



UNIVERSIDAD SANTO TOMÁS

Proyecto de grado

**Desarrollo De Un Sistema De Monitoreo Industrial Para Un Sistema
Solar Fotovoltaico De 100 kW Basado En Protocolo MODBUS**

Desarrollado por:

Angie Sofia Aponte Sanguino

Universidad Santo Tomás

Facultad de Ingeniería Electrónica

Tunja, Colombia

2023



UNIVERSIDAD SANTO TOMÁS

Desarrollo De Un Sistema De Monitoreo Industrial Para Un Sistema Solar Fotovoltaico De 100 kW Basado En Protocolo MODBUS

Desarrollado por:

Angie Sofia Aponte Sanguino

Dirigido por:

Mg. Carlos Alberto Cardona Coy

PhD. Cesar Mauricio Galarza Bogotá

Trabajo de grado para optar por el título de Ingeniero Electrónico

Universidad Santo Tomás

Tunja, 2023

Angie Sofia Aponte Sanguino, 2023

Los conceptos e ideas presentados en este trabajo son de responsabilidad exclusiva de los autores. Esto no representa la opinión de la Universidad Santo Tomás ni la de la Facultad de Ingeniería Electrónica.

Angie Sofia Aponte Sanguino

Notas de Aceptación

Firma del director

Firma del codirector

Firma del Jurado

Firma del Jurado

Dedicado a mi familia por su apoyo durante este proceso tan importante de mi vida, a mi abuelito quien siempre soñó verme graduada, a mis amigas en México que fueron parte de un proceso de crecimiento para mí y, a las personas que me guiaron y enseñaron lo necesario para llegar a ser quien soy actualmente.

Angie Sofia Aponte Sanguino

Agradecimientos

Agradezco a mis papás por apoyarme siempre, por darme los consejos y las fuerzas para cumplir mis sueños.

A mis compañeros, por compartir conmigo una etapa que sin duda es de las más importantes en el desarrollo profesional y por enseñarme que hay amistades que vale la pena conservar.

Finalmente, un especial agradecimiento a los docentes que con su conocimiento aportaron significativamente a mi crecimiento profesional y personal.

Tabla de Contenidos

Agradecimientos	7
Tabla de Contenidos	8
Lista de figuras.....	11
Lista de tablas	13
Resumen.....	14
1 Introducción.....	15
2 Presentación del Problema.....	16
3 Justificación	18
4 Objetivos.....	19
4.1 Objetivo General	19
4.2 Objetivos Específicos	19
5 Marco Referencial	21
5.1 Marco teórico	21
5.1.1 Monitoreo industrial.....	21
5.1.2 Internet de las cosas (IoT)	21
5.1.3 Protocolo de comunicación	24
5.1.4 Arquitecturas de comunicación.....	25

5.1.5	Topologías de redes.....	27
5.1.6	Estándar RS-485.....	31
5.1.7	Modbus.....	32
5.1.8	Máquina de estados finitos.....	42
5.1.9	Controlador lógico programable (PLC)	43
5.2	Estado del arte	47
5.3	Fundamentos legales	50
6	Desarrollo del proyecto	51
6.1	Materiales y componentes.....	51
6.2	Metodología de desarrollo.....	53
6.2.1	Investigación previa	53
6.2.2	Diseño y definición del sistema.	54
6.2.3	Implementación en el PLC.....	64
6.2.4	Simulación, pruebas y documentación del sistema.....	68
7	Resultados.....	73
7.1	Datos simulados en el PLC	73
7.2	Datos en plataforma	75
7.3	Puesta en marcha en sitio	76
8	Actividades desarrolladas en la pasantía	79

8.1	Acompañamiento a proyectos	79
8.2	Desarrollo de herramientas para operación y mantenimiento	79
9	Conclusiones.....	80
9.1	Aspectos por mejorar	80
9.2	Conclusiones del proyecto	80
	Referencias.....	83
	Anexos	91
	Manual de instalación del gabinete de control.....	91

Lista de figuras

<i>Figura 1. Elementos de una plataforma IoT</i>	22
<i>Figura 2. Protocolos de comunicación usados en sistemas de monitoreos.</i>	24
<i>Figura 3. Estructura arquitectura cliente-servidor</i>	26
<i>Figura 4. Topologías de redes más usadas</i>	28
<i>Figura 5. Diagrama general de protocolo Modbus.</i>	33
<i>Figura 6. Zonas de memoria Modbus.</i>	34
<i>Figura 7. Esquema de solicitud/respuesta Modbus RTU</i>	36
<i>Figura 8. Estructura de las tramas de solicitud/respuesta</i>	37
<i>Figura 9. Estructura de las tramas de datos.</i>	38
<i>Figura 10. Integración Modbus con TCP/IP</i>	39
<i>Figura 11. Estructura general de las tramas de datos de Modbus TCP/IP</i>	40
<i>Figura 12. Trama de datos en Modbus TCP/IP</i>	41
<i>Figura 13. Estructura de un controlador lógico programable.</i>	43
<i>Figura 14. Instrucción en TIA PORTAL para comunicación Modbus.</i>	44
<i>Figura 15. Instrucción en TIA PORTAL para declaración de maestro Modbus</i>	46
<i>Figura 16. Metodología de desarrollo de software aplicada</i>	53
<i>Figura 17. Conexiones del inversor</i>	56
<i>Figura 18. Conexiones de la estación meteorológica.</i>	58
<i>Figura 19. Diagrama general de comunicaciones</i>	63
<i>Figura 20. Diagrama de máquina de estados finitos</i>	65
<i>Figura 21. Variables y constantes del programa</i>	66
	11

<i>Figura 22. Configuración de parámetros modbus</i>	67
<i>Figura 23. Configuración del simulador</i>	68
<i>Figura 24. Gabinete de control de prueba</i>	70
<i>Figura 25. Mapa modbus equipo de monitoreo</i>	71
<i>Figura 26. Verificación de datos leídos.</i>	73
<i>Figura 27. Datos visualizados en la plataforma</i>	75
<i>Figura 28. Sistema solar fotovoltaico por monitorear.</i>	76
<i>Figura 29. Gabinete de control y conexión de la clavija de comunicaciones.</i>	77
<i>Figura 30. Monitoreo del sistema instalado.</i>	78

Lista de tablas

<i>Tabla 1. Funciones del protocolo Modbus</i>	35
<i>Tabla 2. Configuración necesaria en equipos conectados a red Modbus</i>	36
<i>Tabla 3. Parámetros de configuración de instrucción Modbus</i>	45
<i>Tabla 4. Parámetros de configuración de maestro Modbus</i>	46
<i>Tabla 5. Lista de materiales para el sistema de monitoreo</i>	52
<i>Tabla 6. Requerimientos del sistema de monitoreo</i>	55
<i>Tabla 7. Especificaciones de comunicaciones de la estación meteorológica.</i>	58
<i>Tabla 8. Datos de ejemplo para la simulación</i>	69

Resumen

Se ha desarrollado un sistema de monitoreo industrial para un sistema solar fotovoltaico de 100 kW, que permite adquirir datos como voltajes y corrientes AC y DC, energía total generada, irradiancia, temperatura ambiente y temperatura del panel solar, esto basado en el protocolo Modbus. Para esto se inicia con el análisis de las arquitecturas y topologías que permiten una mejor adquisición de datos con este protocolo, para después implementar la programación en el PLC Siemens S7 1200 y, realizar las pruebas y simulaciones correspondientes para verificar su funcionamiento. Lo anterior se verifica en una plataforma web de la empresa ERCO Energía, la cual es administrada por un área diferente, en donde gráficamente se pueden observar los datos por horas, días y meses. De esta manera, en el documento se observa el procedimiento realizado junto con evidencias de la puesta en marcha en campo del proyecto.

1 Introducción

El uso de nuevas tecnologías para monitorear sistemas solares fotovoltaicos ha ido en aumento con la creciente demanda de nuevas estrategias para la transición energética [1]. Dentro de este proyecto, se busca desarrollar un sistema que permita adquirir los datos de dispositivos que componen el sistema solar fotovoltaico de 100 kW, como inversores y estación meteorológica, con el fin de ser enviados a un servidor propio de la empresa, en donde se almacenan los datos y poder visualizar su gráficamente el comportamiento operativo del mismo. Esto resulta de ayuda tanto para el equipo de mantenimiento, quienes pueden revisar si el sistema opera correctamente o si ha entrado en falla, pero también estos datos benefician a los clientes, puesto que pueden ver que el sistema se encuentra en funcionamiento, además de ver las estadísticas de generación. Sin embargo, en este documento se presentan los resultados en la plataforma para el equipo de mantenimiento.

Se presenta como se diseña el proyecto, desde el tipo de arquitectura y topología que se va a usar, recordando que la base del sistema de comunicación debe ser Modbus. Una vez definido este aspecto, se realiza la programación en el PLC que permita obtener estos datos, la cual debe ser realizada con precisión, puesto que los dispositivos que se van a leer poseen diferentes direcciones de memoria, un aspecto clave para la comunicación con Modbus.

Posteriormente, se realizan las simulaciones, un aspecto clave que se debe tener en cuenta para asegurarse de que el sistema adquiere los datos de forma correcta. Este proyecto se plantea hasta la simulación y verificación de datos en la plataforma, pero finalmente se da la posibilidad de hacer la visita de campo e instalar el sistema de monitoreo, lo que da paso a tomar los datos de un inversor y estación meteorológica, y ver los datos reales.

2 Presentación del Problema

A medida que ha aumentado la demanda de energía eléctrica, los sistemas solares fotovoltaicos se han consolidado como una solución efectiva para reducir costos durante la generación de energía y así reducir la emisión y producción de gases de efecto invernadero. Por esta razón, se han convertido en una tecnología competitiva a nivel mundial en comparación con las formas tradicionales de generación de energía [2].

Por este motivo, se determina la existencia de generar y utilizar nuevas tecnologías para monitorear el funcionamiento de los sistemas solares fotovoltaicos. Dependiendo de la potencia instalada y las necesidades de la instalación eléctrica, se pueden requerir dispositivos IoT, como Raspberry Pi, Arduino u otros, así como equipos más avanzados como los autómatas programables (PLC), que permiten establecer sistemas de comunicación para monitorear el estado y funcionamiento de los sistemas solares fotovoltaicos [1].

En este contexto, los proveedores de sistemas solares fotovoltaicos se esfuerzan cada vez más por ofrecer a sus clientes nuevos servicios IoT que permitan visualizar los datos en la web u otras plataformas. Esto les brinda una forma de verificar la generación diaria, mensual o anual y así asegurarse de que sus sistemas estén operativos y generando la potencia esperada para cubrir sus necesidades energéticas.

Por lo tanto, se propone el planteamiento del diseño de un sistema de monitoreo industrial para sistemas solares fotovoltaicos. Este sistema debe ser capaz de adquirir información crucial sobre el funcionamiento, como la potencia generada, los voltajes y las corrientes AC y DC, así como el estado de operación, de los diversos dispositivos que componen el sistema fotovoltaico, como inversores y estaciones meteorológicas. Esta información se puede almacenar en una base

de datos, con el fin de realizar el respectivo procesamiento y análisis, para finalmente ser presentados en la página web para realizar un seguimiento de los proyectos realizados.

Teniendo en cuenta esta información, la pregunta problema a solucionar con el proyecto es ¿Cómo realizar la comunicación entre los inversores solares, la estación de monitoreo y el PLC para obtener la información de operación del sistema solar fotovoltaico?

3 Justificación

Mirando a la situación actual de las energías renovables, un sector que ha presentado un aumento considerable de capacidad instalada en América Latina, alcanzando un 65,93% en Colombia [3] y en donde ERCO Energía, que con más de 3500 proyectos solares alcanzando 124 MW instalados [4] ofrece a sus clientes la posibilidad de visualizar los datos de funcionamiento de su sistema solar fotovoltaico. Para brindar esta solución, resulta fundamental disponer de herramientas tecnológicas que posibiliten establecer redes de IoT para obtener la información necesaria.

De esta manera, el sistema de monitoreo que se propone pretende dar una solución que permita obtener de forma continua los datos esenciales que permiten conocer variables de funcionamiento y operación del sistema, como generación total de energía (kWh), corrientes AC y DC, tensiones AC y DC, y la temperatura de paneles, estado de operación de inversores, entre otros, de manera que se le facilite a otra área de la empresa, almacenando la información en una base de datos y posteriormente su presentación en una plataforma web existente de ERCO Energía, en donde finalmente se pueden visualizar los datos adquiridos.

A favor de cumplir con lo mencionado anteriormente, el sistema de monitoreo industrial tiene como dispositivo principal de adquisición de datos a un PLC Siemens S7 1200 - 1212 C el cual, de acuerdo con sus características y diversas opciones de programación, permite establecer una lectura constante de la operación y funcionamiento del sistema solar fotovoltaico por medio del protocolo de comunicación industrial MODBUS, facilitando la rápida detección de fallas en los inversores o en el sistema en general [5].

4 Objetivos

4.1 Objetivo General

Desarrollar un sistema de monitoreo industrial para un sistema solar fotovoltaico de 100 kW mediante el uso de PLC y comunicación con protocolo MODBUS, que permita enviar los datos de operación y funcionamiento a la página web gestionada por otra área de la empresa, de manera que por medio de esta interfaz se pueda verificar la recepción de los datos adquiridos utilizando un usuario de prueba previamente proporcionado por ERCO Energía.

4.2 Objetivos Específicos

1. Analizar las arquitecturas de comunicación asociadas a las redes de datos, llegando a proponer una que garantice la transferencia de información de los distintos dispositivos del sistema solar fotovoltaico como inversores y estación meteorológica por medio del protocolo MODBUS.
2. Diseñar el código de programación en el PLC utilizando el modelo de máquinas de estado finitas, que permita al S7 1200 en físico adquirir datos de inversores y estación meteorológica generados por el simulador de protocolo Modbus MbSlave, como voltajes y corrientes en AC y DC, energía generada, estado de operación u otros.
3. Realizar las pruebas de integración el código de programación implementado en el PLC con la plataforma web existente de ERCO energía, llegando a observar los datos adquiridos en las simulaciones desarrolladas de manera que se verifique el funcionamiento del sistema.

4. Identificar los aspectos principales del funcionamiento del sistema, llegando a elaborar el manual de instalación del gabinete de control que ilustre al equipo técnico las conexiones que se requieren para el sistema de monitoreo, además de funcionar como documento base para revisión de problemas que se puedan presentar.

5 Marco Referencial

5.1 Marco teórico

Para desarrollar el proyecto, se deben comprender los siguientes conceptos de comunicaciones industriales y sistemas solares fotovoltaicos para poder ejecutar de forma correcta la programación del sistema de monitoreo.

5.1.1 Monitoreo industrial

El monitoreo industrial se refiere a la supervisión y análisis sistemático de los procesos de producción en una industria con el fin de mejorar su rendimiento, eficiencia y seguridad [6]. En el contexto de un parque solar, el monitoreo industrial puede incluir la supervisión de la producción de energía en tiempo real, el seguimiento del desempeño del equipo, la detección temprana de fallas y la optimización del mantenimiento preventivo de los paneles solares [7].

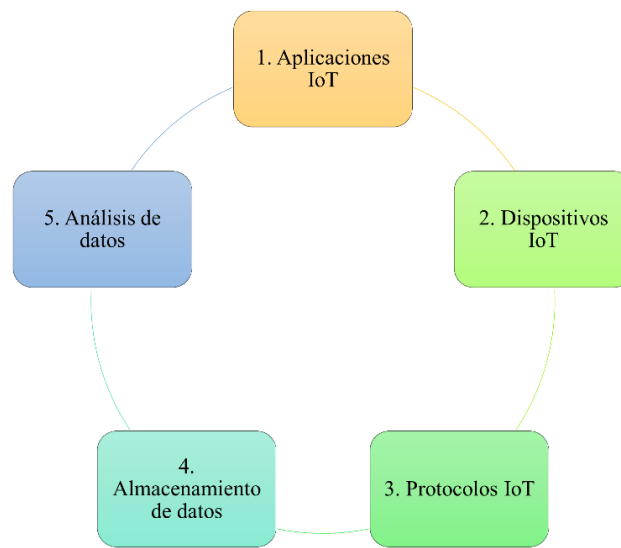
5.1.2 Internet de las cosas (IoT)

IoT se ha hecho conocer como una tecnología emergente que utiliza Internet y proporciona conectividad entre dispositivos por medio del uso de sensores y redes de comunicación adecuados, con los cuales se pueden obtener datos y ofrecer diversos servicios a las personas [8].

La Figura 5 muestra los distintos componentes de una plataforma IoT. Al planificar una aplicación IoT, el primer paso para diseñar sistemas IoT, la selección de los componentes de IoT, como el dispositivo sensor, el protocolo de comunicación, el almacenamiento de datos y la computación, debe ser adecuada para la aplicación prevista. Los dispositivos IoT, que son los segundos componentes de las plataformas IoT, pueden tener forma de sensores, actuadores,

pasarelas IoT o cualquier dispositivo que se una al ciclo de recogida, transmisión y procesamiento de datos.

