



EDUCACIÓN
SECRETARÍA DE EDUCACIÓN PÚBLICA



**TECNOLÓGICO
NACIONAL DE MÉXICO**

INSTITUTO TECNOLÓGICO DE CIUDAD MADERO
DIVISIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN
MAESTRÍA EN INGENIERÍA ELÉCTRICA



TESIS

**INTERFAZ GRÁFICA REMOTA PARA EL ANÁLISIS DE LA SEÑAL ELÉCTRICA DE PANEL SOLAR
CON EL INTERNET DE LAS COSAS.**

Que para obtener el grado de
Maestro en Ingeniería Eléctrica

Presenta

Ing. Juan Manuel Del Angel Arredondo

G08071097

No. CVU: 1315700

Director

Dr. Pedro Martín García Vite

No. CVU: 227310

Codirectora

Dra. Ana Lidia Martínez Salazar



Educación
Secretaría de Educación Pública



TECNOLÓGICO
NACIONAL DE MÉXICO



Instituto Tecnológico de Ciudad Madero
Subdirección Académica
División de Estudios de Posgrado e Investigación

Ciudad Madero, Tamaulipas, **29/mayo/2026**

Oficio No.: U2.029/26

Asunto: Autorización de impresión de tesis

C. JUAN MANUEL DEL ÁNGEL ARREDONDO
No. DE CONTROL G08071097
P R E S E N T E

Me es grato comunicarle que después de la revisión realizada por el Jurado designado para su Examen de Grado de Maestría en Ingeniería Eléctrica, se acordó autorizar la impresión de su tesis titulada:

"INTERFAZ GRÁFICA REMOTA PARA EL ANÁLISIS DE LAS SEÑAL ELÉCTRICA DE PANEL SOLAR CON EL INTERNET DE COSAS"

El Jurado está integrado por los siguientes catedráticos:

PRESIDENTE:	DR. PEDRO MARTÍN GARCÍA VITE
SECRETARIO:	DRA. ANA LIDIA MARTÍNEZ SALAZAR
VOCAL:	M.C. AARÓN GONZÁLEZ RODRÍGUEZ
SUPLENTE:	M.C. RAFAEL CASTILLO GUTIÉRREZ
DIRECTOR DE TESIS:	DR. PEDRO MARTÍN GARCÍA VITE
CO-DIRECTOR:	DRA. ANA LIDIA MARTÍNEZ SALAZAR

Es muy satisfactorio para la División de Estudios de Posgrado e Investigación compartir con usted el logro de esta meta. Espero que continúe con éxito su desarrollo profesional y dedique su experiencia e inteligencia en beneficio de México.

ATENTAMENTE

Excelencia en Educación Tecnológica®

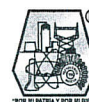
"Por mi patria y por mi bien"®

JOSÉ AARÓN MELO BANDA
JEFE DE LA DIVISIÓN DE ESTUDIOS DE
POSGRADO E INVESTIGACIÓN

ccp. Archivo
JAMB/ RCI/aecr



2026
año de
Margarita
Maza



Dedicatoria

A mi esposa, por haberme impulsado a emprender este camino académico y brindarme su apoyo incondicional en cada etapa. Sin su respaldo, este logro no habría sido posible.

A mis hijos, quienes con paciencia comprendieron mis ausencias y sacrificios durante el tiempo dedicado a la maestría.

A mi madre, por su constante compañía y fortaleza, siempre al pie del cañón.

A mi padre, por sus oraciones y su apoyo sincero que me dieron ánimo en los momentos más difíciles.

A mi hermano, ejemplo de perseverancia y superación, cuya actitud me inspira a seguir adelante.

Agradecimientos

En primer lugar, agradezco a Dios por todas las bendiciones recibidas y por su presencia constate en cada etapa de mi vida.

Al Dr. Pedro Martín García Vite, director de esta tesis, por su guía, por compartir su experiencia y conocimiento, por sus valiosas retroalimentaciones y por el apoyo brindado durante el desarrollo de este proyecto.

A mi esposa, Claudia Lizbeth Núñez Anguiano, por su apoyo económico, su amor, paciencia y comprensión, pilares fundamentales que hicieron posible la culminación de esta maestría.

A mis hijos, Juan Manuel Del Angel Núñez y Valentina Del Angel Núñez, quienes son mi mayor motivo de alegría e inspiración en cada paso que doy. Mis logros serán siempre también los suyos.

A mi padre, Juan Manuel Del Angel Mar, por sus oraciones y por estar siempre pendiente de mí. A mi madre, Clotilde Arredondo Chavarría, por su apoyo incondicional y por no permitirme rendirme en los momentos difíciles. A mi hermano, Andrés Alberto Del Angel Arredondo, por ser un ejemplo de perseverancia y superación, recordándome que siempre es posible luchar por los sueños.

A mis amigos y compañeros del posgrado en Maestría en Ingeniería Eléctrica: Christian, Luis, Gustavo, Alejandro, Alberto y Manríquez, por sus consejos y apoyo a lo largo de esta etapa académica.

Agradecimientos especiales:

Al Consejo Nacional de Ciencia y Tecnología, por el apoyo brindado a través de la beca 4036359.

Al Tecnológico Nacional de México/Instituto Tecnológico de Ciudad Madero, por ser la institución que me permitió desarrollar este proyecto y alcanzar este logro académico.

Interfaz gráfica remota para el análisis de la señal eléctrica de panel solar con el internet de las cosas.

Resumen

Los paneles solares tienen varias ventajas para el cuidado del medio ambiente, ya que generan energía renovable y no contribuyen al calentamiento global, favoreciendo el equilibrio de los ecosistemas. Una de las grandes desventajas es que son de naturaleza variable y necesitan un monitoreo continuo para saber si realizan su funcionamiento correcto y óptimo. El problema surge cuando la mayoría de los paneles solares utilizan aplicaciones que necesitan estar conectados a la red Wifi para monitorear la producción de energía, detectar posibles fallos y optimizar el rendimiento.

Este proyecto tiene como objetivo desarrollar un prototipo que monitoree las variables que inciden directamente la producción de energía de un panel solar fotovoltaico. La medición se envía de forma remota y se muestra posteriormente en una interfaz gráfica.

El sensor encargado de medir el voltaje es el FZ0430; para la corriente se utiliza el ACS712T 20A y la potencia se calcula en el programa mediante instrucciones. La medición de la temperatura y la humedad se realiza mediante el sensor DHT11, que permite monitorear las condiciones ambientales alrededor del panel solar. La irradiancia es monitoreada por un sensor RIK-A 200-04.

El conjunto de sensores está conectado a los pines de entrada analógica (Convertidor Analógico-Digital) y digital del Arduino. Estos valores digitales son procesados mediante algoritmos específicos en el código de un microcontrolador para obtener las variables medidas, que pueden ser almacenadas, visualizadas o transmitidas a otros dispositivos para su análisis. Los datos de los sensores, después de ser procesados por el microcontrolador, son transmitidos de forma remota a un dispositivo digital Raspberry Pi mediante módulos XBee.

Se ha implementado un sistema de monitoreo que proporciona una segunda medición independiente para la evaluación de la eficiencia del panel solar fotovoltaico (PV). Este sistema redundante permite la confiabilidad del monitoreo de la eficiencia del panel solar. La integración de múltiples sensores y la transmisión remota de datos aseguran una supervisión continua y precisa, facilitando la detección temprana de posibles anomalías y optimizando el rendimiento del sistema solar.

Remote Graphical Interface for the Analysis of the Electrical Signal of a Solar Panel with the Internet of Things.

Abstract

Solar panels offer several environmental advantages, as they generate renewable energy and do not contribute to global warming, helping preserve ecosystem balance. One of their main drawbacks is their variable nature, which requires continuous monitoring to ensure proper and optimal operation. The problem arises when most solar panels rely on applications that must remain connected to a Wi-Fi network in order to monitor energy production, detect potential failures, and optimize performance.

This project aims to develop a prototype capable of monitoring the variables that directly influence the energy production of a photovoltaic solar panel. The measurements are transmitted remotely and subsequently displayed through a graphical interface.

Voltage is measured using the FZ0430 sensor; current is obtained through the ACS712T 20A sensor, and power is calculated in the program through specific instructions. Temperature and humidity measurements are performed using the DHT11 sensor, which allows monitoring of the environmental conditions surrounding the solar panel. Solar irradiance is monitored by a RIK-A 200-04 sensor.

The set of sensors is connected to the analog (Analog-to-Digital Converter) and digital input pins of the Arduino. These digital values are processed through specific algorithms in the microcontroller code to obtain measured variables, which can then be stored, visualized, or transmitted to other devices for analysis. After being processed by the microcontroller, the sensor data is transmitted remotely to a Raspberry Pi digital device using XBee modules.

A monitoring system has been implemented to provide a second independent measurement for evaluating the efficiency of the photovoltaic (PV) panel. This redundant system enhances the reliability of the efficiency assessment. The integration of multiple sensors and the remote transmission of data ensure continuous and accurate supervision, enabling early detection of potential anomalies and optimizing the performance of the solar energy system.

Índice general

Resumen.....	III
Abstract.....	V
Índice de Tablas	X
Índice de Figuras	XI
Nomenclatura	XIII
Capítulo 1	1
1.1.- Antecedentes del problema.....	2
1.2.- Planteamiento del problema.....	4
1.3.- Objetivos.....	6
1.3.1.- Objetivo general.....	6
1.3.2.- Objetivos específicos	6
1.4.- Justificación	7
1.5.- Alcances y delimitaciones	9
1.5.1.- Alcances	9
1.5.2.- Delimitaciones	9
Capítulo 2	10
2.2.- Energías renovables y no renovables.....	11
2.4.- Energía solar fotovoltaica	13
2.5.- Celdas fotovoltaicas.....	13
2.5.1- Parámetros de las celdas fotovoltaicas.....	14
2.6.- Efecto de la irradiancia	18
2.7.- Efecto fotovoltaico.....	19
2.8.- Módulo fotovoltaico	19
2.9.- Sistema fotovoltaico	20
2.9.1.- Tipos de sistemas fotovoltaicos	20
2.10.- Componentes de un sistema fotovoltaico.	23

2.11.- Sistema de adquisición de datos (SAD).....	24
2.12.- Sensores usados en sistemas fotovoltaicos.	24
2.13.- Acondicionamiento de señales.....	26
2.14.- XBee.	26
2.14.1.- Características principales.....	26
2.14.2.- Topologías de red.....	27
2.14.3.- Arquitectura de red ZigBee.....	28
2.14.4.- Ventajas y desventajas.	28
2.15.- Raspberry Pi.....	29
2.16.- Pantallas Nextion.	30
2.16.1.- Nextion Editor.....	31
2.17.- Internet de las cosas (IoT).....	31
2.17.1.- Componentes fundamentales del IoT.....	32
2.17.2.- Características Esenciales.	33
2.17.3.- Ámbitos de Aplicación.	33
2.18.- Firebase Realtime Database.....	34
2.18.1.- Características Principales.	34
2.19.- Estado del Arte	35
Capítulo 3	36
3.1.- Arquitectura general del sistema.....	37
3.2.- Propuesta para la estructura del Sistema de Adquisición de Datos (SAD).....	38
3.3.- Adaptación de variables.....	40
3.4.- Comunicación a distancia mediante XBee.	41
3.5.- Comunicación Punto a Punto con XBee Serie 2.....	42
3.6.- Procesamiento en la Raspberry.....	44
3.6.1.- Interfaz Gráfica Nextion.	45
3.6.2.- Base de Datos SQLite.	45
3.6.3.- Firebase Realtime Database.	46

