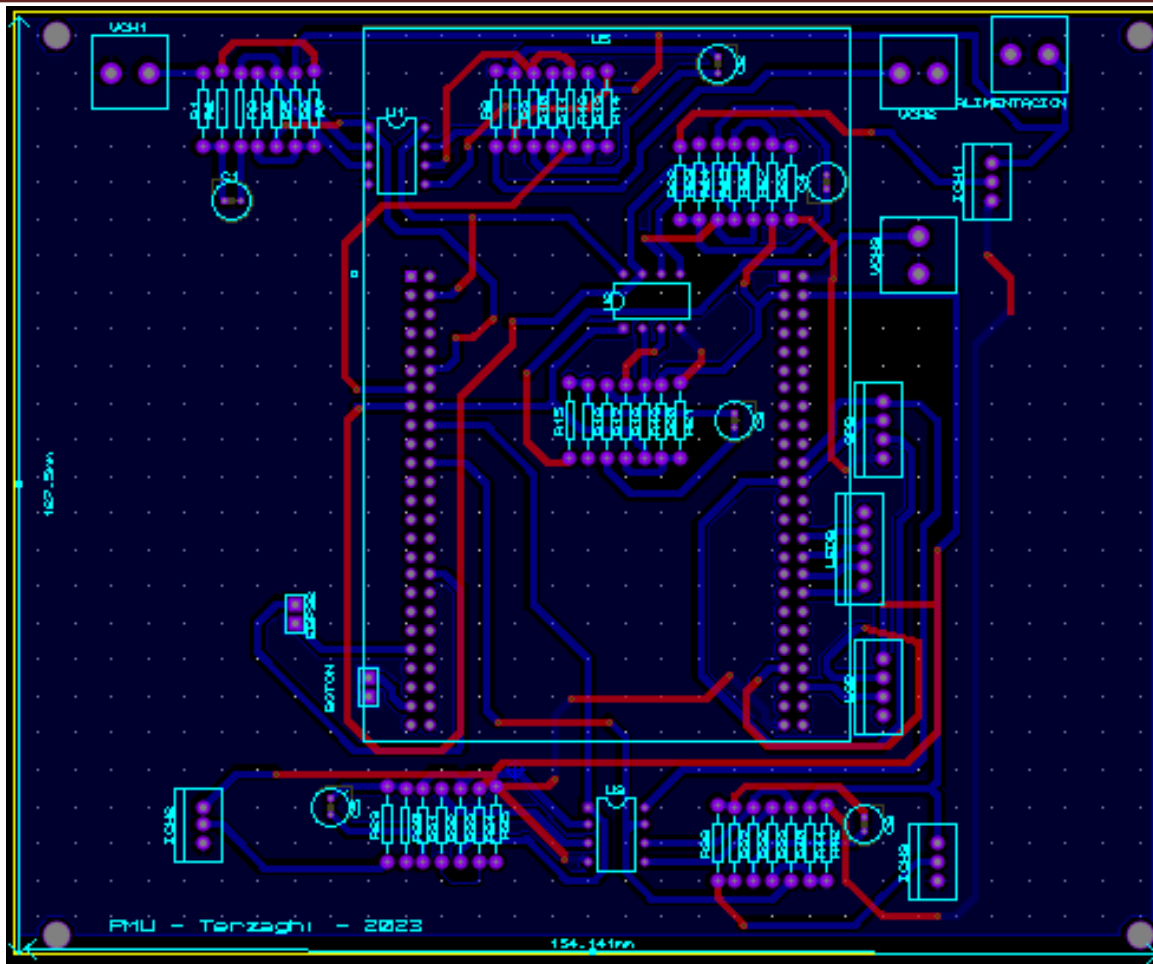


Figura 3.3: Tarjeta STM32F407

Para el hardware diseñado en la figura 3.3, se diseñó un circuito capaz de adaptar la señal transformada a los valores de tensión soportados por los ADC de la placa de desarrollo Discovery STM32F407VG. En la Figura 3.4 se presenta dicho circuito.

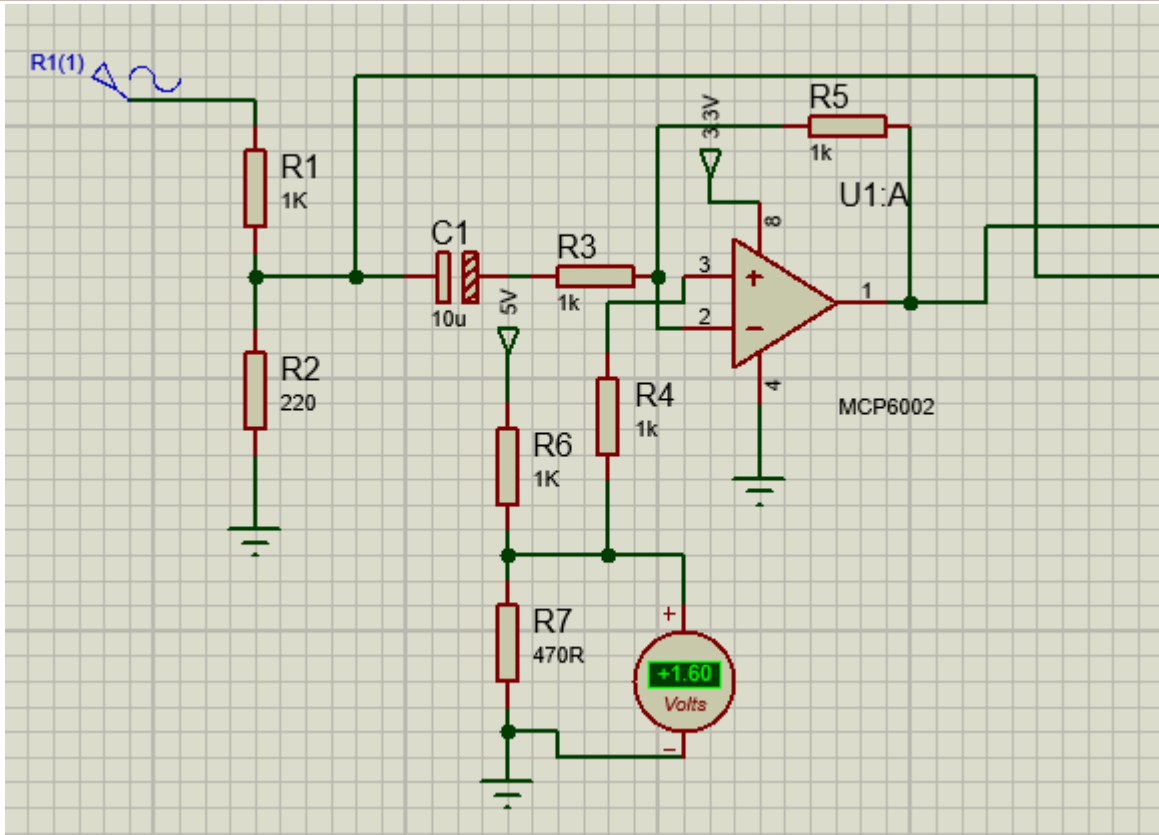


Figura 3.4: Adaptación para señal de tensión al ADC empleado

En el mismo, se puede ver como con el divisor resistivo R6/R7 damos el offset necesario para poder fijar el “0” en la mitad del rango de medición del ADC que es de 0 V a 3.3 V. El divisor resistivo R1/R2 es el que me permite llevar la señal dada por el TV al valor de 3.3 V_{pp} soportado por el ADC. Dicho resultado mencionado se puede ver en la Figura 3.5, donde la señal de entrada es la azul y la señal de salida captada por el ADC es la amarilla.

Para el divisor R6/R7 se realizó lo siguiente:

$$V_0 = V_{IN} * \frac{R_7}{R_6 + R_7} \rightarrow R_6 = 1K\Omega; V_0 = 1,5V; V_{IN} = 5V$$

$$R_7 = \frac{R_6 * \frac{1,5V}{5V}}{1 - \frac{1,5V}{5V}} = 428,571\Omega \cong 470\Omega$$

Para el divisor R1/R2:

$$V_0 = V_{IN} * \frac{R_2}{R_1 + R_2} \rightarrow R_1 = 1K\Omega; V_0 = 3,3V; V_{RMS} = 6V; V_{IN} = 2 * \sqrt{2} * V_{RMS} \cong 17V$$

$$R_2 = \frac{R_1 * \frac{3.3V}{17V}}{1 - \frac{3.3V}{17V}} = 240,876 \approx 220\Omega$$

Digital Oscilloscope

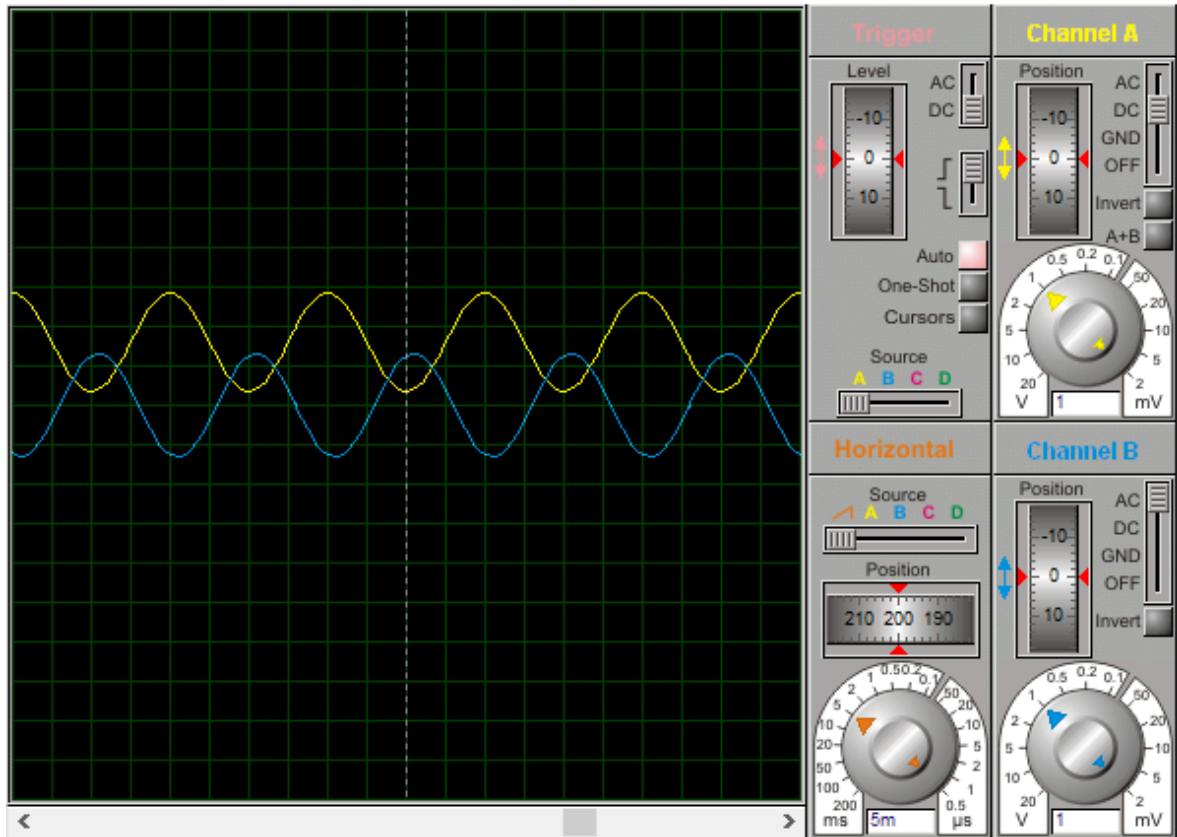


Figura 3.5: Señal de medición del ADC para señales de tensión

Para los canales de corriente también se llegó a un circuito similar, con la diferencia que la señal que ingresa al circuito no proviene desde el TI, sino que desde dicho transformador pasamos por un sensor de efecto Hall para poder pasar los niveles de corriente a niveles de tensión, los cuales sí pueden ser captados por el ADC de la placa utilizada.

Para éstos, se realizaron los siguientes cálculos:

$$I_{TI-pp} = 45 A$$

El sensor de efecto Hall tiene una sensibilidad de 66 mV/A, por ende la salida de dicho sensor es:

Sincrofasores y monitoreo de energía eléctrica trifásica

$$V_{s-0} = I_{TI-pp} * 66mV = 45A * 0.066V = 2.97 V_{pp}$$

Por ésto, en vez de utilizar un divisor resistivo (R1/R2), vamos a aplicar directamente dicha señal para ocupar el rango del ADC.

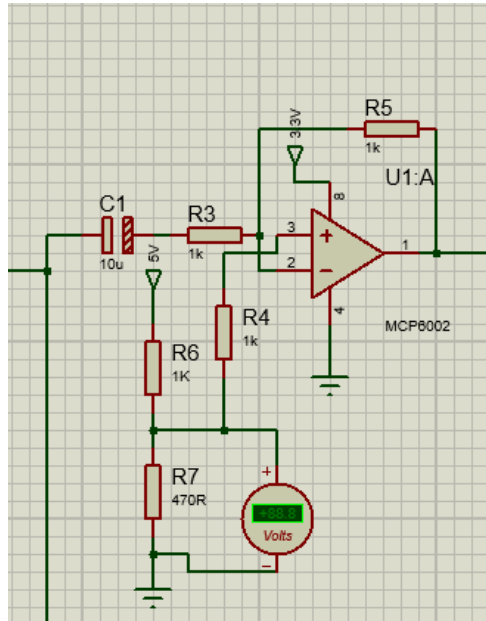


Figura 3.6: Adaptación para la medición de corriente

El resultado de la simulación se muestra a continuación, en la Figura 3.7:

Sincrofasores y monitoreo de energía eléctrica trifásica

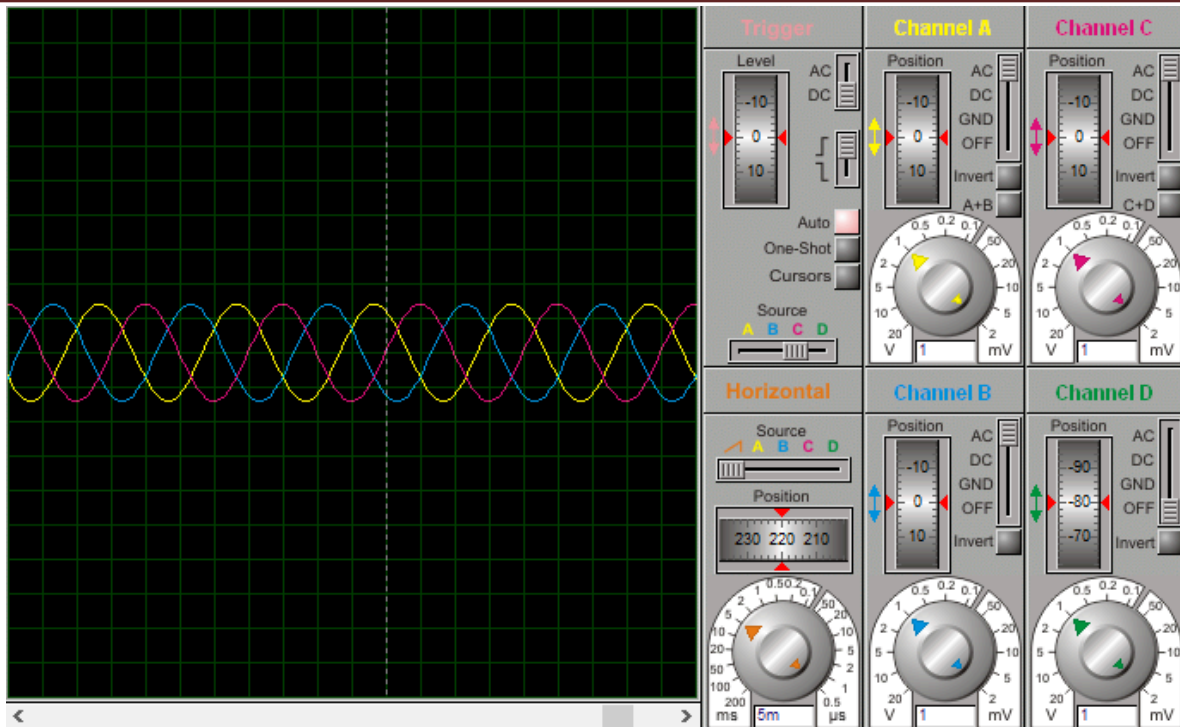


Figura 3.7: Señal de medición de ADC para canales de corriente

La señal amarilla es la señal de salida luego del circuito seguidor de tensión el cual es la protección del canal del ADC (en esta se puede ver que tiene $2.97V_{pp}$ lo cual es admitida por el circuito de protección y captada por el ADC), luego, tenemos la señal azul, la cual es la señal del efecto Hall con diferencia de fase 120° , y por ultimo tenemos la señal rosa, la cual es la señal de salida del tercer sensor de efecto Hall, con diferencia de ángulo de 240° con respecto a la amarilla. Se puede observar que las adaptaciones a los canales no modifican los ángulos de fase entre las señales. Esto se hace para poder aprovechar el total de la resolución del ADC y poder captar sin problemas las sobrecorrientes en el estado estacionario del sistema.

Una vez resuelto esto, se implementó el hardware de placa de control, el cual es el encargado de tomar las muestras, y enviar dichos resultados por el puerto serie a la placa ESP32, la cual realiza los cálculos y envía los resultados al software de PDC mediante el WIFI incorporado que tiene dicha placa.

Cabe aclarar, que una posibilidad considerada en un momento fue mediante el puerto Ethernet del módulo DP83848, y no está descartado del proyecto, sino que es considerado como una conexión auxiliar mediante un driver aparte para poder conectarse al PDC en caso de no contar con una red WIFI en el lugar del ensayo.

Sincrofasores y monitoreo de energía eléctrica trifásica

Características de la placa ESP32-NodeMCU

La ESP32 NodeMCU es una placa de desarrollo que combina la potencia del microcontrolador ESP32 con conectividad Wi-Fi y Bluetooth. El ESP32 es el sucesor del popular ESP8266 y ha mejorado significativamente en términos de potencia de procesamiento, capacidades de conectividad y versatilidad. A medida que avanzamos en este artículo, exploraremos las siguientes características clave de la ESP32 NodeMCU:

- **Microcontrolador ESP32:** El corazón de la ESP32 NodeMCU es el microcontrolador ESP32, basado en el núcleo Tensilica Xtensa LX6 de doble núcleo. Con velocidades de reloj de hasta 240 MHz, una amplia gama de periféricos y capacidades de procesamiento avanzadas, el ESP32 es un chip de alto rendimiento que permite una variedad de aplicaciones.
- **Wi-Fi y Bluetooth Integrados:** La placa está equipada con conectividad Wi-Fi 802.11 b/g/n y Bluetooth 4.2 BLE (Low Energy). Esta combinación de Wi-Fi y Bluetooth proporciona la capacidad de conectarse a redes inalámbricas y dispositivos Bluetooth, lo que permite una comunicación inalámbrica efectiva y versátil.
- **GPIO y Periféricos:** La ESP32 NodeMCU cuenta con una variedad de pines GPIO (General Purpose Input/Output) que permiten la interacción con sensores, actuadores y otros dispositivos. Además, incluye periféricos como UART, SPI, I2C, PWM y ADC para una amplia variedad de aplicaciones.
- **Memoria y Almacenamiento:** La placa ofrece 4 MB de memoria flash para el almacenamiento de programas y datos, lo que proporciona suficiente espacio para aplicaciones complejas. Además, tiene 520 KB de SRAM para el almacenamiento en tiempo de ejecución.
- **Sistemas Operativos en Tiempo Real (RTOS):** La ESP32 NodeMCU es compatible con varios sistemas operativos en tiempo real, incluyendo FreeRTOS y ESP-IDF (Espressif IoT Development Framework), lo que facilita el desarrollo de aplicaciones multitarea y conectadas a Internet.