Figura 1. Elementos de una plataforma IoT



Fuente: Elaborada por el autor

Los protocolos de comunicación, que constituyen el tercer componente de la plataforma IoT, permiten que los dispositivos se comuniquen y compartan datos con los controladores o los centros de toma de decisiones [8]. Las plataformas IoT ofrecen la flexibilidad de seleccionar el tipo de tecnologías de comunicación a partir de los requerimientos de la aplicación como, por ejemplo, Wi-Fi, Bluetooth, ZigBee y tecnología celular como las redes LTE-4G y 5G [9].

El almacenamiento de datos resulta ser el cuarto componente de la plataforma IoT que permite gestionar los datos recogidos de los sensores. En principio, los datos recogidos de los dispositivos son muy grandes [10]. Esto hace necesario planificar un almacenamiento de datos eficiente que puede estar en servidores en la nube o en el borde de una red IoT [11]. Los datos

almacenados que se utilizan con fines analíticos constituyen el quinto componente de las plataformas IoT.

El análisis de datos puede realizarse fuera de línea tras almacenar los datos o en forma de análisis en tiempo real. La analítica de datos se realiza tomar decisiones de funcionamiento usando como base los datos tomados. Y, a partir de esto, el análisis de datos puede realizarse off-line o en tiempo real. En la analítica off-line, los datos almacenados se recogen primero y luego se visualizan in situ mediante herramientas de visualización. En el caso de la analítica en tiempo real, la nube o los servidores de borde se utilizan para proporcionar la visualización, por ejemplo, la analítica de flujo [10].

5.1.2.1 IoT en sistemas solares fotovoltaicos

El internet de las cosas ha presentado una gran ventaja para el sector de la energía solar fotovoltaica, debido a la necesidad de recopilar datos en tiempo real y optimizar la producción y consumo de energía. Viéndolo de esa manera, los equipos de los sistemas fotovoltaicos han mejorado sus tecnologías las cuales permiten implementar IoT, lo que permite conectar más dispositivos y diferentes tipos de sensores para ello. Estas mejoras posibilitan la recopilación y análisis simultáneo de datos, lo que resulta en una herramienta muy buena para medir la eficiencia en la generación de energía y bajar el costo del mantenimiento (Cobo, 2021).

Por otro lado, los usuarios pueden tener la posibilidad de conocer el consumo energético de su hogar y así ajustar los hábitos de uso de la energía, lo cual se logra con el monitoreo en tiempo real del sistema fotovoltaico [13].

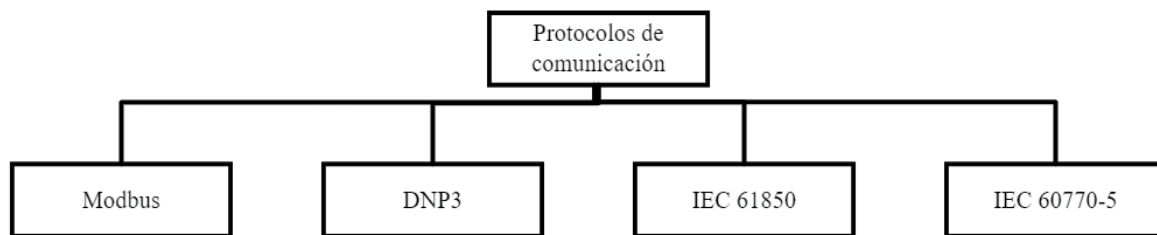
En adición, se pueden integrar sistemas de control y seguridad que funcionen automáticamente a partir de los datos, aumentando la eficiencia del sistema y protegiendo los datos

del usuario, un aspecto clave en toda red IoT. Además, de que por la facilidad que algunos equipos presentan, se pueden detectar las fallas, lo que mejora la confiabilidad y extiende la vida útil de los componentes del sistema, puesto que se pueden tratar a tiempo (Kamble, 2021).

5.1.3 Protocolo de comunicación

El buen funcionamiento del sistema de comunicación va a depender de los protocolos que se utilicen, para lo cual se requiere evaluar diferentes parámetros de red, lo que al final se ve reflejado en qué tan bueno es el servicio que se está brindando con la aplicación IoT creada [15].

Figura 2. Protocolos de comunicación usados en sistemas de monitoreos.



Fuente: Elaborada por el autor

Al hablar de los protocolos de comunicación, se hace referencia a las diferentes reglas y normativas que permiten el envío entre los dispositivos. Es relevante destacar que estos protocolos, gracias a las reglas establecidas, tienen la capacidad de recuperar datos perdidos durante la transmisión de información[16]. Algunos de los protocolos más usados para diferentes aplicaciones industriales se encuentran Modbus, IEC60870-5, Profibus, Profinet, IEC61850, DNP3, entre otros, (figura 2). Debido a que en este documento el proyecto se centra sobre el protocolo Modbus, este se explica en secciones posteriores.

5.1.4 Arquitecturas de comunicación

La arquitectura de comunicación se refiere al conjunto de reglas, protocolos y estándares que definen la manera en que los distintos dispositivos intercambian información cuando están conectados en una red. Estas reglas y normas son fundamentales para establecer un marco de referencia común que permita la interoperabilidad y cooperación efectiva entre los dispositivos conectados [17].

La arquitectura de comunicación abarca diversos aspectos cruciales, como la codificación de datos, el enrutamiento, la multiplexación, la detección de errores y el nivel de seguridad en la comunicación. Dichos elementos resultan esenciales para garantizar que los datos se empaqueten, transmitan y reciban de manera eficiente y confiable entre los dispositivos de la red, permitiendo así una comunicación efectiva y sin interrupciones [18].

Además, se destaca que la arquitectura de comunicación se enfoca en el nivel lógico y los protocolos que posibilitan la comunicación, independientemente de cómo estén físicamente conectados los dispositivos [19]. Esto significa que, independientemente de las características específicas de los dispositivos o la forma física que se utilice para la conexión, la arquitectura de comunicación proporciona un marco general para garantizar una comunicación fluida y confiable en una red.

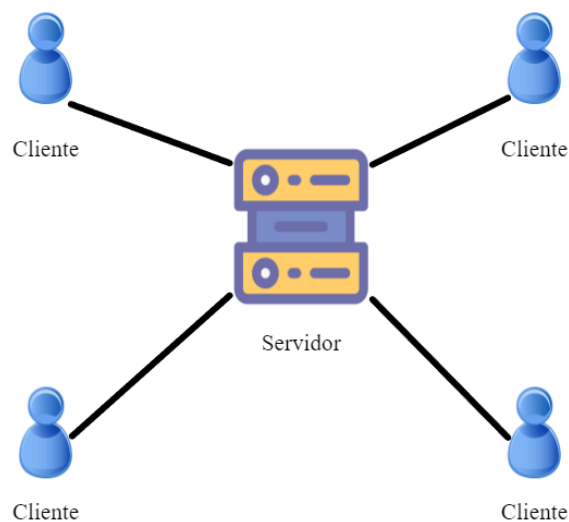
5.1.4.1 Cliente-Servidor

La arquitectura cliente-servidor es un modelo de conexión en la que los dispositivos pueden tener el rol de cliente o de servidor. Este enfoque es ampliamente utilizado en las redes de computadoras y las aplicaciones distribuidas de manera que sea más fácil ejecutar la comunicación

entre dispositivos (Ver figura 3). En este modelo, el cliente envía solicitudes al servidor y espera respuestas para cumplir con sus necesidades específicas, mientras que el servidor proporciona los servicios, recursos o datos solicitados por los clientes [20].

Esta comunicación se basa en un protocolo definido que permite la interacción. Por ejemplo, se utiliza el protocolo Modbus TCP para la comunicación industrial. Otros protocolos comunes para esta arquitectura incluyen HTTP/HTTPS para la web y FTP para la transferencia de archivos.

Figura 3. Estructura arquitectura cliente-servidor



Fuente: Elaborada por el autor

La arquitectura cliente-servidor ofrece varias ventajas. En primer lugar, permite agregar clientes y servidores de acuerdo con la necesidad del sistema, lo que le da un punto por su alta escalabilidad. Además, al separar las responsabilidades entre clientes y servidores, el mantenimiento y las actualizaciones se pueden realizar de manera más simplificada y sin afectar necesariamente a la otra parte del sistema. También ofrece medidas de seguridad al intentar

acceder a los recursos en el servidor y centraliza los datos críticos en un único lugar, lo que facilita la gestión y la copia de seguridad [21].

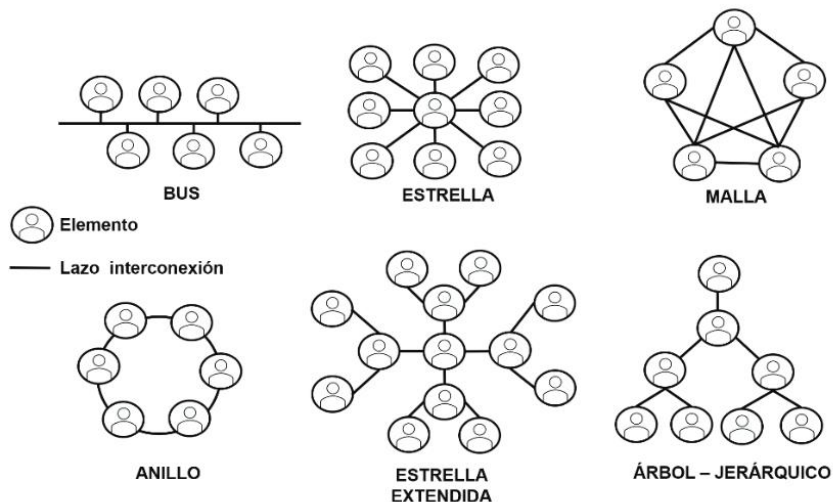
Sin embargo, la arquitectura cliente-servidor también presenta desventajas. Configurar y mantener servidores dedicados puede ser costoso en términos de hardware, software y personal especializado. Un aspecto importante es que la capacidad y la confiabilidad del servidor se ve directamente relacionado con el rendimiento del sistema. Además, la comunicación a través de una red puede introducir latencia, lo que puede ser problemático en aplicaciones sensibles al tiempo. Por último, la gestión de múltiples clientes y servidores puede volverse compleja, especialmente en sistemas distribuidos a gran escala [22].

5.1.5 Topologías de redes

La topología de red es la manera en que se distribuyen los diferentes dispositivos de la red [23]. Al elegir una topología, es fundamental garantizar el funcionamiento óptimo y continuo de la red, considerando las posibles expansiones a futuro. La forma en la que se conectan los dispositivos está relacionada directamente con el flujo de los datos, por lo tanto, se requiere tener en cuenta factores como costo, complejidad y cantidad de datos, los cuales pueden intervenir en las topologías de red [15].

Se clasifican en anillo, estrella, bus, árbol, malla y estrella extendida, las cuales se presentan a continuación (figura 4).

Figura 4. Topologías de redes más usadas



Fuente: Tomado de “Protocolos y topologías utilizadas en los sistemas de comunicación de las microrredes eléctricas”, por M. Aybar Mejía [16]

5.1.5.1 Topología en bus

En esta topología, los nodos se conectan al mismo bus mediante un transceptor que controla la comunicación en la red. En envío de información se realiza por medio de un único bus, de manera que los dispositivos la aceptan o no dependiendo de si les corresponde o no [24]. En esta configuración, no es posible que dos dispositivos intenten transmitir información al mismo tiempo.

Ventajas de la Red en Bus:

- Facilidad de implementación y expansión.
- Costos de implementación bajos.

Desventajas de la Red en Bus:

- Un cable roto puede interrumpir la red.
- El rendimiento puede verse afectado al agregar más dispositivos.

- Puede tener problemas de seguridad debido a que todo se transmite por un mismo bus.

También es ampliamente en la industria para conectar dispositivos como PLC, tarjeta electrónica, computadora, sensores, etc.

5.1.5.2 Topología en estrella

En esta topología, todos los dispositivos se conectan a un equipo principal encargado de transmitir y recibir la información a los demás dispositivos de la red. Esta topología presenta la ventaja de reducir el riesgo de fallos, de modo que, si uno de los nodos experimenta problemas en su conexión cableada con el dispositivo principal, esto no va a afectar a los demás, permitiendo que el resto del sistema opere correctamente [25]. Las ventajas de la red en estrella son:

- Fácil adición de dispositivos en la red.
- La detección de fallos es más sencilla.
- La desconexión de un dispositivo no paraliza el resto de la red.
- Se puede cambiar, agregar o quitar un dispositivo de forma sencilla.

Los inconvenientes de la red en estrella:

- Aumenta el costo en comparación a una red de topología de bus.
- Su funcionamiento depende del switch debido a que, si se encuentra defectuoso, toda la red puede quedar inactiva.
- Permite el uso de switch para conectarse a más redes o dispositivos.

5.1.5.3 Topología en malla

Es una configuración en la cual cada dispositivo de la red se interconecta entre sí. De esta manera, se eliminan las interrupciones porque los datos pueden ser enviados por las diferentes

rutas que se generan en las conexiones entre equipos por lo que esta configuración garantiza una comunicación más segura [26].

5.1.5.4 Topología en anillo

En esta topología, los dispositivos forman un bucle cerrado, es decir, que se conectan en serie. Cada nodo se encuentra con un equipo que repite y envía la información al siguiente dispositivo en la fila. Es importante destacar que, en una red en anillo, la transmisión de datos se realiza en una dirección específica, lo que proporciona un flujo ordenado y secuencial de la información a lo largo del anillo. Además, en caso de que un nodo falle, el resto de los nodos en el anillo pueden seguir operando sin interrupciones [27].

5.1.5.5 Topología en estrella extendida

En esta se conectan múltiples redes en una configuración de estrella, gracias al uso de hubs o switches. Al enlazar varias redes en una sola estructura el alcance se amplía. Cada red individual funciona como una estrella independiente, con su propio hub o switch central, y estas estrellas individuales se conectan entre sí mediante la interconexión de los dispositivos centrales. Esto permite una comunicación más eficiente y una mayor capacidad de expansión de la red, al tiempo que proporciona una mayor flexibilidad para la incorporación de nuevos nodos o redes adicionales en el futuro [28].

5.1.5.6 Topología en árbol

En esta topología, la información se transmite de manera jerárquica, lo que significa que hay un dispositivo principal que envía la información a dispositivos de menor nivel [23]. Al

transmitir la información de manera jerárquica, se establece una estructura de comunicación eficiente y organizada, donde los nodos principales actúan como puntos centrales de control y distribución de la información. Esta característica de la topología contribuye a garantizar una transferencia confiable y rápida de los datos, debido a que los nodos principales se encargan de gestionar y asegurar que la información sea entregada a su destino de manera adecuada [29].

5.1.6 Estándar RS-485

En 1993, la Asociación de Industrias Electrónicas (EIA, por sus siglas en inglés) introdujo el estándar EIA-485, el cual ha desempeñado un papel fundamental en las comunicaciones multipunto. Este estándar ha sido diseñado específicamente para situaciones en las que el número de estaciones involucradas es menor a 32 y donde se requiere una transmisión de datos moderada, priorizando la integridad de los datos transmitidos [30].

El estándar EIA-485 ha demostrado ser altamente confiable y eficaz en una variedad de aplicaciones. Además, una de sus características clave es su capacidad para permitir la comunicación a lo largo de distancias de hasta 1200 metros, convirtiéndolo en una elección ideal para entornos que requieren conexiones a larga distancia. Asimismo, la velocidad de transmisión es de 10 Mbits/s, lo que proporciona un flujo de datos significativo para muchas aplicaciones [31].

La integridad de los datos es una preocupación fundamental en muchas aplicaciones de comunicación, y EIA-485 aborda este desafío de manera efectiva [32]. La información se transmite a través de dos señales complementarias enviadas a través de dos hilos, a menudo denominados A y B. La diferencia de tensión entre estos dos hilos es la que transmite la información, en lugar de la tensión entre un solo hilo y tierra. Esto hace que los sistemas RS-485 sean muy resistentes al

ruido en modo común. Por esta razón, este estándar es especialmente adecuado para entornos industriales y comerciales donde la fiabilidad de la transmisión de datos es esencial [33].

Para diseñar una red en donde se requiera usar RS-485, se deben tener en cuenta varios factores clave:

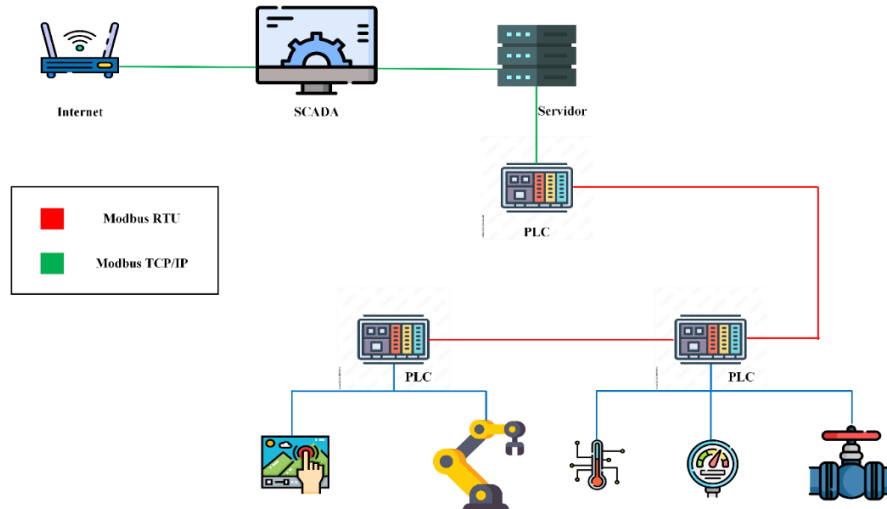
- **Cableado de la red:** La forma de cablear la red es crucial para el éxito de la transmisión de la señal. Se debe utilizar un par trenzado para +(B), -(A) y un cable adicional que termine en REF, DGND o COM.
- **Blindaje:** Utilizar un apantallamiento para proteger las transmisiones de señal de interferencias externas.
- **Polaridad:** Establecer con precisión la polaridad de la red para garantizar una correcta transmisión de la señal.
- **Velocidad de transmisión:** Determina el ritmo de la comunicación.
- **Longitud de la red:** La longitud de la red puede afectar a la calidad de transmisión de la señal.
- **Topología:** La forma de conexión física entre dispositivos, puede influir en la eficacia de la comunicación.