Capítulo 4.....	47
4.1.- Introducción al desarrollo experimental	48
4.2.- Sistema de Adquisición de Datos.	48
4.3.- Red Inalámbrica XBee.....	51
4.4.- Configuración e Implementación en la Raspberry Pi.	54
4.5.- Programa en Python para la gestión del sistema de monitoreo.....	55
4.5.1.- Recepción de datos desde el módulo inalámbrico XBee.	56
4.5.2.- Actualización en tiempo real de la pantalla Nextion.	56
4.5.3.- Almacenamiento local mediante SQLite.	59
4.6.- Interfaz Gráfica Nextion.	64
4.7.- Test de usabilidad de la interfaz gráfica.	66
4.8.- Recolección de datos en campo.	69
4.9.- Conclusiones.....	71
Capítulo 5.....	72
5.1.- Introducción al análisis de datos.	73
5.2.- Análisis en Google Colab.	74
5.3.- Clasificación de condiciones solares mediante K-means.	75
5.4.- Generación de reglas mediante Árbol de Decisión.....	78
5.5.- Acciones correctivas recomendadas.	80
5.6.- Conclusiones.....	81
Capítulo 6.....	83
6.1.- Conclusiones Generales.....	84
6.2.- Contribuciones.....	84
6.3.- Recomendaciones para trabajos futuros.....	85
6.4.- Actividades realizadas.	86
6.5.- Publicaciones.	86
Bibliografía.	87

Anexos.....	93
Anexo A.....	94
Anexo B.....	103

Índice de Tablas

Tabla 3.1: Componentes principales del proyecto.	39
---	----

Índice de Figuras

Figura 2. 1: Representación gráfica de las principales fuentes de energía no renovable del planeta: combustibles fósiles (carbón, petróleo, gas natural) y energía nuclear.	12
Figura 2. 2: Representación gráfica de las principales fuentes de energías renovables.	12
Figura 2. 3: Parte de una celda solar.	14
Figura 2. 4: Curvas características IV y PV de una celda fotovoltaica, mostrando los parámetros eléctricos fundamentales bajo condiciones estándar.	17
Figura 2. 5: Condiciones de prueba estándar para paneles solares.	18
Figura 2. 6: Diferentes tipos de paneles solares.	19
Figura 2. 7: Diagrama general de la clasificación de los sistemas fotovoltaicos.	20
Figura 2. 8: Configuraciones más comunes en los SFA.	21
Figura 2. 9: Esquema básico de una instalación FV conectada a la red.	23
Figura 2. 10: Topologías ZigBee.	27
Figura 2. 11: Raspberry Pi 4.	30
Figura 2. 12: Pantalla Nextion 7 pulgadas.	31
Figura 2. 13: Internet de las cosas.	32
Figura 2. 14: Desarrollo del Firebase.	34
Figura 3. 1: Diagrama general de la Metodología implementada.	37
Figura 3. 2: Arquitectura general del sistema.	38
Figura 3. 3: Circuito electrónico del SAD.	41
Figura 3. 4: Diagrama de flujo de la lógica de control implementada en el subsistema de adquisición de datos.	42
Figura 3. 5: Procesamiento y Gestión de Datos en la Raspberry Pi.	44
Figura 4. 1: Prototipo SAD.	49
Figura 4. 2: Diseño del circuito (PCB) en formato 3D.	50
Figura 4. 3: Circuito SAD con sensores conectados.	51
Figura 4. 4: Parámetros XBee Router.	52
Figura 4. 5: Parámetros XBee Coordinador.	53
Figura 4. 6: Monitor conectado a la Raspberry Pi.	55
Figura 4. 7: Mensaje mostrado al ingresar un usuario incorrecto.	57
Figura 4. 8: Mensaje mostrado al ingresar una contraseña incorrecta.	57
Figura 4. 9: Acceso exitoso después de ingresar el usuario y la contraseña correctas.	58

Figura 4. 10: Interfaz principal del sistema mostrando las variables del panel fotovoltaico en tiempo real.....	59
Figura 4. 11: Datos recopilados.....	61
Figura 4. 12: Carcasa impresa en 3D para interfaz gráfica y Raspberry Pi 4.....	65
Figura 4. 13: Monitoreo de Panel Fotovoltaico.....	70
Figura 5. 1: Distribución de Irradiancia Registrada.	75
Figura 5. 2: Gráfica de las mediciones de irradiancia obtenidas.	76
Figura 5. 3: Clasificación de condiciones solares mediante K-means.	77
Figura 5. 4: Árbol de Decisión para Clasificación de condiciones solares.	79

Nomenclatura

PV	Fotovoltaico (del inglés, photovoltaic)
SAD	Sistema de adquisición de datos
I	Corriente
V	Voltaje
P	Potencia
W	Watts
STC	Condiciones de Prueba Estándar (del inglés, Standard Test Condition)
P_{max}	Potencia máxima
Wp	Potencia pico
AM1.5	Condición estándar de masa de aire 1.5 (del inglés, Air Mass 1.5)
V_{oc}	Voltaje de circuito abierto (del inglés, Voltage Open Circuit)
I_{sc}	Corriente de cortocircuito (del inglés, Short Circuit Current)
E_i	Campo eléctrico interno (del inglés, Electric internal)
V_{mp}	Voltaje en el punto de máxima potencia
I_{mp}	Corriente en el punto de máxima potencia
ADC	Convertidor Analógico-Digital (del inglés, Analog-to-Digital Converter)
IoT	Internet de las Cosas (del inglés, Internet of Things)
GUI	Interfaz Gráfica de Usuario (del inglés, Graphical User Interface)
HMI	Interfaz Hombre-Máquina (del inglés, Human Machine Interface)
SFA	Sistema Fotovoltaico Autónomo
GEI	Gases de Efecto Invernadero
FF	Factor de Llenado (del inglés, Fill Factor)
MPPT	Seguimiento del punto de Máxima Potencia (del inglés, Maximum Power Point Tracking)
PWM	Modulación por Ancho de Pulso (del inglés, Pulse Width Modulation)
CD	Corriente Directa
CA	Corriente Alterna
STC	Condiciones de Prueba Estándar (del inglés, Standard Test Conditions) SHS Sistemas Domésticos (del inglés, Solar Home Systems)

DAQ Sistema de Adquisición de Datos (del inglés, Data Acquisition System)

IDE Entorno de Desarrollo Integrado (del inglés, Integrated Development Enviroment)

WYSIWYG Lo que ves es lo que obtienes (del inglés, What You See Is What You Get)

UART Transmisor-Receptor Asíncrono Universal (del inglés, Universal Asynchronous Receive/Transmitter)

I2C Protocolo de comunicación (del inglés, Inter-Integrated Circuit)

SPI Interfaz Periférica Serial (del inglés, Serial Peripheral Interface)

GPIO Puerto de Entrada/Salida de Propósito General (del inglés, General Purpose Input/Output)

RTD Detector de Temperatura por Resistencia (del inglés, Resistance Temperature Detector)

EVA Etilvinilacetato (del inglés, Ethylene Vinyl Acetate)

LCD Pantalla de Cristal Líquido (del inglés, Liquid Crystal Display)

Capítulo 1

Introducción

En este capítulo se presenta una introducción al trabajo desarrollado, en el cual se muestra los antecedentes del problema, el planteamiento del problema, los objetivos, la justificación, los alcances y delimitaciones.

1.1.- Antecedentes del problema

La industria eléctrica ha sido el principal motor del crecimiento de la economía mundial desde principios del siglo XX hasta el presente, realizando un excelente trabajo al construir, operar y mantener una infraestructura compleja que transporta billones de kWh de electricidad a través de miles de kilómetros. La correlación entre el uso y la disponibilidad de electricidad y el nivel de vida es fuerte y sigue siendo un elemento crítico en las economías en desarrollo de todo el mundo [1].

El aumento de demanda de energía por urbanización, el desarrollo de la sociedad, el crecimiento demográfico y las emisiones de gases de efecto invernadero provenientes del sector eléctrico, principalmente carbón, gas y petróleo, son el motor de la transformación de la industria eléctrica, que implica una serie de cambios fundamentales y un proceso de transición, con la incorporación nuevas fuentes alternas.

Estas fuentes alternas son fuentes de energía renovables, como la eólica, la hidroeléctrica, la volcánica, la solar y los biocombustibles, y pueden utilizarse para disminuir la huella de carbono, mejorar la eficiencia energética, abordar la escasez de energía y apoyar un crecimiento respetuoso con el medio ambiente [2].

Con el fin de lograr el equilibrio y el progreso, los países en desarrollo utilizan la energía solar para satisfacer las demandas energéticas individuales y sociales. Un panel solar, un módulo fotovoltaico, se utiliza para convertir la energía solar en corriente eléctrica gracias al efecto fotovoltaico, por medio de células o celdas fotovoltaicas (PV), almacenando esta energía en baterías [3].

Pese a las grandes ventajas, esta fuente de energía alternativa, al igual que todas las energías renovables, tiene una producción variable, ya que es afectada por factores como el clima, las variables meteorológicas adicionales como la humedad, la temperatura y ángulo de montaje. Para lograr una mejor eficiencia, es decir una generación de energía óptima, cuidar nuestra economía al invertir en estos dispositivos, mantener la seguridad del equipo y la confiabilidad de este, es necesario medir en tiempo cuasi-real estas variables, sus efectos en el sistema e implementar estrategias de control. Por lo anterior, se considera que el buen desempeño de un panel solar depende en gran medida de las tecnologías de comunicación que se emplean para transmitir estos datos.

No solo se necesita una medición de las variables más significativas que interactúan en un panel solar, sino que también es necesario una buena adquisición y procesamiento de datos apoyados en la implementación de un sistema de comunicación. Este sistema de comunicación será el medio para la transmisión de datos y concentrado de los mismos, permitiendo implementar acciones para mejorar el funcionamiento del panel solar.

1.2.- Planteamiento del problema

La energía es fundamental tanto en el problema del cambio climático como en su solución. Gran parte de los gases de efecto invernadero que atrapan el calor del sol provienen de la producción de energía mediante la quema de combustibles fósiles para generar electricidad y calor. Estos combustibles, como el carbón, el petróleo y el gas, son los principales responsables del cambio climático global, contribuyendo a más del 75% de las emisiones totales de gases de efecto invernadero y alrededor del 90% de las emisiones de dióxido de carbono [4].

Para evitar los impactos más negativos del cambio climático, es esencial reducir las emisiones a casi la mitad para el año 2030 y alcanzar cero netos para el 2050 [5]. El 14 de septiembre de 2016, el Senado mexicano ratificó el Acuerdo de París sobre el cambio climático en donde se comprometió a tomar acciones que contribuyan a la reducción de emisiones de gases de efecto invernadero [6].

Para lograrlo, debemos dejar de depender de los combustibles fósiles e invertir en fuentes de energía alternativas que sean limpias, accesibles, asequibles, sostenibles y confiables.

Las energías renovables, abundantes en nuestro entorno, son una solución clave. Estas fuentes de energía, como la solar, eólica, hidroeléctrica, geotérmica y de biomasa, se renuevan naturalmente y emiten pocos o ningún contaminante al aire. Aunque los combustibles fósiles aún representan más del 80 % de la producción de energía en todo el mundo, las energías limpias están ganando terreno. Actualmente, cerca del 29 % de la electricidad proviene de fuentes renovables [5]. Existen varias razones por las cuales acelerar la transición hacia energías limpias beneficia tanto al presente como a las generaciones futuras.

Los paneles solares, también conocidos como módulos fotovoltaicos, son una fuente de energía limpia y sostenible. Sin embargo, para maximizar su eficiencia y prolongar su vida útil, es fundamental llevar a cabo un análisis y monitoreo continuo de la señal eléctrica que generan.

Los beneficios que obtendremos al tener un monitoreo por mencionar algunos son, detectar la capacidad de generación de energía fotovoltaica, así como posibles fallas o

degradación en los paneles, como daños físicos, grietas o roturas en las células solares, también permite detectar suciedad o acumulación de polvo, lo que afecta directamente a la eficiencia.

Al identificar problemas en etapas tempranas, se pueden tomar medidas correctivas para mantener el rendimiento óptimo. Se puede ajustar la orientación y ángulo de los paneles según los datos recopilados, maximizando la captación de luz solar y, por ende, la generación de electricidad.

Se podrá evaluar el retorno de inversión, al evaluar la relación entre costo inicial y la producción de energía a lo largo del tiempo, los consumidores pueden determinar si la inversión en paneles más eficientes vale la pena, ya que los paneles solares más eficientes pueden ser más costosos, pero también generar más energía. El monitoreo constante garantiza que la inversión sea rentable a lo largo del tiempo.

1.3.- Objetivos

1.3.1.- Objetivo general

Diseñar e implementar un sistema de monitoreo eficiente y confiable que permita la recolección y transmisión de datos relevantes sobre el rendimiento de un panel PV, como la irradiancia, la calidad de la energía producida, la temperatura y la potencia generada, mediante instrumentación electrónica y dispositivos digitales.