Una de las características más destacadas de la ESP32 NodeMCU es su capacidad de procesamiento de datos. El microcontrolador ESP32 tiene dos núcleos de procesador Xtensa LX6 que pueden funcionar de manera independiente o en paralelo. Esto permite una mayor eficiencia en la ejecución de tareas y una mayor capacidad de respuesta en aplicaciones exigentes. Algunas de las capacidades de procesamiento clave incluyen:

Sincrofasores y monitoreo de energía eléctrica trifásica

1. Doble Núcleo de Procesamiento

El ESP32 NodeMCU cuenta con dos núcleos de procesador, lo que permite la ejecución de múltiples tareas de manera simultánea. Esto es especialmente útil en aplicaciones que requieren procesamiento en tiempo real y tareas de fondo.

2. Velocidad de Reloj Ajustable

Los núcleos del ESP32 pueden operar a velocidades de reloj ajustables, lo que permite optimizar el rendimiento y el consumo de energía según las necesidades de la aplicación.

3. Capacidad de Procesamiento de Punto Flotante (FPU)

El ESP32 está equipado con una unidad de procesamiento de punto flotante (FPU), lo que lo hace adecuado para aplicaciones que requieren cálculos matemáticos intensivos, como el procesamiento de señales y la inteligencia artificial.

4. Optimización de Código

Espressif ha desarrollado un conjunto de herramientas y bibliotecas que optimizan el código para el ESP32, lo que facilita la programación eficiente y la maximización del rendimiento del microcontrolador.

5. Coprocesadores Especializados

El ESP32 incluye coprocesadores especializados para tareas como el cifrado y descifrado de datos, lo que libera la carga de trabajo de la CPU principal y mejora el rendimiento en aplicaciones de seguridad y comunicaciones.

La capacidad de transmisión de datos por Wi-Fi es una de las características más destacadas de la ESP32 NodeMCU. Esta capacidad permite a la placa conectarse a redes Wi-Fi, intercambiar datos con servidores y dispositivos en la nube, y comunicarse con otros dispositivos en la red local. A continuación, se detallan las capacidades clave de transmisión de datos por Wi-Fi:

1. Conectividad Wi-Fi Integrada

La ESP32 NodeMCU tiene conectividad Wi-Fi integrada con soporte para los estándares 802.11 b/g/n. Esto permite la conexión a redes Wi-Fi domésticas, empresariales o industriales con diferentes niveles de seguridad y autenticación.

2. Modos de Funcionamiento

Sincrofasores y monitoreo de energía eléctrica trifásica

La placa puede operar en diferentes modos de funcionamiento Wi-Fi, incluyendo estación (cliente), punto de acceso (AP), o en modo estación y punto de acceso simultáneamente. Esto permite una flexibilidad excepcional en la configuración de redes.

3. Seguridad Wi-Fi

El ESP32 NodeMCU admite una variedad de protocolos de seguridad Wi-Fi, como WPA/WPA2, lo que garantiza la protección de datos sensibles durante la transmisión.

4. Bibliotecas y Protocolos de Comunicación

Espressif proporciona bibliotecas y protocolos de comunicación que facilitan la implementación de aplicaciones conectadas a Internet, como HTTP, MQTT y WebSocket.

5. Actualizaciones de Firmware OTA (Over-The-Air)

La ESP32 NodeMCU permite actualizaciones de firmware OTA, lo que significa que puede recibir y aplicar actualizaciones de software a través de la conexión Wi-Fi, lo que simplifica la administración y el mantenimiento de dispositivos en campo.

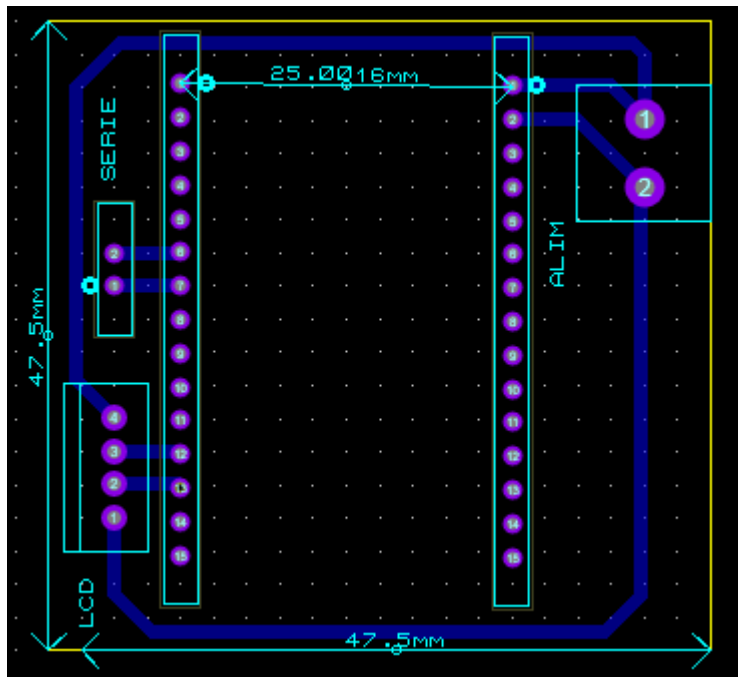


Figura 3.8: Placa Comunicación y calculo ESP32

Por último, se diseñó el hardware de cálculo y comunicación mediante la ESP32. La placa consiste en las conexiones a medida para la ESP32 y tres conectores, uno de alimentación externa, los pines de

Sincrofasores y monitoreo de energía eléctrica trifásica

comunicación serial por donde la ESP recibirá las muestras desde la STM32F407VG con la marca temporal y los pines comunicación I2C para el Display LCD, en donde la PMU muestra la hora recibida en la trama y la IP tanto de Bluetooth como de Internet, cuando esta se encuentre conectada a una red, para ofrecer al usuario información de la comunicación y sincronización de los datos.

Generador de señales UNI-T UTG2025A

El generador de señales UNI-T UTG2025A se destaca como una herramienta esencial para los profesionales de la electrónica y la ingeniería. Con su diseño innovador y una amplia gama de características funcionales, este generador ofrece una solución eficiente y confiable para aplicaciones que requieren señales senoidales de 50Hz con desplazamiento de fase y el uso de disparo externo. Este texto busca analizar exhaustivamente las capacidades y características de este dispositivo, así como su aplicación práctica en entornos de laboratorio y aplicaciones industriales.

Características del UNI-T UTG2025A

1. Funciones de Generación de Señales

El UNI-T UTG2025A ofrece una amplia gama de funciones de generación de señales que lo convierte en una herramienta versátil y adaptable para diversas aplicaciones. Desde señales senoidales básicas hasta formas de onda más complejas, como cuadradas y triangulares, este generador proporciona flexibilidad en la generación de formas de onda precisas y estables. La capacidad de producir señales senoidales de 50Hz de alta calidad, con una distorsión armónica baja y una excelente precisión en la frecuencia, lo convierte en una opción óptima para aplicaciones que requieren alta fidelidad en la señal.

2. Desplazamiento de Fase

El UNI-T UTG2025A se destaca por su capacidad para aplicar desplazamientos de fase precisos, lo que permite a los usuarios ajustar la relación temporal entre múltiples señales. Esta función resulta especialmente útil en aplicaciones donde se requiere el estudio de fenómenos de interferencia y superposición de señales. Con una interfaz de usuario intuitiva y controles precisos, los usuarios pueden configurar fácilmente el desplazamiento de fase según sus requisitos específicos, lo que garantiza una precisión excepcional en los resultados de las pruebas y mediciones.

3. Disparo Externo

La capacidad de sincronización externa del UNI-T UTG2025A permite una integración perfecta con otros dispositivos y equipos de prueba. Al utilizar el disparo externo, los usuarios pueden coordinar la generación de señales con precisión milimétrica con otros instrumentos de medición y sistemas complejos, lo que facilita la realización de experimentos y pruebas sincronizadas en entornos de laboratorio. Esta funcionalidad mejora la eficiencia operativa y la precisión de las mediciones, convirtiéndolo en una herramienta esencial para aplicaciones que requieren una coordinación precisa entre varios equipos.

4. Pantalla y Controles Intuitivos

El generador UNI-T UTG2025A está equipado con una pantalla LCD de alta resolución que proporciona una visualización clara y detallada de los parámetros de la señal generada. Los controles ergonómicos y fáciles de usar permiten a los usuarios ajustar rápidamente los parámetros de la señal, lo que garantiza una operación sencilla y eficiente. Además, la pantalla muestra información crucial, como la frecuencia, la amplitud y el desplazamiento de fase, lo que facilita el monitoreo en tiempo real y la verificación de los parámetros de la señal durante las pruebas.

5. Precisión y Estabilidad

Con una alta precisión en la generación de señales y una estabilidad excepcional, el UNI-T UTG2025A garantiza resultados de prueba confiables y consistentes. La tecnología de generación de señales avanzada y los componentes de alta calidad incorporados en el dispositivo aseguran una baja distorsión armónica y una excelente precisión en la frecuencia, lo que resulta fundamental para aplicaciones que requieren mediciones precisas y confiables. Además, la estabilidad a largo plazo de las señales generadas asegura una coherencia constante durante las pruebas prolongadas, lo que reduce la necesidad de ajustes frecuentes y mejora la eficiencia del proceso de prueba.

Aplicación del UNI-T UTG2025A

La simulación de una onda trifásica es fundamental en aplicaciones de ingeniería eléctrica y sistemas de potencia. Los generadores de señales UNI-T UTG2025A, con sus capacidades avanzadas y su alta precisión, permiten a los profesionales de la industria electrónica y eléctrica realizar simulaciones precisas y confiables de sistemas trifásicos. En este texto, se proporcionará una guía paso a paso sobre cómo conectar y configurar

Sincrofasores y monitoreo de energía eléctrica trifásica

tres generadores UNI-T UTG2025A para simular una onda trifásica con un desfase de 120 grados entre cada fase.

Pasos para Conexión y Configuración

1. Conexión de los Generadores

Conecte los tres generadores UNI-T UTG2025A a una fuente de alimentación adecuada y asegúrese de que estén en un entorno bien conectado a tierra para evitar problemas de interferencia.

Utilice cables de conexión adecuados para establecer una conexión firme entre los generadores y los dispositivos que reciben la onda trifásica simulada.

2. Configuración de Cada Generador

Generador 1:

Encienda el primer generador UTG2025A y navegue hasta la opción de onda senoidal.

Ajuste la amplitud a 8.5Vpp y la frecuencia a 50Hz.

Configure un desfase de 0 grados utilizando la función de ajuste de fase.

Generador 2:

Encienda el segundo generador UTG2025A y seleccione la opción de onda senoidal.

Ajuste la amplitud a 8.5Vpp y la frecuencia a 50Hz.

Configure un desfase de 120 grados utilizando la función de ajuste de fase.

Generador 3:

Encienda el tercer generador UTG2025A y acceda a la opción de onda senoidal.

Ajuste la amplitud a 8.5Vpp y la frecuencia a 50Hz.

Configure un desfase de 240 grados utilizando la función de ajuste de fase.

Esto se realizó con los canales 1 de los tres generadores. Del mismo modo se configuraron los canales 2 para las señales de corriente, con la diferencia que estas la amplitud son de 2.9Vpp.

3. Sincronización de los Generadores

Utilice un método de sincronización adecuado, como un disparador externo o una referencia de tiempo común, para garantizar que los generadores estén sincronizados entre sí y generen una onda trifásica coherente y precisa.

Sincrofasores y monitoreo de energía eléctrica trifásica

Configurar los ajustes de sincronización en cada generador para asegurarse que las señales estén perfectamente coordinadas en términos de tiempo y fase.

4. Verificación de la Onda Trifásica

Conecte un osciloscopio u otro dispositivo acorde de medición a las salidas de los generadores para verificar la forma de onda trifásica generada.

Ajuste los parámetros según sea necesario para garantizar que la forma de onda sea una representación precisa de una onda trifásica con las características requeridas.

Para esto se utilizó un osciloscopio de la marca Tectronix modelo MISO2024 para los canales de tensión y otro osciloscopio de la misma marca Tektronix TDS-3054 y modelo Las figuras 3.9-a y 3.9-b muestran las señales de entrada al instrumento, y con las cuales se realizaron las pruebas detalladas en el capítulo 5.

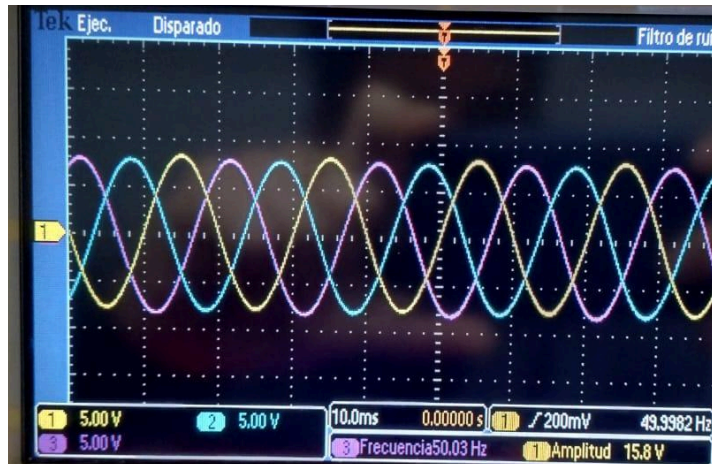


Figura 3.9-a: Canales de tensión IN

Sincrofasores y monitoreo de energía eléctrica trifásica

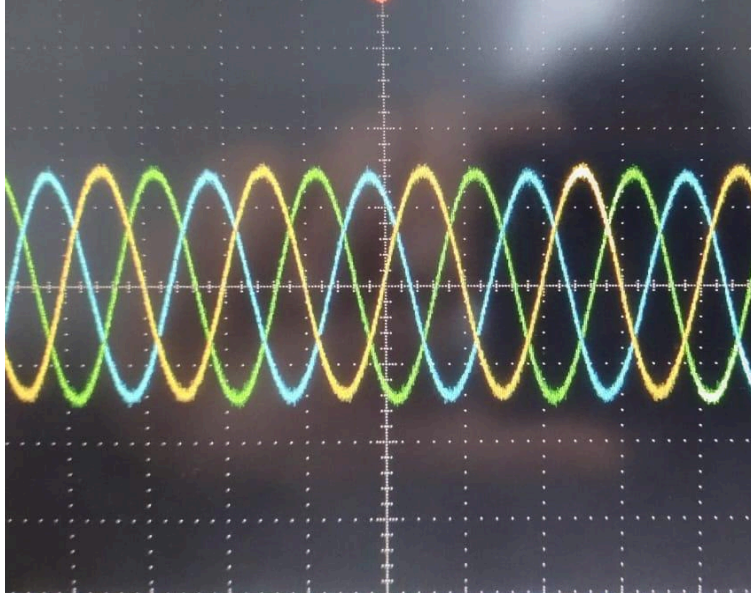


Figura 3.9-b: Canales de corriente IN

Se puede ver lo generado por dichos generadores que respeta la configuración explicada anteriormente, siendo la onda amarilla la señal de fase 0° , la azul de 120° y la rosa-verde (tensión-corriente) de 240° .

A continuación, en la figura 3.10, se pueden ver los tres generadores conectados y sincronizados funcionando.

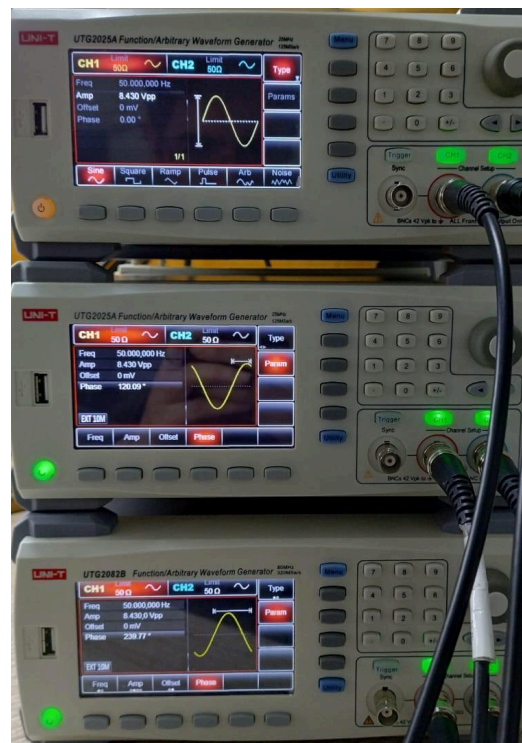


Figura 3.10: Generadores de onda trifásica.

Sincrofasores y monitoreo de energía eléctrica trifásica

El funcionamiento de los mismos se puede verificar en la imagen, siendo que cada uno de los canales esta encendido con la luz testigo de los mismos.

Módulo GPS

Como se ha mencionado, la sincronización de los fasores calculados por la PMU se realiza mediante la utilización de un GPS. El receptor de la marca U-BLOX está integrado por las siguientes partes:

- ◇ Módulo Receptor.
- ◇ Antena GPS.

Módulo receptor

El módulo receptor GPS U-BLOX NEO-6M se ha convertido en una solución líder en el mercado para la recepción y procesamiento precisos de señales GPS. Con su combinación de tecnología avanzada y características funcionales superiores, el módulo NEO-6M de U-BLOX ha ganado reconocimiento por su capacidad para proporcionar datos de ubicación precisos y en tiempo real en una variedad de aplicaciones que requieren una geolocalización fiable. Este texto explora detalladamente las características clave y el funcionamiento del módulo receptor GPS U-BLOX NEO-6M, destacando su versatilidad y aplicación en diversas industrias y aplicaciones que dependen de una localización precisa y confiable.

Funcionamiento del Módulo Receptor GPS U-BLOX NEO-6M

El módulo receptor GPS U-BLOX NEO-6M funciona utilizando una combinación de antenas sensibles y un sistema de procesamiento avanzado para adquirir señales de los satélites GPS y calcular con precisión la ubicación del receptor. El proceso general de funcionamiento se puede describir en los siguientes pasos:

Adquisición de Señal: El módulo receptor NEO-6M utiliza una antena altamente sensible para adquirir señales de los satélites GPS en la constelación, calculando el tiempo y la ubicación basándose en la información recibida de múltiples satélites en órbita.

Procesamiento de Datos: Una vez adquiridas las señales, el módulo realiza un procesamiento interno de los

Sincrofasores y monitoreo de energía eléctrica trifásica

datos recibidos, utilizando algoritmos avanzados para filtrar y corregir posibles errores en las señales de los satélites y calcular la ubicación precisa del receptor en función de la información de tiempo y posición recibida.

Generación de Datos de Ubicación: El módulo receptor GPS U-BLOX NEO-6M genera datos de ubicación precisos en formato de coordenadas geográficas, que se pueden utilizar para rastrear y seguir la posición del receptor en tiempo real con una alta precisión, lo que lo convierte en una solución confiable para aplicaciones de navegación y geolocalización.

Características Principales del Módulo Receptor GPS U-BLOX NEO-6M

1. Sensibilidad y Precisión de Recepción

El módulo receptor GPS U-BLOX NEO-6M se distingue por su alta sensibilidad y precisión en la recepción de señales GPS, lo que garantiza una adquisición rápida y confiable de datos de ubicación incluso en entornos desafiantes y con obstáculos de señal, lo que resulta fundamental para aplicaciones que dependen de una geolocalización precisa y en tiempo real.