5.1.7 Modbus

Modbus es un protocolo de comunicación de uso libre que maneja un modelo de solicitud-respuesta utilizado para transmitir información entre dispositivos interconectados mediante redes o buses seriales, de Ethernet o de forma inalámbrica [34]. Este protocolo se enfoca en el sector industrial (Figura 5), y se utiliza comúnmente para establecer comunicaciones entre interfaces

HMI, sistemas SCADA, controladores lógicos programables (PLC) y otros dispositivos, lo que facilita la interoperabilidad en procesos industriales [35].

Figura 5. Diagrama general de protocolo Modbus.



Fuente: Elaborada por el autor

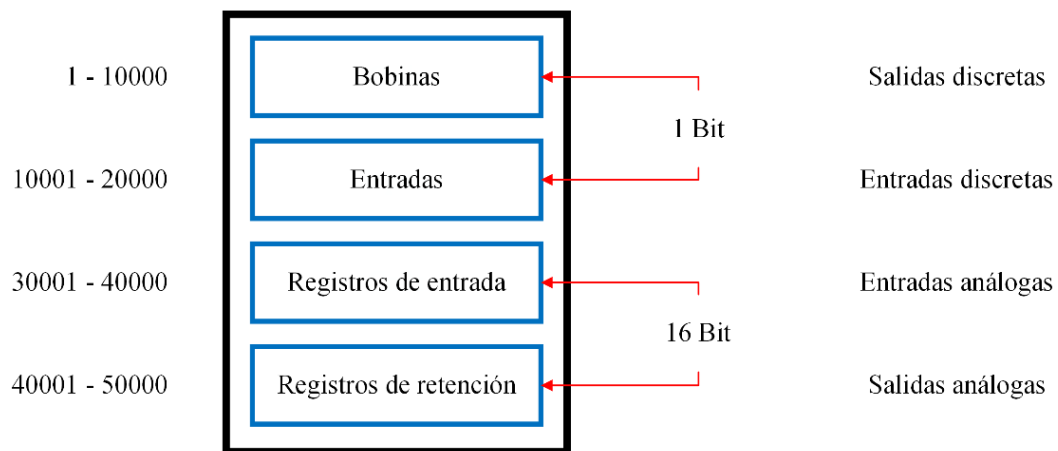
En la comunicación entre distintos controladores (PLC), siempre se asigna la función de "Maestro" a un dispositivo, mientras que los demás actúan como esclavos en una comunicación de uno a muchos. Es importante que cada esclavo tenga un ID único, porque cuando el maestro solicita información, la trama de datos de solicitud llega a cada uno de los esclavos y solo responde el dispositivo que tiene el ID requerido por el maestro.

5.1.7.1 Mapa general de memoria Modbus

Para comprender como se conforman las tramas de información de solicitud/respuesta, se hace necesario considerar el mapa de memoria Modbus (Figura 6), que permite ubicar las

direcciones de memoria donde se almacenan los diferentes tipos de datos consultados por los dispositivos esclavos.

Figura 6. Zonas de memoria Modbus.



Fuente: Elaborada por el autor

Es importante tener en cuenta la organización de las zonas de memoria al programar dispositivos, pues cada zona almacena diferentes tipos de datos. En las direcciones del rango de 1 a 10000, se almacenan los estados de las bobinas correspondientes a salidas discretas, que pueden tomar valores de 1 o 0. En el rango de 10001 a 20000, se almacena el estado de las entradas discretas, también con valores de 1 o 0. Los valores almacenados para valores que se deban almacenar en los registros de entrada se encuentran entre las posiciones 30001 y 40000, mientras que los registros de retención se encuentran de 40001 a 50000. Cabe aclarar que el mapa Modbus con las direcciones reales de los dispositivos puede variar según el fabricante.

5.1.7.2 Códigos de función Modbus

Por otro lado, Modbus cuenta con funciones específicas que deben usarse según el tipo de dato solicitado, es decir, se indica al esclavo qué tipo de dato consultar según las direcciones de memoria. A continuación, se presenta la tabla de funciones:

Tabla 1. Funciones del protocolo Modbus

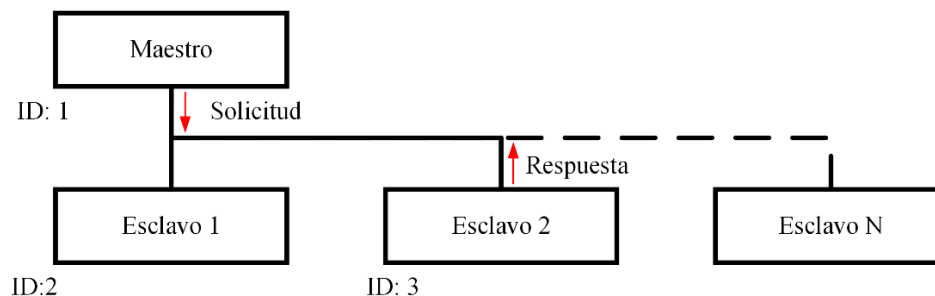
Código	Función
01	Leer estado de las bobinas.
02	Leer estado de las entradas discretas.
03	Leer registros de retención.
04	Leer registros de entrada.
05	Escribir el estado de una bobina.
06	Escribir un valor en un registro de retención.
07	Lee el tipo de excepción.
15	Escribe múltiples estados de bobinas.
16	Escribe múltiples valores en registros de retención.

Fuente: Tomado de Mastering Modbus RS485 Network Communication, por Emile Ackbarali

5.1.7.3 Modbus RTU

Modbus RTU se ejecuta en una red serial RS485 por medio de un formato de trama binaria compacto para la comunicación. Además, Modbus RTU es un protocolo maestro/esclavo, lo que significa que un dispositivo controla la comunicación sobre el resto de los dispositivos (ver figura 7), solicitando la información a un dispositivo con un ID específico [36]

Figura 7. Esquema de solicitud/respuesta Modbus RTU



Fuente: Elaborada por el autor

Al configurar dispositivos para conectarse a una red RS485, es fundamental asegurarse de que los siguientes parámetros sean idénticos para todos ellos:

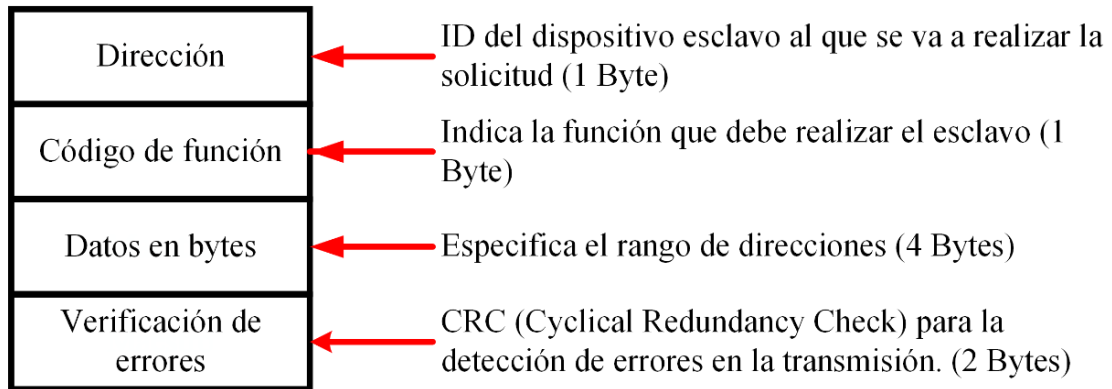
Tabla 2. Configuración necesaria en equipos conectados a red Modbus

Parámetro	Valor
Velocidad de transmisión (Baudios)	9600
Numero de bits	8
Numero de bits de parada	1
Paridad	none

Fuente: Elaborada por el autor

El ciclo solicitud/respuesta (Figura 8) opera de la siguiente manera: El maestro envía una petición (consulta) en forma de un flujo de bytes y el esclavo responde mediante un flujo de bytes similar (pero diferente) que consta de cuatro secciones diferentes formadas por bits, que sirven para diferentes propósitos.

Figura 8. Estructura de las tramas de solicitud/respuesta



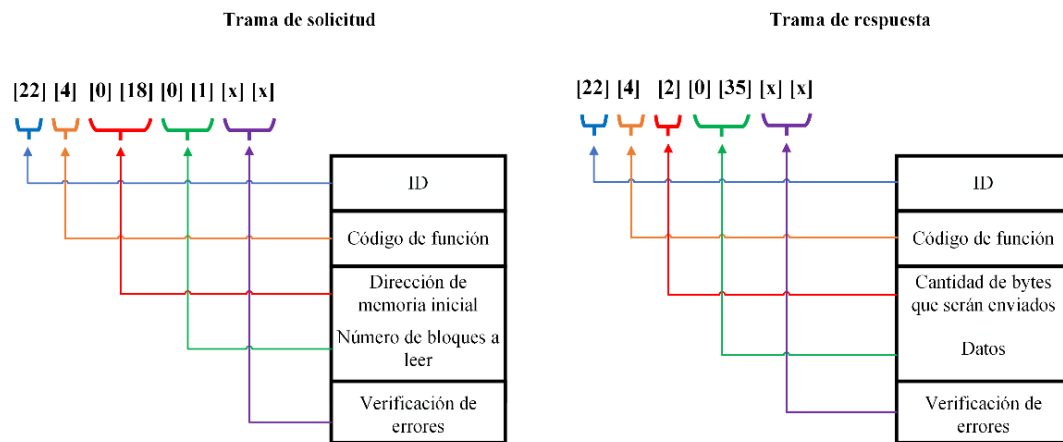
Fuente: Elaborada por el autor

Al observar la trama de datos generada por la solicitud del maestro (Figura 8), se indica el ID del esclavo al que se le va a preguntar, el código de la función que indica el tipo de función que se espera que el esclavo realice y, en la sección “Datos en bytes”, se debe indicar la dirección de memoria desde donde se quiere iniciar a leer los datos, seguida de la cantidad de bloques que se quieren leer, asegurando así la lectura de datos específicos según sea necesario. Por este motivo, es importante conocer las zonas de memoria de Modbus. Los últimos 2 bytes que se envían en la trama de solicitud corresponden a los bytes generados por la detección de errores, bytes que no se conocen hasta que el algoritmo CRC de resultado. En resumen, se maneja de la siguiente manera:

- **Campo de dirección:** Indica la dirección del dispositivo esclavo con el que se quiere comunicar.
- **Código de función:** Indica la función que el maestro solicita al esclavo.
- **Campo de datos:** Contiene los datos que han sido solicitados.
- **CRC o LRC:** Se refiere a los procesos para la identificación de errores (Puede referirse a una sección posterior para observar más a detalle de este tema).

Ahora bien, cuando se analiza la respuesta del esclavo (Figura 9), se observa que el ID y el código de función son iguales a los indicados por el maestro. En la sección de “Datos en bytes”, el esclavo responde indicando la cantidad de bytes que va a enviar, seguido por los bytes de información que han sido solicitados y, finalmente, los 2 bytes de la detección de errores. A continuación, se presenta un ejemplo de estas tramas:

Figura 9. Estructura de las tramas de datos.

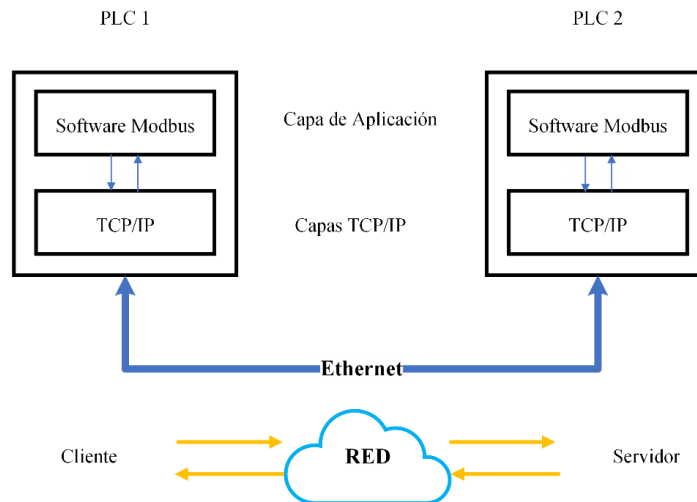


Fuente: Elaborada por el autor

5.1.7.4 Modbus TCP/IP

Por otro lado, Modbus TCP/IP se ejecuta en una red Ethernet utilizando el protocolo TCP/IP, lo que permite la comunicación a través de redes más grandes y complejas. Es decir, que al ser un protocolo cliente/servidor, los dispositivos cliente envían solicitudes a los servidores para obtener información que ha sido leída por la parte compuesta por Modbus, es decir, que en esta variación se tiene una migración de los datos (Figura 10) generados por el software de Modbus para ser enviados con la estructura de datos del TCP/IP [37].

Figura 10. Integración Modbus con TCP/IP



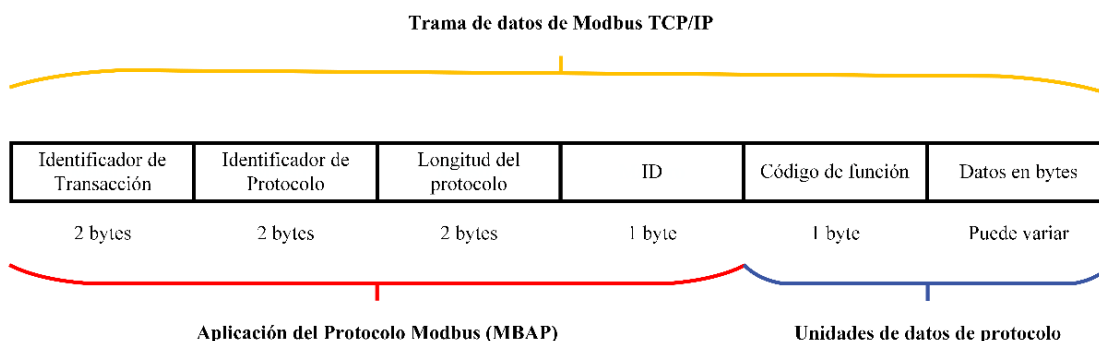
Fuente: Elaborada por el autor

Una de las ventajas que presenta esta variación de Modbus es que puede tener múltiples maestros (clientes) y múltiples esclavos (servidores), lo que implica que se puede generar una comunicación muchos a muchos. Esto significa que un cliente puede funcionar como servidor y un servidor como cliente, aunque su limitación viene dada por la red a la que se conecten.

Además, se debe considerar que, en este protocolo, dos o más clientes no pueden solicitar información al mismo tiempo porque puede generarse una colisión de datos en la red. Para esto se usan técnicas en donde todos los equipos escuchan el medio, si ningún otro dispositivo está enviando datos, el equipo envía lo que le fue solicitado y espera una respuesta del otro dispositivo. El equipo que recibe el dato hace la verificación de errores y de estar todo en orden, envía un visto bueno [38]. En otras palabras, esta técnica hace que los dispositivos se mantengan atentos y den un tiempo de espera para hacer la solicitud de información por la red.

En la figura 11, la trama de datos de Modbus TCP/IP, la cual consta de las siguientes partes [39]:

Figura 11. Estructura general de las tramas de datos de Modbus TCP/IP



Fuente: Elaborada por el autor

- **Identificador de transacción:** Es un campo de dos bytes que se utiliza para identificar la respuesta correspondiente a la solicitud enviada por el dispositivo maestro. Es decir, cuando un dispositivo maestro envía una solicitud a un dispositivo esclavo, esta incluye un identificador de transacción que permite al dispositivo maestro identificar cuál de sus solicitudes ha recibido una respuesta determinada.

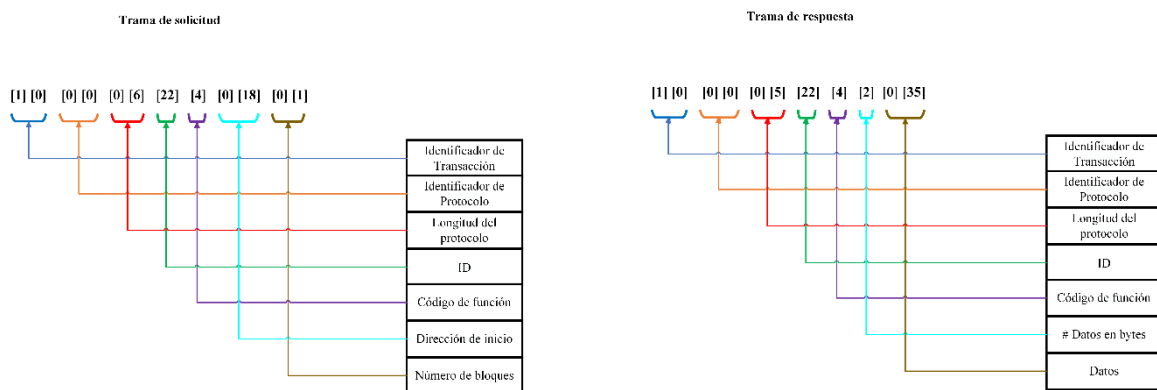
Este identificador permite a los dispositivos maestros manejar múltiples transacciones simultáneamente, ayudando a otorgar mayor eficiencia a la comunicación entre dispositivos.