1.3.2.- Objetivos específicos

- Medir y registrar las variables de interés que son el voltaje, corriente, potencia, irradiancia, temperatura y humedad adquiridas en el panel solar.
- Implementar un sistema de adquisición de datos en tiempo real basado en tecnología de interconexión inalámbrica.
- Mostrar y registrar, mediante una interfaz gráfica, los datos obtenidos de un panel fotovoltaico.
- Realizar una validación experimental del prototipo con el fin de recolectar datos reales.

1.4.- Justificación

En la búsqueda de un futuro más sostenible y amigable con el medio ambiente, se ha tomado la decisión de favorecer la reducción del dióxido de carbono mediante el uso de energías limpias. Entre estas, destaca la generación solar fotovoltaica como una alternativa clave.

Sin embargo, los paneles solares tienen muchas variables que afectan su rendimiento óptimo, ya que existe una naturaleza variable en la calidad y cantidad de la energía, como lo son la temperatura, la irradiancia, la humedad y el ángulo de inclinación [7]. Otros agentes externos que también impactan a estos dispositivos son el polvo, el viento o las nubes, el factor más influyente son las condiciones climáticas, que a su vez son afectadas por las estaciones del año. Dificultando la tarea de mantener un equilibrio adecuado entre el consumo y la generación de energía.

Todos estos agentes dificultan mantener un rendimiento óptimo, por lo tanto, al tener un análisis y monitoreo gráfico de forma remota de la señal eléctrica que produce el panel solar, se podrá conocer la eficiencia, la cantidad disponible y la calidad de la energía, la optimización del rendimiento, si existe alguna falla o degradación, con base a esto se podrán tomar acciones correctivas que permitan que el sistema genere su máximo potencial de salida y maximizaría su vida útil.

Además, se realizarán medidas basadas en los datos recopilados, permitiendo mitigar efectos como el sombreado parcial o la acumulación de polvo.

El presente proyecto de investigación propone monitorear en tiempo real el estado de optimización de un panel solar fotovoltaico mediante dispositivos que brindan la posibilidad de medir y registrar en tiempo real el estado de optimización de un panel solar fotovoltaico mediante dispositivos que permiten medir y registrar variables eléctricas y meteorológicas implicadas en el proceso. Esto se hará de forma inalámbrica mediante el protocolo ZigBee, que ofrece un amplio rango de comunicación, bajo costo y una implementación sencilla.

Los datos se visualizarán mediante una interfaz gráfica que permitirá correlacionar las condiciones ambientales (irradiancia, temperatura) con los parámetros eléctricos del panel (voltaje, corriente, potencia). Esta correlación facilitará la identificación de anomalías operativas, como la disminución del rendimiento en condiciones de alta irradiancia, lo cual puede indicar problemas como suciedad en el panel, degradación de celdas o fallos en el sistema. La detección temprana de estas discrepancias permitirá implementar acciones correctivas oportunas, optimizando la generación de energía y maximizando el retorno de la inversión.

1.5.- Alcances y delimitaciones

1.5.1.- Alcances

- Diseño e implementación de un prototipo de un sistema de monitoreo para un panel solar fotovoltaico.
- Medición y registro de las variables de voltaje, corriente, potencia, irradiancia, temperatura y humedad del panel solar.
- Implementación de un sistema de comunicación inalámbrica para la transmisión de datos mediante el protocolo ZigBee.
- Desarrollo de una interfaz gráfica para la visualización y el registro de los datos adquiridos en tiempo real.
- Validación experimental del funcionamiento del prototipo en un entorno controlado.

1.5.2.- Delimitaciones

- El sistema está diseñado para monitorear un único panel solar.
- La comunicación ZigBee está sujeta a las limitaciones de distancia.
- El sistema se limita a la adquisición, transmisión y visualización de datos. No se implementan acciones de control automático en el panel.
- La validación del prototipo se realiza en un período de tiempo acotado, no siendo un estudio de degradación a largo plazo.
- El sistema desarrollado no calcula directamente el retorno de inversión, sin embargo, proporciona las variables fundamentales para que un usuario o instalador pueda estimarlos con precisión, ya que permite conocer la energía real generada por el panel bajo condiciones específicas.

Capítulo 2

Marco Teórico

En este capítulo se establecen los fundamentos teóricos esenciales sobre energía solar fotovoltaica, caracterización de paneles solares, sistemas de adquisición de datos, comunicación inalámbrica ZigBee y tecnologías IoT. Incluye una revisión del estado del arte en sistemas de monitoreo para energía solar.

2.1.- Introducción a los sistemas fotovoltaicos

El crecimiento en el uso de sistemas fotovoltaicos ha incrementado la necesidad de contar con mecanismos de supervisión capaces de proporcionar información remota, precisa y oportuna sobre el comportamiento de los paneles solares. Debido a la naturaleza variable de la energía solar, factores como la irradiancia, la temperatura, la humedad, el sombreado y la degradación física pueden afectar significativamente la producción eléctrica. Por ello, resulta indispensable fortalecer los procesos de adquisición y transmisión de datos que permitan identificar fallas, detectar disminuciones en la eficiencia, anticipar limitaciones de capacidad y responder adecuadamente ante condiciones ambientales cambiantes. La disponibilidad de información en tiempo real no solo mejora la confiabilidad del sistema, sino que también posibilita un análisis proactivo del rendimiento del panel fotovoltaico, permitiendo implementar acciones correctivas antes de que se presenten pérdidas energéticas significativas. En este contexto, el monitoreo continuo mediante tecnologías de instrumentación electrónica, comunicación inalámbrica e interfaces gráficas se convierte en una herramienta clave para optimizar la operación y prolongar la vida útil de los sistemas solares.

2.2.- Energías renovables y no renovables

La transición energética global se fundamenta en la distinción crítica entre fuentes de energía renovables y no renovables. Las fuentes no renovables, principalmente los combustibles fósiles (carbón, petróleo, gas natural) y el uranio para energía nuclear, se caracterizan por existir en reservas finitas en la Tierra y por tiempos de regeneración extremadamente largos, lo que las hace agotables [8]. Su utilización está asociada a impactos ambientales significativos, como la emisión de Gases de Efecto Invernadero (GEI) que impulsan el cambio climático, la contaminación local del aire y la generación de residuos de difícil gestión, como los residuos radiactivos [9]. Algunos ejemplos de estas fuentes de energías se ilustran en la Figura 2.1.



Figura 2. 1: Representación gráfica de las principales fuentes de energía no renovable del planeta: combustibles fósiles (carbón, petróleo, gas natural) y energía nuclear.

Fuente: Elaboración propia.

Por el contrario, las energías renovables se derivan de flujos naturales continuos o repetitivos, como la radiación solar, el viento, el ciclo del agua y el calor interno de la Tierra, lo que las hace inherentemente inagotables [10]. Tecnologías como la solar fotovoltaica, la eólica, la hidroeléctrica, la geotérmica y la de biomasa sostenible, presentan la ventaja clave de generar electricidad con emisiones de GEI nulas o mínimas durante su operación [11]. Si bien su fabricación e instalación conllevan impactos ambientales y el desafío de la intermitencia, estos son manejables mediante estrategias de diseño, almacenamiento de energía y sistemas de previsión, posicionándolas como la piedra angular de un sistema energético sostenible, seguro y descentralizado. Algunos ejemplos de estas tecnologías se presentan en la Figura 2.2.



Figura 2. 2: Representación gráfica de las principales fuentes de energías renovables.

Fuente: Elaboración propia.

2.3.- Energía solar

El Sol, como una fuente de energía inagotable, emite energía en forma de radiación electromagnética. Gracias a nuestra distancia al Sol, recibimos luz y calor. Este recurso natural es considerado sostenible y alternativo, reduciendo significativamente el impacto ambiental en comparación con las energías sucias y contaminantes, como los combustibles fósiles.

La energía solar se define como aquella que proviene de la radiación del sol. Llega a nosotros a través de la radiación electromagnética y así puede ser aprovechada para la obtención de energía eléctrica [12]. Para lograrlo es necesario utilizar paneles solares, los cuales captan la radiación y la convierten en una energía para consumo humano.

Al ingresar la radiación solar en la atmósfera terrestre, ésta se absorbe, se refleja, se refracta y se transmite hacia la superficie terrestre, fenómeno que atenúa el valor de la irradiancia. Así la radiación que incide sobre un captador está formada por dos componentes: la directa y la difusa. La suma de las dos formas el concepto de irradiancia global o total [13].

2.4.- Energía solar fotovoltaica

La energía solar fotovoltaica es una tecnología renovable que transforma directamente la luz solar en electricidad a través del efecto fotovoltaico en semiconductores, como el silicio. Este proceso se da en las celdas o células fotovoltaicas, que al juntarse forman paneles solares que generan electricidad en forma de corriente directa, la cual puede transformarse en corriente alterna para uso doméstico, comercial o industrial [11].

2.5.- Celdas fotovoltaicas.

Las celdas solares constituyen el componente fundamental de los sistemas de generación de energía solar y son las unidades mínimas de conversión fotovoltaica. Con semiconductores tipo-n y tipo-P, y con contactos eléctricos en la parte superior y en la inferior se forman las celdas solares. En la Figura 2.3 se muestra una estructura simple de una celda solar en donde se ilustra el principio básico de la conversión fotovoltaica.

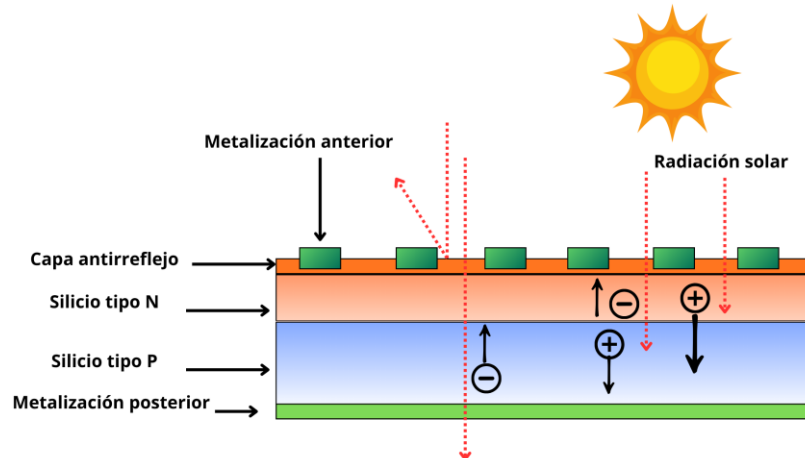


Figura 2. 3: Parte de una celda solar.

Fuente: Adaptado de [14].

La unión entre las capas “N” (lado negativo) y “P” (lado positivo) es la responsable de que se forme un campo eléctrico interno E_i que radica principalmente en la unión.

El principio de funcionamiento de la célula se basa en la absorción de fotones por su estructura semiconductor. Cuando un fotón incide sobre la capa p y posee la energía suficiente, es capaz de liberar un electrón. Este electrón es transportado a través del circuito externo, lo que da lugar a la generación de una corriente eléctrica. Las células se encuentran interconectadas a través de tiras metálicas que constituyen arreglos eléctricos en serie y paralelo dentro del módulo fotovoltaico [15]. La unidad de medición es el voltio.

2.5.1- Parámetros de las celdas fotovoltaicas.

Se identifican diversos parámetros de las células fotovoltaicas que influyen en la efectividad de la conversión de luz solar a electricidad. Es muy habitual que los fabricantes proporcionen estos parámetros bajo condiciones de (STC).

- Corriente de cortocircuito (I_{SC}): Se define como la corriente máxima que puede generar una célula. La unidad de medida utilizada es el Amper (A) o miliamperios (mA) [17].
- Voltaje de circuito abierto (V_{OC}): Es el voltaje máximo generado por una celda solar. La unidad de medida es el voltio (V).