2. Eficiencia Energética

Con un consumo de energía optimizado y modos de suspensión de bajo consumo, el módulo receptor GPS NEO-6M de U-BLOX garantiza una duración prolongada de la batería en dispositivos portátiles y aplicaciones que operan con limitaciones de energía, lo que lo convierte en una solución eficiente y confiable para aplicaciones de seguimiento y navegación a largo plazo.

3. Integración Sencilla

El módulo receptor GPS U-BLOX NEO-6M está diseñado para una integración sencilla en una variedad de dispositivos y sistemas, con soporte para una interfaz de comunicación estándar, lo que facilita su implementación en sistemas existentes y acelera el proceso de desarrollo y despliegue de productos que requieren capacidades de geolocalización.

4. Compatibilidad con Múltiples Satélites

El módulo receptor es compatible con múltiples constelaciones de satélites, incluyendo GPS, GLONASS y GALILEO, lo que amplía su cobertura global y mejora la precisión y la disponibilidad de señales en diversas

Sincrofasores y monitoreo de energía eléctrica trifásica

regiones del mundo, lo que resulta esencial para aplicaciones que requieren una localización global precisa y confiable.



Figura 3.11: Modulo receptor GPS NEO6-M

Antena GPS

La antena del GPS NEO-6M de U-BLOX desempeña un papel crucial en la recepción y el procesamiento precisos de señales de satélites GPS. Con su diseño avanzado y su alta sensibilidad, la antena garantiza una adquisición rápida y confiable de señales en una variedad de entornos, lo que la convierte en un componente esencial para aplicaciones que requieren una geolocalización precisa y estable. En este texto, se detallarán minuciosamente las características clave y el funcionamiento de la antena del GPS NEO-6M de U-BLOX, destacando su importancia en el proceso de recepción de señales y su impacto en la precisión y confiabilidad de los datos de ubicación.

Características Principales de la Antena

1. Sensibilidad de Recepción Mejorada

La antena del GPS NEO-6M de U-BLOX está diseñada con una sensibilidad mejorada que le permite captar señales débiles de los satélites GPS, lo que resulta en una adquisición rápida y confiable de datos de ubicación

Sincrofasores y monitoreo de energía eléctrica trifásica

incluso en entornos con obstáculos de señal y condiciones atmosféricas desafiantes. Esta característica garantiza una mayor precisión y estabilidad en la determinación de la ubicación del receptor, lo que resulta esencial para aplicaciones que dependen de una geolocalización precisa y en tiempo real.

2. Cobertura de Múltiples Bandas

La antena del GPS NEO-6M de U-BLOX ofrece soporte para múltiples bandas de frecuencia, lo que le permite recibir señales de diferentes constelaciones de satélites, como GPS, GLONASS y GALILEO. Esta capacidad de cobertura de múltiples bandas amplía su alcance global y mejora la precisión y la disponibilidad de señales en diversas regiones, lo que resulta fundamental para aplicaciones que requieren una localización global precisa y confiable.

3. Diseño Compacto y Duradero

La antena está diseñada para ser compacta y duradera, lo que la hace adecuada para una variedad de aplicaciones en entornos industriales y comerciales. Su diseño robusto y resistente a la intemperie garantiza un rendimiento confiable incluso en condiciones ambientales adversas, convirtiéndola en una solución fiable y duradera para aplicaciones que operan en entornos exteriores exigentes.

4. Fácil Integración

Con su diseño versátil y su compatibilidad con una variedad de dispositivos y sistemas, la antena del GPS NEO-6M de U-BLOX se integra fácilmente en una amplia gama de productos y soluciones que requieren capacidades de geolocalización. Su compatibilidad con interfaces estándar simplifica su implementación en sistemas existentes, lo que acelera el proceso de desarrollo y despliegue de productos que dependen de una recepción de señal GPS precisa y estable.

Funcionamiento de la Antena

La antena del GPS NEO-6M de U-BLOX funciona utilizando un diseño de antena activa que amplifica la señal recibida de los satélites GPS y la transmite al módulo receptor para su procesamiento. El proceso general de funcionamiento se puede describir en los siguientes pasos:

Recepción de Señales: La antena recibe señales de múltiples satélites GPS y utiliza su sensibilidad mejorada para captar señales débiles y fuertes con precisión y estabilidad.

Sincrofasores y monitoreo de energía eléctrica trifásica

Amplificación de Señales: La antena activa amplifica las señales recibidas para mejorar la calidad y la fuerza de la señal, lo que garantiza una transmisión confiable al módulo receptor para su procesamiento.

Transmisión de Datos: Una vez que la señal se amplifica, se transmite al módulo receptor para su posterior procesamiento y cálculo de la ubicación precisa del receptor en función de la información recibida de los satélites GPS.



Figura 3.12: Antena externa del módulo GPS

4. Capítulo 4: Firmware y Software

Introducción

En este capítulo se presenta el desarrollo del software implementado para la realización de este proyecto, que consta de una unidad de medición fasorial sincronizada vía satélite.

De manera detallada, se muestran las etapas necesarias para el desarrollo del PMU, entre las cuales se tienen los algoritmos de medición de las señales, la determinación de fasores de voltaje y corriente, obtención de la frecuencia, sincronización mediante el uso del GPS y la visualización de los datos.

Firmware de la STM32F407VG

Al firmware de esta placa, se lo puede describir mediante sus tres componentes principales. La unidad de medición, el sistema de obtención de la hora y el envío de los datos a la unidad de procesamiento y comunicación.

Para medir los seis canales necesarios se utilizan dos ADCs, cada uno con tres canales. A cada ADC, se le asocia un DMA. Estos DMAs, están configurados para tomar un conjunto de 150 muestras (50 muestras por canal), cada una separadas por intervalos de 40us. Cuando el DMA finaliza el muestreo, dispara una interrupción que habilita el procesamiento. Esto da paso, al envío de los datos a la unidad complementaria. Luego de enviados los datos, se reinicia la medición. Los datos se almacenan intercalados, en su correspondiente arreglo, diferenciando entre tensión y corriente. Es decir, los primeros valores del arreglo de tensión o corriente, quedarían como se indica en la imagen:

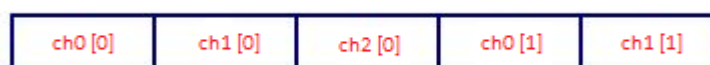


Figura 4.1: Ubicación de los datos de los sucesivos en el arreglo asociado al DMA.

Sincrofasores y monitoreo de energía eléctrica trifásica

Para el envío de los datos, se utiliza una comunicación serie por protocolo RS232, con una velocidad de 9600 baudios 8N1 (8 bits, sin paridad, y un bit de stop). Se envían primero todas las mediciones sucesivas de tensión, y luego las de corriente. Los datos se separan con un carácter ‘,’ (coma). Para diferenciar entre ráfagas de tensión y corriente, se añade un signo ‘;’ (punto y coma) al final de cada ráfaga. Luego de las mediciones, se envía la hora en su formato estándar (HH:MM:SS). Se añade un signo ‘&’ (ampersand) antes de la hora para indicar el inicio de la misma, y un signo ‘|’ (barra vertical) para indicar el final de la secuencia de transmisión.

Para la obtención de la hora, se estableció una comunicación serie con el GPS, como se indicó previamente. Una vez recibida la cadena completa que incorpora todas las tramas, se envía esta cadena a una función. La función despeja las tramas GPRMC y GPGGA (que incorporan los datos horarios), y obtiene hora, minuto y segundo, de las mismas, en la posición correcta según corresponda. Esto sí, y solo sí, la trama llega correctamente y sin corrupción. De lo contrario, devuelve estado nulo, indicando de información.

Firmware de la ESP32

El firmware de esta placa está definido por dos grandes partes: los cálculos y la comunicación. Se describen primero los cálculos.

Luego de generadas las matrices con los valores ordenados (se detalla este proceso en la descripción de la comunicación), se comienza calculando la frecuencia. Para esto, primeramente se normalizan los valores, convirtiéndolos a un rango entre -1 y 1. El algoritmo de cálculo de frecuencia, obtiene los cruces por cero de la señal, diferencia entre aquel cruce producido por un flanco positivo (“cruce1”), y el producido por un flanco negativo (“cruce2”). Para lograr una mayor aproximación, reconoce cuando es la muestra más cercana a cero, y la elige como representante del cruce. Utilizando la diferencia de muestras, y conociendo el intervalo entre estas, se calcula la frecuencia como la inversa de esta diferencia en tiempo. Para mejorar la precisión, se multiplica por 10 la diferencia entre muestras, y se divide por 10 al resultado obtenido. Esto no afecta la exactitud, pero si mejora la precisión.

Sincrofasores y monitoreo de energía eléctrica trifásica

```
double detectar_frecuencia(double *x, uint8_t *cruce1, uint8_t *cruce2, double frecs[N_FRECS], uint8_t index_array_freq) {  
    /*  
     * Con un periodo entre muestras de 40us, calcula frecuencia.  
     */  
    double frecuencia = 0;  
    uint32_t suma_freces = 0;  
    uint8_t i = 0;  
  
    detectar_cruces(x, cruce1, cruce2);  
  
    if (*cruce1 - *cruce2 != 0) {  
        // Para evitar que de infinito  
        double diferencia = abs(*cruce1 - *cruce2) * 0.00004;  
        frecuencia = 1.0 / (diferencia * 2);  
    }  
    frecuencia /= 10.0;  
  
    frecs[index_array_freq] = frecuencia;  
    for (i = 0; i < N_FRECS; i++) {  
        suma_freces += frecs[i];  
    }  
  
    return suma_freces / (double)N_FRECS;  
}
```

Figura 4.2: Captura del algoritmo de cálculo de frecuencia.

Luego, se calculan las magnitudes. Para esto, se convierten las mediciones originales a valores RMS, tanto de tensión o corriente. Se procesan estos valores, según el cálculo detallado previamente, y se obtiene para cada canal, el valor RMS de magnitud fasorial.

```
double obtener_magnitud(double *x) {  
    uint8_t i;  
    uint32_t suma_valores = 0;  
    for(i = 0; i < N; i++) {  
        suma_valores += x[i] * x[i];  
    }  
  
    return sqrt(suma_valores / N);  
}
```

Figura 4.3: Captura del algoritmo de cálculo de la magnitud.

Se continúa buscando la diferencia de ángulo entre fases. El algoritmo de obtención de fase, reutiliza los cruces por cero, calculando la fase entre señales, como la diferencia de muestras entre cruces relativos para

Sincrofasores y monitoreo de energía eléctrica trifásica

fases sucesivas, en relación al periodo. Es decir, para el mismo cruce (flanco positivo), se calcula la diferencia de tiempo (diferencia de muestra por intervalo entre ellas), se divide sobre el periodo, y multiplica por 360 para obtener la relación en grados.

```
double obtener_angulo_entre_fases(uint8_t cruce1_x1, uint8_t cruce1_x2) {
    int8_t dif_cruces = cruce1_x1 - cruce1_x2;
    double fase = (abs(dif_cruces) * 0.0004) / PERIODO;
    fase = fase * 360; // Fase en grados
    if (dif_cruces < 0) {
        fase = 360 - fase;
    }

    return fase / 2.0;
}
```

Figura 4.4: Captura del algoritmo de detección de fase entre señales.

A esto se le agrega una etapa de cálculo redundante de fases. Se obtienen las fases entre ambas señales relacionadas a la de cálculo actual, y se selecciona la más confiable. Para esto, se elige la más cercana a 120 grados.

```
double obtener_fase(uint8_t cruce1_x0, uint8_t cruce1_x1, uint8_t cruce1_x2, uint8_t index) {
    double fase, fase1, fase2;
    if (index == 0) {
        fase1 = obtener_angulo_entre_fases(cruce1_x0, cruce1_x1);
        fase2 = obtener_angulo_entre_fases(cruce1_x2, cruce1_x0);
    } else if (index == 1) {
        fase1 = obtener_angulo_entre_fases(cruce1_x0, cruce1_x1);
        fase2 = obtener_angulo_entre_fases(cruce1_x1, cruce1_x2);
    } else if (index == 2) {
        fase1 = obtener_angulo_entre_fases(cruce1_x2, cruce1_x0);
        fase2 = obtener_angulo_entre_fases(cruce1_x1, cruce1_x2);
    } else {
        fase1 = 0;
        fase2 = 0;
    }
    // Calculo las distintas fases redundantemente para elegir aquella
    // que este mas cerca de 120. Esto se hace porque cuando una fase se ve
    // alterada y se cae, se puede utilizar la otra para calcular la fase actual

    if (comparar_igualdad(abs(120 - fase1), abs(120 - fase2), 2)) fase = fase1;
    else if (abs(120 - fase1) > abs(120 - fase2)) fase = fase2;
    else fase = fase1;

    return fase;
}
```

Figura 4.5: Captura del algoritmo de redundancia de fases

Sincrofasores y monitoreo de energía eléctrica trifásica

Finalmente, el último parámetro deseado es el factor de potencia para cada fase. Esto se calcula simplemente, como el coseno de la diferencia angular de fase entre tensión y corriente para líneas sucesivas.

```
double obtener_cos_fi(uint8_t cruceV, uint8_t cruceI) {  
    int8_t dif_cruces = (cruceV - cruceI) +4;  
    double fi = (dif_cruces * 0.0004) / PERIODO;  
    fi = fi * 360; // Fase en grados  
  
    return cos((fi * PI)/180.0);  
}
```

Figura 4.6: Algoritmo de cálculo de cos Phi.

El coseno de Phi se calcula si y sólo si se detecta presencia de tensión y corriente para la fase en cuestión.

En este algoritmo se puede observar una compensación en los cruces recibidos de cuatro muestras. Esto se debe a que ocurre un desfase previo entre las ondas de tensión y corriente de los respectivos canales. Este desfase se puede ver en la figura 4.6.a:

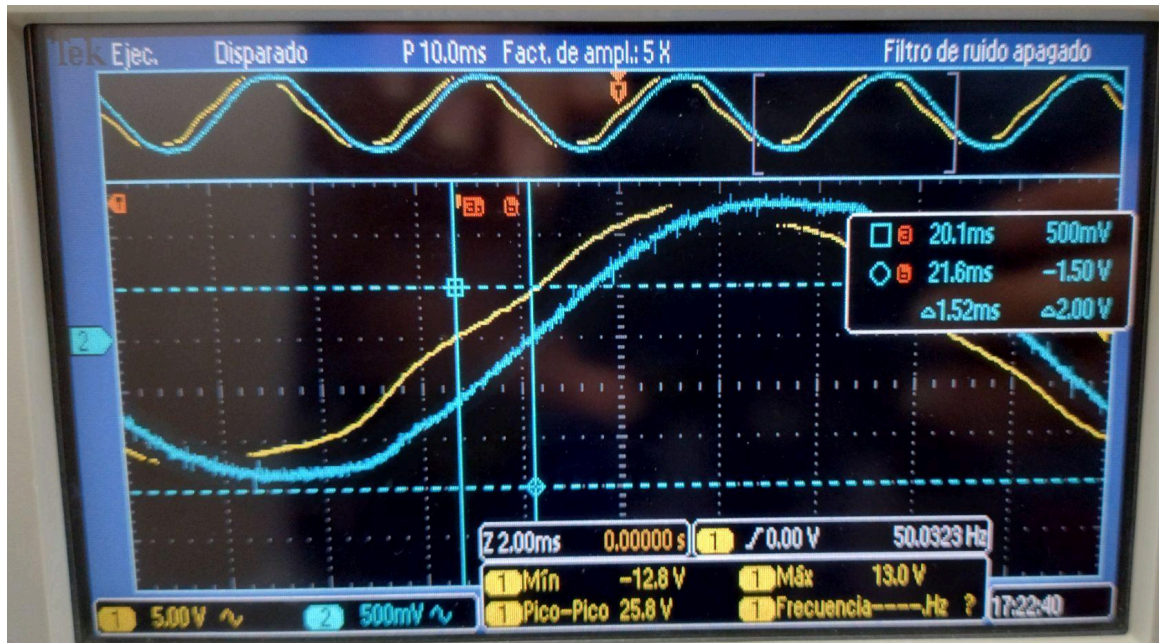


Figura 4.6.a: Desfasaje entre canales de tensión (amarilla) y corriente (azul).

Esto se debe a que los canales de tensión están atenuados mediante el uso de transformadores, para tener un aislamiento galvánico entre la entrada de 220VAC y la salida de 6VAC, las cuales ingresan a las etapas de adaptación a los ADC para ser muestreadas. Esto se puede ver en la figura 4.6.b.



Figura 4.6.b: Desfasaje de tensión 220 VAC (verde) y tensión 6 VAC (amarilla) de los transformadores de tensión.

De esta manera queda expuesto y justificado el algoritmo realizado con la compensación vista para el correcto funcionamiento del dispositivo.

Para la parte de comunicación, se utilizan dos medios. Bluetooth, para comunicar con una aplicación móvil, que permite configurar la red WiFi a la que se conecta el dispositivo. Al iniciar, el sistema muestra en el display LCD, la dirección MAC de la antena Bluetooth. En la aplicación, debe introducirse este dato, para conectar el dispositivo móvil con el equipo. Se eligió este sistema por ser sólido y seguro. Al concretar exitosamente la conexión, el equipo envía una cadena que incorpora todas las SSIDs de las redes WiFi disponibles. Esto se presenta en la aplicación móvil, y luego de que el usuario elija la red a la que quiere que el equipo se conecte, junto a su contraseña, se envían los datos y se intenta la conexión. El equipo envía un dato en respuesta, indicando éxito o fracaso en el intento de conectarse. En el caso de conexión exitosa, se deshabilitan todas las opciones en la aplicación, dejando solo disponible información de la MAC Bluetooth, y al SSID de la red conectada.

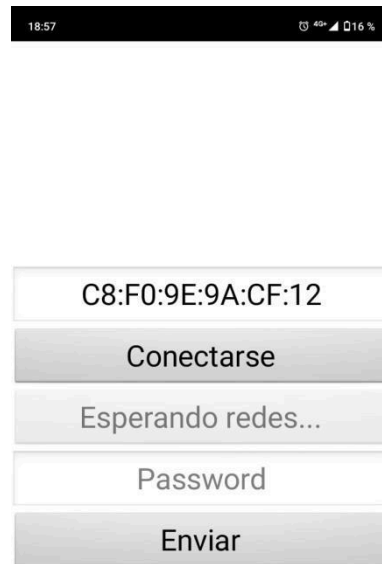


Figura 4.7: Captura de la aplicación móvil.

En el lado del firmware, una vez que finalizó esta etapa se desactiva la comunicación Bluetooth, y se deja sólo activa la comunicación WiFi. En el ciclo principal, la placa espera recibir datos por el puerto serie. Al recibirlos, los procesa, almacenando los valores en dos matrices, una para tensión y otra para corriente. Luego, se realizan los cálculos descritos previamente, y al finalizar se genera una cadena que incluye toda la información obtenida de los cálculos. Estas cadenas se envían por WiFi, en un puerto fijo, que reconoce el software del lado de la PC.

Las cadenas formateadas son dos, una para cada parámetro, y se componen de 13 datos. El primero es la hora, en su formato estándar. Se agrega un símbolo ‘&’ (ampersand) como separador. Y luego se agregan los siguientes datos, separados por coma:

- Frecuencia del primer canal.
- Magnitud del primer canal.
- Fase angular del primer canal, respecto al segundo.

Sincrofasores y monitoreo de energía eléctrica trifásica

- Frecuencia del segundo canal.
- Magnitud del segundo canal.
- Fase angular del segundo canal, respecto al tercero.
- Frecuencia del tercer canal.
- Magnitud del tercer canal.
- Fase angular del tercer canal, respecto al primero.
- Factor de potencia del primer canal.
- Factor de potencia del segundo canal.
- Factor de potencia del tercer canal.