- **Identificador de protocolo:** Indica que el protocolo siendo utilizado es Modbus. Este campo se utiliza para identificar la comunicación de Modbus TCP/IP en las redes Ethernet [40]. Se identifica con el 0.

- **Longitud del protocolo:** Este campo de dos bytes indica la cantidad de bytes que están siendo transmitidos.
- **ID:** Este campo de un byte indica la dirección del dispositivo esclavo.
- **Código de función:** Este campo de un byte indica la función que está siendo solicitada por el dispositivo maestro.
- **Datos en bytes:** Este campo contiene datos específicos de la función.

Un ejemplo de estas tramas de datos se presenta a continuación:

Figura 12. Trama de datos en Modbus TCP/IP



Fuente: Elaborada por el autor

5.1.7.5 Comprobación de errores

En Modbus se pueden definir dos modos para la comprobación de errores [41]:

- **Chequeo de paridad:** Se implementa un procedimiento en el cual se aplica una operación a cada byte transmitido, con el propósito de calcular el bit de paridad correspondiente a cada byte y lo agregue al mensaje. Cuando el receptor recibe el mensaje, realiza un cálculo de paridad y verifica que corresponda con el bit que fue

enviado por el transmisor. Si no hay coincidencia, el receptor descarta el mensaje. En esta situación, el dispositivo esclavo no envía una respuesta al dispositivo maestro y el temporizador del dispositivo maestro llegará a su límite de tiempo establecido.

- **Chequeo de CRC / LRC:** Cuando el transmisor utiliza este algoritmo en la trama, genera un campo que se agrega a la trama previamente a su envío. Luego, el receptor también aplica el mismo algoritmo y verifica que ambos resultados sean iguales, es decir, que el resultado del transmisor sea igual al receptor. Si no hay coincidencia, se descarta el mensaje.

5.1.8 Máquina de estados finitos

En el ámbito de la programación de PLC, una máquina de estados finitos (FSM) es una técnica de programación que utiliza un conjunto de estados que definen el comportamiento del PLC. Una FSM se puede definir como un diagrama dirigido de estados, donde los nodos indican los estados y las aristas las transiciones entre los estados.

Una vez se enciende el equipo, este analiza las condiciones iniciales, lo que hace que empiece a evaluar los diferentes estados, de manera que, se calculan las expresiones lógicas acordes para cada transición de estados. Si al realizar estas operaciones el autómatas programable obtiene un valor lógico “0”, este se va a mantener en el mismo estado. En caso contrario pasa al siguiente estado acorde al diagrama de estados que fue definido (Mandado Pérez, 2009).

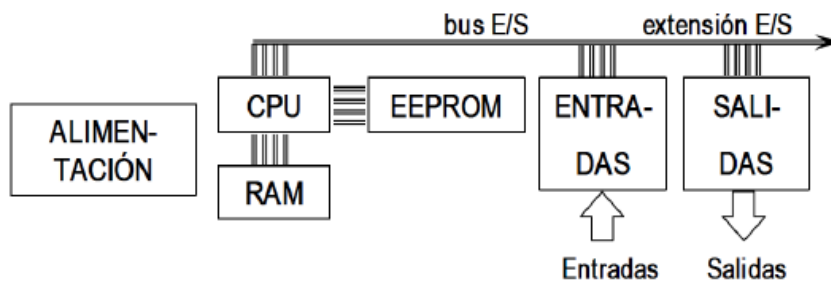
De acuerdo con [43], “las máquinas de estados finitos son una técnica popular en el diseño de controladores para aplicaciones industriales debido a su robustez y facilidad de implementación” (p. 152).

5.1.9 Controlador lógico programable (PLC)

Los Controladores Lógicos Programables (PLC) son dispositivos electrónicos programables en un lenguaje no informático que se utilizan como herramientas para solucionar problemas de automatización y controlar diferentes tipos de procesos en tiempo real. Su funcionamiento parte con base a los sensores y a la programación realizada, lo que le permite ejecutar una acción de control sobre los diferentes actuadores [44].

El PLC está diseñado para entornos industriales lo que lo ubica como una unidad de control de manera que para que funcione, como todo dispositivo electrónico programable, se compone de el hardware y el software. Estos componentes se disponen de diferentes formas y configuraciones como se muestra en la figura 13.

Figura 13. Estructura de un controlador lógico programable.



*Fuente: Tomado de Diseño y desarrollo de una red MODBUS RTU basada en Arduino,
por J. Jose Rabadan Barastegui.*

Dependiendo del fabricante y de la gama del dispositivo, se pueden agregar módulos extras de entradas y salidas analógicas y/o digitales, módulos de comunicaciones con los que se pueden utilizar diferentes protocolos de comunicaciones como Profinet, Profibus, Modbus, entre otros,

siendo muy útil al momento de diseñar redes de comunicaciones para monitorear y controlar diferentes tipos de dispositivos en una industria.

Particularmente, para el desarrollo de este proyecto se requiere usar el protocolo Modbus en un controlador lógico programable, específicamente con un controlador de la gama SIMATIC S7-1200 1212C de Siemens. Para este fin, es importante conocer los bloques de función en TIA PORTAL que permiten realizar la comunicación por medio del protocolo Modbus RS485.

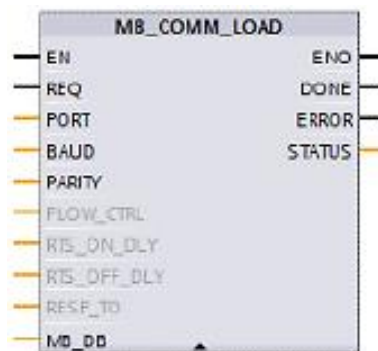
5.1.9.1 PLC Siemens S7 1200 1212C

Para utilizar el protocolo de comunicación Modbus RTU en este PLC, se requiere agregar un módulo de comunicaciones CM 1241 RS485.

Por otro lado, para programar la lectura y/o escritura de dispositivos modbus RTU RS485, hay instrucciones específicas que debe llevar el programa, motivo por el cual es importante tenerlas en cuenta antes de llevar a cabo cualquier tipo de programación [45]:

- **MB_COMM_LOAD:** Se utiliza para configurar al equipo para usar el protocolo Modbus RTU, viéndose reflejada en la instrucción **MB_MASTER** o **MB_SLAVE** (Figura 14).

Figura 14. Instrucción en TIA PORTAL para comunicación Modbus.



Fuente: Tomado de Controlador programable S7-1200, por Siemens

A continuación, en la tabla 3 se presenta la descripción de algunos parámetros de la instrucción:

Tabla 3. Parámetros de configuración de instrucción Modbus

Parámetro	Tipo de datos	Descripción
<i>REQ</i>	Bool	True activa el proceso.
<i>PORT</i>	Port	Número del puerto físico.
<i>BAUD</i>	UDInt	Velocidad de transmisión: 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 76800, 115200.
<i>PARITY</i>	UInt	Selección de paridad: 0 – Ninguna 1 – Impar 2 – Par
<i>DONE</i>	Bool	TRUE si no hay errores.
<i>ERROR</i>	Bool	TRUE si hay errores después de un ciclo.
<i>STATUS</i>	Word	Código de condición de ejecución

Fuente: Tomado de Controlador programable S7-1200, por Siemens

- **MB_MASTER:** Permite que el equipo se identifique como un maestro y se pueda comunicar con los esclavos Modbus (Figura 15).

Figura 15. Instrucción en TIA PORTAL para declaración de maestro Modbus



Fuente: Tomado de Controlador programable S7-1200, por Siemens

A continuación, en la tabla 4 se presenta la descripción algunos parámetros de la instrucción:

Tabla 4. Parámetros de configuración de maestro Modbus

Parámetro	Tipo de datos	Descripción
<i>REQ</i>	Bool	0 = Sin peticiones 1 = Petición de transmisión de datos
<i>MB_ADDR</i>	UInt	Dirección de estación Modbus RTU
<i>MODE</i>	USInt	lectura, escritura o diagnóstico.
<i>DATA_ADDR</i>	UDInt	Dirección inicial de memoria.
<i>DATA_LEN</i>	UInt	Cantidad de direcciones a leer.
<i>BUSY</i>	Bool	0 – No está siendo utilizado 1 – Está siendo utilizado.
<i>ERROR</i>	Bool	TRUE durante un ciclo tras haberse finalizado la última petición con un error
<i>STATUS</i>	Word	Código de condición de ejecución

Fuente: Tomado de Controlador programable S7-1200, por Siemens

Para más información, consultar el manual de especificaciones del equipo [45].

5.2 Estado del arte

En una revisión de la literatura y artículos científicos y técnicos publicados, se muestra que existen algunos trabajos previos sobre el monitoreo en las plantas solares.

Según el artículo publicado por [5], se propone la implementación de un sistema de monitoreo y detección de fallas en un sistema fotovoltaico, que utiliza sensores para medir variables meteorológicas, consumo energético, nivel de tensión y potencias de los dispositivos. Lo que se busca con el sistema es detectar posibles fallas o interrupciones en la generación de energía solar para prevenir interrupciones imprevistas y mejorar la eficiencia y confiabilidad del sistema fotovoltaico.

Por otro lado, [46] señala como se puede manejar el protocolo Modbus en conjunto con los servicios de IoT, por medio una arquitectura de interconexión entre dispositivos Modbus y una plataforma IoT basada en estándares oneM2M. La arquitectura utiliza una entidad proxy de interconexión (IPE) para vincular los dispositivos Modbus y la capa de servicio IoT, lo que puede ser útil para la implementación de sistemas de gestión de energía y otros sistemas industriales que utilizan dispositivos Modbus como un PLC y que necesiten integrarse con plataformas IoT.

La integración de plataformas Modbus e IoT para servicios industriales es significativa porque permite la interoperabilidad y la comunicación sin fisuras entre dispositivos Modbus y aplicaciones IoT. Modbus es un protocolo ampliamente utilizado en dispositivos electrónicos industriales, pero sus datos no pueden ser entendidos por aplicaciones IoT que utilizan protocolos diferentes. Mediante la integración de dispositivos Modbus con una plataforma de servicios IoT, las organizaciones industriales pueden conectar y gestionar sus dispositivos dentro de una red IoT.

Esta integración permite la supervisión en tiempo real, proporcionando capacidades mejoradas de automatización y control en el ámbito industrial.

Otro trabajo previo, mencionado por [47], se propone el establecimiento de los fundamentos para un sistema de monitoreo remoto de parques solares, en el cual debe establecer un bus de comunicaciones entre los dispositivos de medición y el Gateway. Además, se hace referencia a la visualización de las mediciones del parque solar fotovoltaico y al control a distancia del interruptor central, así como a la configuración del software PowerLogic ION Enterprise para exportar información hacia otros sistemas SCADA a través del protocolo de comunicación MODBUS RTU. El uso del sistema SCADA es muy importante porque es una tecnología que permite supervisar y controlar sistemas complejos, como parques solares, a través de la adquisición de información en tiempo real, la visualización de información y la generación de informes.

De forma similar, los autores [48] proponen implementación de un sistema de telemetría para una planta solar, siendo de gran importancia por varias razones. En primer lugar, se permite conocer de forma continua las variables del sistema fotovoltaico, tales como las curvas de los paneles solares, regulador y la carga de las baterías. Esto permite obtener información muy importante para generar estadísticas sobre el rendimiento y la eficiencia del sistema, lo que facilita la detección y actuación rápida sobre problemas o fallos y la implementación de procesos para corregir el error de manera oportuna.

Se compone por el Arduino Uno y el módulo de transmisión de datos por medio de una SIM y el protocolo GSM/GPRS. Este microcontrolador recopila la información leída por los sensores y enviarlos al servidor web a través del módulo de transmisión GSM/GPRS. Una vez allí,

los datos son procesados y utilizados según sea necesario en la plataforma Xively para su visualización.

Sin embargo, [49] muestra el desarrollo de un sistema de monitoreo industrial para plantas entre 3 y 50kW con el cual plantea establecer un plan de mantenimiento preventivo, aspecto importante para la correcta operación de una planta solar. Para lograr este objetivo, es necesario supervisar los parámetros del sistema, como corriente, voltaje y potencia, así como los parámetros ambientales, como radiación y temperatura. Los sistemas industriales disponen de una red wifi, lo que facilita mucho más el uso de un módulo wifi para la supervisión. Además, con el avance de las plataformas digitales y las redes, la conexión a internet es una propiedad casi indispensable para estos sistemas.

En otro artículo, [50] propone utilizar un PLC como pasarela de datos y un sistema SCADA basado en el protocolo Modbus TCP con el cual se pretende elaborar un sistema de monitoreo industrial para aplicaciones domésticas. El PLC actúa como el dispositivo de control y la pasarela de datos, mientras que el SCADA se utiliza para la visualización y el control de la configuración del dispositivo. El sistema propuesto permite la comunicación con otros dispositivos como VE. Bus, Can Bus y Modbus TCP. Sin embargo, se destaca el uso de Modbus TCP debido a su amplia adopción y capacidad para implementar sistemas SCADA de terceros y combinar dispositivos de diferentes proveedores. El sistema de monitoreo se basa en una estructura SCADA desarrollada utilizando el software TIA Portal WinCC Advanced 16 de SIEMENS. Este sistema permite la visualización de datos y el control de los ajustes del dispositivo. Además, se propone utilizar MS SQL Server para el registro de datos en diferentes formatos de archivo.

5.3 Fundamentos legales

El Acuerdo 1602, emitido por la Comisión de Regulación de Energía y Gas (CREG) en Colombia, es un documento normativo de vital importancia que establece los diferentes aspectos requeridos para conectar sistema de generación al Sistema de Interconexión Nacional (SIN) del país.

El propósito central de este acuerdo es asegurar la confiabilidad y seguridad del SIN al permitir la integración de sistemas de generación distribuida y renovable, como plantas solares fotovoltaicas y parques eólicos, en la red eléctrica nacional.

Para lograr estos objetivos, el Acuerdo 1602 establece requisitos específicos de protecciones que varían según la potencia nominal del sistema de generación (Consejo nacional de operación del sector eléctrico, CNO, 2018). Estos requisitos se dividen en dos categorías principales: para potencias menores y para potencias mayores a 250 kilovatios (kW).

- Menores o iguales a 0.25 MW: Al estar basados en inversores y mientras estén conectados a los niveles de tensión 1, 2 y pueden basarse en las protecciones que poseen los inversores.
- Mayores a 0.25 MW y menores a 5 MW: Deben utilizar dispositivos de protección para proteger la instalación eléctrica del generador, de acuerdo con las características de la red.

Es fundamental tener en cuenta esta normativa, porque en el caso de que el proyecto propuesto cumpla con estos requisitos, es necesario incluir en la programación la lectura y control de estas protecciones. Sin embargo, dado que el proyecto planteado es para el monitoreo de una planta de 100 kW, solo aplican las protecciones del inversor, tal como se indica para plantas menores a 250 kW.

6 Desarrollo del proyecto

En este proyecto, maneja una metodología de investigación cuantitativa de tipo experimental, debido a que se requiere una adquisición de datos que será probada por medio de simulaciones y dispositivos físicos. En el caso del proyecto de diseño de un sistema de monitoreo industrial para un parque solar de 100 kW, se aplicaron diversas etapas, como la revisión bibliográfica, el diseño experimental y la recopilación de datos.

Es fundamental utilizar herramientas como PLC y simuladores del protocolo Modbus para capturar información sobre la generación de energía del sistema fotovoltaico. Con base en los resultados obtenidos, se pueden extraer conclusiones y realizar recomendaciones o protocolos para mejorar el sistema de monitoreo.

La metodología elegida se ajusta a las necesidades del proyecto, debido a que implica una etapa inicial de investigación para comprender como crear máquinas de estados y los diferentes tipos de comunicaciones requeridos, la cual se ve reflejada en el marco teórico de este documento. Luego, se procede con la implementación de la máquina de estados finitos en el PLC y la programación del sistema bajo el protocolo de comunicación Modbus, siendo estos objetivos específicos abordados mediante la aplicación práctica de la investigación.

Además, permite la realización de simulaciones y pruebas con las cuales se puede obtener la retroalimentación necesaria para mejorar iterativamente el ciclo de investigación y desarrollo.