- Punto de máxima potencia (P_{max}): Se define como la potencia máxima que puede generar una celda solar bajo condiciones estándar de prueba (STC, por sus siglas en inglés). La unidad de medida es el watt (W). Una celda fotovoltaica alcanzará su máxima potencia únicamente cuando funcione a una determinada corriente y voltaje. El punto de máxima potencia en términos de expresión P_m se define como:

$$P_m = I_m \cdot V_m \quad (1)$$

- La forma típica de la curva de potencia es ascendente hasta P_{max} y luego descendente, formando una parábola asimétrica. Este comportamiento permite identificar el voltaje en el punto de máxima potencia (V_{mp}) y la corriente correspondiente (I_{mp}), parámetros fundamentales para el dimensionamiento de inversores y sistemas de seguimiento del punto de máxima potencia (MPPT).
- Corriente en el punto de máxima potencia (I_m): La corriente generada por la celda fotovoltaica en el punto de máxima potencia será siempre inferior a la corriente de cortocircuito (I_{SC}). La corriente eléctrica se mide en Amperios (A) o miliamperios (mA).
- Voltaje en el punto máxima (V_m): El voltaje que la célula fotovoltaica generará al operar en el punto de máxima potencia es, por definición, inferior al voltaje de circuito abierto (V_{OC}) y se expresa en volts.
- Factor de llenado (FF): Se establece una relación entre las áreas ocupadas por el rectángulo $I_m - V_m$ y el área correspondiente al rectángulo $I_{SC} - V_{OC}$. La ecuación se presenta a continuación:

$$FF = \frac{I_m \cdot V_m}{I_{SC} \cdot V_{OC}} \quad (2)$$

O

$$FF = \frac{P_{max}}{I_{SC} \cdot V_{OC}} \quad (3)$$

P_{max} puede escribirse alternativamente en términos de I_{SC} , V_{OC} y FF como:

$$P_m = I_{SC} \cdot V_{OC} \cdot FF \quad (4)$$

- Eficiencia (η): La eficiencia de una celda fotovoltaica se define como la relación entre la potencia máxima de salida (P_{max}) y la potencia de entrada (P_{in}). La eficiencia de una

celda se expresa en términos de porcentaje (%), lo que indica que dicho porcentaje de la potencia de radiación incidente se transforma en potencia eléctrica. El valor de P_{in} para STC se considera 1000W/m². La potencia de entrada se define como densidad de potencia, es decir, la potencia dividida por área. Por consiguiente, para calcular la eficiencia utilizando la potencia de entrada (P_{in}) en condiciones estándar de prueba STC, es necesario multiplicar esta densidad por el área de la celda fotovoltaica [17]. Por lo tanto, la eficiencia puede expresarse de la siguiente manera:

$$\eta = \frac{P_m}{P_{in}} = \frac{I_{sc} \cdot V_{oc} \cdot FF}{P_{in} \cdot A} \quad (5)$$

La curva de potencia de una celda o módulo fotovoltaico representa gráficamente la relación entre la potencia generada y el voltaje aplicado, bajo condiciones específicas de irradiancia y temperatura. Esta curva es derivada directamente de la curva IV (corriente – voltaje), ya que la potencia eléctrica se calcula como el producto de la corriente (I) y el voltaje (V), es decir:

$$P = I \cdot V \quad (6)$$

A medida que el voltaje varía desde cero hasta el voltaje de circuito abierto (V_{oc}), la corriente también cambia, y por lo tanto la potencia generada presenta un comportamiento no lineal. Inicialmente, cuando el voltaje es bajo, la corriente es alta por lo que la potencia es baja. Conforme el voltaje aumenta, la potencia también incrementa hasta alcanzar el punto de máxima potencia (P_{max}), donde el producto $I \cdot V$ es máximo. Este punto es crítico para el diseño y operación de sistemas fotovoltaicos, ya que representa el estado óptimo de generación eléctrica.

El análisis de la curva de potencia es esencial para evaluar el rendimiento de los paneles solares en condiciones reales. Factores como la irradiancia, la temperatura ambiente, el envejecimiento de las celdas, la acumulación de suciedad pueden alterar la forma de la curva, desplazando el punto P_{max} y reduciendo la eficiencia del sistema. Por ello, el monitoreo continuo de esta curva permite detectar anomalías, optimizar la captación de energía y prolongar la vida útil del sistema fotovoltaico [19].

El comportamiento eléctrico de una celda fotovoltaica puede analizarse mediante sus curvas características. La curva de potencia frente a voltaje muestra cómo varía la potencia generada en función del voltaje aplicado, revelando el punto de máxima potencia. En ella, el voltaje de circuito abierto (V_{OC}) suele ser cercano a 0.6 V para celdas de silicio, y los valores del eje horizontal representan los distintos puntos de operación de voltaje, mientras que el eje vertical cuantifica la potencia eléctrica generada en cada uno de esos puntos.

En la Figura 2.4 se muestra simultáneamente la curva de corriente (IV) y la curva de potencia (PV) de una celda fotovoltaica, destacando los parámetros clave como I_{SC} , V_{OC} , I_{mp} , V_{mp} y el punto de máxima potencia P_{max} , lo que permite visualizar de forma integrada el comportamiento eléctrico del dispositivo bajo condiciones estándar.

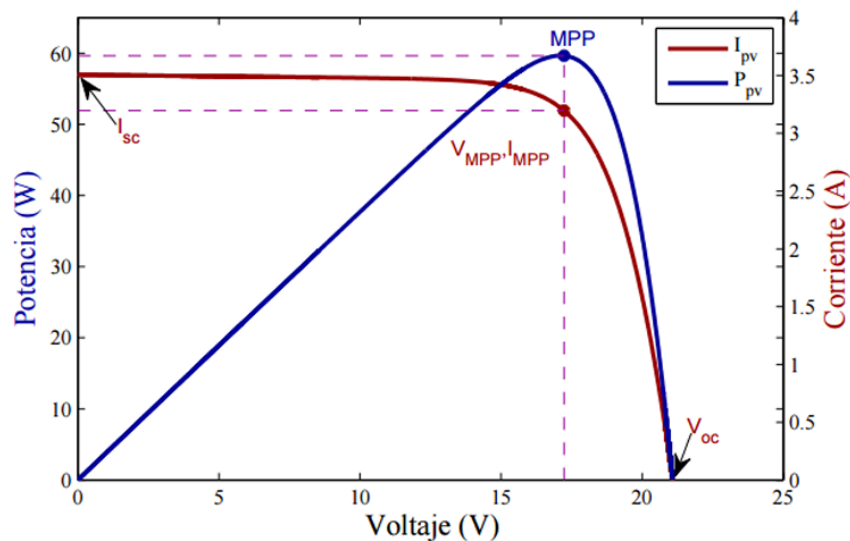


Figura 2. 4: Curvas características IV y PV de una celda fotovoltaica, mostrando los parámetros eléctricos fundamentales bajo condiciones estándar.

Fuente: Citado de [20].

2.6.- Efecto de la irradiancia

La irradiancia solar es el parámetro fundamental que determina la generación de electricidad en un sistema fotovoltaico se simboliza por la letra G . En el sistema internacional la unidad de energía es el joule, J.

Para la caracterización y comparación estandarizada del desempeño de celdas y paneles solares, la industria emplea las Condiciones de Prueba Estándar (STC), en la Figura 2.5 se muestra una representación de las 3 condiciones de prueba estándar para paneles solares. Estas condiciones, definidas internacionalmente, especifican una irradiancia de 1000 W/m^2 (100mW/cm^2) con un espectro de luz solar equivalente a una masa de aire de 1.5 y una temperatura de celda de 25°C [16]. La potencia máxima (P_{max}) que un módulo puede entregar bajo estas condiciones se denomina “potencia pico” y se expresa en Watts pico (Wp).

Sin embargo, en condiciones de operación reales, la irradiancia rara vez es constante e igual a 1000 W/m^2 . Varía continuamente debido a la posición del sol (hora del día y época del año), condiciones atmosféricas (nubosidad, neblina, contaminación) y la orientación e inclinación del panel.

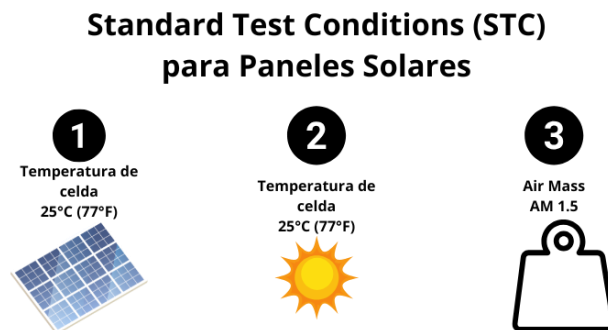


Figura 2. 5: Condiciones de prueba estándar para paneles solares.

Fuente: Citado de [21].

2.7.- Efecto fotovoltaico.

Es el proceso de convertir la luz solar en electricidad y ocurre en un dispositivo constituido por semiconductores, en el cual la luz solar es absorbida y transformada directamente en electricidad del tipo de corriente directa (c.d.), sin la intervención de procesos intermedios.

En una celda solar el efecto fotovoltaico se manifiesta a través de la generación de voltaje en sus terminales en presencia de iluminación. Al conectar un dispositivo eléctrico, como un motor, a las terminales de una celda solar, se cierra el circuito, la tensión generada provoca que el motor gire como resultado del trabajo realizado por la corriente al fluir a través de su embobinado. Esto constituye la evidencia tangible del efecto fotovoltaico [17].

2.8.- Módulo fotovoltaico

Un módulo fotovoltaico comúnmente llamada panel solar, está formado por la interconexión en serie y en algunos casos, en paralelo de múltiples celdas fotovoltaicas encapsuladas para protegerlas del ambiente. Su estructura típica incluye las celdas, una capa de EVA como encapsulante, un vidrio templado frontal, una lámina posterior resistente a la intemperie y un marco de aluminio, lo que garantiza una vida útil de 25 a 30 años con una degradación anual menor al 1% [18]. Esta configuración permite obtener niveles de voltaje y corriente adecuados para aplicaciones prácticas, alcanzando potencias desde unos cuantos watts hasta cientos de watts, dependiendo del número y tipo de celdas utilizadas, como las monocristalinas, policristalinas o de capa fina.

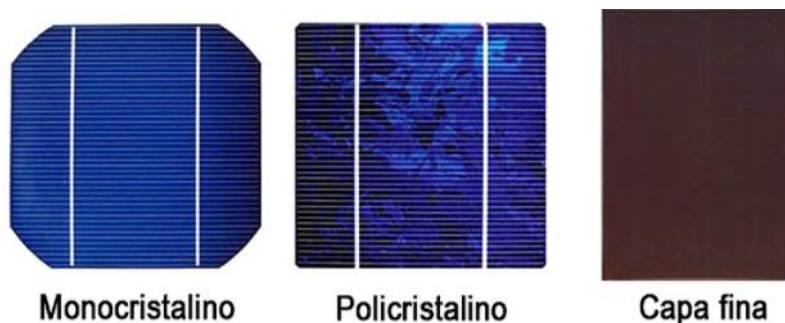


Figura 2. 6: Diferentes tipos de paneles solares.

Fuente: Citado de [22].

2.9.- Sistema fotovoltaico

Un sistema fotovoltaico es un conjunto de dispositivos diseñados para convertir la energía solar en energía mediante módulos fotovoltaicos, almacenarla en baterías, regular su flujo a través de un controlador de carga y suministrarla eficientemente mediante un inversor. Dependiendo de la escala de la instalación, estos sistemas pueden incluir fusibles, protecciones avanzadas equipos de medición y control. Los sistemas fotovoltaicos se clasifican principalmente en autónomos, que operan sin conexión a la red y son comunes en zonas remotas, y en sistemas conectados a la red, que trabajan en conjunto con la infraestructura eléctrica pública.

2.9.1.- Tipos de sistemas fotovoltaicos

Los sistemas solares fotovoltaicos pueden clasificarse en tres categorías principales: sistemas aislados, sistemas conectados a la red y sistemas híbridos. Esta clasificación permite comprender las distintas configuraciones y aplicaciones que pueden adoptar los sistemas FV según sus requerimientos de operación, disponibilidad de infraestructura eléctrica y necesidades energéticas del usuario. En la Figura 2.7 se presenta un diagrama general que resume esta clasificación y sirve como base para describir cada tipo de sistema.



Figura 2. 7: Diagrama general de la clasificación de los sistemas fotovoltaicos.

Fuente: Elaboración propia.