PDC (Processing Data Center - Centro de Procesamiento de Datos)

Este es un software que permite obtener y procesar, con el fin de visualizar, las mediciones obtenidas de la red a analizar. Está compuesto por dos grandes partes. Por un lado, los gráficos que presentan los fasores en módulo ($\text{Valor}_{\text{RMS}}$) y ángulo, de cada una de las líneas. Y por el otro, un conjunto de indicadores que representan la información específica en cuanto a frecuencia, factor de potencia, sincronismo, entre otros parámetros relevantes. Además, incorpora un registro de datos.

Pasando a la descripción del código, el mismo se centra en la herramienta de comunicación por sockets UDP, que al recibir datos, dispara eventos que actualizan la información en los gráficos e indicadores. Además, estas mismas instancias almacenan los datos recibidos en los registros.

En los registros, no sólo se guardan las mediciones convencionales, sino que además, se registran en un archivo aparte las perturbaciones. En la pantalla se puede ver también un indicador en la zona lateral derecha, que indica la hora en la que ocurrió por última ocasión uno de estos eventos (notado como “Última alarma”).

Si bien depende el equipo en el cual se ejecuta, además de la condiciones de la conexión inalámbrica, se notó que en ocasiones las mediciones se acumulaban en el buffer serie de la comunicación WiFi, produciendo que al intentar procesar esto, el software perdiera información o se trabara. Por ésto se incorporó un buffer, que divide las cadenas acumuladas para un procesamiento intercalado. Esto soluciona el problema y disminuye la tendencia a errores del software.

Sincrofasores y monitoreo de energía eléctrica trifásica

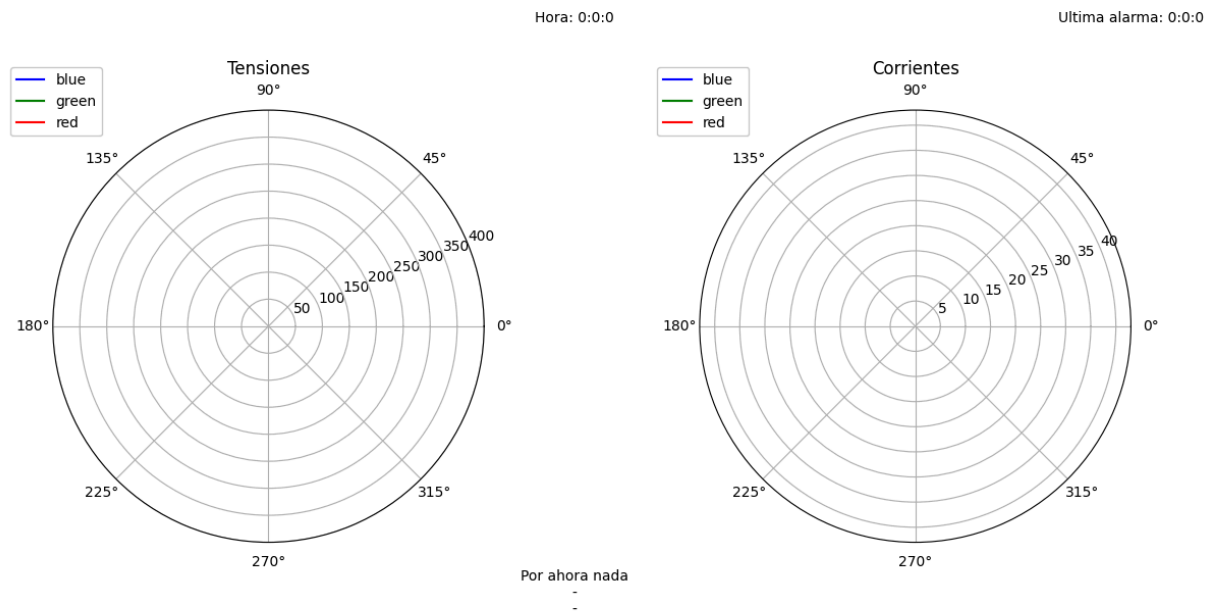


Figura 4.8: Captura del software sin información.

5. Capítulo 5: Ensayos

En este capítulo se analiza la Unidad de Medición Fasorial, mostrando resultados de las mediciones realizadas por la misma. Se hace el análisis de la respuesta ante diferentes señales de prueba introducidas por el Generador de Señales.

5.1 Introducción

Como se mencionó anteriormente, es necesario definir, con señales en tiempo real, un tiempo de referencia para medir los ángulos de fase en forma sincronizada. El estándar IEEE 1344-1995 define el inicio de un segundo como el tiempo de referencia para establecer el valor del ángulo del fasor. La utilidad de un fasor aumenta si pueden ser definidos en una base de tiempo común, logrando ésto si la señal de entrada medida en varios sitios se sincroniza una respecto a la otra. El estándar mencionado define una forma de onda en estado estacionario como aquella cuya magnitud, frecuencia y ángulo de fase no varían con el tiempo.

5.2. Funcionamiento del instrumento

Como primera prueba del correcto funcionamiento, se inició el ensayo con el conexionado del instrumento. En la figura 5.1 se observa al instrumento conectado y funcionando.



Figura 5.1: Instrumento conectado y funcionando

En la imagen presentada se muestra la información de la red mostrada (IP: 192.168.0.101) asignada por el router WiFi y también la hora UTC como hora de sincronismo de las mediciones realizadas.

Por otro lado, se observan 4 leds, cada uno de los cuales indica una etapa de funcionamiento del equipo. Esto se detalla a continuación:

- Rojo: Indica la correcta recepción de las tramas de GPS para ser procesadas y así obtener la hora UTC.
- Verde: Este marca la correcta medición y llenado del buffer con los datos adquiridos por los ADCs con la utilización de los DMA.
- Amarillo: Es el indicador del correcto envío de tramas de medicion para la ESP.
- Naranja: Este es el encargado de indicar el correcto funcionamiento del sistema completo.

Una vez comprobado ésto, se continuó con el ingreso de señales externas para corroborar, en primera instancia, el correcto procesamiento y envío de los parámetros a obtener para un buen análisis de la red eléctrica. La figura 5.2 y 5.3 muestra las señales de tensión y corriente ingresados a cada uno de los canales de medición del instrumento.

Sincrofasores y monitoreo de energía eléctrica trifásica

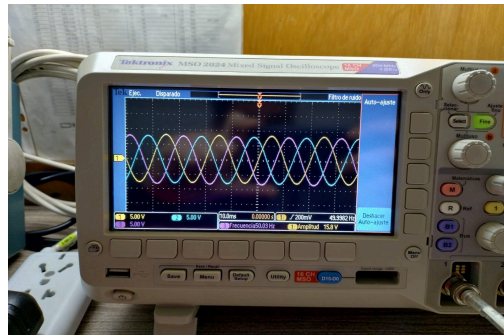


Figura 5.2: Señal de tensión ingresada

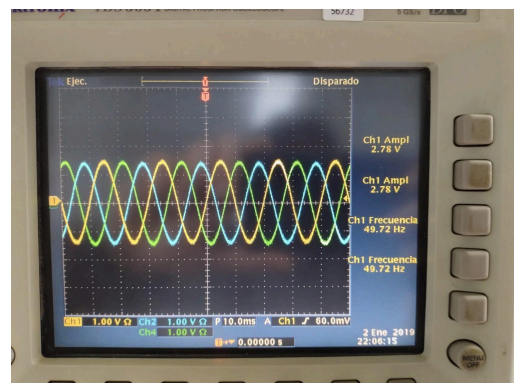


Figura 5.3: Señal de corriente ingresada

En las figuras presentadas se observan las señales ingresadas para la medición, con valores adaptados para el correcto funcionamiento del equipo, simulando una red eléctrica trifásica estable y sin perturbaciones.

Con la inyección de estas señales, y conectado el instrumento a la red WiFi, se comenzó con los ensayos de funcionamiento en cuanto a veracidad de mediciones y correcto envío de las tramas. La figura 5.4 demuestra el software con la recepción de las tramas ingresadas.

Sincrofasores y monitoreo de energía eléctrica trifásica

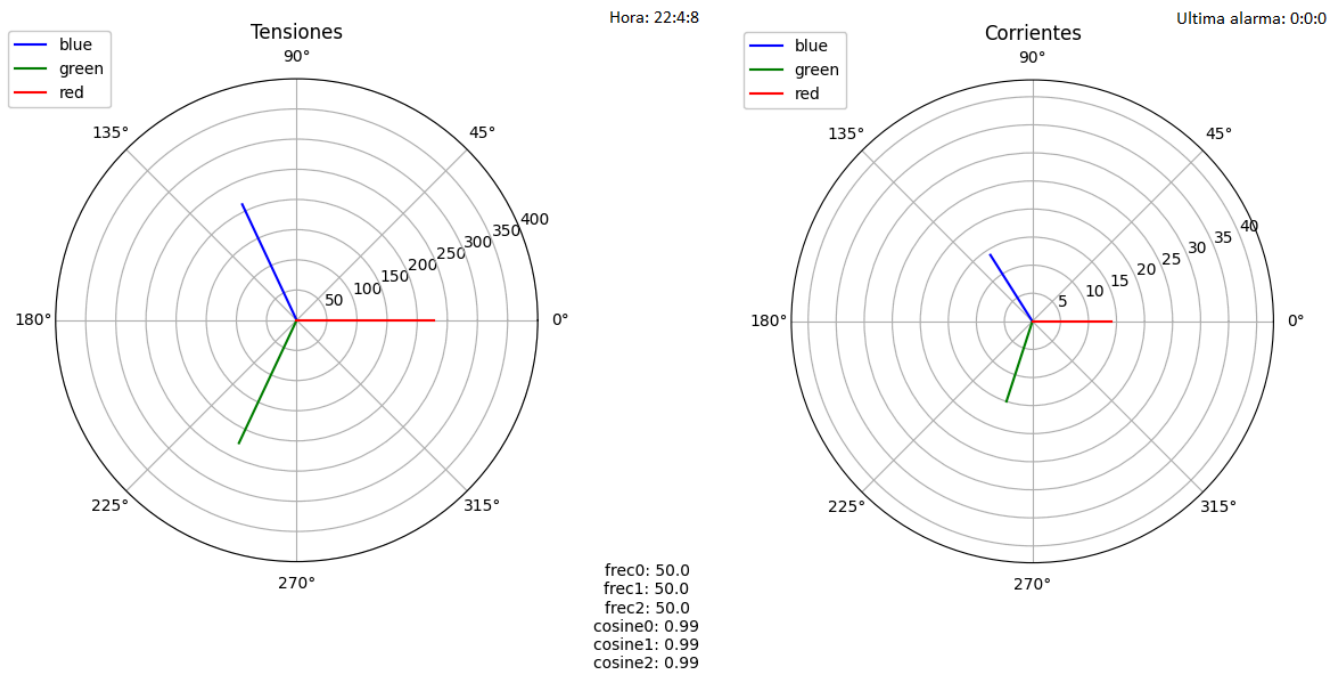


Figura 5.4: Sistema de monitoreo funcionando correctamente

Se puede observar que el instrumento responde de buena manera al ingreso de las señales, tomando las muestras de las señales, discriminándolas por canal y obteniendo así una resolución de cálculos coherente y claramente representada en el software de PDC.

Con ésto visto, se procedió al control del Logger, o registro de datos obtenidos, con lo que se obtuvieron los resultados que se observan a continuación, según se muestra en la figura 5.5:

Sincrofasores y monitoreo de energía eléctrica trifásica

```
23:15:50,54.35,211.98,115.2,52.08,221.91,129.6,46.3,225.23,115.2,0.51,0.7,0.51,tensiones
23:15:50,52.08,14.53,115.2,52.08,14.59,129.6,46.3,14.25,115.2,0.51,0.7,0.51,corrientes
23:15:55,52.08,211.57,122.4,46.3,222.23,122.4,54.35,225.65,115.2,0.99,0.7,0.51,tensiones
23:15:55,52.08,14.63,122.4,48.08,14.83,129.6,56.82,14.35,108.0,0.99,0.7,0.51,corrientes
23:15:55,46.3,212.67,122.4,46.3,222.46,122.4,52.08,226.28,115.2,0.99,0.7,0.51,tensiones
23:15:55,48.08,14.56,122.4,46.3,14.7,129.6,54.35,14.39,108.0,0.99,0.7,0.51,corrientes
23:15:55,54.35,14.56,122.4,56.82,14.73,115.2,48.08,14.25,122.4,0.7,0.99,0.99,corrientes
23:15:55,54.35,212.0,122.4,46.3,222.41,122.4,52.08,225.92,115.2,0.99,0.99,0.51,tensiones
23:15:55,48.08,14.87,122.4,52.08,14.73,122.4,48.08,14.21,115.2,0.99,0.99,0.51,corrientes
23:15:55,46.3,213.37,122.4,54.35,221.65,122.4,48.08,225.82,115.2,0.99,0.99,0.51,tensiones
23:15:55,48.08,14.83,122.4,52.08,14.7,129.6,46.3,14.07,108.0,0.99,0.7,0.51,corrientes
23:15:55,52.08,212.59,122.4,52.08,222.28,122.4,48.08,225.58,115.2,0.99,0.99,0.51,tensiones
23:15:55,52.08,14.9,122.4,46.3,14.42,122.4,52.08,14.21,115.2,0.99,0.99,0.51,corrientes
23:15:55,54.35,211.93,122.4,46.3,221.93,122.4,52.08,225.77,115.2,0.99,0.99,0.51,tensiones
23:15:55,46.3,212.34,122.4,54.35,221.65,122.4,54.35,226.04,115.2,0.99,0.99,0.51,tensiones
23:15:55,46.3,14.87,115.2,50.0,14.49,129.6,48.08,14.59,115.2,0.99,0.99,0.51,corrientes
23:15:55,48.08,212.38,122.4,46.3,221.8,129.6,54.35,226.03,108.0,0.99,0.7,0.51,tensiones
23:15:55,52.08,14.76,122.4,48.08,14.28,129.6,52.08,14.59,108.0,0.51,0.7,0.38,corrientes
```

Figura 5.5: Registro de datos comunes normales de la red eléctrica

En la imagen presentada se puede ver los registros identificados por canal, distribuidos de la siguiente manera: Tensión-Corriente – Ch1, Tensión-Corriente – Ch2, Tensión-Corriente – Ch3.

Del mismo modo se realiza con una acción de alarma. En un archivo aparte se registran los datos de alarma que hacen alertar y realizar un análisis puntual de cada una de las líneas monitoreadas para poder determinar el desperfecto que hace a la perturbación observada en el PDC. En la figura 5.6, se observa una porción de dicho archivo:

```
21:20:32,56.42,7.06,16.03,46.25,43.33,177.18,58.92,55.04,85.26,1.0,1.0,1.0,corrientes
21:20:33,51.56,72.27,124.29,45.1,64.57,89.38,55.96,8.92,97.61,1.0,1.0,1.0,corrientes
21:20:34,48.08,69.35,47.53,41.41,31.82,74.48,43.29,15.01,153.99,1.0,1.0,1.0,corrientes
21:20:35,47.61,51.24,96.47,49.61,80.16,136.02,46.24,88.08,3.45,1.0,1.0,1.0,corrientes
21:20:36,51.11,40.44,11.54,44.56,20.98,128.25,47.93,80.22,98.42,1.0,1.0,1.0,corrientes
21:20:37,41.18,10.41,61.12,51.88,83.53,130.27,49.78,82.88,27.73,1.0,1.0,1.0,corrientes
21:20:38,43.1,78.66,81.18,49.92,29.14,163.6,57.49,98.2,125.75,1.0,1.0,1.0,corrientes
21:20:39,44.18,15.82,48.71,58.95,11.34,143.57,40.93,42.31,74.88,1.0,1.0,1.0,corrientes
21:20:40,43.84,79.78,147.25,45.54,20.08,3.02,46.22,3.28,168.27,1.0,1.0,1.0,corrientes
21:20:41,49.3,32.9,106.46,40.11,63.77,145.45,59.34,2.64,51.27,1.0,1.0,1.0,corrientes
21:20:42,48.69,32.92,64.47,40.7,58.24,165.2,53.76,78.33,32.9,1.0,1.0,1.0,corrientes
21:20:43,42.36,79.31,156.92,40.1,49.21,58.69,49.07,42.65,173.37,1.0,1.0,1.0,corrientes
21:20:44,41.52,5.42,127.04,59.47,91.11,92.86,46.98,88.99,160.78,1.0,1.0,1.0,corrientes
21:20:45,57.22,2.77,22.91,57.28,26.2,92.19,52.52,95.85,94.84,1.0,1.0,1.0,corrientes
21:20:46,49.71,16.02,159.72,52.41,6.62,16.78,53.24,45.08,117.66,1.0,1.0,1.0,corrientes
21:20:47,47.93,40.73,144.24,43.87,45.35,106.92,59.53,17.84,71.26,1.0,1.0,1.0,corrientes
21:20:48,45.93,78.34,18.22,41.47,6.21,9.14,44.14,49.83,78.89,1.0,1.0,1.0,corrientes
21:20:49,47.74,77.79,165.06,46.79,48.57,136.68,56.75,78.18,50.68,1.0,1.0,1.0,corrientes
```

Figura 5.6: Archivo de detección de alertas o alarmas.

Sincrofasores y monitoreo de energía eléctrica trifásica

Como muestra la figura 5.6, en este caso las alertas son por sobrecorriente en las líneas de tendido eléctrico por lo que sólo muestra con un cartel al final de cada cadena “corrientes”. En el caso de que la alarma sea por sobretensión, entonces la alerta será con la indicación “tensión” al final de cada cadena.

5.3 Medición de frecuencia

En primera instancia, se pone a prueba el algoritmo de medición de frecuencia. Como se explicó anteriormente, éste funciona detectando los cruces de la onda por el “0”, lo cual nos da un error bajo aunque considerable en las mediciones instantáneas, pero en el promedio de muestras detectadas en 1 segundo se puede ver un error suficientemente bajo, lo cual garantiza la veracidad de la medición. A continuación, en la tabla 5.1 se observa el resultado de lo explicado.

Fase	Frecuencia	Frecuencia IN
B	50,632	50Hz
G	49,244	
R	50,805	

Tabla 5.1: Medición de frecuencia en la ventana temporal de 1 seg

Con dichos datos obtenidos, se procede a calcular el error en la medición. Según la norma, el error es el considerado en la ventana temporal del dispositivo, por lo que en este caso, al poseer un refresco de 1 segundo entonces el error en este algoritmo es el siguiente

$$E_{fB} = \frac{f_m - f_{teo}}{f_m} * 100 = 1,264\% \rightarrow E_{fG} = 1,512\% \rightarrow E_{fR} = 1,61\%$$

Se puede ver que todas las fases están por debajo de 2% de error, lo cual califica dentro del rango establecido por la norma.

5.4 Valor RMS

Para el valor RMS de tensión se procedió de la misma manera que para la medición de frecuencia. En la tabla 5.2 se puede ver los parámetros obtenidos.

Fase	Frecuencia	Vrms	Vrms IN
B	50,632	212,345	220V
G	49,244	222,074	
R	50,805	225,911	

Tabla 5.2: Medición de tensión RMS

Se puede observar una amplitud de valores promedio más amplio por fase, lo que pone a prueba aún más al prototipo realizado para garantizar la independencia de los canales.

$$E_{fBrms} = ((V_{rmsB} - V_{rmsIN})/V_{rmsIN}) = 3,5\% \rightarrow E_{fGrms} = 1.24\% \rightarrow E_{fRrms} = 2.68\%$$

5.5 Ensayos en redes reales

Una vez resuelto los ensayos controlados mediante los generadores mencionados, se procede al ensayo en campo, el cual fue realizado con la red trifásica instalada en el laboratorio del 3er piso de la Facultad.