6.1 Materiales y componentes

Para desarrollar el proyecto de acuerdo con las etapas que son mencionadas más adelante, se requieren los siguientes materiales:

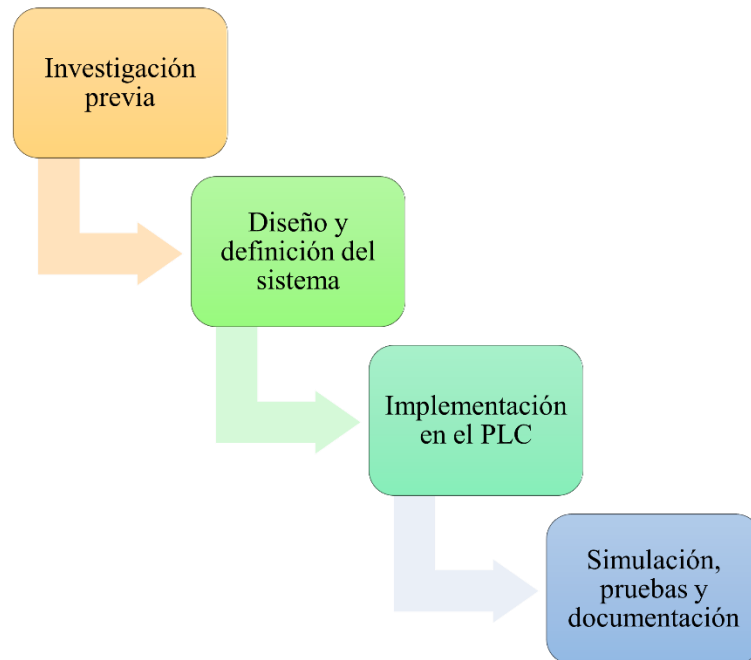
Tabla 5. Lista de materiales para el sistema de monitoreo

Equipos	Cantidad
Computador con mínimo 8 GB de memoria RAM, en donde se pueda instalar TIA PORTAL V17, MbSlave, Python.	1
Gabinete en Acero Cold Rolled 600x400x250 DKC	1
Interruptor Termomagnético 1X3A / BKN_b 1P C3	1
Interruptor Termomagnético 2X3A / BKN_b 2P C3	1
Fuente Sitop PSU 2,5A, 24VDC / 6EP1332-5BA00	1
CPU 1212C, DC/DC/Relé 8DIX6DO / 6ES7212-1HE40-0XB0	1
Scalance XB005 / 6GK5005-0BA00-1AB2	1
Módulo CM 1241, RS485, Macho / 6ES7241-1CH32-0XB0	1
Relé Enchufable Con Base - 115VAC, 8 Pines / LZS: RT4A4S15	6

Fuente: Elaborada por el autor

6.2 Metodología de desarrollo

Figura 16. Metodología de desarrollo de software aplicada



Fuente: Elaborada por el autor

Se plantea realizar el proyecto en 4 fases (Figura 16), las cuales se detallan a continuación:

6.2.1 Investigación previa

En esta fase se pretende adquirir y potenciar conocimientos específicos para llevar a cabo la implementación del sistema utilizando un controlador lógico programable (PLC).

Dentro de esta se realizan cursos e investigaciones sobre las diferentes variaciones del protocolo Modbus, como Modbus RTU, que utiliza una representación binaria de los datos enviados, y TCP/IP, que utiliza un formato de datos basado en texto. Además, se profundiza en el uso de las máquinas de estado finitas, que constituyen una herramienta útil para describir estados

y transiciones en sistemas y procesos que, para este proyecto, se aplica en el desarrollo del software en el PLC.

Esta fase se ve reflejada en el marco teórico de este documento y fue parte de la etapa de entrenamiento que la empresa proporcionó en el inicio de la pasantía empresarial.

6.2.2 Diseño y definición del sistema.

Para llevar a cabo el diseño del proyecto, luego de tener conocimientos más claros sobre las arquitecturas y topologías de red, así como del funcionamiento y operación del protocolo Modbus, se deben identificar los siguientes aspectos:

- ✓ **Marca y cantidad de inversores:** Esto es importante para establecer el mapa Modbus a la hora de implementar la programación.
- ✓ **Inversores centralizados:** De esto depende que el sistema de monitoreo utilice uno o más gabinetes de control.
- ✓ **Limitador de potencia:** Este sistema se encarga de controlar la cantidad de energía que fluye del sistema fotovoltaico hacia la red, limitando la cantidad de potencia que puede ser entregada en un momento dado. Por tanto, es importante preguntar si el proyecto requiere de un limitador.
- ✓ **Calidad de la señal en el lugar de instalación:** Este dato es importante porque permite evaluar si la señal es buena o mala, con lo cual se puede definir si se requiere usar GSM o si se tiene un punto de red fijo en el lugar.
- ✓ **Potencia instalada:** Muy importante para verificar si el proyecto cumple o no con el CNO.

- ✓ **Uso de planta diésel, mando motorizado o totalizador:** En ocasiones se requiere ejecutar acciones de control sobre estos dispositivos, por lo que es necesario conocer si aplica para el proyecto o no.
- ✓ **Estación meteorológica:** Se utiliza para monitorear las condiciones ambientales como temperatura ambiente, radiación solar y temperatura del panel con el fin de tomar decisiones y verificar el rendimiento de la planta solar.

Una vez organizada la reunión con el equipo de ingeniería, encargados de diseño del sistema fotovoltaico, se identificaron los requisitos del sistema de monitoreo:

Tabla 6. Requerimientos del sistema de monitoreo

Requisitos del sistema	Descripción
Cantidad de inversores	1
Marca de inversores	Huawei 100kW x1 100KTL-M1
Planta centralizada	Si
Potencia instalada	100 KW
Cumplimiento CNO	No
Estación meteorológica	Si
Totalizador, mando motorizado	No
Planta diésel	No
Calidad de la señal	Buena
Limitación de potencia	No

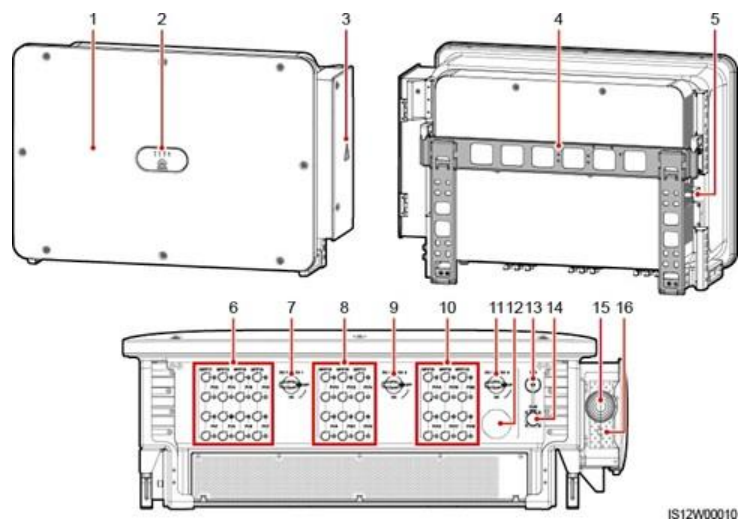
Fuente: Elaborada por el autor

Después de tener definidos los requisitos para diseñar el sistema de monitoreo, se procede a analizar las arquitecturas y topologías de redes con el fin de poder establecer la mejor opción para el sistema de monitoreo de este sistema solar fotovoltaico. Para ello es importante estudiar las hojas de datos de los equipos, es decir, del inversor Huawei y de la estación meteorológica, de manera que se comprenda como se puede establecer la comunicación entre estos y el PLC.

6.2.2.1 Consideraciones del inversor Huawei

Por un lado, antes de comenzar a planear la arquitectura a utilizar en el sistema de monitoreo, se debe conocer el dispositivo que se va a utilizar, siendo en este caso un inversor Huawei 100KTL-M1.

Figura 17. Conexiones del inversor



Fuente: Tomado de SUN2000-(100KTL, 110KTL, 125KTL) Series User Manual, por Huawei

En un primer momento, se revisan las conexiones del equipo con el fin de identificar el puerto de comunicaciones y comprobar que el equipo se pueda comunicar por medio del protocolo

modbus, así como el tipo de bornera o clavija que se utiliza. En la figura 17, se identifica el puerto de comunicaciones correspondiente al puerto 14, puerto que soporta la comunicación RS485, lo que significa que sí se puede realizar la comunicación mediante el protocolo Modbus.

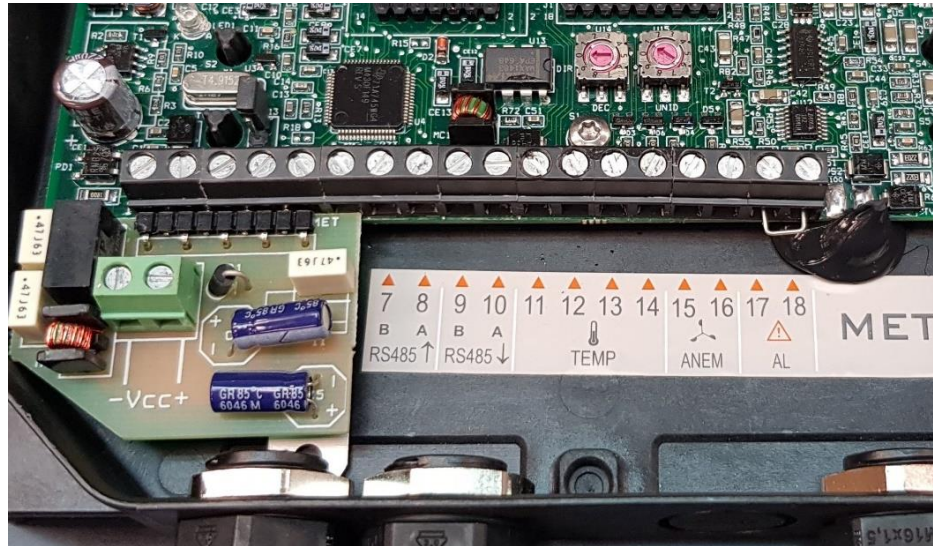
Como se sabe, modbus requiere establecer un parámetro para la transmisión de datos llamado baudios. En el manual del inversor se menciona que se pueden configurar a 4800, 9600 y 19200. Este es un dato muy importante porque al diseñar el sistema se debe garantizar que todos los dispositivos involucrados se encuentren configuración con la misma velocidad de transmisión, de lo contrario los dispositivos del sistema no se podrán comunicar entre sí.

También se indica que, para evitar problemas en la velocidad de respuesta de la comunicación, se recomienda usar menos de 30 inversores en la conexión a los puertos de comunicación y, que la distancia entre estos no exceda los 1000 m. La importancia de tener esto en cuenta radica en el hecho de que, en algún proyecto, por ejemplo, en gran escala, puede que se requieran conectar una gran cantidad de inversores, estos no deben exceder esta cantidad de inversores debido a que el protocolo de comunicación puede presentar fallos. Sin embargo, para el proyecto en cuestión, al tratarse de un solo inversor, no existe la necesidad de realizar la conexión a más dispositivos aparte del PLC en el gabinete de control. Por lo tanto, solo se debe considerar la distancia del inversor hasta el gabinete.

6.2.2.2 Consideraciones de la estación meteorológica MET ATERSA

Por otro lado, es importante conocer también algunas especificaciones técnicas asociadas a la comunicación de la estación meteorológica que se va a utilizar en el proyecto que, para este caso, es una ATERSA MU38S.

Figura 18. Conexiones de la estación meteorológica.



Fuente: Tomado de Datasol MET Manual de Operación y de instalación, por Atersa

Como se puede observar en la figura 18, la estación meteorológica cuenta con un bus de comunicaciones RS485 In/Out, así como las debidas conexiones para el sensor de temperatura, anemómetro y estados de alarmas. Sin embargo, para el proyecto solo se requiere el puerto de comunicaciones y la bornera para el sensor de temperatura, debido a que no se contempla el uso de un anemómetro para este sistema de monitoreo.

Tabla 7. Especificaciones de comunicaciones de la estación meteorológica.

Especificación	Descripción
Estándar	RS485
Protocolo de comunicaciones	Modbus RTU
Velocidad de transmisión	9.600 / 19.200 / 38.400 / 57.600 / 115.200 bps

Distancia máxima	500 m
Modo	Esclavo.
Máximo número de nodos.	32

Fuente: Tomado de Datasol MET Manual de Operación y de instalación, por Atersa

6.2.2.3 Análisis de las topologías de comunicaciones del sistema

De acuerdo con lo mencionado anteriormente, se tiene que la comunicación se puede realizar mediante el protocolo modbus, en donde se configuren los equipos a la misma velocidad de transmisión (Baudios) y se asigne a cada uno un ID único y diferente de 0, preferiblemente en orden numérico para que hacer más fácil el mantenimiento y revisión de los equipos a futuro.

En lo que se refiere a la topología a utilizar en el sistema de monitoreo, al conocer el funcionamiento y características de operación del protocolo modbus para esta aplicación en particular, se tienen las siguientes consideraciones:

Si se analiza, por una parte, si el cliente requiere de una expansión de la planta, se debe prever que la topología que se decida utilizar, es decir, la conexión física de los inversores sea viable para agregar más dispositivos sin presentar inconvenientes, que permita establecer comunicación punto a punto o punto a multipunto y, sea adecuado para el uso del protocolo Modbus.

A partir de estas consideraciones, y de las topologías de red existentes, se analiza cuáles son aptas para utilizar Modbus RTU y cuáles no:

- **Topología de bus:** Los dispositivos comparten un único medio de comunicaciones, lo que facilita el cableado de los equipos. Además, es ideal para utilizarla con el protocolo

Modbus, porque este es un protocolo que transmite los datos de forma serial, y de acuerdo con la distribución física que ofrece esta topología, se encuentra la conexión serial entre los dispositivos [54].

Otro aspecto que le da un punto positivo a esta topología es que permite comunicar los dispositivos punto a punto o punto a multipunto. Esto se logra mediante el direccionamiento de los dispositivos, asignando direcciones únicas a cada uno y, por tanto, solicitando información a un dispositivo a la vez.

Por supuesto, también supone algunas desventajas como limitaciones en las distancias o en el hecho de que, al estar en serie, si algún dispositivo presenta una falla, los otros dispositivos de la red se van a ver afectados.

- **Topología de Estrella:** Esta topología puede funcionar con Modbus RTU si se utiliza un nodo central, como un convertidor de medios o un controlador de comunicación para dirigir el tráfico de solicitudes y recepciones entre los dispositivos. En esta topología, el nodo central, que en este caso puede ser el PLC, es el que administra la comunicación punto a multipunto entre los dispositivos, lo que lo hace viable para que Modbus RTU funcione [55].

Otro aspecto que le da cierta ventaja es el hecho de que en esta topología la falla de un dispositivo no va a afectar toda la comunicación del sistema, por lo que el nodo central puede gestionar otras rutas alternativas siempre y cuando se hayan considerado.

- **Topología tipo malla:** Esta topología no es apta para crear una red Modbus RTU debido a su nivel de complejidad, en donde cada dispositivo se conecta a múltiples dispositivos,

lo que genera en cada punto conexiones multipunto y, causar una posible redundancia en la comunicación.

Si bien, la redundancia en las comunicaciones es buena, siempre y cuando se tengan rutas alternativas de comunicaciones, pero en el caso de Modbus RTU, se pueden presentar colisiones de datos, retrasos en la entrega de mensajes y pérdida de información. En otras palabras, resulta una estructura compleja para lo que Modbus realmente necesita para operar.

- **Topología en anillo:** Esta topología, aunque ofrece una alta redundancia en la comunicación, puede ser demasiado para las necesidades del protocolo Modbus. Es cierto que, si uno de los dispositivos falla, la dirección en la que se hace la comunicación puede cambiar, pero si llega a fallar otro dispositivo se tendría una pérdida total de la comunicación. Además, para que funcione de esta manera supone una serie de configuraciones y gestiones para sincronizar los mensajes, por lo que realmente no es algo que modbus requiera y tampoco algo que lo favorezca.
- **Topología en estrella extendida y árbol:** Como se analizó previamente, la topología estrella puede funcionar para aplicar el protocolo modbus, y en este sentido la topología de árbol, por la forma en la que se distribuyen los dispositivos, también puede ser una opción.

Pero modbus es un protocolo que ofrece simplicidad y, en este caso, si se enfoca en el análisis para el proyecto que se busca desarrollar, estas dos topologías son complejas para esta aplicación, lo supone más dispositivos de los necesarios y, por tanto, no son topologías que beneficien la implementación con este protocolo.

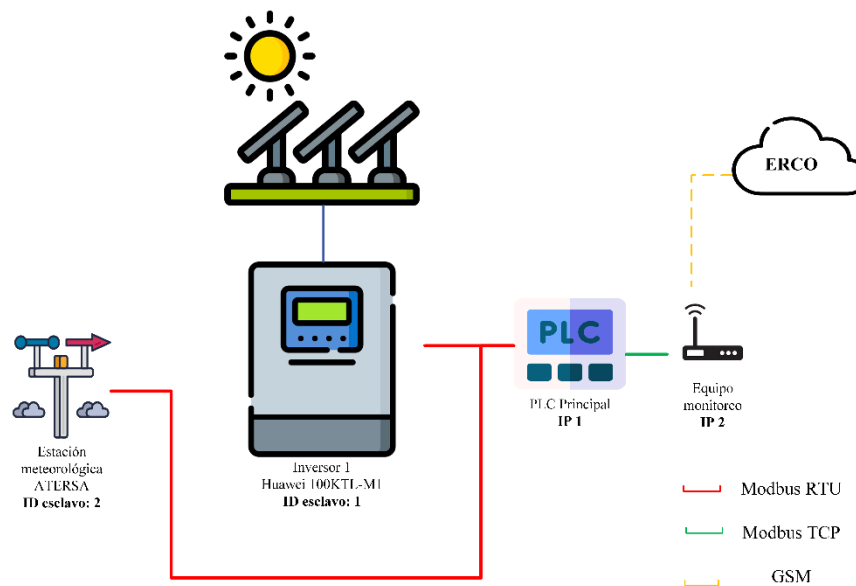
Es importante aclarar, que la topología tipo árbol se puede usar cuando se requieren varios PLC, por ejemplo, cuando la planta solar no se encuentre centralizada o cuando se tengan varias etapas,

Considerando lo mencionado previamente, en el contexto de la automatización industrial y para la aplicación que se busca desarrollar en este documento, correspondiente a un sistema de monitoreo para un sistema solar fotovoltaico, y de acuerdo a las características vistas sobre el protocolo Modbus, en particular en su configuración RTU RS485, se determina que la topología más viable en términos de conexiones, facilidad de pasar de comunicación de punto a punto a punto multipunto y eficiencia, es la tipología tipos bus para interconectar los inversores y el PLC. La estación meteorológica se puede conectar en paralelo a estos equipos, lo que formaría también una topología tipo estrella. En resumen, se puede tener una topología mixta.