Un sistema fotovoltaico autónomo (SFA) es un conjunto de dispositivos o componentes que aprovechan y utilizan la energía solar para producir electricidad. Estos sistemas funcionan independientemente de la red eléctrica convencional y se utilizan principalmente en áreas

remotas o donde no hay acceso a la red eléctrica. En la Figura 2.8, se muestra cuatro configuraciones más comunes en los SFA.

Los sistemas domésticos (SHS), suelen incorporar únicamente cargas en continua, por esta razón no es necesario que incluya un inversor. Están compuestos por el generador, un acumulador electroquímico y un regulador de carga y descarga (Figura 2.8A). Cuando el consumo incluye cargas de alterna es necesario que el SFA incluya un inversor (Figura 2.8B) [23].

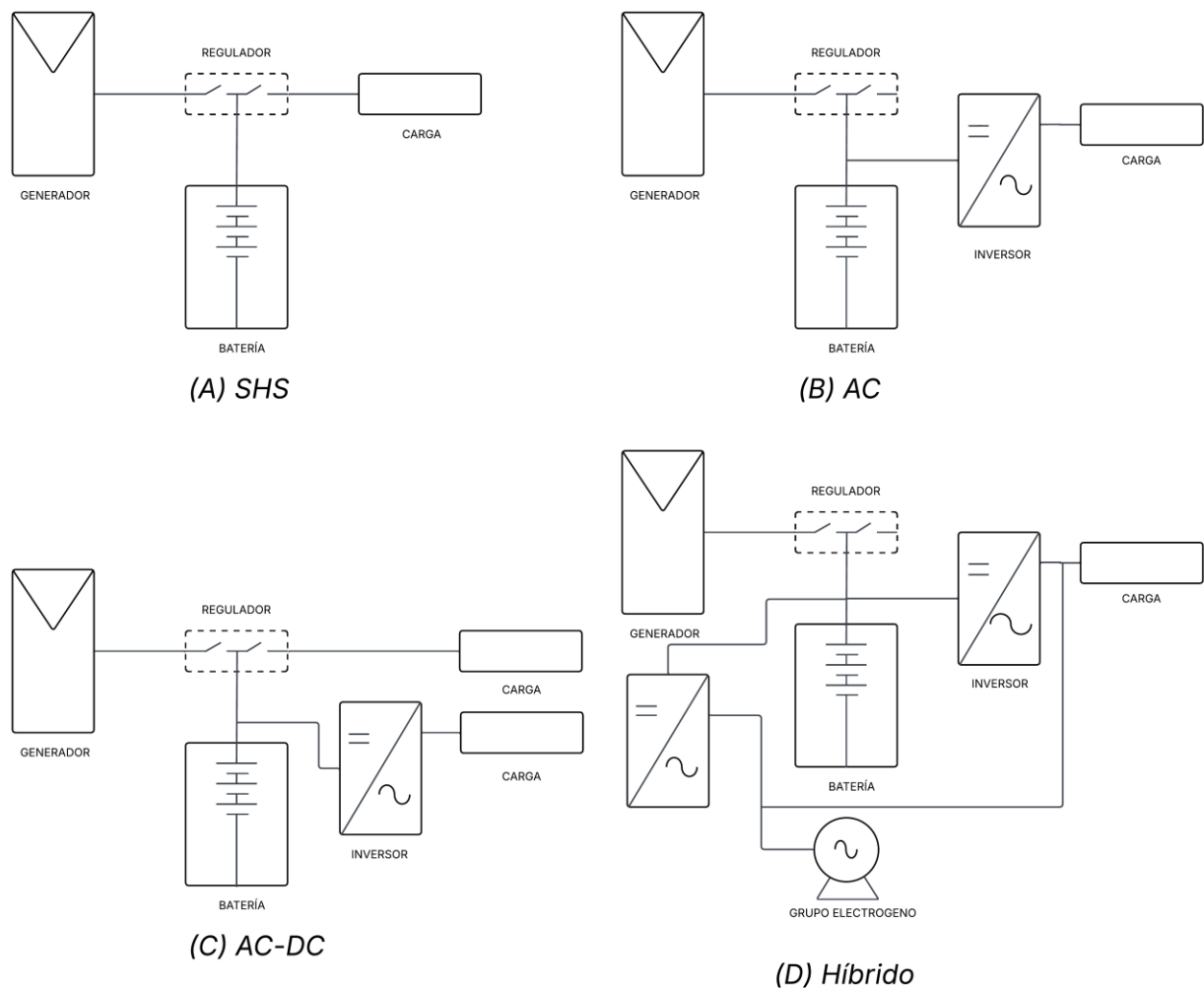


Figura 2. 8: Configuraciones más comunes en los SFA.

Fuente: Citado de [23].

Cabe la posibilidad de que el consumo esté compuesto por cargas en continua y en alterna (Figura 2.8C), o exclusivamente por cargas en alterna. El funcionamiento del inversor puede provocar la circulación de corrientes transitorias que el regulador no puede gestionar correctamente. Por este motivo, es recomendable que el inversor esté conectado directamente a la batería, y no a la salida del regulador. Los inversores para SFA suelen incorporar un mecanismo de regulación de descarga que permite esta conexión. El funcionamiento de estos inversores es, en muchos aspectos, similar al de los inversores de conexión a red, pero con varias peculiaridades. La principal diferencia está en su salida: dado que estos inversores no están conectados a una red con la que sincronizar, deben funcionar como fuentes de tensión (y no como fuentes de corriente, caso habitual en los sistemas fotovoltaicos conectados a la red). Por lo general, no incluyen un buscador del punto de máxima potencia [23].

Como caso especial, destacan los sistemas híbridos (Figura 2.8D). Como se entenderá en la descripción de los métodos de dimensionado de los SFA, existe una probabilidad no nula de fallo de suministro. Así, durante un año típico, es previsible que un porcentaje de la energía demandada por la red de consumo no pueda ser correspondida por el SFA. El dimensionado de un SFA consiste, por tanto, en elegir los tamaños de generador y acumulador como una solución de compromiso entre mínima probabilidad de fallo y mínimo coste [23].

Un sistema fotovoltaico conectado a la red consiste en un generador fotovoltaico acoplado a un inversor que opera en paralelo con la red eléctrica convencional. En la Figura 2.9 se observa un diagrama de los componentes principales de un sistema de conexión a red. Su modo de operación se basa en que el generador fotovoltaico capta la radiación solar y la transforma en energía eléctrica, la cual ya no se almacenará en baterías, como en los sistemas aislados e híbridos, se puede usar directamente en el consumo o entregarla a la red eléctrica de distribución.

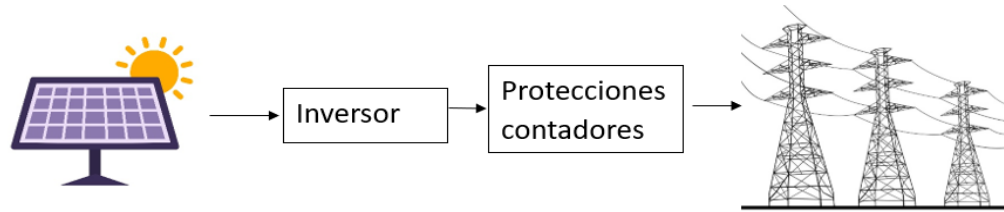


Figura 2. 9: Esquema básico de una instalación FV conectada a la red.

Fuente: Citado de [23].

2.10.- Componentes de un sistema fotovoltaico.

Los inversores fotovoltaicos son dispositivos electrónicos encargados de convertir la energía en corriente directa (CD) generada por los módulos solares en corriente alterna (CA), permitiendo su uso en cargas domésticas, comerciales o su inyección a la red eléctrica [24]. Su principio de funcionamiento se basa en el uso de convertidores electrónicos de potencia que modulan la señal de salida mediante técnicas como PWM para obtener una forma de onda senoidal adecuada para el sistema [27].

Los inversores pueden clasificarse según su configuración en tres tipos principales: inversores conectados a la red, que sincronizan su salida con la red pública; inversores autónomos, utilizados en sistemas aislados con bancos de baterías; e inversores híbridos, capaces de operar tanto con la red como con almacenamiento energético [28]. Además, los inversores modernos incorporan algoritmos de seguimiento del punto de máxima potencia (MPPT), los cuales ajustan continuamente el punto de operación del módulo fotovoltaico para extraer la máxima potencia disponible bajo condiciones variables de irradiancia y temperatura [24]. Gracias a estas funciones, los inversores constituyen un componente esencial para garantizar la eficiencia, estabilidad y aprovechamiento óptimo de la energía generada por los sistemas fotovoltaicos.

2.11.- Sistema de adquisición de datos (SAD).

Un Sistema de Adquisición de Datos (SAD), también conocido por sus siglas en inglés como DAQ (Data Acquisition System), es un conjunto de hardware y software diseñado para capturar, medir, registrar y analizar señales físicas del entorno (como voltaje, corriente, temperatura, irradiancia, y humedad entre otras) en tiempo real o a lo largo del tiempo [24].

Este sistema funciona mediante sensores y transductores que convierte las magnitudes físicas en señales eléctricas analógicas, las cuales son posteriormente acondicionadas (por ejemplo, amplificadas o filtradas) y digitalizadas mediante un convertidor analógico-digital (ADC). Una vez en formato digital, los datos pueden ser procesados, almacenados, visualizados o transmitidos a dispositivos electrónicos u ordenadores para su análisis y toma de decisiones.

De esta forma, un SAD permite transformar fenómenos físicos variables en información digital utilizable, facilitando tareas de monitoreo, control y análisis en aplicaciones técnicas y científicas, como el seguimiento del rendimiento de un panel fotovoltaico.

2.12.- Sensores usados en sistemas fotovoltaicos.

Un sensor se define como un dispositivo que provee una salida utilizable en respuesta a una cantidad física, química o biológica específica de entrada [34]. En esencia, constituyen la interfaz primaria entre el mundo físico y los sistemas electrónicos de medición y control, siendo un componente fundamental en cualquier Sistema de Adquisición de Datos (SAD).

La operación de un sensor se fundamenta en principios de transducción, donde un cambio en la propiedad física (la cantidad medida) se convierte en un cambio en una propiedad eléctrica, como la resistencia, la capacitancia, el voltaje o la corriente [35]. Esta señal eléctrica resultante, a menudo de bajo nivel y susceptible al ruido, requiere típicamente de un proceso de acondicionamiento de señal que incluye amplificación, filtrado y linealización para que sea apta para su digitalización y posterior procesamiento [36].

Existen múltiples criterios para clasificar a los sensores. Una clasificación fundamental los divide en analógicos y digitales. Los sensores analógicos proporcionan una señal de salida continua (generalmente un voltaje o corriente) proporcional a la magnitud medida. Por el contrario, los sensores digitales entregan la información directamente en formato digital,

mediante protocolos de comunicación I²C, SPI o UART, lo que simplifica su interfaz con microcontroladores y reduce la susceptibilidad al ruido en la transmisión [37].

Otra clasificación relevante los categoriza según el principio de funcionamiento y la variable física que miden. Entre los tipos más comunes en aplicaciones de instrumentación y monitoreo se encuentran:

- Sensores de temperatura: Miden la energía térmica. Los principios de funcionamiento comunes incluyen cambios de resistencia (RTD y los termistores), o la generación de un voltaje (termopares) [38].
- Sensores de magnitudes radiométricas: Diseñados para medir la intensidad de la radiación electromagnética. En el contexto de la energía solar, los piranómetros o sensores de irradiancia son cruciales para cuantificar la potencia de la luz solar incidente por unidad de área (W/m²) [35].
- Sensor de humedad: Mide el contenido de vapor de agua en el aire (humedad relativa). Utiliza un elemento capacitivo cuya constante dieléctrica cambia con la humedad [34].

Al seleccionar un sensor para una aplicación específica, es crítico considerar sus especificaciones técnicas. Entre los parámetros más importantes se encuentran[38]:

- Rango: Los valores mínimo y máximo de la variable que puede medir.
- Exactitud y Precisión: La proximidad de la medición al valor real y la repetibilidad de la medida.
- Resolución: El cambio más pequeño en la variable de entrada que puede detectar.
- Sensibilidad: La relación entre el cambio en la señal de salida y el cambio en la variable de entrada.
- Linealidad: La medida en que la curva de respuesta del sensor se aproxima a una línea recta.
- Tiempo de Respuesta: La rapidez con la que la salida del sensor reacciona a un cambio en la entrada.
- Estabilidad: La capacidad de mantener un rendimiento consistente a lo largo del tiempo.