Para ello fue necesaria la construcción de un diseño reductor y adaptador tanto de tensión como de corriente.

Para la reducción de la onda de tensión, se implementó una conexión estrella con el primario de tres transformadores de 220V/6V, como se puede observar en la figura 5.7

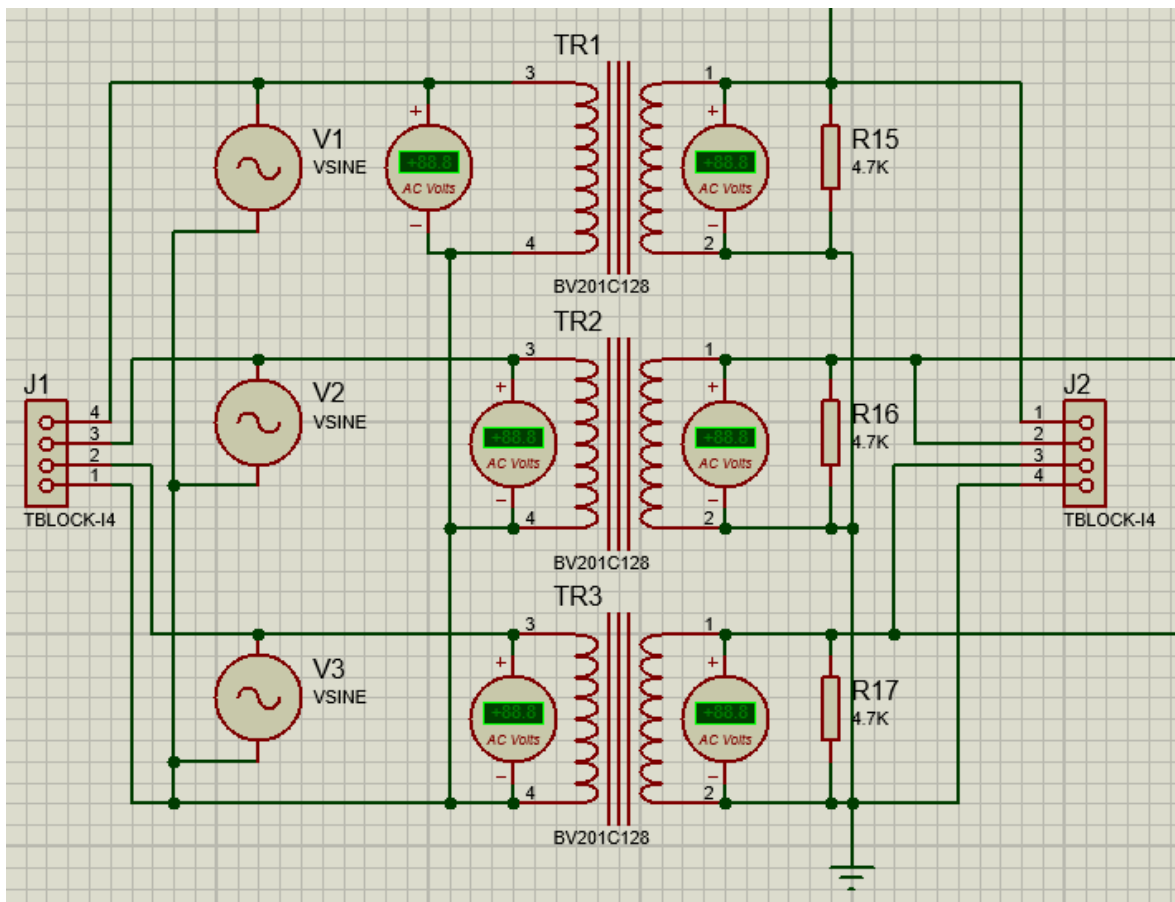


Figura 5.7: Conexión de transformadores para la reducción de la onda de tensión con aislación galvánica

En la figura se puede observar que los tres transformadores están conectados a cada una de las fases, de forma simétrica mientras que el terminal restante de cada uno de los primarios están unidos. En primer lugar, la simetría de la conexión se debe a que se necesita que los transformadores tengan su salida ascendente al mismo tiempo, al igual que su salida descendente. En otras palabras, se necesita que estos dispositivos entreguen la misma salida frente a una misma entrada. Luego se colocó una carga resistiva fija a la salida de cada transformador, con el fin de evitar la saturación del núcleo de los mismos, ya que estos dispositivos no son del tipo de medición, con la particularidad de éstos de que su núcleo no se satura.

Para la etapa de corriente, se diseñó una conexión simple con módulos sensores de efecto Hall (ACS712). Los mismos se conectaron de tal forma que cada uno interrumpe una fase de la red, es decir que para medir las corrientes de cada una de las líneas se cuenta con un módulo sensor de efecto Hall independiente. {Esto se debe a que con uno solo no se pueden captar las tres ondas de corriente a la vez.

Sincrofasores y monitoreo de energía eléctrica trifásica

Estos sensores tienen una sensibilidad particular, dependiendo del modelo que se utilice. A continuación, en la tabla 5.3, se presentan los modelos disponibles que se encuentran en el mercado.

Part Number	T _A (°C)	Optimized Range, I _p (A)	Sensitivity, Sens (Typ) (mV/A)
ACS712ELCTR-05B-T	-40 to 85	±5	185
ACS712ELCTR-20A-T	-40 to 85	±20	100
ACS712ELCTR-30A-T	-40 to 85	±30	66

Tabla 5.3: Modelos de sensores de efecto Hall

Reproducido desde "<https://www.sparkfun.com/datasheets/BreakoutBoards/0712.pdf>"

En esta tabla se puede ver que al variar el rango admisible con el que cada modelo puede trabajar, varía su sensibilidad. Para el ensayo se utilizó el modelo ACS712ELCTR-05B-T, debido a que las cargas para cada línea del tendido no es de gran consumo, por lo cual se necesita una gran sensibilidad para que la señal de salida del sensor no quede por debajo del piso de ruido del sistema.

Como se explicó, las etapas de transformación de la onda pura se encuentran en un gabinete aparte, comunicado mediante canales. Esto es por una cuestión de seguridad en el diseño que se extrapola a la seguridad de los ensayos para el usuario y para el módulo Master del prototipo.

Una vez resuelto esto, se comenzó a ensayar el equipo. Se utilizaron tres cargas resistivas puras, para poder corroborar la medición del factor de potencia en cada una de las líneas y la frecuencia, contrastando el resultado con el analizador de redes de la marca Meatrol modelo ME440.

En primer lugar, en las imágenes 5.8 y 5.9, se muestran las cargas utilizadas para el ensayo:



Figura 5.8: Cargas resistivas

Estas cargas son las encargadas de simular un sistema estable y balanceado, Con el motor de la figura 5.9, se produce un desbalance en el sistema el cual veremos a continuación con el prototipo y contrastado con el Analizador de redes.

Sincrofasores y monitoreo de energía eléctrica trifásica



Figura 5.9: Carga inductiva

La siguiente es una captura de este analizador, sin conectar a la red:



Figura 5.10: Captura del analizador de redes ME440.

Para la misma condición, se incorpora una captura del prototipo desarrollado:

Sincrofasores y monitoreo de energía eléctrica trifásica

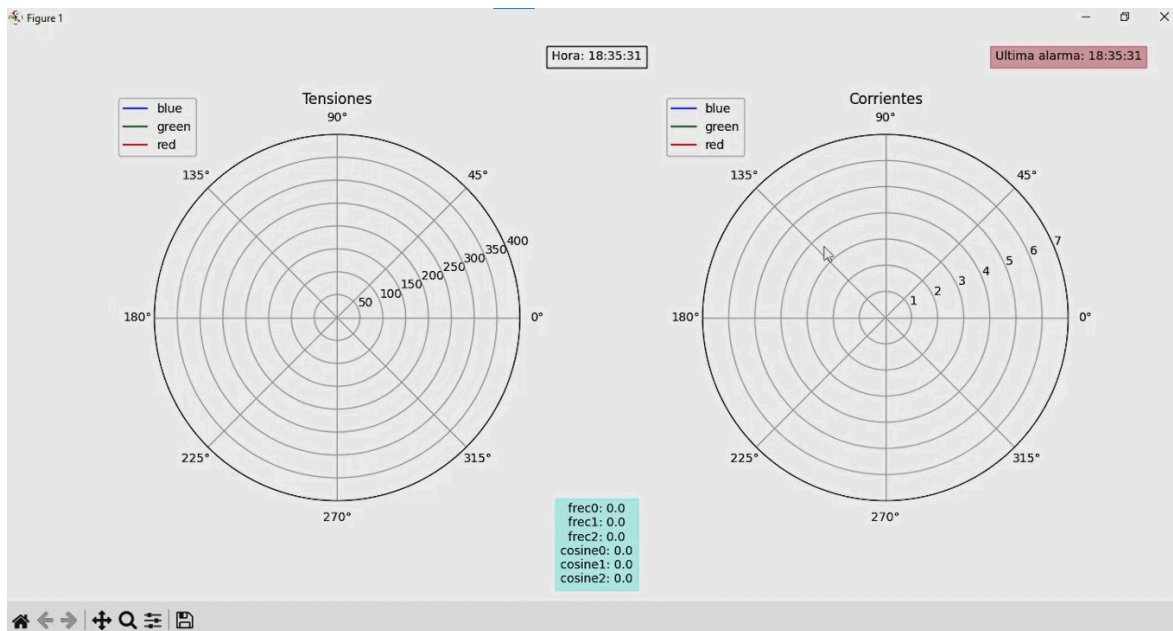


Figura 5.11: Captura prototipo PDC desarrollado

En la imagen se puede observar cómo son captadas las ondas de tensión y corriente que luego mediante los algoritmos antes descritos en el capítulo 2, se procesan y se convierten al dominio fasorial con su estampa de tiempo para que las mediciones sean sincrónicas.

Primero, se adjuntan capturas con cargas resistivas puras:

Sincrofasores y monitoreo de energía eléctrica trifásica



Figura 5.12: Captura del analizador de redes

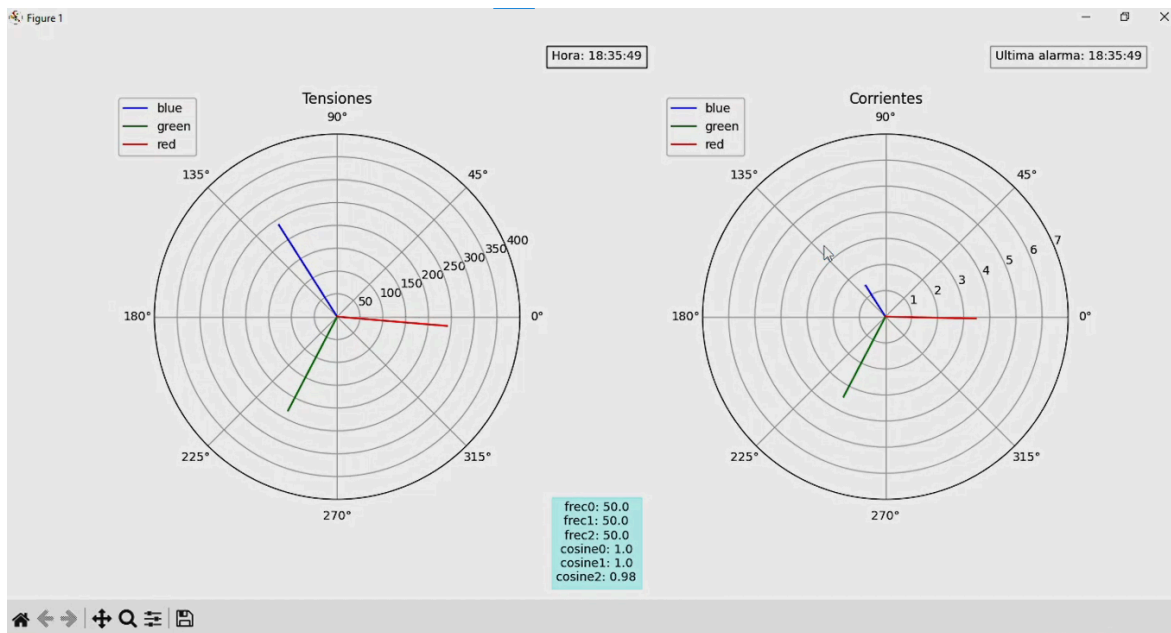


Figura 5.13: Captura del prototipo

Luego, se hicieron ensayos con un motor eléctrico monofásico.



Figura 5.14: Captura del analizador de redes, para contrastar mediciones.

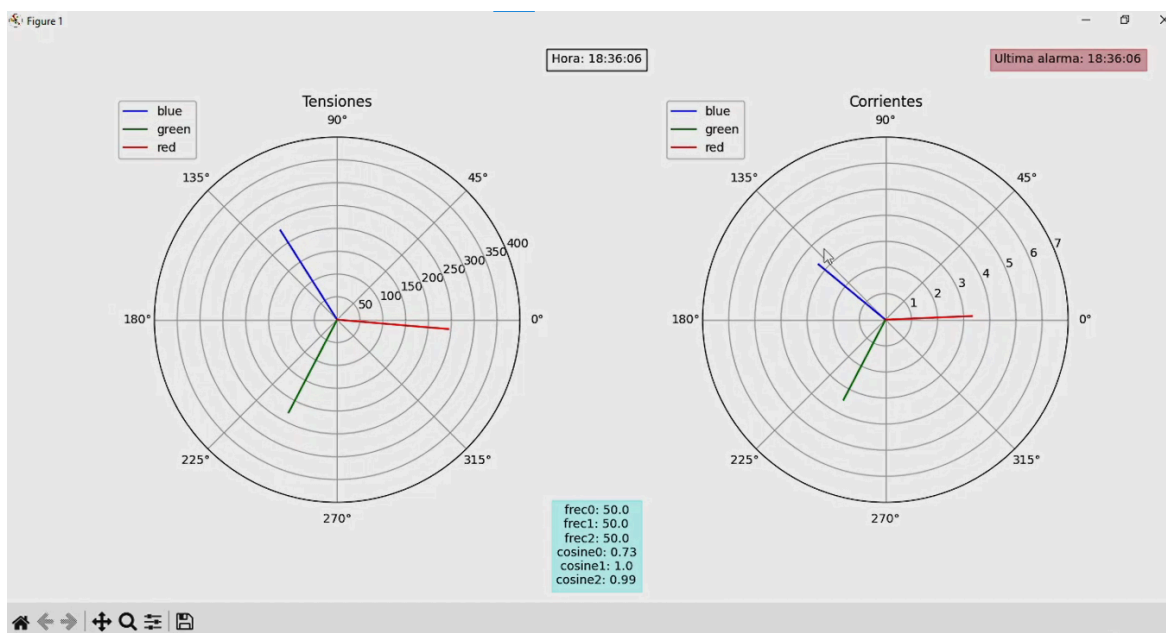


Figura 5.15: Ensayo con motor monofásico

Con este ensayo se puede observar como el prototipo responde a las alteraciones que sufren las líneas de la red trifásica, manteniendo cada una de las fases independientemente del resto de la red, y actualizando las visualizaciones con una ventana menor a 20 segundos, como contempla la norma para equipos como este.

6. Conclusiones

El monitoreo de ondas de energía trifásica es una aplicación crítica en muchas industrias y sistemas de energía eléctrica. Los ADC en la placa STM32F407VG proporcionan la capacidad necesaria para adquirir y procesar señales de tensión y corriente con precisión en sistemas trifásicos. La configuración adecuada de los ADC, la aplicación de filtros y la gestión de interrupciones son fundamentales para garantizar mediciones precisas y la seguridad del sistema.

Con la STM32F407VG y sus ADC, es posible monitorear y controlar sistemas de energía trifásica de manera eficiente y confiable, lo que contribuye a la eficiencia energética y la seguridad de las operaciones en una amplia variedad de aplicaciones industriales y de energía eléctrica.

La ESP32 NodeMCU es una placa de desarrollo versátil y potente que combina la capacidad de procesamiento de datos y la transmisión de datos por Wi-Fi en un solo paquete. Su microcontrolador ESP32 de doble núcleo, junto con su conectividad Wi-Fi y Bluetooth, la convierte en una opción de elección para una amplia gama de aplicaciones en el mundo de la electrónica. Con ejemplos prácticos que abarcan desde hogares inteligentes hasta aplicaciones industriales como es en este caso, y sensores inalámbricos, la ESP32 NodeMCU es una herramienta esencial para crear proyectos conectados a Internet de manera eficiente y efectiva.

La combinación de capacidad de procesamiento y conectividad inalámbrica hace que la ESP32 NodeMCU sea una plataforma altamente valorada en el mundo de la electrónica. A medida que continúa evolucionando, es probable que veamos aún más aplicaciones que aprovechen al máximo su potencial.

Al seguir cuidadosamente los pasos de la guía, se pudo configurar fácilmente tres generadores UNI-T UTG2025A para simular una onda trifásica con un desfase de 120 grados entre cada fase. La simulación precisa y confiable de ondas trifásicas resulta crucial en la aplicación de simulación trifásica para el instrumento diseñado. En el campo de la ingeniería eléctrica y la electrónica de potencia, contribuye al desarrollo y la implementación efectiva de sistemas de energía y equipos eléctricos avanzados y eficientes.

Con su funcionamiento de alta sensibilidad, eficiencia energética y capacidad de integración versátil, el módulo receptor GPS U-BLOX NEO-6M sigue desempeñando un papel fundamental en el campo de la

Sincrofasores y monitoreo de energía eléctrica trifásica

geolocalización y la navegación, ofreciendo una solución confiable y precisa para una amplia gama de aplicaciones en diversas industrias y entornos. Su capacidad para proporcionar datos de ubicación precisos y en tiempo real continúa siendo esencial para el desarrollo de tecnologías de posicionamiento avanzadas y sistemas de seguimiento eficientes, contribuyendo al avance y la innovación en el campo de la telemática y la geolocalización.

El equipo analizador de redes ME440 de Meatrol ha demostrado ser una herramienta confiable y precisa para contrastar mediciones en el ámbito de las redes eléctricas. Su reputación se fundamenta en su capacidad para proporcionar mediciones precisas y comparables, lo que permite evaluar con exactitud el rendimiento y la eficiencia de las redes eléctricas.

Este dispositivo se destaca por su capacidad para medir una amplia gama de parámetros eléctricos con una precisión excepcional. Sus capacidades abarcan desde la medición de voltaje, corriente, frecuencia, potencia activa, reactiva y aparente hasta la identificación de armónicos y el análisis de la calidad de la energía. Esta versatilidad le otorga la capacidad de ofrecer mediciones detalladas y precisas en diversos entornos y condiciones de red.

Uno de los aspectos más notables del ME440 es su capacidad para contrastar mediciones. No solo proporciona datos puntuales de cada parámetro eléctrico, sino que también ofrece la posibilidad de comparar y contrastar estas mediciones en diferentes momentos, condiciones y puntos de la red. Esto permite identificar fluctuaciones, anomalías o variaciones que puedan afectar el rendimiento general de la red.

El equipo analizador de redes ME440 de Meatrol destaca por su capacidad para contrastar mediciones, ofreciendo datos precisos y comparables que permiten evaluar y optimizar el rendimiento de las redes eléctricas. El respaldo del fabricante mediante pruebas, estándares de calidad y soporte técnico refuerza la confiabilidad y veracidad de las mediciones obtenidas con este dispositivo.