Ahora bien, en la definición del protocolo se habla de que su arquitectura de comunicación es maestro-esclavo, pero como el sistema requiere enviar datos a la nube por medio de internet o GSM, se requiere también una arquitectura cliente-servidor debido a que el PLC se debe interconectar con un equipo de monitoreo encargado de enviar los datos a la plataforma de la empresa para lo cual se requiere usar de modbus TCP/IP, motivo por el cual es necesaria esta arquitectura.

6.2.2.4 Definición de la comunicación del sistema

Figura 19. Diagrama general de comunicaciones



Fuente: Elaborada por el autor

A partir del análisis realizado, se propone un diagrama que muestra la arquitectura comunicaciones (Ver figura 19) en donde de forma gráfica y sencilla se presentan las conexiones entre los dispositivos modbus del sistema.

Como se menciona, la empresa usa un equipo de monitoreo para leer los datos del PLC y enviarlos a la nube de la empresa para ser tratados según se requiera. Por este motivo, se requiere también del uso de Modbus TCP/IP y, adicionalmente, una conexión inalámbrica para el envío de datos de GSM, lo que implica la ubicación de una antena para establecer la comunicación entre diversos puntos.

En esta etapa es importante asignar el ID a cada dispositivo, recordando que para los dispositivos esclavos que se leen por medio de Modbus RTU se les asigna un número entero

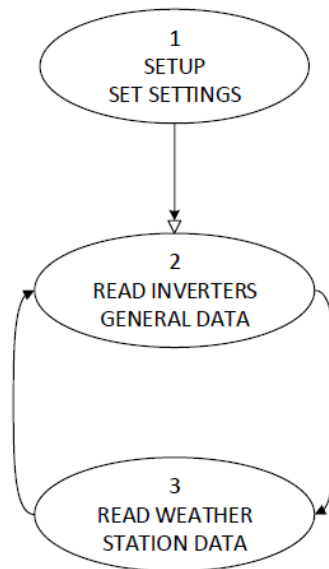
diferente de cero y, para mayor orden, de forma ascendente. De esta manera el inversor queda con ID 1 y la estación meteorológica con ID 2. Para el caso de los dispositivos que se comunican por medio de Modbus TCP/IP, se le asigna un ID y una IP a cada dispositivo, recordando que deben estar definidas en el mismo segmento de red.

6.2.3 Implementación en el PLC

Teniendo claro la arquitectura y la topología a utilizar en el sistema de monitoreo, mostrada en el diagrama de la figura 19, se puede diseñar la máquina de estados finitos con la cual se establece la lógica del programa en el PLC (Figura 20). Este enfoque facilita el diseño y establecimiento del orden en que se va a ejecutar el programa, es decir, en que orden va a solicitar la información de los dispositivos.

Para su desarrollo, se definan los estados que se quieren realizar para cumplir con la lectura de los dispositivos siendo estos: Configuración de parámetros Modbus, lectura de datos de inversores y lectura de estación meteorológica. Esta técnica de programación no requiere de un gran nivel de detalle, puesto que su propósito es ayudar a identificar y organizar el programa para que sea más fácil desarrollarlo y comprenderlo.

Figura 20. Diagrama de máquina de estados finitos



Fuente: Elaborada por el autor

Con esto en mente, se realiza el código de programación, para lo cual es importante conocer las zonas de memoria del dispositivo, es decir, las direcciones que se deben consultar para obtener la información deseada. Estas direcciones de memoria dependen de los fabricantes, por lo que el inversor tendrá direcciones diferentes a la estación meteorológica.

De acuerdo con los datos que se requieren para la plataforma, las direcciones de memoria son:

- Inversor: dirección 32008 y 119 registros.
- Estación meteorológica: 96 y 3 registros.

En el anexo 1 se pueden observar las tablas con las direcciones de memoria que se van a leer para ambos equipos.

6.2.3.1 Programación realizada

Debido a que este programa fue realizado a partir de lo proporcionado por la empresa, no se puede presentar todo el código realizado porque es material confidencial que solo se utiliza para los proyectos. Motivo por el cual se presentan algunas partes en donde se pretende mostrar la relevancia del uso del protocolo Modbus el cual es la base del sistema de monitoreo.

Figura 21. Variables y constantes del programa

	Name	Data type	Value
■	INITIAL_STATE	UInt	1
■	SETUP_STATE	UInt	1
■	READ_INVERTER_GENERAL_DATA_STATE	UInt	2
■	READ_WHEATER_STATION_STATE	UInt	3
■	MODBUS_RTU_BAUDRATE	UInt	9600
■	MODBUS_MODE_R	Int	0
■	INVERTER_FIRST_ID	UInt	1
■	INVERTER_LAST_ID	UInt	1
■	INVERTER_TOTAL_REGISTERS	UInt	119
■	INVERTER_REGISTER_ADDRESS_INIT	DInt	432009
■	WEATHER_STATION_ID	UInt	2
■	WHEATER_STATION_REGISTER_ADDRESS_INIT	UDInt	40097
■	WHEATER_STATION_TOTAL_REGISTERS	UInt	12
■	RECONNECTION_TIMEOUT	DInt	10000

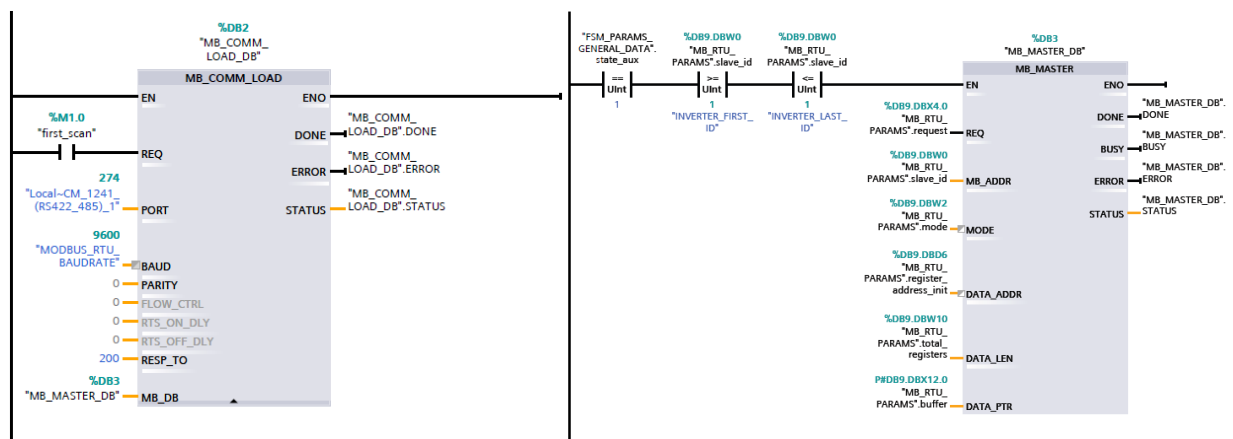
Fuente: Elaborada por el autor

En un primer momento, se crean las variables y constantes necesarias para el funcionamiento del programa. En esta se definen los estados establecidos en la máquina de estados finitos, el ID de los dispositivos, las direcciones de memoria, entre otros, tal y como se muestra en la figura 21. Se debe considerar que en donde se define la dirección inicial de memoria, se debe agregar la función con la cual se va a leer el dispositivo. En este caso, los dispositivos se leen con función 3, pero en TIA PORTAL se debe poner un 4 seguido de la dirección. Aplica de la misma manera para la estación meteorológica. Y, la dirección que se asigne se debe ingresar con un offset

de 1 con el fin de leer adecuadamente las posiciones deseadas, es decir que, si se quiere leer desde la posición 32008, en TIA PORTAL se debe poner la dirección 32009.

Una vez declaradas, se configuran las instrucciones en TIA PORTAL que van a permitir establecer la comunicación con los dispositivos. Estos se programan en bloques de funciones, en donde cada bloque corresponde a un estado de la máquina de estados finitos mencionada. Cabe resaltar que, en estas instrucciones, como por ejemplo *MB_MASTER*, se logra configurar y armar la trama de datos para la solicitud modbus a partir de las constantes mencionadas en la figura 21. Esta trama de solicitud es diferente tanto para el inversor como para la estación meteorológica por lo que, en cada bloque de función, se debe crear la instrucción configurando la trama que corresponda a cada dispositivo.

Figura 22. Configuración de parámetros modbus



Fuente: Elaborada por el autor

Como se observa en la configuración de la instrucción de *MB_MASTER*, los datos se van a almacenar en un buffer cada vez que se realice la lectura de datos, de manera que se crea un

register de datos para poder visualizarlo de forma separada, es decir, que se puedan ver un array de datos por cada dispositivo modbus y poder verificarlos de acuerdo con el dispositivo.

Cuando se tienen creados todos los cloques de función necesarios, se agregan en el *Main* junto con la instrucción de Modbus TCP/IP y se compila para verificar que no existan errores en las diferentes declaraciones de las instrucciones. El siguiente paso es simular.

6.2.4 Simulación, pruebas y documentación del sistema

6.2.4.1 Configuración del simulador Modbus

Una vez se tiene la programación del PLC en el TIA PORTAL, se crean las simulaciones con programas como MbSlave. Este programa permite simular la memoria de los dispositivos modbus, lo que facilita la emulación de los datos recibidos en el PLC provenientes de los inversores (Figura 23 (a)) y de la estación meteorológica (Figura 23 (b)).

Figura 23. Configuración del simulador

The image shows two side-by-side screenshots of the 'Slave Definition' dialog box in a software application. Both windows show the same fields: Slave ID, Function (03 Holding Register (4x)), Address, Quantity, View options (Rows, Hide Alias Columns, PLC Addresses (Base 1)), Display (Signed), and Error Simulation options (Skip response, Insert CRC/LRC error, Return exception 06, Busy). In the left window (a), Slave ID is 1, Address is 32008, Quantity is 119, and 100 rows are selected. In the right window (b), Slave ID is 2, Address is 96, Quantity is 12, and 20 rows are selected.

(a) Inversor

(b) estación meteorológica

Fuente: Elaborada por el autor

MbSlave y ModSim ofrecen la posibilidad de seleccionar diferentes funciones que componen el protocolo (como por ejemplo las funciones de lectura y escritura) para simular el flujo de información. En esta configuración es importante establecer correctamente las direcciones de memoria y la cantidad de registros de manera que coincidan con la programación en TIA PORTAL, de lo contrario, la lectura por modbus no va a funcionar debido a que el PLC va a intentar leer posiciones que no existen. Por eso es importante prestar atención a la configuración del simulador. Para la lectura de los dispositivos Modbus se requiere el uso de la Función 03 (Read Holding Registers) [52]. Los datos que se esperan leer dentro del rango de direcciones establecidas son: voltajes y corrientes en AC y DC, energía total generada, estado de operación, irradiancia, temperatura ambiente, temperatura del panel.

Lo anterior se evidencia en la figura 23. Lo que procede es ingresar los valores en cada posición de memoria según corresponda, para que al realizar el envío de los datos simulados a la plataforma se puedan visualizar gráficamente.

Tabla 8. Datos de ejemplo para la simulación

72	Current Ia [0.001A]	6000		
73		0		
74	Current Ib [0.001A]	7000		
75		0		
76	Current Ic [0.001A]	8000		
77		0		
78		0		
79		0		
80	Active Power [0.001kW]	5000		
81		0		
82	Reactive Power [0.001kvar]	1000		
83		0		
84	Power Factor [0.001]	1000		
85	Frequency [0.01Hz]	6000		
86		0		
87	T. panel [0.1°C]	552		
88		0		
89	Device status	512		

2				0
3				0
4				0
5				0
6			T. ambiente [°C]	15
7			T. panel [°C]	25
8				
9				
10				
11				
12				
13				
14				
15				
16	Radiacion medida [W/m2]	1000		
17		0		
18		0		
19		0		

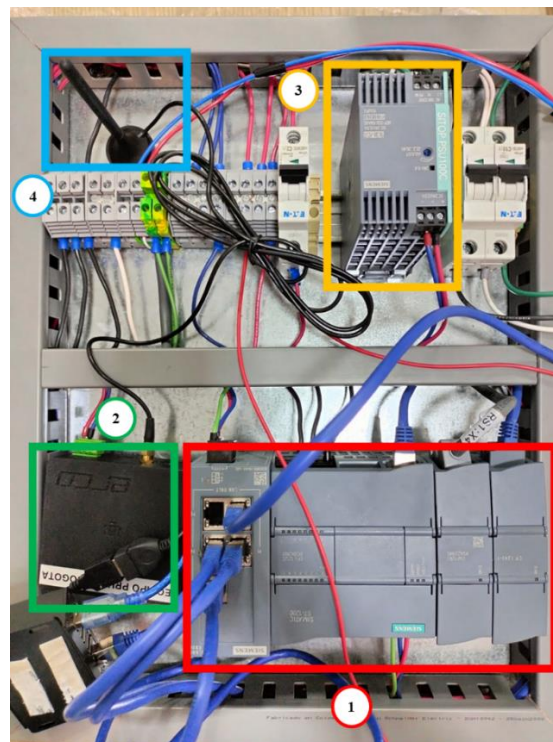
Fuente: Elaborada por el autor

Debido a la gran cantidad de datos, considerando que se deben leer 119 registros, solo se muestra una parte y se pueden simular los valores que se deseen. Si se quiere una mayor exactitud en la simulación, es importante tener en cuenta las unidades en la que los dispositivos entregan la información.

6.2.4.2 Pruebas con el PLC físico

Una vez se crean las simulaciones de los inversores y de la estación meteorológica en MbSlave, se pueden ejecutar las pruebas con el PLC S71200 en físico por medio de una conexión serial. Y a su vez, se puede establecer la conexión con la estación de monitoreo, la cual enviará los datos a la plataforma web de ERCO Energía.

Figura 24. Gabinete de control de prueba



Fuente: Elaborada por el autor

En la figura 24 se observa el gabinete de control con el que se realizan las pruebas necesarias en conjunto con el simulador en el computador. En este se observan el PLC Siemens S7 1200 1212C con el módulo de comunicación RS485, el switch Ethernet (1), el equipo de monitoreo NEU+ (2), la fuente 24 V SITOP (3) y la antena de comunicaciones (4); el resto de los materiales son los mencionados en la tabla 5.

Esta prueba es crucial, debido a que permite comprobar la lectura de los inversores mediante el protocolo Modbus desde el PLC, almacenando correctamente los datos obtenidos y ajustándolos para su posterior envío a la estación.

Figura 25. Mapa modbus equipo de monitoreo

*ID:1,data:8-
119,TP_I:7,"OS":89,"F1":8,"F2":9,"F3":10,"F4":118,"GEN":106.1.00,"PF":84.3.000,"PS":80.1.3.000,"AP":82.1.3.000,"VDC_1":16.0,"VDC_2":18.0,"VDC_3":20.0,"VDC_4":22.0,"VDC_5":24.0,"VDC_6":26.0,"VDC_7":28.0,"VDC_8":30.0,"VDC_9":32.0,"VDC_10":34.0,"VDC_11":36.0,"VDC_12":38.0,"VDC_13":40.0,"VDC_14":42.0,"VDC_15":44.0,"VDC_16":46.0,"VDC_17":48.0,"VDC_18":50.0,"VDC_19":52.0,"VDC_20":54.0,"IDC_1":17.3.00,"IDC_2":19.3.00,"IDC_3":21.3.00,"IDC_4":23.3.00,"IDC_5":25.3.00,"IDC_6":27.3.00,"IDC_7":29.3.00,"IDC_8":31.3.00,"IDC_9":33.3.00,"IDC_10":35.3.00,"IDC_11":37.3.00,"IDC_12":39.3.00,"IDC_13":41.3.00,"IDC_14":43.3.00,"IDC_15":45.3.00,"IDC_16":47.3.00,"IDC_17":49.3.00,"IDC_18":51.3.00,"IDC_19":53.3.00,"IDC_20":55.3.00,"VAC_1":66.0,"VAC_2":67.0,"VAC_3":68.0,"IAC_1":72.1.3.000,"IAC_2":74.1.3.000,"IAC_3":76.1.3.000,"TI":87.0,"F":85.00
ID:2,data:0-11,TP_I:5,"IRR":0,"TP":11.0,"TE":10.0*

Fuente: Elaborada por el autor

El equipo de monitoreo lee los datos del PLC y los almacena en un mapa modbus con el formato mostrado en la figura 25, en donde se tiene un mapa para el inversor y uno para la estación meteorológica, debidamente identificados con el ID que fue asignado desde el diagrama general de comunicaciones, para luego ser agrupados en un archivo JSON y se enviados a la nube por GSM. Este equipo se basa basado en una Raspberry pi y programación Python. Si los datos se

envían correctamente y se actualizan en la plataforma web con el usuario de prueba, se considera que el sistema de monitoreo funciona correctamente. Esto se evidencia en la sección de resultados.