En el ámbito del monitoreo de sistemas fotovoltaicos, la correcta selección de sensores, considerando estas características y su compatibilidad con el sistema de adquisición, es

fundamental para garantizar la fiabilidad y precisión de los datos recabados, los cuales son la base para un análisis robusto del rendimiento y la detección de anomalías.

2.13.- Acondicionamiento de señales.

El acondicionamiento de señal es una etapa fundamental que prepara la salida de un sistema de medición para su posterior procesamiento. Su objetivo es adaptar la señal cruda a los requisitos del siguiente componente, lo que puede incluir: incrementar su nivel mediante amplificación, purificarla mediante filtrado de interferencias, corregir su comportamiento mediante linealización, realizar la conversión analógico-digital (o viceversa), o incluso modificar su naturaleza, como convertir una señal de voltaje en una de corriente [39].

2.14.- XBee.

Los módulos XBee son dispositivos de comunicación inalámbrica desarrollados por Digi International, basados en el estándar IEEE 802.15.4, que define las capas físicas y de control de acceso al medio para redes inalámbricas de área personal de bajo consumo (LR-WPANs) [40]. Estos módulos están diseñados para aplicaciones que requieren un intercambio de datos moderado, baja latencia y una comunicación confiable en distancias que van desde decenas hasta cientos de metros, dependiendo de la versión del módulo y las condiciones del entorno [41].

2.14.1.- Características principales.

Los módulos XBee operan comúnmente en la banda de frecuencia de 2.4 GHz, con una velocidad de transmisión típica de 250 kbps [40]. Entre sus características más relevantes se encuentran:

- Bajo consumo de energía, ideal para aplicaciones autónomas o alimentadas por batería.
- Capacidad de conectar hasta 65,000 dispositivos en una misma red, dependiendo de la topología utilizada.

- Comunicación bidireccional confiable, con soporte para modos de operación como transparente y API (Application Programming Interface) [42].
- Versatilidad en la selección de antenas (chip, cable, RPSMA, etc.) y configuraciones de potencia, como los modelos XBee Pro que ofrecen mayor alcance [43].

2.14.2.- Topologías de red.

Los módulos XBee permiten implementar diversas topologías de red, lo que facilita su adaptación a distintas necesidades de monitoreo y control. Las topologías más comunes son:

- Punto a punto (Peer-to-Peer): Comunicación directa entre dos dispositivos.
- Estrella (Star): Un nodo central (coordinador) se comunica con múltiples nodos periféricos.
- Malla (Mesh): Los nodos pueden comunicarse entre sí y actuar como repetidores para extender el alcance de la red [40].

Las principales topologías empleadas en redes ZigBee, se ilustran en la Figura 2.10, donde se muestran las configuraciones punto a punto, estrella y malla.

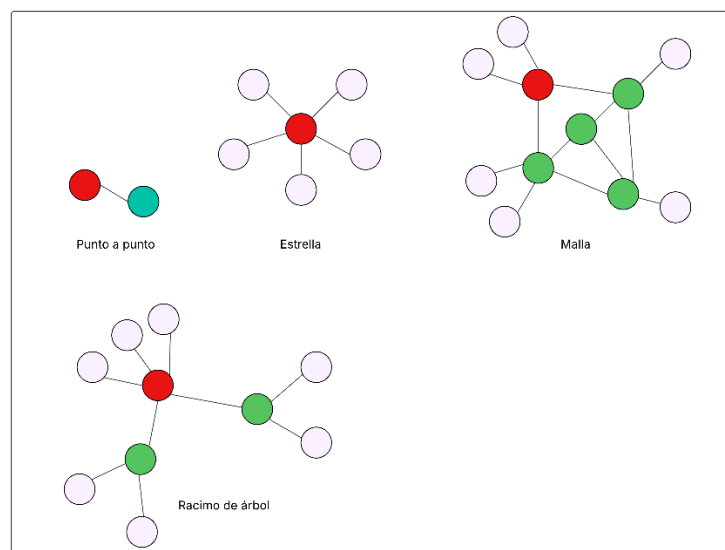


Figura 2. 10: Topologías ZigBee.

Fuente: Adaptado de [44].

Esta flexibilidad topológica es especialmente útil en sistemas de adquisición de datos distribuidos, como el propuesto en esta tesis, donde se requiere recopilar información desde múltiples sensores de forma remota [44].

2.14.3.- Arquitectura de red ZigBee.

Los módulos XBee implementan el protocolo ZigBee, que define tres tipos de dispositivos en una red:

- Coordinador (ZigBee Coordinador): Es el nodo que inicia y gestiona la red, asignado direcciones y gestionando la seguridad.
- Router (ZigBee Router): Actúa como intermediario, retransmitiendo mensajes entre nodos que están fuera del alcance directo.
- Dispositivos Final (End Device): Nodos de bajo consumo que se comunican a través de su nodo padre (coordinador router) [42].

Esta arquitectura jerárquica permite la autoorganización y autoregeneración de la red, características clave para aplicaciones de monitoreo continuo y entornos variables [43] .

2.14.4.- Ventajas y desventajas.

Entre las principales ventajas de los módulos XBee se encuentran:

- Bajo costo en comparación con otras tecnologías inalámbricas.
- Bajo consumo de energía, lo que permite mayor autonomía.
- Facilidad de integración con microcontroladores como Arduino y Raspberry Pi.
- Soporte para redes flexibles y extensibles [40].

Por otro lado, entre sus limitaciones destacan:

- Velocidad de transmisión reducida en comparación con Wi-Fi o Bluetooth.
- Alcance limitado en entornos con obstáculos.
- Complejidad en la configuración de redes mesh en comparación con topologías más simples [40].

2.15.- Raspberry Pi.

La Raspberry Pi es una computadora de placa única (Single-Board Computer-SBC) de bajo costo, desarrollada en el Reino Unido por la Raspberry Pi Foundation [45]. Su objetivo principal es promover la educación en ciencias de la computación y facilitar el acceso a la tecnología digital en entornos educativos, industriales y de desarrollo experimental. Gracias a su diseño compacto, bajo consumo energético y versatilidad, se ha consolidado como una plataforma de referencia para proyectos de Internet de las Cosas (IoT), automatización, robótica y adquisición de datos.

En el contexto de este proyecto, la Raspberry Pi actúa como el núcleo de procesamiento y control del sistema de monitoreo. Sus capacidades de cómputo permiten ejecutar sistemas operativos basados en Linux, lo que facilita la implementación de entornos de programación y aplicaciones especializadas. Entre sus características más relevantes para esta aplicación se destacan:

- Conectividad y expansión: Cuenta con un conector GPIO de 40 pines que habilita interfaces de comunicación como I2C, SPI, UART, esenciales para la interconexión con sensores y módulos externos como el ADC AD1115 y el sensor DHT11.
- Capacidades de red: Integra puerto Ethernet y conectividad Wi-Fi y Bluetooth, lo que permite el monitoreo remoto y la integración con servicios en la nube o dashboards accesibles desde la red local.
- Bajo consumo energético: Ideal para aplicaciones autónomas o alimentadas por sistemas fotovoltaicos, ya que opera con tensiones de 5V y consume una potencia reducida en comparación con sistemas computacionales convencionales.
- Ecosistemas de software: Es compatible con una amplia gama de entornos de desarrollo y herramientas de código abierto, entre los que destaca Node-RED, utilizado en este proyecto para implementación de la lógica de adquisición, procesamiento y visualización de datos.

Estas características convierten a la Raspberry Pi en una plataforma idónea para el desarrollo de prototipos en el ámbito de las energías renovables, lo que permite integrar hardware y software de manera económica, flexible y escalable. Su uso en este sistema, cuyo componente central se muestra en la Figura 2.11, no solo reduce los costos de implementación,

sino que también facilita la adaptación a futuras mejoras o la replicación del proyecto en otros contextos académicos o experimentales.

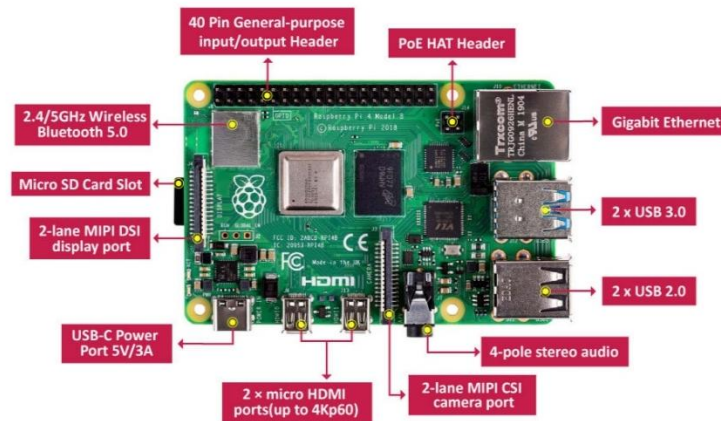


Figura 2. 11: Raspberry Pi 4.

Fuente: Citado de [45].

2.16.- Pantallas Nextion.

Las pantallas Nextion representan una solución integral en el ámbito de las interfaces hombre-máquina (HMI, por sus siglas en inglés), ya que combinan una pantalla táctil LCD, un controlador y memoria en un solo dispositivo. Un ejemplo de esta pantalla se muestra en la Figura 2.12. A diferencia de las pantallas convencionales que requieren un control constante y el envío de grandes cantidades de datos desde un microcontrolador principal (como Arduino o Raspberry Pi), Nextion funciona de manera más inteligente. Su principal ventaja radica en que puede almacenar los elementos gráficos de la interfaz (botones, imágenes, gráficos) en su memoria interna y manejar localmente las interacciones táctiles. Esto libera de una carga significativa de procesamiento a la unidad central, con la que se comunica mediante un protocolo serie simple (UART), intercambiando solo comandos de alto nivel en lugar de datos de píxeles crudos [46].



Figura 2. 12: Pantalla Nextion 7 pulgadas.

Fuente: Citado de [46].

2.16.1.- Nextion Editor.

Nextion Editor es el entorno de desarrollo integrado (IDE) oficial y especializado para el diseño, configuración y programación de las pantallas Nextion. Según el fabricante, este software está específicamente diseñado para implementar una filosofía de desarrollo “What You See Is What You Get” (WYSIWYG), permitiendo crear interfaces de usuario complejas a través de una interfaz visual de arrastrar y soltar [47].

La arquitectura de trabajo del editor se basa en dos pilares fundamentales la pantalla de diseño visual y el editor de código de eventos. En la primera, el desarrollador coloca y configura objetos (botones, textos, gráficos) que componen la interfaz. En el segundo, se programa la lógica de interacción asociando scripts en el lenguaje de comandos Nextion a eventos táctiles de cada objeto. Estos comandos permiten, por ejemplo, actualizar el valor de un display numérico o enviar datos a través del puerto serie cuando un usuario presiona un botón, facilitando la creación de interfaces dinámicas para monitoreo en tiempo real [48].

2.17.- Internet de las cosas (IoT).

El término “Internet de las Cosas” (IoT) fue acuñado por Kevin Ashton en 1999 durante sus investigaciones en el Auto ID Center del MIT, donde exploraba la identificación por radiofrecuencia (RFID) y tecnologías de sensores emergentes [49]. Sin embargo, el concepto se

popularizó aproximadamente una década después, con el desarrollo masivo de tecnologías inalámbricas, sensores de bajo costo y la expansión global de Internet.

El Internet de las cosas se define como una red global e interconectada de objetos físicos o “cosas” equipados con tecnologías que les permite detectar, comunicar y compartir datos a través de Internet, sin requerir necesariamente la intervención humana directa [50]. Estos objetos, que pueden ser desde dispositivos domésticos hasta maquinaria industrial, se convierten en elementos inteligentes capaces de interactuar tanto con el entorno físico como con el mundo digital. En la Figura 2.13 se puede observar algunos dispositivos en los cuales se pueden implementar el internet de las cosas.



Figura 2. 13: Internet de las cosas.

Fuente: Adaptado de [51].