Finalmente, concluimos el presente trabajo destacando la importancia que tiene la correcta aplicación de los algoritmos presentados con un muestreo de las señales implementado de forma óptima con el fin de obtener la mayor información de cada línea de la red trifásica y poder procesarla con el menor error posible para representar dicha información de manera legible y de fácil entendimiento para el usuario como es un fasor que, sincronizado en el tiempo, le permite a este tener un panorama específico del sistema evaluado.

Sincrofasores y monitoreo de energía eléctrica trifásica

Esta correcta implementación hace que el prototipo cumpla con el 80% de las normas necesarias para la certificación del prototipo, ya que los elementos de hardware utilizados y los procesos de fabricación realizados no responden en su totalidad a las normas de seguridad y confiabilidad del instrumento, como así se logró con las normas de funcionamiento y procesamiento de datos para el correcto monitoreo.

7. Apéndice A: Código implementado en STM32F407VG

```
C/C++
/* USER CODE BEGIN Header */
/*
   Autor: Terzaghi, Benjamín.
   Fecha: 29/12/2023
   Placa: STM32F407VG
   Proyecto: PMU
   Versión: V2023-12-29
*/
/* USER CODE END Header */
/* Includes -----*/
#include "main.h"
/* Private includes -----*/
/* USER CODE BEGIN Includes */
#include "string.h"
#include "stdio.h"
#include "stdlib.h"
/* USER CODE END Includes */
/* Private typedef -----*/
/* USER CODE BEGIN PTD */
#define N 50 // Cantidad de muestras
#define UPDATE_GPS 1001 // Intervalo entre tramas GPS
#define ADC_VOLTAJES_N 3 // Cantidad de muestras por canal
#define ADC_CORRIENTES_N 3 // Cantidad de muestras por canal
#define LED_VERDE GPIO_PIN_2
#define LED_NARANJA GPIO_PIN_6
#define LED_ROJO GPIO_PIN_0
#define LED_AMARILLO GPIO_PIN_4
```

```
//#define LED_VERDE          GPIO_PIN_14
//#define LED_NARANJA        GPIO_PIN_13
//#define LED_ROJO           GPIO_PIN_0
//#define LED_AMARILLO       GPIO_PIN_12
/* USER CODE END PTD */

/* Private define -----*/
/* USER CODE BEGIN PD */
/* USER CODE END PD */

/* Private macro -----*/
/* USER CODE BEGIN PM */
/* USER CODE END PM */

/* Private variables -----*/
ADC_HandleTypeDef hadc1;
ADC_HandleTypeDef hadc2;
DMA_HandleTypeDef hdma_adc1;
DMA_HandleTypeDef hdma_adc2;
TIM_HandleTypeDef htim2;
TIM_HandleTypeDef htim3;
TIM_HandleTypeDef htim4;
UART_HandleTypeDef huart1;
UART_HandleTypeDef huart3;
/* USER CODE BEGIN PV */
uint32_t interrupciones_DMA = 0;
uint8_t gps_funcionando = 0;
uint8_t continuar_medicion = 1;
uint8_t update_gps_flag = 0;
uint8_t enviar_medicion = 0;
char receivedData[60];
struct GpsTime {
    int hora;
```

```
    int min;
    int seg;
} gps_time;
typedef struct GpsTime GpsTime;
uint16_t ADC_DMA_Buffer_Voltajes[N * ADC_VOLTAJES_N] = { 0 };
uint16_t ADC_DMA_Buffer_Corrientes[N * ADC_CORRIENTES_N] = { 0 };
/* USER CODE END PV */
/* Private function prototypes -----*/
void SystemClock_Config(void);
static void MX_GPIO_Init(void);
static void MX_DMA_Init(void);
static void MX_ADC1_Init(void);
static void MX_TIM3_Init(void);
static void MX_ADC2_Init(void);
static void MX_TIM4_Init(void);
static void MX_USART1_UART_Init(void);
static void MX_USART3_UART_Init(void);
static void MX_TIM2_Init(void);
/* USER CODE BEGIN PFP */
/* USER CODE END PFP */
/* Private user code -----*/
/* USER CODE BEGIN 0 */
void sendDataToProcessor() {
    /*
    * Funcion que envia las tramas de datos a la ESP.
    *
    * El formato es el siguiente:
    * - Los valores se separan por comas.
    * - Tension y corriente se separan por punto y coma.
    * - Los valores medidos se separan de los del GPS por &.
    */
}
```

Sincrofasores y monitoreo de energía eléctrica trifásica

```
* - Al final se envia un | para indicar fin de la trama.
*/
uint8_t i;
char buffer[15] = {'\0'};
for (i = 0; i < ADC_VOLTAJES_N * N; i++) {
    sprintf(buffer, "%hu,", ADC_DMA_Buffer_Voltajes[i]);
    HAL_UART_Transmit(&huart3, buffer, strlen(buffer), 1000);
}
// Valores de tension
HAL_UART_Transmit(&huart3, "|", 1, 100);
for (i = 0; i < ADC_CORRIENTES_N * N; i++) {
    sprintf(buffer, "%hu,", ADC_DMA_Buffer_Corrientes[i]);
    HAL_UART_Transmit(&huart3, buffer, strlen(buffer), 1000);
}
// Valores de corriente
sprintf(buffer, "%d:%d:%d|", gps_time.hora, gps_time.min, gps_time.sec);
// GPS y Fin de la trama
HAL_UART_Transmit(&huart3, buffer, strlen(buffer), 1000);
}

GpsTime extraerTiempo(char cadenaNMEA[200], GpsTime ultimoTiempo) {
    /*
    * Funcion que extrae el tiempo de la trama del GPS.
    */
    GpsTime tiempo = ultimoTiempo;
    char *cadenas = strtok(cadenaNMEA, "$");
    while (cadenas != NULL) {
        char primer_termino[10];
        sprintf(primer_termino, "%c%c%c%c%c", cadenas[0], cadenas[1],
            cadenas[2], cadenas[3], cadenas[4]);
    }
}
```

```
        if (strcmp(primer_termino, "GPRMC") == 0 || strcmp(primer_termino,
"GPGGA") == 0) {

            // ACA VA LA HORA
            char cadena_tiempo[100];
            strcpy(cadena_tiempo, cadenas);

            // Lo paso a un arreglo porque si no strtok no funciona
            strtok(cadena_tiempo, ",");
            char *segundo_termino = strtok(NULL, ",");
            char hora[3] = { '\0' };
            char minuto[3] = { '\0' };
            char segundo[3] = { '\0' };
            sprintf(hora, "%c%c", segundo_termino[0], segundo_termino[1]);
            sprintf(minuto, "%c%c", segundo_termino[2], segundo_termino[3]);
            sprintf(segundo, "%c%c", segundo_termino[4], segundo_termino[5]);
            tiempo.hora = atoi(hora);
            tiempo.min = atoi(minuto);
            tiempo.seg = atoi(segundo);
            break;
        }
        cadenas = strtok(NULL, "$");
    }
    return tiempo;
}

void update_gps_callback() {
    /*
    * Funcion que recibe los datos del GPS.
    */
    static uint8_t intentos = 0;
    if (HAL_UART_Receive(&huart1, (uint8_t*) receivedData,
        sizeof(receivedData) - 1, 500) == HAL_OK) {
```

```
gps_time = extraerTiempo(receivedData, gps_time);
if (gps_time.hora == 0 && gps_time.min == 0 && gps_time.seg == 0) {
    gps_funcionando = 0;
    HAL_GPIO_WritePin(GPIOD, LED_ROJO, GPIO_PIN_RESET);
} else {
    gps_funcionando = 1;
    intentos = 0;
    HAL_GPIO_WritePin(GPIOD, LED_ROJO, GPIO_PIN_SET);
}
} else {
    intentos++;
    if (intentos > 5) {
        gps_funcionando = 0;
        gps_time.hora = 0;
        gps_time.min = 0;
        gps_time.seg = 0;
        HAL_GPIO_WritePin(GPIOD, LED_ROJO, GPIO_PIN_RESET);
    }
}
}

void SysTick_Callback(void) {
    /*
    * Callback del systick.
    */
    static uint32_t counter = 0;
    counter++;
    if (counter % UPDATE_GPS == 0) {
        // Actualizacion del GPS
        update_gps_flag = 1;
        counter = 0;
    }
}
```

```
}
if (continuar_medicion == 1) {
    HAL_ADC_Start_DMA(&hadc2, (uint32_t*) ADC_DMA_Buffer_Corrientes,
        ADC_CORRIENTES_N * N);
    HAL_ADC_Start_DMA(&hadc1, (uint32_t*) ADC_DMA_Buffer_Voltajes,
        ADC_VOLTAJES_N * N);
    continuar_medicion = 0;
}
}

void HAL_DMA_Callback(uint8_t adc_index) {
    /*
    * Callback del DMA.
    */
    static uint8_t adc1_ready, adc2_ready = 0;
    if (adc_index == 0) {
        adc1_ready = 1;
    } else if (adc_index == 1) {
        adc2_ready = 1;
    }
    if (adc1_ready && adc2_ready) {
        continuar_medicion = 0;
        adc1_ready = adc2_ready = 0;
    }
}

void TIM2_Callback() {
    /*
    * Callback del Timer 2. Dispara a una frecuencia de 0.5ms.
    */
    static uint8_t counter = 0;
    if (counter >= 200) {
```



```
        // Se actualiza cada 100ms.
        enviar_medicion = 1;
        counter = 0;
    }
    counter++;
}
/* USER CODE END 0 */
/**
 * @brief The application entry point.
 * @retval int
 */
int main(void)
{
    /* USER CODE BEGIN 1 */
    /* USER CODE END 1 */
    /* MCU Configuration-----*/
    /* Reset of all peripherals, Initializes the Flash interface and the Systick. */
    HAL_Init();
    /* USER CODE BEGIN Init */
    /* USER CODE END Init */
    /* Configure the system clock */
    SystemClock_Config();
    /* USER CODE BEGIN SysInit */
    /* USER CODE END SysInit */
    /* Initialize all configured peripherals */
    MX_GPIO_Init();
    MX_DMA_Init();
    MX_ADC1_Init();
    MX_TIM3_Init();
    MX_ADC2_Init();
```

```
MX_TIM4_Init();
MX_USART1_UART_Init();
MX_USART3_UART_Init();
MX_TIM2_Init();
/* USER CODE BEGIN 2 */
    HAL_ADC_Start_DMA(&hadc1, (uint32_t*) ADC_DMA_Buffer_Voltajes,
                      ADC_VOLTAJES_N * N);
    HAL_ADC_Start_DMA(&hadc2, (uint32_t*) ADC_DMA_Buffer_Corrientes,
                      ADC_CORRIENTES_N * N);
    HAL_TIM_Base_Start_IT(&htim2);
    HAL_TIM_Base_Start(&htim3);
    HAL_TIM_Base_Start_IT(&htim4);
    HAL_GPIO_WritePin(GPIOD, LED_NARANJA, GPIO_PIN_SET);
/* USER CODE END 2 */
/* Infinite loop */
/* USER CODE BEGIN WHILE */
    while (1) {
/* USER CODE END WHILE */
/* USER CODE BEGIN 3 */
        if (enviar_medicion) {
            HAL_GPIO_WritePin(GPIOD, LED_AMARILLO, GPIO_PIN_SET);
            sendDataToProcessor();
            HAL_GPIO_WritePin(GPIOD, LED_AMARILLO, GPIO_PIN_RESET);
            HAL_GPIO_WritePin(GPIOD, LED_VERDE, GPIO_PIN_RESET);
            HAL_ADC_Stop_DMA(&hadc1);
            HAL_ADC_Stop_DMA(&hadc2);
            HAL_Delay(10);
            HAL_ADC_Start_DMA(&hadc1, (uint32_t*) ADC_DMA_Buffer_Voltajes,
ADC_VOLTAJES_N * N);
```

```
        HAL_ADC_Start_DMA(&hadc2, (uint32_t*) ADC_DMA_Buffer_Corrientes,
ADC_CORRIENTES_N * N);
        HAL_GPIO_WritePin(GPIOD, LED_VERDE, GPIO_PIN_SET);
        // Reinicia el DMA
        continuar_medicion = 1;
        enviar_medicion = 0;
    }
    if (update_gps_flag) {
        update_gps_callback();
        update_gps_flag = 0;
    }
}
/* USER CODE END 3 */
}
/**
 * @brief System Clock Configuration
 * @retval None
 */
void SystemClock_Config(void)
{
    RCC_OscInitTypeDef RCC_OscInitStruct = {0};
    RCC_ClkInitTypeDef RCC_ClkInitStruct = {0};
    /** Configure the main internal regulator output voltage
     */
    __HAL_RCC_PWR_CLK_ENABLE();
    __HAL_PWR_VOLTAGESCALING_CONFIG(PWR_REGULATOR_VOLTAGE_SCALE1);
    /** Initializes the RCC Oscillators according to the specified parameters
     * in the RCC_OscInitTypeDef structure.
     */
    RCC_OscInitStruct.OscillatorType = RCC_OSCILLATORTYPE_HSI;
```

```
RCC_OscInitStruct.HSIState = RCC_HSI_ON;
RCC_OscInitStruct.HSICalibrationValue = RCC_HSICALIBRATION_DEFAULT;
RCC_OscInitStruct.PLL.PLLState = RCC_PLL_NONE;
if (HAL_RCC_OscConfig(&RCC_OscInitStruct) != HAL_OK)
{
    Error_Handler();
}
/** Initializes the CPU, AHB and APB buses clocks
 */
RCC_ClkInitStruct.ClockType = RCC_CLOCKTYPE_HCLK|RCC_CLOCKTYPE_SYSCLK
    |RCC_CLOCKTYPE_PCLK1|RCC_CLOCKTYPE_PCLK2;
RCC_ClkInitStruct.SYSCLKSource = RCC_SYSCLKSOURCE_HSI;
RCC_ClkInitStruct.AHBCLKDivider = RCC_SYSCLK_DIV1;
RCC_ClkInitStruct.APB1CLKDivider = RCC_HCLK_DIV4;
RCC_ClkInitStruct.APB2CLKDivider = RCC_HCLK_DIV2;
if (HAL_RCC_ClockConfig(&RCC_ClkInitStruct, FLASH_LATENCY_0) != HAL_OK)
{
    Error_Handler();
}
}
/**
 * @brief ADC1 Initialization Function
 * @param None
 * @retval None
 */
static void MX_ADC1_Init(void)
{
    /* USER CODE BEGIN ADC1_Init 0 */
    /* USER CODE END ADC1_Init 0 */
    ADC_ChannelConfTypeDef sConfig = {0};
```

```
/* USER CODE BEGIN ADC1_Init 1 */
/* USER CODE END ADC1_Init 1 */
/** Configure the global features of the ADC (Clock, Resolution, Data Alignment
and number of conversion)
*/
hadc1.Instance = ADC1;
hadc1.Init.ClockPrescaler = ADC_CLOCK_SYNC_PCLK_DIV8;
hadc1.Init.Resolution = ADC_RESOLUTION_12B;
hadc1.Init.ScanConvMode = ENABLE;
hadc1.Init.ContinuousConvMode = DISABLE;
hadc1.Init.DiscontinuousConvMode = DISABLE;
hadc1.Init.ExternalTrigConvEdge = ADC_EXTERNALTRIGCONVEDGE_RISING;
hadc1.Init.ExternalTrigConv = ADC_EXTERNALTRIGCONV_T3_TRGO;
hadc1.Init.DataAlign = ADC_DATAALIGN_RIGHT;
hadc1.Init.NbrOfConversion = 3;
hadc1.Init.DMAContinuousRequests = DISABLE;
hadc1.Init.EOCSelection = ADC_EOC_SEQ_CONV;
if (HAL_ADC_Init(&hadc1) != HAL_OK)
{
    Error_Handler();
}
/** Configure for the selected ADC regular channel its corresponding rank in the
sequencer and its sample time.
*/
sConfig.Channel = ADC_CHANNEL_0;
sConfig.Rank = 1;
sConfig.SamplingTime = ADC_SAMPLETIME_84CYCLES;
if (HAL_ADC_ConfigChannel(&hadc1, &sConfig) != HAL_OK)
{
    Error_Handler();
}
```

```
}
/** Configure for the selected ADC regular channel its corresponding rank in the
sequencer and its sample time.
*/
sConfig.Channel = ADC_CHANNEL_3;
sConfig.Rank = 2;
if (HAL_ADC_ConfigChannel(&hadc1, &sConfig) != HAL_OK)
{
    Error_Handler();
}
/** Configure for the selected ADC regular channel its corresponding rank in the
sequencer and its sample time.
*/
sConfig.Channel = ADC_CHANNEL_4;
sConfig.Rank = 3;
if (HAL_ADC_ConfigChannel(&hadc1, &sConfig) != HAL_OK)
{
    Error_Handler();
}
/* USER CODE BEGIN ADC1_Init 2 */
/* USER CODE END ADC1_Init 2 */
}
/**
 * @brief ADC2 Initialization Function
 * @param None
 * @retval None
 */
static void MX_ADC2_Init(void)
{
    /* USER CODE BEGIN ADC2_Init 0 */
```

```
/* USER CODE END ADC2_Init 0 */
ADC_ChannelConfTypeDef sConfig = {0};
/* USER CODE BEGIN ADC2_Init 1 */
/* USER CODE END ADC2_Init 1 */

/** Configure the global features of the ADC (Clock, Resolution, Data Alignment
and number of conversion)
*/
hadc2.Instance = ADC2;
hadc2.Init.ClockPrescaler = ADC_CLOCK_SYNC_PCLK_DIV8;
hadc2.Init.Resolution = ADC_RESOLUTION_12B;
hadc2.Init.ScanConvMode = ENABLE;
hadc2.Init.ContinuousConvMode = DISABLE;
hadc2.Init.DiscontinuousConvMode = DISABLE;
hadc2.Init.ExternalTrigConvEdge = ADC_EXTERNALTRIGCONVEDGE_RISING;
hadc2.Init.ExternalTrigConv = ADC_EXTERNALTRIGCONV_T3_TRGO;
hadc2.Init.DataAlign = ADC_DATAALIGN_RIGHT;
hadc2.Init.NbrOfConversion = 3;
hadc2.Init.DMAContinuousRequests = DISABLE;
hadc2.Init.EOCSelection = ADC_EOC_SEQ_CONV;
if (HAL_ADC_Init(&hadc2) != HAL_OK)
{
    Error_Handler();
}

/** Configure for the selected ADC regular channel its corresponding rank in the
sequencer and its sample time.
*/
sConfig.Channel = ADC_CHANNEL_5;
sConfig.Rank = 1;
sConfig.SamplingTime = ADC_SAMPLETIME_84CYCLES;
if (HAL_ADC_ConfigChannel(&hadc2, &sConfig) != HAL_OK)
```

```
{
    Error_Handler();
}
/** Configure for the selected ADC regular channel its corresponding rank in the
sequencer and its sample time.
*/
sConfig.Channel = ADC_CHANNEL_6;
sConfig.Rank = 2;
if (HAL_ADC_ConfigChannel(&hadc2, &sConfig) != HAL_OK)
{
    Error_Handler();
}
/** Configure for the selected ADC regular channel its corresponding rank in the
sequencer and its sample time.
*/
sConfig.Channel = ADC_CHANNEL_8;
sConfig.Rank = 3;
if (HAL_ADC_ConfigChannel(&hadc2, &sConfig) != HAL_OK)
{
    Error_Handler();
}
/* USER CODE BEGIN ADC2_Init 2 */
/* USER CODE END ADC2_Init 2 */
}
/**
 * @brief TIM2 Initialization Function
 * @param None
 * @retval None
 */
static void MX_TIM2_Init(void)
```



```
{
/* USER CODE BEGIN TIM2_Init 0 */
/* USER CODE END TIM2_Init 0 */
TIM_ClockConfigTypeDef sClockSourceConfig = {0};
TIM_MasterConfigTypeDef sMasterConfig = {0};
/* USER CODE BEGIN TIM2_Init 1 */
/* USER CODE END TIM2_Init 1 */
htim2.Instance = TIM2;
htim2.Init.Prescaler = 0;
htim2.Init.CounterMode = TIM_COUNTERMODE_UP;
htim2.Init.Period = 39999;
htim2.Init.ClockDivision = TIM_CLOCKDIVISION_DIV1;
htim2.Init.AutoReloadPreload = TIM_AUTORELOAD_PRELOAD_DISABLE;
if (HAL_TIM_Base_Init(&htim2) != HAL_OK)
{
    Error_Handler();
}
sClockSourceConfig.ClockSource = TIM_CLOCKSOURCE_INTERNAL;
if (HAL_TIM_ConfigClockSource(&htim2, &sClockSourceConfig) != HAL_OK)
{
    Error_Handler();
}
sMasterConfig.MasterOutputTrigger = TIM_TRGO_UPDATE;
sMasterConfig.MasterSlaveMode = TIM_MASTERSLAVEMODE_DISABLE;
if (HAL_TIMEx_MasterConfigSynchronization(&htim2, &sMasterConfig) != HAL_OK)
{
    Error_Handler();
}
/* USER CODE BEGIN TIM2_Init 2 */
/* USER CODE END TIM2_Init 2 */
}
```

```
}
/**
 * @brief TIM3 Initialization Function
 * @param None
 * @retval None
 */
static void MX_TIM3_Init(void)
{
    /* USER CODE BEGIN TIM3_Init 0 */
    /* USER CODE END TIM3_Init 0 */
    TIM_ClockConfigTypeDef sClockSourceConfig = {0};
    TIM_MasterConfigTypeDef sMasterConfig = {0};
    /* USER CODE BEGIN TIM3_Init 1 */
    /* USER CODE END TIM3_Init 1 */
    htim3.Instance = TIM3;
    htim3.Init.Prescaler = 319;
    htim3.Init.CounterMode = TIM_COUNTERMODE_UP;
    htim3.Init.Period = 9;
    htim3.Init.ClockDivision = TIM_CLOCKDIVISION_DIV1;
    htim3.Init.AutoReloadPreload = TIM_AUTORELOAD_PRELOAD_DISABLE;
    if (HAL_TIM_Base_Init(&htim3) != HAL_OK)
    {
        Error_Handler();
    }
    sClockSourceConfig.ClockSource = TIM_CLOCKSOURCE_INTERNAL;
    if (HAL_TIM_ConfigClockSource(&htim3, &sClockSourceConfig) != HAL_OK)
    {
        Error_Handler();
    }
    sMasterConfig.MasterOutputTrigger = TIM_TRGO_UPDATE;
```

```
sMasterConfig.MasterSlaveMode = TIM_MASTERSLAVEMODE_DISABLE;
if (HAL_TIMEx_MasterConfigSynchronization(&htim3, &sMasterConfig) != HAL_OK)
{
    Error_Handler();
}
/* USER CODE BEGIN TIM3_Init 2 */
/* USER CODE END TIM3_Init 2 */
}
/**
 * @brief TIM4 Initialization Function
 * @param None
 * @retval None
 */
static void MX_TIM4_Init(void)
{
    /* USER CODE BEGIN TIM4_Init 0 */
    /* USER CODE END TIM4_Init 0 */
    TIM_ClockConfigTypeDef sClockSourceConfig = {0};
    TIM_MasterConfigTypeDef sMasterConfig = {0};
    /* USER CODE BEGIN TIM4_Init 1 */
    /* USER CODE END TIM4_Init 1 */
    htim4.Instance = TIM4;
    htim4.Init.Prescaler = 0;
    htim4.Init.CounterMode = TIM_COUNTERMODE_UP;
    htim4.Init.Period = 41999;
    htim4.Init.ClockDivision = TIM_CLOCKDIVISION_DIV1;
    htim4.Init.AutoReloadPreload = TIM_AUTORELOAD_PRELOAD_DISABLE;
    if (HAL_TIM_Base_Init(&htim4) != HAL_OK)
    {
        Error_Handler();
    }
}
```

```
}
sClockSourceConfig.ClockSource = TIM_CLOCKSOURCE_INTERNAL;
if (HAL_TIM_ConfigClockSource(&htim4, &sClockSourceConfig) != HAL_OK)
{
    Error_Handler();
}
sMasterConfig.MasterOutputTrigger = TIM_TRGO_UPDATE;
sMasterConfig.MasterSlaveMode = TIM_MASTERSLAVEMODE_DISABLE;
if (HAL_TIMEx_MasterConfigSynchronization(&htim4, &sMasterConfig) != HAL_OK)
{
    Error_Handler();
}
/* USER CODE BEGIN TIM4_Init 2 */
/* USER CODE END TIM4_Init 2 */
}
/**
 * @brief USART1 Initialization Function
 * @param None
 * @retval None
 */
static void MX_USART1_UART_Init(void)
{
    /* USER CODE BEGIN USART1_Init 0 */
    /* USER CODE END USART1_Init 0 */
    /* USER CODE BEGIN USART1_Init 1 */
    /* USER CODE END USART1_Init 1 */
    huart1.Instance = USART1;
    huart1.Init.BaudRate = 9600;
    huart1.Init.WordLength = UART_WORDLENGTH_8B;
    huart1.Init.StopBits = UART_STOPBITS_1;
```

```
huart1.Init.Parity = UART_PARITY_NONE;
huart1.Init.Mode = UART_MODE_TX_RX;
huart1.Init.HwFlowCtl = UART_HWCONTROL_NONE;
huart1.Init.OverSampling = UART_OVERSAMPLING_16;
if (HAL_UART_Init(&huart1) != HAL_OK)
{
    Error_Handler();
}
/* USER CODE BEGIN USART1_Init 2 */
/* USER CODE END USART1_Init 2 */
}
/**
 * @brief USART3 Initialization Function
 * @param None
 * @retval None
 */
static void MX_USART3_UART_Init(void)
{
    /* USER CODE BEGIN USART3_Init 0 */
    /* USER CODE END USART3_Init 0 */
    /* USER CODE BEGIN USART3_Init 1 */
    /* USER CODE END USART3_Init 1 */
    huart3.Instance = USART3;
    huart3.Init.BaudRate = 115200;
    huart3.Init.WordLength = UART_WORDLENGTH_8B;
    huart3.Init.StopBits = UART_STOPBITS_1;
    huart3.Init.Parity = UART_PARITY_NONE;
    huart3.Init.Mode = UART_MODE_TX;
    huart3.Init.HwFlowCtl = UART_HWCONTROL_NONE;
    huart3.Init.OverSampling = UART_OVERSAMPLING_16;
```

```
if (HAL_UART_Init(&huart3) != HAL_OK)
{
    Error_Handler();
}
/* USER CODE BEGIN USART3_Init 2 */
/* USER CODE END USART3_Init 2 */
}
/**
 * Enable DMA controller clock
 */
static void MX_DMA_Init(void)
{
    /* DMA controller clock enable */
    __HAL_RCC_DMA2_CLK_ENABLE();
    /* DMA interrupt init */
    /* DMA2_Stream0_IRQn interrupt configuration */
    HAL_NVIC_SetPriority(DMA2_Stream0_IRQn, 0, 0);
    HAL_NVIC_EnableIRQ(DMA2_Stream0_IRQn);
    /* DMA2_Stream2_IRQn interrupt configuration */
    HAL_NVIC_SetPriority(DMA2_Stream2_IRQn, 0, 0);
    HAL_NVIC_EnableIRQ(DMA2_Stream2_IRQn);
}
/**
 * @brief GPIO Initialization Function
 * @param None
 * @retval None
 */
static void MX_GPIO_Init(void)
{
    GPIO_InitTypeDef GPIO_InitStruct = {0};
```

```
/* USER CODE BEGIN MX_GPIO_Init_1 */
/* USER CODE END MX_GPIO_Init_1 */

/* GPIO Ports Clock Enable */
__HAL_RCC_GPIOH_CLK_ENABLE();
__HAL_RCC_GPIOA_CLK_ENABLE();
__HAL_RCC_GPIOB_CLK_ENABLE();
__HAL_RCC_GPIOD_CLK_ENABLE();
__HAL_RCC_GPIOC_CLK_ENABLE();

/*Configure GPIO pin Output Level */
HAL_GPIO_WritePin(GPIOD, GPIO_PIN_0|GPIO_PIN_2|GPIO_PIN_4|GPIO_PIN_6,
GPIO_PIN_RESET);

/*Configure GPIO pin : PD15 */
GPIO_InitStruct.Pin = GPIO_PIN_15;
GPIO_InitStruct.Mode = GPIO_MODE_IT_RISING;
GPIO_InitStruct.Pull = GPIO_PULLUP;
HAL_GPIO_Init(GPIOD, &GPIO_InitStruct);

/*Configure GPIO pin : PC9 */
GPIO_InitStruct.Pin = GPIO_PIN_9;
GPIO_InitStruct.Mode = GPIO_MODE_AF_OD;
GPIO_InitStruct.Pull = GPIO_NOPULL;
GPIO_InitStruct.Speed = GPIO_SPEED_FREQ_VERY_HIGH;
GPIO_InitStruct.Alternate = GPIO_AF4_I2C3;
HAL_GPIO_Init(GPIOC, &GPIO_InitStruct);

/*Configure GPIO pin : PA8 */
GPIO_InitStruct.Pin = GPIO_PIN_8;
GPIO_InitStruct.Mode = GPIO_MODE_AF_OD;
GPIO_InitStruct.Pull = GPIO_NOPULL;
GPIO_InitStruct.Speed = GPIO_SPEED_FREQ_VERY_HIGH;
GPIO_InitStruct.Alternate = GPIO_AF4_I2C3;
HAL_GPIO_Init(GPIOA, &GPIO_InitStruct);
```

```
/*Configure GPIO pins : PD0 PD2 PD4 PD6 */
GPIO_InitStruct.Pin = GPIO_PIN_0|GPIO_PIN_2|GPIO_PIN_4|GPIO_PIN_6;
GPIO_InitStruct.Mode = GPIO_MODE_OUTPUT_PP;
GPIO_InitStruct.Pull = GPIO_NOPULL;
GPIO_InitStruct.Speed = GPIO_SPEED_FREQ_LOW;
HAL_GPIO_Init(GPIOD, &GPIO_InitStruct);
/* EXTI interrupt init*/
HAL_NVIC_SetPriority(EXTI15_10_IRQn, 0, 0);
HAL_NVIC_EnableIRQ(EXTI15_10_IRQn);
/* USER CODE BEGIN MX_GPIO_Init_2 */
/* USER CODE END MX_GPIO_Init_2 */
}
/* USER CODE BEGIN 4 */
/* USER CODE END 4 */
/**
 * @brief Period elapsed callback in non blocking mode
 * @note This function is called when TIM6 interrupt took place, inside
 * HAL_TIM_IRQHandler(). It makes a direct call to HAL_IncTick() to increment
 * a global variable "uwTick" used as application time base.
 * @param htim : TIM handle
 * @retval None
 */
void HAL_TIM_PeriodElapsedCallback(TIM_HandleTypeDef *htim)
{
/* USER CODE BEGIN Callback 0 */
/* USER CODE END Callback 0 */
if (htim->Instance == TIM6) {
HAL_IncTick();
}
/* USER CODE BEGIN Callback 1 */
```



```
/* USER CODE END Callback 1 */
}
/**
 * @brief This function is executed in case of error occurrence.
 * @retval None
 */
void Error_Handler(void)
{
/* USER CODE BEGIN Error_Handler_Debug */
    /* User can add his own implementation to report the HAL error return state */
    __disable_irq();
    while (1) {
    }
/* USER CODE END Error_Handler_Debug */
}

#ifdef USE_FULL_ASSERT
/**
 * @brief Reports the name of the source file and the source line number
 * where the assert_param error has occurred.
 * @param file: pointer to the source file name
 * @param line: assert_param error line source number
 * @retval None
 */
void assert_failed(uint8_t *file, uint32_t line)
{
/* USER CODE BEGIN 6 */
/* User can add his own implementation to report the file name and line number,
ex: printf("Wrong parameters value: file %s on line %d\r\n", file, line) */
/* USER CODE END 6 */
}
```

```
#endif /* USE_FULL_ASSERT */
```