6.2.4.3 Manual de instalación

Una vez se tiene certeza del funcionamiento del sistema, se elabora un manual de instalación del gabinete de control para que el equipo técnico tenga a la mano una herramienta que le permita comprender cómo se deben realizar las conexiones entre dispositivos. Además de algunos detalles que conexión en el gabinete de control. Este documento se presenta en el anexo.

7 Resultados

7.1 Datos simulados en el PLC

Una vez terminado el programa en TIA PORTAL y cargado en el PLC, se configuran las simulaciones como se muestra previamente y, se configura el equipo de monitoreo con el mapa modbus y otros aspectos importantes para realizar el envío de datos a la base de datos de la empresa.

Al poner en modo online el TIA PORTAL, para ver el comportamiento del programa, lo primero que se hace es verificar que en la instrucción del *MB_MASTER*, tanto para el inversor como para la estación meteorológica, se encuentre operando correctamente, es decir, que el *ERROR* se encuentre en *FALSO* y el *STATUS* en *16#0000* lo que indica que la comunicación Modbus se ejecuta de forma correcta. Esto de acuerdo con lo mostrado en la tabla 4.

Figura 26. Verificación de datos leídos.

	Name	Data type	Offset	Start value	Monitor value	Retain
67	INVERTER_1[64]	UInt	128.0	0	0	☐
68	INVERTER_1[65]	UInt	130.0	0	4750	☐
69	INVERTER_1[66]	UInt	132.0	0	4760	☐
70	INVERTER_1[67]	UInt	134.0	0	4800	☐
71	INVERTER_1[68]	UInt	136.0	0	6210	☐
72	INVERTER_1[69]	UInt	138.0	0	6200	☐
73	INVERTER_1[70]	UInt	140.0	0	6220	☐
74	INVERTER_1[71]	UInt	142.0	0	0	☐
75	INVERTER_1[72]	UInt	144.0	0	402	☐
76	INVERTER_1[73]	UInt	146.0	0	0	☐
77	INVERTER_1[74]	UInt	148.0	0	409	☐
78	INVERTER_1[75]	UInt	150.0	0	0	☐
79	INVERTER_1[76]	UInt	152.0	0	399	☐
80	INVERTER_1[77]	UInt	154.0	0	0	☐
81	INVERTER_1[78]	UInt	156.0	0	0	☐
82	INVERTER_1[79]	UInt	158.0	0	0	☐
83	INVERTER_1[80]	UInt	160.0	0	0	☐
84	INVERTER_1[81]	UInt	162.0	0	0	☐

	Name	Data type	Offset	Start value	Monitor value	Retain
1	Static					☐
2	INVERTER_1	Array(0..199) of UInt	0.0			☐
3	WHEATER_STATION_1	Array(0..199) of UInt	400.0			☐
4	WHEATER_STATION_1	UInt	400.0	0	23	☐
5	WHEATER_STATION_1	UInt	402.0	0	0	☐
6	WHEATER_STATION_1	UInt	404.0	0	0	☐
7	WHEATER_STATION_1	UInt	406.0	0	0	☐
8	WHEATER_STATION_1	UInt	408.0	0	0	☐
9	WHEATER_STATION_1	UInt	410.0	0	0	☐
10	WHEATER_STATION_1	UInt	412.0	0	0	☐
11	WHEATER_STATION_1	UInt	414.0	0	0	☐
12	WHEATER_STATION_1	UInt	416.0	0	0	☐
13	WHEATER_STATION_1	UInt	418.0	0	0	☐
14	WHEATER_STATION_1	UInt	420.0	0	15	☐
15	WHEATER_STATION_1	UInt	422.0	0	31	☐
16	WHEATER_STATION_1	UInt	424.0	0	0	☐
17	WHEATER_STATION_1	UInt	426.0	0	0	☐
18	WHEATER_STATION_1	UInt	428.0	0	0	☐

(a) Inversor

(b) estación meteorológica

Fuente: Elaborada por el autor

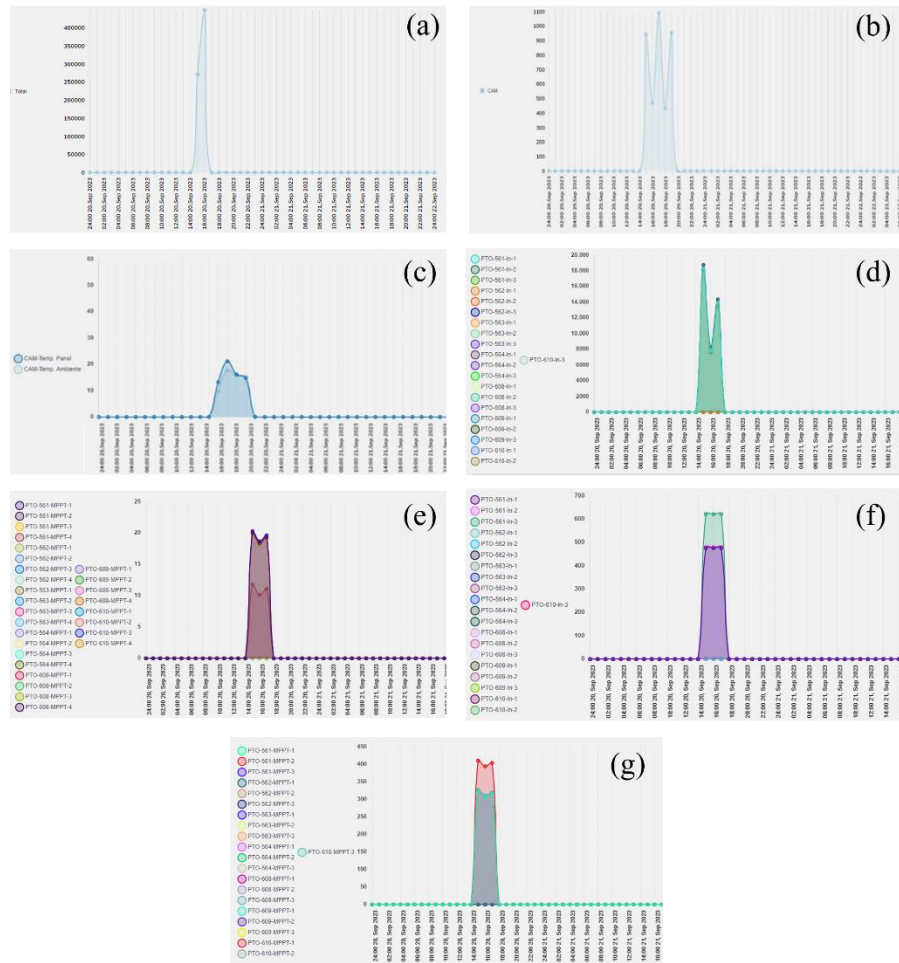
Luego, se verifica en el register del PLC los datos leídos en las direcciones de memorias especificadas. En la figura 26 se muestran algunos registros obtenidos de la simulación creada en

MbSlave que, en la imagen mostrada (a), corresponden a los registros de corrientes y voltajes AC para el inversor y, en (b) corresponden a radiación medida, temperatura ambiente y temperatura de panel para la estación meteorológica.

Cuando se verifican la variación de estos registros, es decir, que no se haya presentado un error en el tráfico de datos o que se haya bloqueado la conexión serial entre el simulador y el PLC, se procede a configurar el equipo de monitoreo con el mapa modbus mostrado previamente. Se ejecuta el código del equipo de monitoreo y se envían los datos a la plataforma.

7.2 Datos en plataforma

Figura 27. Datos visualizados en la plataforma



Fuente: Elaborada por el autor

El propósito del sistema de monitoreo es obtener los datos del sistema solar fotovoltaico para enviarlos a la plataforma web que es administrada por otra área de la empresa. Sin embargo, para dar por finalizado el sistema de monitoreo, se debe verificar que los datos se visualicen en la página. Como se ha mencionado previamente, estos datos corresponden a: (a) energía total

generada, (b) irradiancia, (c) temperatura del panel y temperatura ambiente, (d) corriente AC, (e) corriente DC, (f) tensión AC y, (g) tensión DC, los cuales se presentan en la figura 27.

Se recalca que estos datos no son reales sino simulados, por lo que sus valores pueden no ser cercanos a la realidad. Pero es un acercamiento a cómo se verían visualizados gráficamente.

7.3 Puesta en marcha en sitio

Después de terminadas las simulaciones, se da la posibilidad de poner en marcha en proyecto de forma remota, debido a cumplimientos de cronograma del proyecto. Sin embargo, también se da la oportunidad de conocer el proyecto en persona y hacer las respectivas revisiones de funcionamiento, de esta manera, se visita la Clínica Norte en Cúcuta en donde se conocen las instalaciones. El sistema solar es de 100 kW instalados y se compone por 215 paneles solares ubicados en el techo de la clínica, tal y como se observa en la figura 28.

Figura 28. Sistema solar fotovoltaico por monitorear.

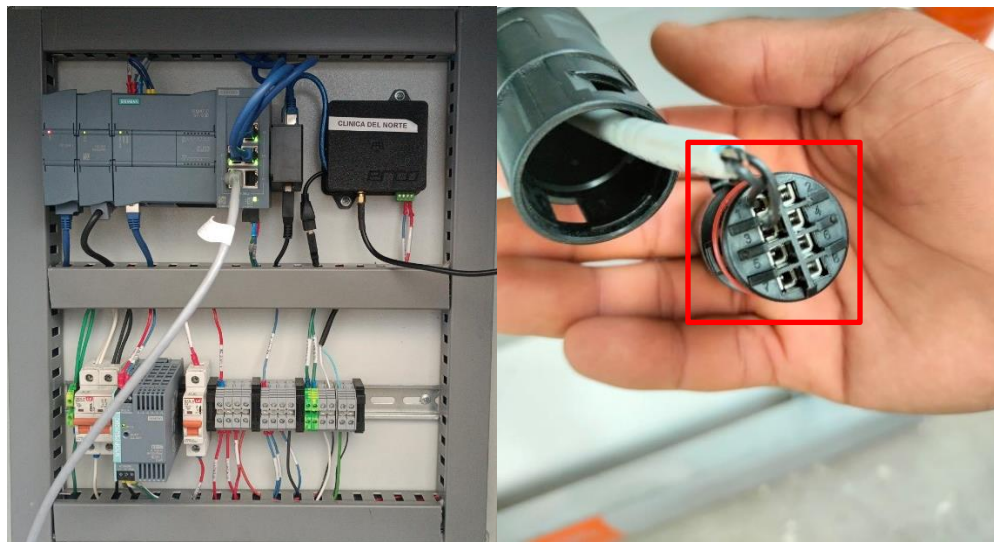


Fuente: Tomada por Erco Energía S.A.S

Para revisar el funcionamiento del sistema de monitoreo, se revisa el cuarto eléctrico en donde se encuentra el gabinete de control y el inversor solar (Figura 30).

Se verifica que la clavija de comunicaciones se encuentre bien conectada, es decir, que los dos cables que permiten que se comuniquen por modbus sean correctos lo cual se revisa en el manual de conexiones [52] y, que las conexiones en el gabinete también se encuentren bien.

Figura 29. Gabinete de control y conexión de la clavija de comunicaciones.

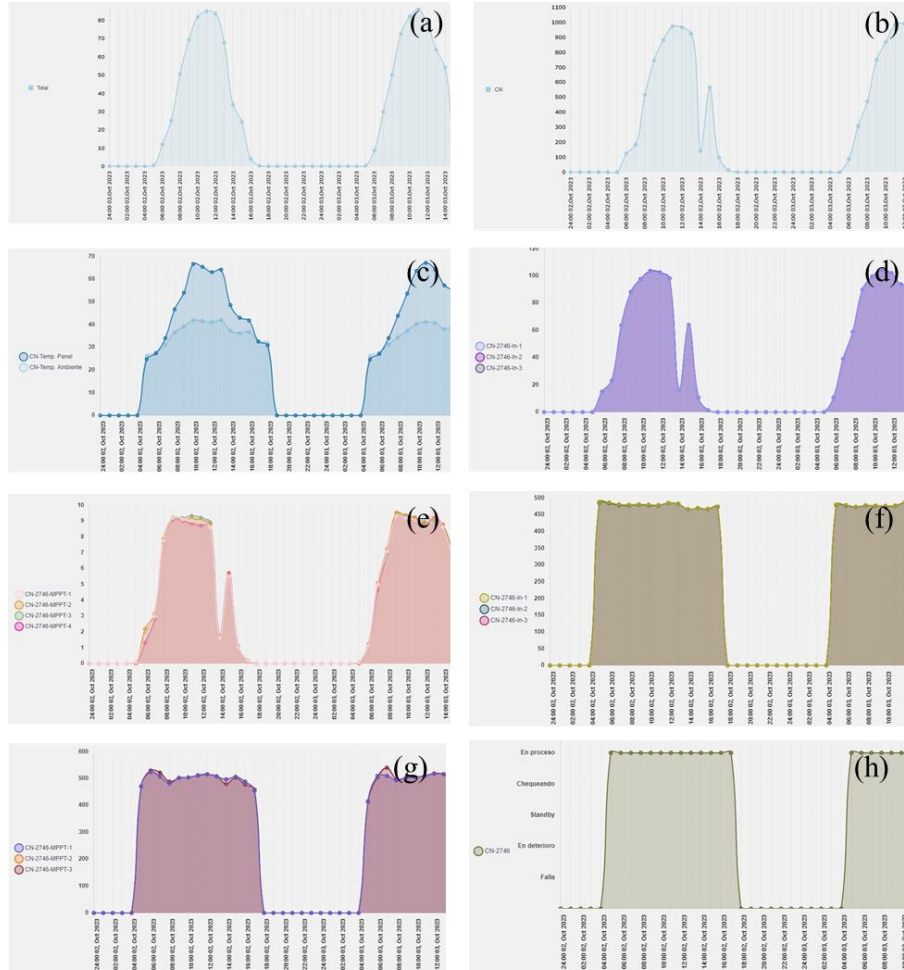


Fuente: Elaborada por el autor

Luego, se ingresa al PLC y al equipo de monitoreo para verificar los datos, de manera que se deja el sistema operando para obtener los datos y verificar que todo se encuentre operando para posteriormente poder hacer entrega del proyecto.

A continuación, se muestran datos de la plataforma después de tener un monitoreo constante del sistema:

Figura 30. Monitoreo del sistema instalado.



Fuente: Elaborada por el autor

Estos datos corresponden a: (a) energía total generada, (b) irradiancia, (c) temperatura del panel y temperatura ambiente, (d) corriente AC, (e) corriente DC, (f) tensión AC, (g) tensión DC, (h) estados de operación, todos siendo datos reales tomados directamente del proyecto en la Clínica Norte.

8 Actividades desarrolladas en la pasantía

8.1 Acompañamiento a proyectos

Después de desarrollar y ejecutar el proyecto mencionado en este documento, se realizaron programas para diferentes proyectos, los cuales cuentan con requerimientos diferentes. Algunos con más inversores y diferentes marcas, proyectos que cuentan con limitación de potencia, con acciones de control por planta diésel, totalizadores y contactores. Todo esto manteniendo como base el uso del protocolo Modbus en sus dos variaciones.

8.2 Desarrollo de herramientas para operación y mantenimiento

También se desarrolla un programa para testear el correcto funcionamiento del PLC tanto para antes de instalar el gabinete de control en sitio como para cuando se requiera hacerle mantenimiento al equipo. Este programa se basa en un web Server por lo que contiene una interfaz gráfica que permite ejecutar la activación o apagado de entradas y salidas, así como la verificación de modbus desde la web. Aún no se implementa puesto que se continua con su desarrollo y mejora.

Además, se realiza la documentación de las guías de instalación para los gabinetes debido a que en cada proyecto se manejan equipos o requerimientos diferentes, lo que implica que las conexiones pueden variar. Esto resulta de ayuda y base para soporte técnico tanto para el equipo instalador como para el equipo de mantenimiento.

9 Conclusiones

9.1 Aspectos por mejorar

Este sistema de monitoreo tiene muchas mejoras, tanto en la parte de programación como en la interfaz final correspondiente a la plataforma web. Por una parte, se puede pensar en implementar un sistema SCADA, sobre todo para los proyectos industriales, en donde se le pueda dar un mejor seguimiento al proyecto desde la planta, además del soporte que se tiene en la plataforma web.

Por otra parte, se puede realizar una mejora en el equipo de monitoreo y buscar un equipo más robusto que permita tener un monitoreo más estable, puesto que la Raspberry en ocasiones se bloquea debido a la gran cantidad de datos, lo que se soluciona con reinicios cada hora, pero no es algo ideal para un equipo de monitoreo industrial.