2.17.1.- Componentes fundamentales del IoT.

Un sistema IoT típico integra cuatro componentes tecnológicos esenciales:

1. Dispositivos y Sensores: Elementos físicos que capturan información del entorno. Incluyen sensores (temperatura, humedad, movimiento, etc.), actuadores (mecanismos que ejecutan acciones) dispositivos de identificación.
2. Conectividad: Medios de comunicación que permiten la transmisión de datos. Puede ser tecnologías cableadas (Ethernet) o inalámbricas (Wi-Fi, Bluetooth, ZigBee, celular), seleccionadas según los requisitos de alcance, consumo energético y ancho de banda.

3. Procesamiento de Datos: Infraestructura que recibe, almacena y analiza la información recolectada. Incluyendo plataformas en la nube, servidores y software especializado que transforma los datos crudos en información útil.
4. Interfaz de Usuario: Medios through los cuales los usuarios interactúan con el sistema, pudiendo ser aplicaciones móviles, portables web, pantallas táctiles o sistemas de notificación

2.17.2.- Características Esenciales.

Los sistemas IoT se caracterizan por:

- Conectividad Omnipresente: Capacidad de mantener conexión constante con la infraestructura de Internet.
- Inteligencia Integrada: Procesamiento local o distribuido que permite cierta autonomía decisional.
- Heterogeneidad: Capacidad de integrar múltiples tecnologías y protocolos diferentes.
- Escalabilidad: Posibilidad de expandir el sistema añadiendo nuevos dispositivos sin afectar su funcionamiento.
- Interoperabilidad: Capacidad de diferentes sistemas y dispositivos de trabajar conjuntamente.
- Seguridad y Privacidad: Consideraciones críticas dada la sensibilidad de los datos recolectados.

2.17.3.- Ámbitos de Aplicación.

El IoT ha encontrado aplicación en numerosos dominios, transformando industrias completas:

- Ciudades inteligentes: Gestión de tráfico, alumbrado público, estacionamiento inteligente.
- Salud: Monitoreo remoto de pacientes, historiales médicos digitales, telemedicina.
- Industria 4.0: Mantenimiento predictivo, optimización de cadenas de suministros, automatización industrial.
- Agricultura: Monitoreo de cultivos, riego automatizado, gestión de invernaderos.

- Hogares Inteligentes: Domótica, seguridad automatizada, eficiencia energética.
- Energía: Redes eléctricas inteligentes (Smart Grids), monitoreo de infraestructura, optimización de consumo.

2.18.- Firebase Realtime Database.

Firebase Realtime Database es un servicio de base de datos en la nube, proporcionado por Google, que permite almacenar y sincronizar datos en tiempo real entre todos los clientes conectados. A diferencia de las bases de datos SQL tradicionales, es una base de datos NoSQL que almacena la información en una estructura de árbol JSON, lo que facilita su uso en aplicaciones que requieren una sincronización de datos inmediata y bidireccional [52]. Como podemos observar en la Figura 2.14 las funciones que tiene el servicio de la base de datos Firebase.

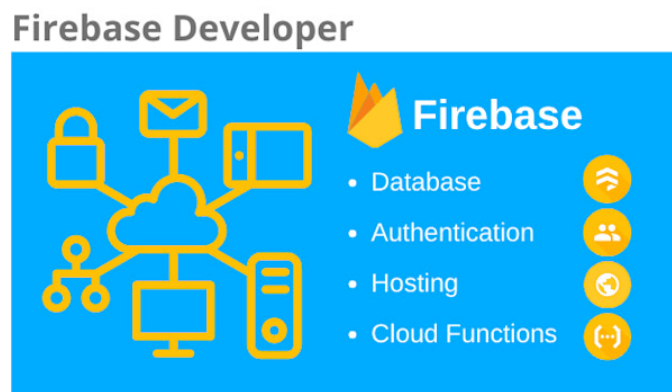


Figura 2. 14: Desarrollo del Firebase.

Fuente: Adaptado de [52].

2.18.1.- Características Principales.

Las características que hacen de Firebase Realtime Database una solución ideal para aplicaciones de monitoreo en tiempo real, como la desarrollada en este proyecto, son:

- Sincronización en Tiempo Real: Cualquier cambio en los datos se propaga instantáneamente a todos los clientes conectados (dispositivos, navegadores web) en cuestión de milisegundos.

- Arquitectura sin Servicio (Serverless): Firebase elimina la necesidad de gestionar infraestructura de servidores, lo que reduce la complejidad y el tiempo de desarrollo. El desarrollador se enfoca en la lógica de la aplicación mientras Google gestiona la escalabilidad y disponibilidad del servicio.

2.19.- Estado del Arte

En los últimos años se han propuesto varias metodologías para supervisar las variables que influyen en el rendimiento de los sistemas fotovoltaicos. Algunas de las últimas contribuciones son sistemas basados en microcontroladores y tecnologías inalámbricas para la captura y transmisión de datos. Por ejemplo, se han desplegado redes de sensores inalámbricos con módulos XBee para monitorizar de forma remota paneles solares y medir voltaje, corriente y temperatura [53]. Otras arquitecturas combinan edge computing con Raspberry Pi para procesamiento local y visualización en tiempo real [54]. También se han creado sistemas de monitorización en la nube con IoT para medir el rendimiento fotovoltaico, almacenar datos históricos y permitir el análisis remoto [55].

Además, se han propuesto sistemas IoT para medir variables ambientales como irradiancia y temperatura, así como parámetros eléctricos, para estimar la eficiencia de los paneles [56]. Revisiones exhaustivas de tecnologías de monitoreo para sistemas FV resaltan el uso de protocolos de comunicación inalámbrica y módulos de procesamiento de datos, haciendo énfasis en la exactitud de la medición [57]. Otras técnicas comprenden trazadores I-V en línea para el mantenimiento predictivo de paneles [58] y sistemas portátiles para medir la eficiencia de paneles solares en condiciones reales [53].

En instrumentación, se han desarrollado sistemas de bajo costo con Arduino para toma de datos en sistemas fotovoltaicos autónomos [54], mientras que otros estudios han incorporado sensores específicos para medir la irradiancia solar [59]. Las últimas propuestas utilizan IA para predecir el rendimiento y detectar fallas con redes neuronales y técnicas de minería de datos [60][61]. Finalmente, se han creado sistemas de monitoreo inteligente que integran mediciones en tiempo real con almacenamiento en la nube para evaluar de forma continua el rendimiento de los paneles fotovoltaicos [62][63].

Capítulo 3

Metodología

En este capítulo se describe la arquitectura completa del sistema dividida en tres subsistemas: el primero es el sistema de adquisición de datos (SAD) con el Arduino Uno y sensores especializados para la medición de variables eléctricas y ambientales. El siguiente subsistema se refiere al sistema de comunicación inalámbrica mediante módulos XBee Pro S2 en topología punto a punto. Y el tercer subsistema es la plataforma de procesamiento con Raspberry Pi4.

3.1.- Arquitectura general del sistema.

La metodología consta de diversas etapas secuenciales, desde el análisis inicial del sistema hasta la validación experimental del prototipo funcional. En la Figura 3.1 se muestra el diagrama general de la Metodología aplicada. Esta etapa se visualiza mediante una interfaz gráfica, implementada en una pantalla touch Nextion, que permite observar la correlación entre variables ambientales y eléctricas para determinar el comportamiento del panel, generando una base de datos en la memoria interna y cargándola posteriormente a una base de datos en la nube para evaluar el rendimiento del panel.

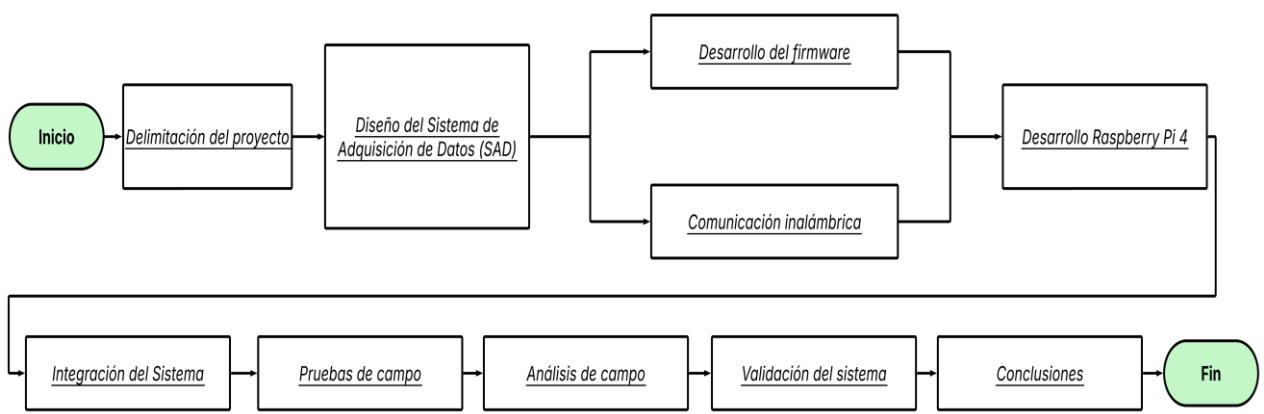


Figura 3. 1: Diagrama general de la Metodología implementada.

Fuente: Elaboración propia.

El sistema propuesto, cuya arquitectura se muestra en la Figura 3.2, se compone de tres subsistemas principales interconectados:

1. Sistema de adquisición de datos (SAD): Compuesto por sensores especializados, un microcontrolador Arduino Uno y un acondicionamiento de señal, para la medición de variables eléctricas y ambientales.
2. Sistema de comunicación inalámbrica: Implementado mediante módulos XBee Pro 2SC configurados en topología punto a punto, que permite la transmisión remota de datos desde el Arduino hacia una Raspberry Pi.
3. Sistema de procesamiento y visualización: Desarrollado en una Raspberry Pi 4, que recibe, almacena y procesa los datos, y los presenta en una interfaz gráfica mediante una pantalla Nextion.

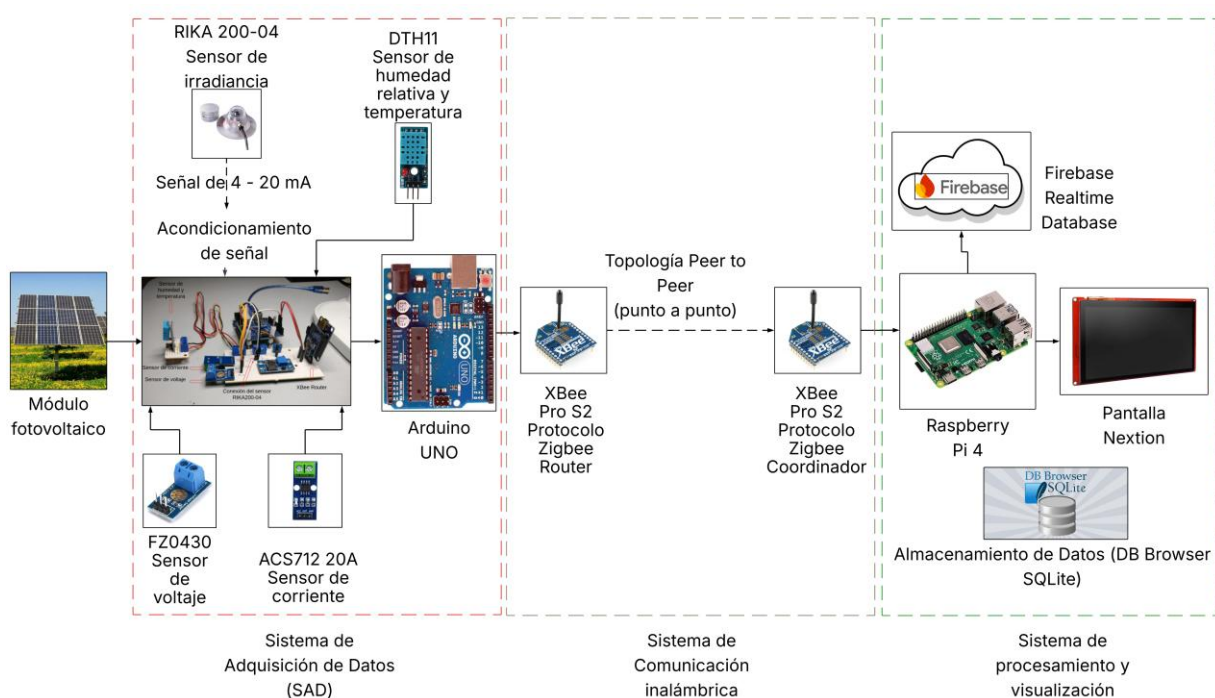


Figura 3. 2: Arquitectura general del sistema.