8. Apéndice B: Código implementado en ESP32-NodeMCU

```
C/C++
/**
 * Autor: Terzaghi, Benjamín.
 * Fecha: 29/12/2023
 * Placa: ESP32
 * Proyecto: PMU
 * Versión: V2023-12-29
 */

#include <LiquidCrystal_I2C.h>
#include "BluetoothSerial.h"
#include <HardwareSerial.h>
#include "AsyncUDP.h"
#include "WiFi.h"
#include <Wire.h>
#include "math.h"

AsyncUDP udp;
LiquidCrystal_I2C lcd(0x27, 16, 2);
BluetoothSerial SerialBT;

// Librerías

#define DEBUG 0
// En el modo DEBUG, se utiliza la red y contraseña fija en el llamado
```

```
// a la funcion (DEBUG = 1). En el otro modo (DEBUG = 0), se debe conectar
// utilizando la aplicacion dedicada.

#define PI 3.14159264

#define N 50 // Numero de muestras
#define N_VOLTAJES 3 // Cantidad de canales de tension
#define N_CORRIENTES 3 // Cantidad de canales de corriente
#define MIN_ADC 0 // Valor minimo de digitalizacion del ADC
#define MAX_ADC 4096 // Valor maximo de digitalizacion del ADC
#define MIN_TRAFO_V -311 // Valor minimo medible por el trafo de tension
#define MAX_TRAFO_V 311 // Valor maximo medible por el trafo de tension
#define MAX_TRAFO_I 5 // Valor maximo medible por el trafo de corriente
#define MIN_TRAFO_I -5 // Valor minimo medible por el trafo de corriente
#define N_COS_FI 5 // Cantidad de mediciones instantaneas de coseno de fi promediados

#define PERIODO 0.02 // 20ms - Periodo de la señal
#define TIEMPO_ENTRE_MUESTRAS 0.0004 // 400us entre muestras

#define IP0 192
#define IP1 168
#define IP2 1
#define IP3 2
// Componentes de la IP del instrumento
#define PORTD 8
// Puerto destino

#define VENTANA 5
// Tamaño ventana de filtrado

#define UART_PORT 2
// Puerto serie de la placa utilizado

// Defines

HardwareSerial SerialPort(UART_PORT);
```

```
uint8_t i;

uint16_t voltajes[N_VOLTAJES][N] = { 0 };
uint16_t corrientes[N_CORRIENTES][N] = { 0 };
// Matrices que almacenan valores ordenados de tension y corriente
uint16_t voltajes_filtrados[N_VOLTAJES][N] = { {0}, {0}, {0} };
uint16_t corrientes_filtrados[N_CORRIENTES][N] = { {0}, {0}, {0} };
// Matrices que almacenan valores filtrados
double frecs_v[N_VOLTAJES] = { 0 };
// Arreglo de frecuencias
double magnitudes_v[N_VOLTAJES] = { 0 };
double magnitudes_i[N_CORRIENTES] = { 0 };
// Arreglos de magnitudes de tension y corriente
double fases_v[N_VOLTAJES] = { {0}, {0}, {0} };
double fases_i[N_CORRIENTES][2] = { {0}, {0}, {0} };
// Arreglos de fase de tension y corriente
double cos_fi[N_VOLTAJES] = { 0 };
double cur_cosfi[N_VOLTAJES] = { 0 };
double prev_cos_fi[N_VOLTAJES] = {0}; // Almacena el ultimo cos fi
double index_cos_fi = 0; // Recuerda la cantidad de promedios de cos fi
// Variables de calculo de cos fi
char tiempo[10];
// Variable para recibir y almacenar el tiempo obtenido del GPS en la STM32F4
char send_v_string[100] = "";
char send_i_string[100] = "";
// Variables de envio de datos
const char *pin = "1234"; // Change this to more secure PIN.
const String device_name = "PMU-BT";
// Parametros de la conexion BT

String password, ssid;
// Variables para almacenar los parametros recibidos por la aplicacion
// para la conexion WiFi

// Variables
```

```
uint8_t comparar_igualdad(double value1, double value2, double tol) {
    /*
     * Compara dos valores con una cierta tolerancia
     */
    if (abs(value1 - value2) > tol) return 0;
    return 1;
}

void normalizar_x(uint16_t *x, double *x_n) {
    /*
     * Normaliza todos los valores de un arreglo entre 0 y 1.
     */
    uint8_t i;
    for (i = 0; i < N; i++) {
        x_n[i] = (x[i] - 2048) / 2048.0;
    }
}

void convertir_x(uint16_t *x, double *x_n, double min_value, double max_value) {
    /*
     * Convierte todos los valores de un arreglo entre un cierto rango.
     */
    uint8_t i;
    for (i = 0; i < N; i++) {
        x_n[i] = convertir_rango(x[i], MIN_ADC, MAX_ADC, min_value, max_value);
    }
}

void detectar_cruces(double *x, uint8_t *cruce1, uint8_t *cruce2) {
    /*
     * Detecta los cruces por cero. Si detecta un solo cruce, supone
     * que el primero es el del índice 0.
     */
    uint8_t i;
    *cruce1 = *cruce2 = 0;
    for (i = 1; i < N; i++) {
```

```
if (x[i - 1] <= 0 && x[i] > 0) {
    // Flanco ascendente
    if (abs(x[i]) < abs(x[i - 1]))
        *cruce1 = i;
    else
        *cruce1 = i - 1;
} else if (x[i - 1] > 0 && x[i] <= 0) {
    // Flanco descendente
    if (abs(x[i]) < abs(x[i - 1]))
        *cruce2 = i;
    else
        *cruce2 = i - 1;
}
}

double detectar_frecuencia(double *x, uint8_t *cruce1, uint8_t *cruce2) {
    /*
    Calcula la frecuencia para un arreglo de muestras.
    */
    double frecuencia = 0;
    uint8_t i = 0;

    detectar_cruces(x, cruce1, cruce2);

    if (*cruce1 - *cruce2 != 0) {
        // Para evitar que de infinito
        double diferencia = abs(*cruce1 - *cruce2) * TIEMPO_ENTRE_MUESTRAS;
        frecuencia = 1.0 / (diferencia * 2);
        // Se multiplica por dos porque se calcula en un semiperiodo
    }

    return (round(frecuencia / 10.0) * 10.0);
    // Se redondea para disminuir la imprecision
}
```

```
double obtener_magnitud(double *x) {
    /*
    Obtiene la magnitud de una señal representada por un arreglo de muestras.
    */
    uint8_t i;
    uint32_t suma_valores = 0;
    for (i = 0; i < N; i++) {
        suma_valores += x[i] * x[i];
    }

    return sqrt(suma_valores / N);
}

double obtener_angulo_entre_fases(uint8_t cruce1_x1, uint8_t cruce1_x2) {
    /*
    Obtiene la diferencia angular entre dos señales representadas por
    un arreglo de muestras.
    */
    int8_t dif_cruces = cruce1_x1 - cruce1_x2;
    double fase = (abs(dif_cruces) * TIEMPO_ENTRE_MUESTRAS) / (PERIODO / 2.0);
    // La fase se calcula en un semiperiodo
    fase = fase * 360;
    // Convierto la fase en grados
    if (dif_cruces < 0) {
        fase = 360 - fase;
    }
    // La ubico en el cuadrante correcto

    return fase;
}

double obtener_cos_fi(uint8_t cruceV, uint8_t cruceI) {
    /*
    Obtiene el valor del coseno de fi para los arreglos recibidos.
    */
    int8_t dif_cruces = (cruceV - cruceI) + 4;
    double fi = (dif_cruces * TIEMPO_ENTRE_MUESTRAS) / PERIODO;
}
```

```
fi = fi * 360;

return cos((fi * PI) / 180.0);
}

void obtener_fase(uint8_t cruce1_x0, uint8_t cruce1_x1, uint8_t cruce1_x2, uint8_t index,
double *fases) {
    double fase, fase1, fase2;
    if (index == 0) {
        fase1 = obtener_angulo_entre_fases(cruce1_x0, cruce1_x1);
        fase2 = obtener_angulo_entre_fases(cruce1_x2, cruce1_x0);
    } else if (index == 1) {
        fase1 = obtener_angulo_entre_fases(cruce1_x0, cruce1_x1);
        fase2 = obtener_angulo_entre_fases(cruce1_x1, cruce1_x2);
    } else if (index == 2) {
        fase1 = obtener_angulo_entre_fases(cruce1_x2, cruce1_x0);
        fase2 = obtener_angulo_entre_fases(cruce1_x1, cruce1_x2);
    } else {
        fase1 = 0;
        fase2 = 0;
    }

    // Calculo las distintas fases redundantemente para elegir aquella
    // que este mas cerca de 120. Esto se hace porque cuando una fase se ve
    // alterada y se cae, se puede utilizar la otra para calcular la fase actual
    if (comparar_igualdad(abs(120 - fase1), abs(120 - fase2), 2)) {
        fases[0] = fase1;
        fases[1] = fase2;
    }
    else if (abs(120 - fase1) > abs(120 - fase2)) {
        fases[0] = fase2;
        fases[1] = fase1;
    }
    else {
        fases[0] = fase1;
        fases[1] = fase2;
    }
}
```



```
}
}

double convertir_rango(uint16_t entrada, uint16_t rangoEntradaMin,
                      uint16_t rangoEntradaMax, double rangoSalidaMin, double
rangoSalidaMax) {
    /*
    Convierte un valor entre un rango a otro valor dentro de otro rango.
    */
    if (entrada < rangoEntradaMin) {
        entrada = rangoEntradaMin;
    } else if (entrada > rangoEntradaMax) {
        entrada = rangoEntradaMax;
    }

    double rangoEntrada = rangoEntradaMax - rangoEntradaMin;
    double rangoSalida = rangoSalidaMax - rangoSalidaMin;
    double salida = ((entrada - rangoEntradaMin) / rangoEntrada) * rangoSalida
        + rangoSalidaMin;

    return salida;
}

uint8_t process_values(uint16_t voltajes[N_VOLTAJES][N], uint16_t
corrientes[N_CORRIENTES][N]) {
    /*
    Procesa los valores de tension y corriente, para obtener frecuencia, magnitud, fase, y
    cos fi.
    */
    double x[N];
    // Arreglo utilizado para almacenar valores auxiliares
    uint8_t cruces1_v[N_VOLTAJES], cruces1_i[N_CORRIENTES];
    uint8_t cruces2_v[N_VOLTAJES], cruces2_i[N_CORRIENTES];
    // Arreglos que almacenan cruces de tension y corriente
    uint8_t bool_status_v[N_VOLTAJES] = { 1, 1, 1 };
    uint8_t bool_status_i[N_CORRIENTES] = { 1, 1, 1 };
```

```
// Arreglo que identifica fases correctas

for (i = 0; i < N_VOLTAJES; i++) {
    normalizar_x(voltajes[i], x);
    frecs_v[i] = detectar_frecuencia(x, &cruces1_v[i], &cruces2_v[i]);
    if (frecs_v[i] <= 0) {
        bool_status_v[i] = 0;
        frecs_v[i] = 0;
    }
}

// Calculo de frecuencias (solo en tension)

for (i = 0; i < N_CORRIENTES; i++) {
    normalizar_x(corrientes[i], x);
    detectar_cruces(x, &cruces1_i[i], &cruces2_i[i]);
    if ((cruces2_i - cruces1_i) == 0) {
        bool_status_i[i] = 0;
    }
}

// Obtencion de cruces para las fases de corriente

for (i = 0; i < N_VOLTAJES; i++) {
    if (bool_status_v[i]) {
        convertir_x(voltajes[i], x, MIN_TRAFO_V, MAX_TRAFO_V);
        magnitudes_v[i] = obtener_magnitud(x);
    } else {
        magnitudes_v[i] = 0;
    }
}

// Calculo de magnitudes de tension

for (i = 0; i < N_CORRIENTES; i++) {
    if (bool_status_i[i]) {
        convertir_x(corrientes[i], x, MIN_TRAFO_I, MAX_TRAFO_I);
        magnitudes_i[i] = obtener_magnitud(x);
    } else {
```

```
    magnitudes_i[i] = 0;
}
}
// Calculo de magnitudes de corriente

fases_v[0] = obtener_angulo_entre_fases(cruces1_v[0], cruces1_v[1]);
fases_v[1] = obtener_angulo_entre_fases(cruces1_v[1], cruces1_v[2]);
fases_v[2] = obtener_angulo_entre_fases(cruces1_v[2], cruces1_v[0]);
// Calculo de fases para la tension (sin redundancia)

obtener_fase(cruces1_i[0], cruces1_i[1], cruces1_i[2], 0, fases_i[0]);
obtener_fase(cruces1_i[0], cruces1_i[1], cruces1_i[2], 1, fases_i[1]);
obtener_fase(cruces1_i[0], cruces1_i[1], cruces1_i[2], 2, fases_i[2]);
// Calculo de fases para la corriente (con redundancia)

for(i = 0; i < N_VOLTAJES; i++) {
    if (!bool_status_v[i] || !bool_status_i[i])
        cur_cosfi[i] += 0;
    else {
        cur_cosfi[i] += obtener_cos_fi(cruces1_v[i], cruces1_i[i]);
    }
}
// Calculo del cos fi

if (cur_cosfi[0] == 0 && cur_cosfi[1] == 0 && cur_cosfi[2] == 0) ;
else {
    index_cos_fi++;

    if (index_cos_fi == N_COS_FI) {
        cos_fi[0] = cur_cosfi[0] / (double) N_COS_FI;
        cos_fi[1] = cur_cosfi[1] / (double) N_COS_FI;
        cos_fi[2] = cur_cosfi[2] / (double) N_COS_FI;

        index_cos_fi = 0;
        cur_cosfi[0] = 0;
        cur_cosfi[1] = 0;
    }
}
```