Sin duda, se puede pensar también en incluir inteligencia artificial que permita hacer un análisis de datos automáticamente y brindar mejores reportes del consumo de la red y del sistema solar fotovoltaico.

Finalmente, se puede organizar una mejora conjunta para la plataforma web, en donde se puedan aprovechar de mejor manera los datos adquiridos con el sistema de monitoreo.

9.2 Conclusiones del proyecto

Se obtiene un sistema de monitoreo para un sistema solar fotovoltaico de 100 kW instalado por ERCO Energía S.A.S, basado en el protocolo Modbus y con el uso como dispositivo principal un PLC S71200, que permite obtener datos de inversores y estación meteorológica para ser visualizados en una plataforma web manejada por otra área de la empresa.

Luego de analizar las arquitecturas y topologías de comunicación asociadas a las redes de datos, se ha propuesto una arquitectura que permite obtener la información de los dispositivos del sistema solar fotovoltaico, como inversores y estación meteorológica, utilizando el protocolo Modbus RTU y TCP/IP. La arquitectura propuesta es una maestro-esclavo, debido al uso de Modbus RTU con una topología de red tipo híbrida compuesto por topología tipo bus y estrella; así como una arquitectura tipo cliente-servidor puesto que se tiene un componente basado en TCP/IP. De esta manera se logra una comunicación confiable entre los componentes del sistema.

Por otra parte, mediante el diseño de un código de programación en el PLC utilizando el modelo de máquinas de estado finitas, se ha logrado que el S7 1200 adquiera datos de los inversores y la estación meteorológica generados por el simulador de protocolo Modbus MbSlave. Estos datos incluyen voltajes y corrientes en AC y DC, energía generada, estado de operación, entre otros. Esto ha permitido obtener información precisa y actualizada sobre el funcionamiento del sistema.

Las pruebas de integración realizadas entre el código de programación implementado en el PLC y la plataforma web existente de ERCO energía han sido exitosas. Durante las simulaciones desarrolladas, se ha observado de manera efectiva los datos adquiridos, verificando así el correcto funcionamiento del sistema. Esto asegura que la plataforma web pueda mostrar de manera precisa y confiable la información generada por el sistema solar fotovoltaico.

Como resultado de la identificación de los aspectos principales del funcionamiento del sistema, se ha elaborado un manual de instalación del gabinete de control. El manual proporciona al equipo técnico la información detallada sobre las conexiones requeridas para el sistema de

monitoreo. Además, el manual servirá como documento base para la revisión de problemas que puedan surgir, facilitando así la resolución eficiente de cualquier incidencia.

Si bien, este proyecto tiene mejoras por realizar, pero constituye una base para aplicaciones IoT enfocadas a monitoreos de sistemas solares fotovoltaicos. Este sistema de monitoreo cumple con los requisitos básicos para obtener una aplicación IoT, siendo estos dispositivos de adquisición de datos, un protocolo definido siendo en este caso modbus, un almacenamiento de datos el cual es manejado por un área diferente de la empresa y, finalmente la parte de analítica de datos, que, si bien tiene mucho por mejorar, se encuentra presente en el sistema.

Referencias

- [1] P. I. Wendy, R. Rivero, R. Luis, B. Horta, y I. Miguel López Leiva, “MONITORIZACIÓN DE UN PARQUE SOLAR FOTOVOLTAICO”, 2020.
- [2] ACCIONA, “¿Qué beneficios tiene la energía solar?” Consultado: el 3 de mayo de 2023. [En línea]. Disponible en: https://www.accion.com/es/energias-renovables/energia-solar/?_adin=02021864894
- [3] N. Vargas, “Colombia es el tercero de la región más avanzado en transición energética”. Consultado: el 4 de agosto de 2023. [En línea]. Disponible en: <https://www.larepublica.co/globoeconomia/colombia-es-el-tercer-pais-de-la-region-que-mas-ha-avanzado-en-la-transicion-energetica-3483686>
- [4] ERCO Energía, “Energía Solar Colombia”. Consultado: el 4 de agosto de 2023. [En línea]. Disponible en: https://erco.energy/co?utm_source=googleads&utm_medium=paid_search&utm_campaign=mofup_boyaca&utm_adgroup=palabrasclavemarca&gad=1&gclid=CjwKCAjw5remBhBiEiwAxL2M90iMjyizUo9wQXboxzbO1ktEZxQ6ILG5pQk15cGl0syj415YgrWPCBoCaDAQAvD_BwE
- [5] J. O. Molina Guerrero y L. G. Sierra Aguilera, “Propuesta de un Sistema de Monitoreo y Detección de Fallas en un Sistema Fotovoltaico”, 2023.
- [6] Z. Li, G. Tian, J. Yang, Y. Yang, y L. Zheng, “Intelligent condition monitoring and process optimization in internet of things (IoT)-based industrial systems”, *J Clean Prod*, 2021.

- [7] C. Chiu, H. K. C., J. Wu, y C. H. Hsei, “Integration of big data analytics and industrial internet of things for condition-based maintenance”, *J Manuf Syst*, vol. 48, pp. 80–93, 2018.
- [8] N. H. Motlagh, M. Mohammadrezaei, J. Hunt, y B. Zakeri, “Internet of things (IoT) and the energy sector”, *Energies (Basel)*, vol. 13, núm. 2, pp. 1–27, 2020, doi: 10.3390/en13020494.
- [9] S. Li, L. Da Xu, y S. Zhao, “5G Internet of Things: A survey”, *J Ind Inf Integr*, vol. 10, pp. 1–9, jun. 2018, doi: 10.1016/J.JII.2018.01.005.
- [10] J. Cheng, W. Chen, F. Tao, y C. L. Lin, “Industrial IoT in 5G environment towards smart manufacturing”, *J Ind Inf Integr*, vol. 10, pp. 10–19, jun. 2018, doi: 10.1016/j.jii.2018.04.001.
- [11] F. Al-Turjman, “5G-enabled devices and smart-spaces in social-IoT: An overview”, *Future Generation Computer Systems*, vol. 92, pp. 732–744, mar. 2019, doi: 10.1016/j.future.2017.11.035.
- [12] Á. L. Cobo, I. Oliveira, y J. M. Pamplona, “A review on the application of internet of things in photovoltaic systems”, *Renewable and Sustainable Energy Reviews*, 2021.
- [13] S. S. Mukrimaa *et al.*, “Los avances tecnológicos iot en sistemas fotovoltaicos”, *Jurnal Penelitian Pendidikan Guru Sekolah Dasar*. [En línea]. Disponible en: <https://www.olania.es/los-avances-tecnologicos-iot-en-sistemas-fotovoltaicos/>
- [14] S. Kamble, R. Kar, M. Sahoo, y R. Padhi, “An IoT-based intelligent monitoring system for solar photovoltaic power plant”, *Sustainable Energy Technologies and Assessments*, 2021.

- [15] S. Marzal, R. Salas, R. González-Medina, G. Garcerá, y E. Figueres, “Current challenges and future trends in the field of communication architectures for microgrids”, *Renewable and Sustainable Energy Reviews*, vol. 82. Pergamon, pp. 3610–3622, el 1 de febrero de 2018. doi: 10.1016/j.rser.2017.10.101.
- [16] M. Aybar Mejía *et al.*, “Protocolos y topologías utilizadas en los sistemas de comunicación de las microrredes eléctricas”, *Ciencia, Ingenierías y Aplicaciones*, vol. 4, núm. 1, pp. 81–95, jul. 2021, doi: 10.22206/cyap.2021.v4i1.pp81-95.
- [17] Cisco, “¿Qué es el monitoreo de red?”, *Cisco*. 2022. Consultado: el 3 de octubre de 2023. [En línea]. Disponible en: https://www.cisco.com/c/es_mx/solutions/automation/what-is-network-monitoring.html#~protocolos-de-monitoreo-de-red
- [18] MICROSOFT, “Comunicación en una arquitectura de microservicio | Microsoft Docs”. Consultado: el 3 de octubre de 2023. [En línea]. Disponible en: <https://learn.microsoft.com/es-es/dotnet/architecture/microservices/architect-microservice-container-applications/communication-in-microservice-architecture>
- [19] W. Stallings, “Comunicaciones y Redes de Computadores”, *Prentice Hall*, p. 896, 2004, [En línea]. Disponible en: http://www.unav.es/SI/manuales/Redes_Internet/indice.html
- [20] R. H. Deng, “Distributed systems, second edition”, *Comput Commun*, vol. 18, núm. 1, pp. 58–59, 1995, doi: 10.1016/0140-3664(95)90078-0.
- [21] A. Ananda y G. Poo, “Distributed systems: Concepts and design”, *Comput Commun*, vol. 18, núm. 7, pp. 521–522, 1995, doi: 10.1016/0140-3664(95)90005-5.
- [22] O. Lizama, “Redes de computadores Arquitectura Cliente-Servidor”, pp. 1–5, 2016, [En línea]. Disponible en:

<http://profesores.elo.utfsm.cl/~agv/elo322/1s16/projects/reports/Proyecto Cliente - Servidor.pdf>

- [23] Area tecnológica;, “Topologías de Red Aprende Fácil”, 2023. Consultado: el 30 de septiembre de 2023. [En línea]. Disponible en:
<https://www.areatecnologia.com/informatica/topologias-de-red.html>
- [24] I. Cerutti, A. M. Behredin, N. Andriolli, O. L. Ladouceur, y P. Castoldi, “Ring versus bus topology: A network performance comparison of photonic integrated NoC”, en *International Conference on Transparent Optical Networks*, IEEE Computer Society, ago. 2016. doi: 10.1109/ICTON.2016.7550706.
- [25] W. Song, Y. Chen, Y. Zhang, A. Wen, y C. Wei, “Start-up and Shut-down Control Scheme of DC Distribution Network based on Star Topology”, en *2019 IEEE PES GTD Grand International Conference and Exposition Asia, GTD Asia 2019*, Institute of Electrical and Electronics Engineers Inc., may 2019, pp. 571–576. doi: 10.1109/GTDAAsia.2019.8715896.
- [26] F. Z. Rachman, N. Yanti, N. Jamal, y E. Purwanto, “Wireless Sensor Network with Mesh Topology to Control Room Devices”, en *Proceedings - 2018 International Conference on Applied Science and Technology, iCAST 2018*, Institute of Electrical and Electronics Engineers Inc., oct. 2018, pp. 5–9. doi: 10.1109/iCAST1.2018.8751589.
- [27] R. Mangharam, D. Weller, R. Rajkumar, P. Mudalige, y F. Bai, “GrooveNet: A Hybrid Simulator for Vehicle-to-Vehicle Networks”, IEEE Explore. Consultado: el 30 de septiembre de 2023. [En línea]. Disponible en:
<https://ieeexplore.ieee.org/abstract/document/4141794>

- [28] Cisco, “Cisco Networking”, p. 282, 2008, [En línea]. Disponible en:
http://www.utez.edu.mx/curriculas/ccna1_ES/CHAPID=knet-1072827538031/RLOID=knet-1072905429421/RIOID=knet-1072905429828/knet/311072827537671/chapterframeset.html
- [29] J. Yu, X. Cheng, H. Jiang, y D. Yu, *Hierarchical Topology Control for Wireless Networks*. 2018. doi: 10.1201/9781315119236.
- [30] A. Álvarez, “Tipos de transmisores”, 2021.
- [31] M. Soltero *et al.*, “RS-422 and RS-485 Standards Overview and System Configurations”, *Configurations*, núm. May, p. 25, 2010.
- [32] Maxim Integrated, “How Far and How Fast Can You Go with RS-485 ?”, pp. 1–10, 2018.
- [33] Editorial Staff, “Understanding RS-485 Communication”. Consultado: el 25 de septiembre de 2023. [En línea]. Disponible en:
<https://instrumentationtools.com/understanding-rs-485-communication/>
- [34] The Modbus Organization, “Modbus”. Consultado: el 8 de mayo de 2023. [En línea]. Disponible en: <https://modbus.org/>
- [35] Paessler, “Modbus - Definition and Details”. Consultado: el 2 de mayo de 2023. [En línea]. Disponible en: <https://www.paessler.com/it-explained/modbus>
- [36] Udemy y Emile Ackbarali, “Mastering Modbus RS485 Network Communication | Udemy”. Consultado: el 2 de mayo de 2023. [En línea]. Disponible en:
<https://www.udemy.com/course/mastering-modbus-rs485-network-communication/learn/lecture/4919984?start=75#overview>

- [37] Udemy y Emile Ackbarali, “Mastering Modbus TCP/IP Network Communication”. Consultado: el 2 de mayo de 2023. [En línea]. Disponible en:
<https://www.udemy.com/course/mastering-modbus-tcpip-network-communication/learn/lecture/5205308?start=75#questions>
- [38] Universidad Autónoma del estado de Hidalgo, “Comunicación en Redes: CSMA/CD”. Consultado: el 7 de junio de 2023. [En línea]. Disponible en:
http://cidecame.uaeh.edu.mx/lcc/mapa/PROYECTO/libro27/4221_csmacd.html
- [39] H. M. Newman, “BACnet: The Global Standard for Building Automation and Control Networks”, *BACnet: The Global Standard for Building Automation and Control Networks*, jul. 2013, doi: 10.5643/9781606502907.
- [40] I. Durneva y N. Ajuha, “Implementation and transmission of Modbus TCP/IP protocol in the monitoring and control system of a universal electric drive”, *IOP Conf Ser Mater Sci Eng*, 2021.
- [41] J. Jose Rabadan Barastegui, “Diseño y desarrollo de una red MODBUS RTU basada en Arduino”, p. 66, 2017, [En línea]. Disponible en:
<http://bibing.us.es/proyectos/abreproy/91400/fichero/Memoria+TFG+JJRB.pdf>
- [42] E. Mandado Perez, J. M. Acevedo, C. Fernandez Silva, y J. Armesto Quiroga, *Automatas Programables y Sistemas de Automatización*, Segunda Edición. Barcelona: Marcombo, 2009.
- [43] R. Ragunathan, C. Jawahar, y V. Sankaranarayanan, “Design of PLC-based elevator control system using Finite State Machines”, *Int J Eng Adv Technol*, vol. 9, pp. 152–155, 2019.

- [44] M. A. T. Guaman quispe, “SISTEMA AUTOMATIZADO PARA EL CONTROL, MONITOREO Y ALMACENAMIENTO DE DATOS DEL MÓDULO FOTOVOLTAICO DE LA FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL”, 2011.
- [45] Siemens, “Controlador programable S7-1200 Manual de sistema”, p. 23, 2016, [En línea]. Disponible en: <https://support.industry.siemens.com/cs/document/109741593/simatic-s7-s7-1200-programmable-controller?dti=0&dl=en&lc=es-WW>
- [46] S. Elamanov, H. Son, B. Flynn, S. K. Yoo, N. Dilshad, y J. Song, “Interworking between Modbus and internet of things platform for industrial services”, *Digital Communications and Networks*, oct. 2022, doi: 10.1016/J.DCAN.2022.09.013.
- [47] P. I. Wendy, R. Rivero, R. Luis, B. Horta, y I. Miguel López Leiva, “Monitorización de un parque solar fotovoltaico”, *Eco Solar*, núm. 73, pp. 28–37, sep. 2020, Consultado: el 3 de mayo de 2023. [En línea]. Disponible en: <http://ecosolar.cubaenergia.cu/index.php/ecosolar/article/view/32>
- [48] J. E. Castro Segura y R. Posso García, “Diseño e implementación de un sistema de telemetría para el monitoreo de un sistema de generación de energía solar fotovoltaico”, Pereira, 2015.
- [49] L. A. Bargaran Vásquez, “Diseño de Sistema Automatizado de Monitoreo y Control con Telemetría para Mantenimiento Preventivo de Sistemas Fotovoltaicos en las industrias de 3 a 50 kW Lambayeque-Perú”, Trujillo, 2021.

- [50] M. Georgiev, D. Gospodinova, y A. Georgieva, “PLC Based Monitoring of Energy System of a Family House”, *2020 12th Electrical Engineering Faculty Conference, BulEF 2020*, sep. 2020, doi: 10.1109/BULEF51036.2020.9326070.
- [51] Consejo nacional de operación del sector eléctrico CNO, “Acuerdo 1602 Por el cual se aprueban los requisitos de protecciones para la conexión de sistemas de generación en el SIN”. Consultado: el 31 de julio de 2023. [En línea]. Disponible en:
<https://www.cno.org.co/content/acuerdo-1602-por-el-cual-se-aprueban-los-requisitos-de-protecciones-para-la-conexion-de>
- [52] Huawei, “SUN2000-(100KTL, 110KTL, 125KTL) Series User Manual”, 2021,
Consultado: el 4 de octubre de 2023. [En línea]. Disponible en: <https://e.huawei.com>
- [53] Atersa, “Datsol MET Adquisición de Radiación, Temperaturas y Velocidad del Viento Manual de Operación y de instalación”, 2008.
- [54] T. Bartlett, *Industrail Automated Systems: Instrumentation and Motion Control*. Delmar/Cengage Learning, 2010.
- [55] K. T. Chui *et al.*, “A Survey of Internet of Things and Cyber-Physical Systems: Standards, Algorithms, Applications, Security, Challenges, and Future Directions”, *Information (Switzerland)*, vol. 14, núm. 7. Multidisciplinary Digital Publishing Institute, p. 388, el 8 de julio de 2023. doi: 10.3390/info14070388.

Anexos

Manual de instalación del gabinete de control