Sincrofasores y monitoreo de energía eléctrica trifásica

```
    cur_cosfi[2] = 0;
}
}
// Aplicacion de promedios en el calculo del cos fi

return 1;
}

void prepare_values() {
    /*
    Genera las cadenas con los datos para enviar por UDP.
    */

    sprintf(send_v_string,
"%s&%.21f,%.21f,%.21f,%.21f,%.21f,%.21f,%.21f,%.21f,%.21f,%.21f,%.21f", tiempo,
    frecs_v[0],
    magnitudes_v[0],
    fases_v[0],
    frecs_v[1],
    magnitudes_v[1],
    fases_v[1],
    frecs_v[2],
    magnitudes_v[2],
    fases_v[2],
    cos_fi[0],
    cos_fi[1],
    cos_fi[2]);

    sprintf(send_i_string,
"%s-%.21f,%.21f,%.21f,%.21f,%.21f,%.21f,%.21f,%.21f,%.21f,%.21f,%.21f", tiempo,
    frecs_v[0],
    magnitudes_i[0],
    fases_i[0][0],
    frecs_v[1],
    magnitudes_i[1],
    fases_i[1][0],
    frecs_v[2],
```

```
        magnitudes_i[2],
        fases_i[2][0],
        cos_fi[0],
        cos_fi[1],
        cos_fi[2]);
}

void generar_matrices(String values_string, uint16_t matrix[N_VOLTAJES][N]) {
    /*
    Recibe las cadenas de valores, y los convierte a matrices de valores ordenados.
    */
    uint8_t i, j_m, j;
    uint16_t last_comma_got = 0;

    uint16_t num_arrays[N_VOLTAJES * N] = { 0 };
    for (i = 0; i < N_VOLTAJES * N; i++) {
        String last_num_string = values_string.substring(last_comma_got,
values_string.indexOf(',', last_comma_got + 1));
        last_comma_got = values_string.indexOf(',', last_comma_got + 1) + 1;

        num_arrays[i] = last_num_string.toInt();
    }
    for (i = 0; i < N_VOLTAJES; i++) {
        for (j_m = i, j = 0; j_m < N_VOLTAJES * N; j_m = j_m + N_VOLTAJES, j++) {
            matrix[i][j] = num_arrays[j_m];
        }
        // i_m salta de N_VOLTAJES en N_VOLTAJES, de forma analoga a i, que salta de 1 en 1
    }
}

void filtrar_datos(uint16_t values[N], int ventana, uint16_t valores_filtrados[]) {
    /*
    Filtra los datos en un arreglo, haciendolos pasar por una ventana.
    */
    for (int i = 0; i < N; i++) {
        int suma = 0;
    }
}
```

```
int inicio = i - ventana / 2 < 0 ? 0 : i - ventana / 2;
int fin = i + ventana / 2 >= N ? N - 1 : i + ventana / 2;

for (int j = inicio; j <= fin; j++) {
    suma += valores[j];
}
valores_filtrados[i] = suma / (fin - inicio + 1);
}
}

void recibir_procesar() {
    /*
    Recibe las cadenas de valores, y los divide:
    1. Valores de tension.
    2. Valores de corriente.
    3. Hora actual.
    Procesa los datos, y los envia.
    */
    String received_string = SerialPort.readStringUntil('|');
    String voltages_string = received_string.substring(0, received_string.indexOf(';'));
    String currents_string = received_string.substring(received_string.indexOf(';') + 1,
received_string.indexOf('&', received_string.indexOf(';')));
    String time_string = received_string.substring(received_string.indexOf("&") + 1,
received_string.length());

    if (time_string.length() <= 11) {
        generar_matrices(voltages_string, voltajes);
        generar_matrices(currents_string, corrientes);
        tiempo.toCharArray(tiempo, time_string.length() + 1);

        lcd.setCursor(4, 1);
        lcd.print("          ");
        if (strcmp(tiempo, "0:0:0") == 0) {
            lcd.setCursor(4, 1);
            lcd.print("Sin datos");
        } else {
```

```
    lcd.setCursor(4, 1);
    lcd.print(tiempo);
}

filtrar_datos(voltajes[0], VENTANA, voltajes_filtrados[0]);
filtrar_datos(voltajes[1], VENTANA, voltajes_filtrados[1]);
filtrar_datos(voltajes[2], VENTANA, voltajes_filtrados[2]);
// Filtra valores de tension

filtrar_datos(corrientes[0], VENTANA, corrientes_filtrados[0]);
filtrar_datos(corrientes[1], VENTANA, corrientes_filtrados[1]);
filtrar_datos(corrientes[2], VENTANA, corrientes_filtrados[2]);
// Filtra valores de corriente

uint8_t response = process_values(voltajes_filtrados, corrientes_filtrados);
// Procesa valores
if (response) {
    /*
     * Que si no recibe señal, no envia nada.
     */
    prepare_values();
    // Prepara cadenas

    udp.broadcastTo(send_v_string, 8);
    Serial.println(send_v_string);
    udp.broadcastTo(send_i_string, 8);
    Serial.println(send_i_string);
    // Envia los datos
}
}
}

// Funciones de procesamiento de datos

uint8_t scan_and_return_networks() {
    /*
     * Escanea buscando redes de WiFi disponibles.
     */
}
```

```
*/
return WiFi.scanNetworks();
}

uint8_t connect_wifi(String ssid, String password) {
  /*
  Realiza la conexión WiFi con la red provista.
  */
  WiFi.mode(WIFI_STA);
  char ssidBuffer[80];
  ssid.toCharArray(ssidBuffer, 80);
  WiFi.begin(ssidBuffer, password);
  if (WiFi.waitForConnectResult() != WL_CONNECTED) {
    Serial.println("WiFi Failed");
    return 0;
  }
  if (udp.connect(IPAddress(IP0, IP1, IP2, IP3), 8)) {
    Serial.println("UDP connected");
  } else {
    return 0;
  }

  return 1;
}

void conectar_wifi() {
  /*
  Realiza el proceso de recibir la red y contraseña, comunicándose con la aplicación por
  BT.
  */
  uint8_t wifi_connected = 0;
  uint8_t select_ready = 0;
  char serial_read;
  uint8_t n_nets;

  while (!wifi_connected) {
```



```
if (!SerialBT.available()) continue;
// Si no hay datos disponibles, continua

serial_read = SerialBT.read();
Serial.println(serial_read);

if (select_ready == 0 & serial_read == 'r') {
    // Escanea las redes y las envia

    n_nets = scan_and_return_networks();
    // Escaneo de redes

    if (n_nets) {
        // Genera la cadena de redes disponibles
        for (uint8_t i = 0; i < n_nets; ++i) {
            WiFi.SSID(i).trim();
            Serial.printf("%s/", WiFi.SSID(i).c_str());
            SerialBT.printf("%s/", WiFi.SSID(i).c_str());
            // Envia cada SSID por BT
        }
        Serial.println();
    }
    SerialBT.flush();

    select_ready = 1;
} else if (select_ready == 1 & serial_read == 'c') {
    // Recibe el SSID - Mandar un caracter adelante
    ssid = SerialBT.readStringUntil('/');
    Serial.print(ssid);
    Serial.print("-");

    // Recibe la contraseña
    password = SerialBT.readStringUntil('/');
    Serial.println(password);

    select_ready = 0;
```

```
// Verifica la conexion y da una señal de afirmacion
wifi_connected = connect_wifi(ssid, password);
if (wifi_connected) {
    Serial.println("Conectado");
    SerialBT.print("1");
    break;
} else {
    Serial.println("No Conectado");
    SerialBT.print("0");
}
}
}
}

// Funciones de conexion WiFi

// Funciones

void setup() {
    Serial.begin(9600);
    SerialPort.begin(115200, SERIAL_8N1, 16, 17);
    SerialPort.flush();
    // Inicializa y limpia el puerto serie
    SerialBT.begin(device_name);
    // Inicializa la comunicacion BT

    WiFi.mode(WIFI_STA);
    WiFi.disconnect();
    // Se desconecta por precaucion

    String BT_MAC = SerialBT.getBtAddressString();
    Serial.print("BT Address: ");
    Serial.println(BT_MAC);
    // Presenta la MAC address por Serial

    lcd.init();
}
```

```
// Inicializa el display
lcd.clear();
lcd.backlight();
lcd.setCursor(2, 0);
// Presenta la MAC address por display

BT_MAC.replace(":", "");
lcd.print(BT_MAC);
// Quita los ':' para que entren en el display

if (DEBUG) { // Revisar constante DEBUG al principio del documento
  connect_wifi("TP-LINK_E1F2", "46050021");
} else {
  conectar_wifi();
}
/*
Procesa la comunicacion Bluetooth y cuando logra conectar al WiFi,
corta y desabilita el BT.
*/

lcd.clear();
lcd.setCursor(2, 0);
lcd.print(WiFi.localIP());
// Presenta la IP WiFi por display

Serial.println("OK - Funcionando...");
delay(1000);
}

void loop() {
  if (SerialPort.available()) {
    recibir_procesar();
  }
}
```

Bibliografía

- [1] Phadke, A. G., & Thorp, J. S. (2008). Synchronized Phasor Measurements and Their Applications. Springer.
- [2] Rogers, G. (2000). Power System Oscillations. Kluwer Academic Publishers.
- [3] CAMMESA S.A. (2018). Los Procedimientos Versión XXVI - Anexo 24.
- [4] Cigré. B111. (1996). Analysis and Control of Power System Oscillations, Task force 07 of Advisory Group 01 of Study Committee 38.
- [5] Kundur, P. (1994). Power System Stability and Control. McGraw-Hill.
- [6] Machowski, J., Bialek, J. W., & Bumby, J. (2008). Power Systems Dynamics. Stability and Control. Wiley.
- [7] Sauer, P., & Pai, M. (2005). Power System Dynamics and Stability. Springer.
- [8] Monti, A., Muscas, C., & Ponci, F. (2016). Phasor Measurement Units and Wide Area Monitoring Systems. Elsevier.
- [9] IEEE Std. 60255-118-1-2018. (2018). IEEE/IEC International Standard - Measuring relays and protection equipment - Part 118-1: Synchrophasor for power systems - Measurements.
- [10] IEEE Std C37.118.2-2011. (2011). IEEE Standard for Synchrophasor Data Transfer for Power Systems.
- [11] IEEE Std. 1344 (R2001). (2001). IEEE Standard for Synchrophasors for Power Systems.
- [12] IEEE Std. C37.111-2013. (2013). IEEE/IEC Measuring relays and protection equipment Part 24: Common format for transient data exchange (COMTRADE) for power systems.
- [13] NASPI. (n.d.). North American SynchroPhasor Initiative. Retrieved from <http://www.naspi.org>.
- [14] NASPI. (2018). Synchrophasor Monitoring for Distribution Systems.
- [15] Lu, C., Shi, B., Wu, X., & Sun, H. (2015). Advancing China's Smart Grid: Phasor Measurement Units in

a Wide-Area Management System. IEEE Power & Energy Magazine, 13(5), 60-71.

[16] Grid Protection Alliance. (n.d.). GPA Grid Protection Alliance. Retrieved from <https://www.gridprotectionalliance.org>.

[17] IEEE Std. C37.244. (2013). IEEE Guide for Phasor Data Concentrator Requirements for Power System Protection, Control, and Monitoring.

Referencias

Phadke, A. G., Thorp, J. S., & Adamiak, M. (1983). Fundamentos teóricos y proceso computacional para el cálculo de fasores en tiempo real. Transformada Discreta de Fourier (DFT) para la obtención de fasores.

Autor. (1988). Primer Unidad de Medición Fasorial (PMU) construida en el Tecnológico de Virginia, sincronizada mediante el Sistema Satelital de Posicionamiento Global (GPS) y comercializada por Macrodyne.

Phadke, A. G. (1993). Descripción de la PMU y formas de sincronización. Destaque de la sincronización mediante el Sistema de Posicionamiento Global (GPS) con una exactitud de 1 ms.

Ota, Y., & Fujita, H. (2000). Evaluación de la estabilidad del sistema de potencia utilizando unidades de medición fasorial con microcontrolador DSP y sincronización mediante pulso por segundo del Sistema de Posicionamiento Global.

San Vicente, R. (2004). Desarrollo de una PMU sincronizada vía satélite en México con microcontrolador DSP de Texas Instruments.

Cuvas, C. (2005). Desarrollo de una PMU sincronizada vía satélite en México con microcontrolador DSP de Freescale.

Hernández, B. V. (2009). Desarrollo de una PMU sincronizada vía satélite en México con microcontrolador de ColdFire.

Sincrofasores y monitoreo de energía eléctrica trifásica

Valdiosera Marroquín, A. (2013). Desarrollo de un medidor inteligente con comunicación bidireccional, microcontrolador ColdFire y receptor GPS para mediciones fasoriales sincronizadas. Comunicación inalámbrica mediante Zig-Bee.

Cárdenas, D. S. (2015). Desarrollo de un dispositivo de medición inteligente para la detección de robo de energía, cumpliendo con los estándares IEEE 1459 y IEEE C37.118, con receptor GPS.