Fuente: Elaboración propia.

Esta arquitectura permite visualizar la relación entre los módulos de adquisición, procesamiento, comunicación y almacenamiento de datos, lo cual sirve como base para las etapas descritas en este capítulo.

3.2.- Propuesta para la estructura del Sistema de Adquisición de Datos (SAD).

Como se mencionó en la Figura 3.2 se presenta el desarrollo del prototipo para el monitoreo de un módulo fotovoltaico, el cual en la primera fase se llevó un análisis de las variables clave de un módulo fotovoltaico, enfocándose en parámetros eléctricos como voltaje, corriente y potencia, además de factores ambientales como temperatura, humedad relativa e irradiancia, todos esenciales para la eficiencia de los paneles o módulos fotovoltaicos. Estas fases se abordaron de manera secuencial, pero con un enfoque iterativo, lo que permitió realizar ajustes durante la integración del hardware con el software. Con base en este estudio, se

seleccionaron los sensores más adecuados y se diseñó un sistema de adquisición de datos compatible con Arduino Uno.

Esta arquitectura propuesta integra los siguientes componentes principales los cuales se muestran en la Tabla 3.1.

Tabla 3.1: Componentes principales del proyecto.

Componentes	Descripción
<i>Panel solar</i>	Fuente de energía renovable que convierte la radiación solar en energía eléctrica de corriente directa. Su desempeño (voltaje y corriente) es monitoreada directamente para caracterizar su eficiencia y comportamiento bajo diferentes condiciones ambientales.
<i>Sensor de voltaje FZ0430</i>	Módulo aislante basado en un amplificador de precisión que mide altos voltajes DC de forma segura. Con un rango de entrada de 0-25V (o ampliable hasta 250V con un divisor de voltaje externo), proporciona una salida analógica lineal de 0-5V, aislada eléctricamente del circuito de medición para proteger la Raspberry Pi.
<i>Sensor de corriente ACS712T20A</i>	Circuito integrado que mide la corriente del panel de forma no intrusiva y aislada galvánicamente. Convierte la intensidad medida en un voltaje analógico de referencia central, protegiendo el resto de la electrónica y facilitando la interfaz con el microcontrolador.
<i>Sensor DHT11</i>	Dispositivo digital que mide la temperatura ambiente y la humedad relativa en el entorno del panel solar. Proporciona mediciones digitales calibradas, lo que simplifica su conexión directa con el Arduino Uno.
<i>Sensor de irradiancia RIK-A 200-04</i>	Sensor de radiación solar global diseñado para aplicaciones meteorológicas y fotovoltaicas. Mide la densidad de potencia de la radiación (W/m^2) que incide directamente sobre el panel, proporcionando el dato fundamental para correlacionar la entrada de energía solar con la salida eléctrica generada.
<i>Arduino Uno</i>	Microcontrolador central que actúa como interfaz entre los sensores (DHT11, ACS712, FZ0430, RK 200-04) y la Raspberry Pi. Se encarga de leer todas las señales analógicas y digitales, preprocesar los datos y enviarlos mediante comunicación serial a la unidad principal de procesamiento.
<i>Módulos XBee Pro S2C</i>	Dispositivos de comunicación que establecen un enlace de comunicación inalámbrico punto a punto entre el Arduino Uno (Router) y la

	Raspberry Pi (Coordinador). Facilitan la transmisión remota de los datos adquiridos por los sensores, permitiendo el monitoreo a distancia del comportamiento del panel fotovoltaico.
<i>Raspberry Pi 4</i>	Unidad central de procesamiento que recibe datos vía XBee, implementa el esquema de almacenamiento dual (SQLite local + Firebase), controla la interfaz Nextion para visualización en tiempo real, gestiona la lógica de promediado cada 1 minuto durante un horario establecido.
<i>Firebase Realtime Database</i>	Servicio de base de datos en la nube Google que proporciona almacenamiento remoto y sincronización en tiempo real de los datos del panel solar, funcionando como respaldo del almacenamiento local SQLite y permitiendo acceso a los datos históricos.
<i>Pantalla Nextion</i>	Interfaz de usuario local que visualiza los parámetros operativos del módulo fotovoltaico, mostrando simultáneamente las seis variables críticas para evaluación instantánea del desempeño.

El panel fotovoltaico seleccionado para implementar en el proyecto es un panel solar multicristalino, con una potencia máxima de 100W, el cual se encuentra localizado en el Laboratorio de Maestría en Ingeniería Eléctrica del Campus II del Instituto Tecnológico de Ciudad Madero.

3.3.- Adaptación de variables.

El sistema emplea sensores específicos para cada variable. Para medir el voltaje se utiliza el sensor FZ0430 el cual permite medir tensiones de hasta 25 V de CD, cuenta con una señal de analógica la cual se puede conectar a cualquier pin de entrada analógica del Arduino, para la corriente se utiliza el sensor ACS712 20A, que permite medir hasta 20 A. Parte de las variables ambientales consideradas esta la irradiancia, la cual se utiliza el sensor de irradiancia RIK-A 200-04, tiene un rango de 0 a 1500 W/m² y proporciona una salida de 4-20mA, para poder lograr la comunicación con el Arduino Uno, esta salida debe de pasar por una etapa de acondicionamiento, para convertir la salida en un voltaje de 0 a 5 V. Para la temperatura y la humedad se usa el módulo DHT11, el cual tiene la capacidad de proporcionar simultáneamente la temperatura y la humedad, contando con un rango de operación de 0°C a 50°C para la temperatura y un rango de operación de entre 20% y 90% para la humedad relativa, su voltaje de operación es de 3.3V a 5V, incorpora un sensor analógico, el cual realiza la conversión

analógica-digital dentro del dispositivo. Tal como se observa en la Figura 3.3, todos los sensores se encuentran conectados al Arduino Uno, que recibe las señales de salida de los instrumentos para procesar la información acumulada.

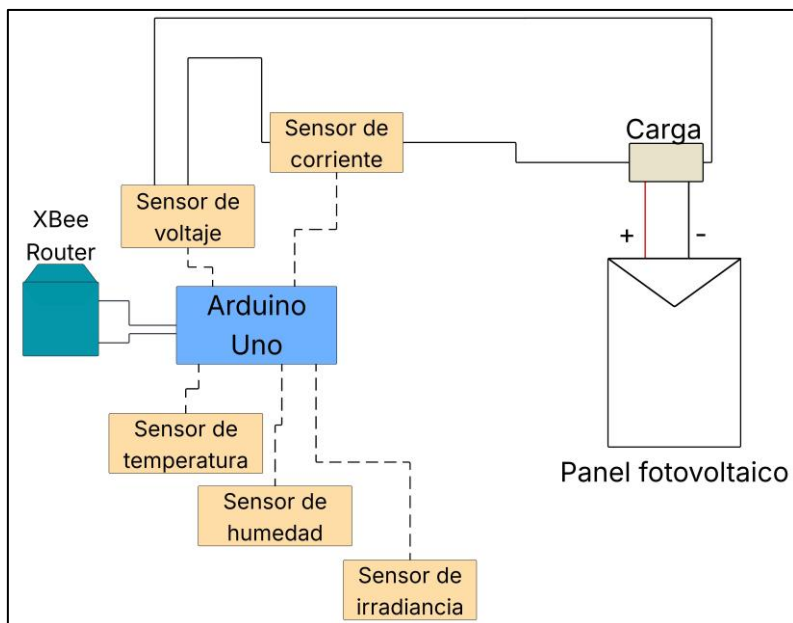


Figura 3. 3: Circuito electrónico del SAD.

Fuente: Elaboración propia.

3.4.- Comunicación a distancia mediante XBee.

La plataforma de Arduino Uno, que incorpora un microcontrolador ATmega328P, se empleó para la transmisión inalámbrica de los datos adquiridos. El firmware, elaborado en el Entorno de Desarrollo Integrado (IDE) y redactado en C++, se encarga de la lectura de las variables analógicas y digitales de los sensores conectados. El microcontrolador transforma las señales analógicas en valores digitales mediante el convertidor analógico-digital (ADC) interno. Después, estos valores son sometidos a procesamiento, convertidos a las unidades requeridas (Voltios, Amperios, Potencia, Grados Centígrados, % de humedad relativa, watts sobre metros cuadrados) y estructurados en una trama o cadena de texto organizada.

Esta trama se transmite por radiofrecuencia de manera inalámbrica a través de un módulo XBee configurado como Router, que se conecta al Arduino mediante una comunicación serial. Para la configuración punto a punto empleada, la estructura de la trama no se adhiere al protocolo particular de tramas API de XBee, sino que se establece como una secuencia de

caracteres legible para facilitar el desarrollo (por ejemplo: V:12.50,I:1.230,P:15.38,T:25.00,H:45.00,R:850.50). Este procedimiento se lleva a cabo de manera cíclica en el bucle principal del software, facilitando la supervisión en tiempo real del estado del panel solar. La Figura 3.4 presenta el diagrama de flujo que ilustra exhaustivamente este proceso.

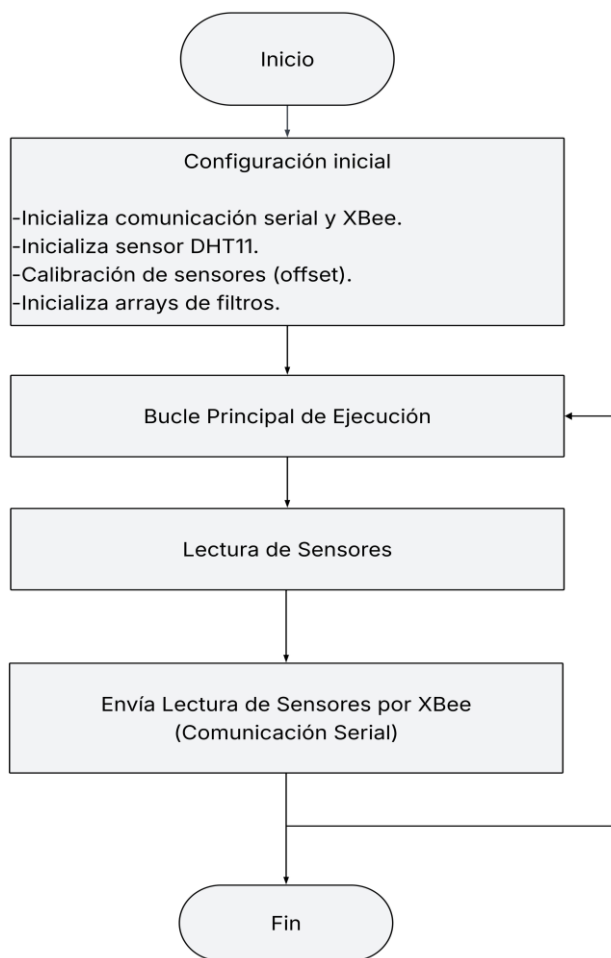


Figura 3. 4: Diagrama de flujo de la lógica de control implementada en el subsistema de adquisición de datos.

Fuente: Elaboración propia.

3.5.- Comunicación Punto a Punto con XBee Serie 2.

La topología punto a punto se refiere a la conexión directa entre dos nodos a través de un enlace dedicado, sin la intervención de intermediarios., esto implica que un nodo emisor equipado con sensores transmite información de manera inalámbrica a un nodo receptor (la Raspberry Pi) a través de módulos XBee. Esta conexión dedicada impide la utilización de

recursos compartidos y canales intermedios, garantizando una privacidad óptima y minimizando la saturación de la red. Los módulos XBee Pro-Serie 2 (ZigBee) respaldan de manera óptima este modelo: por ejemplo, Digi señala que la serie S2C proporciona una comunicación "rápida y robusta" en configuraciones punto a punto.

La topología punto a punto fue seleccionada debido a su sencillez y eficacia en un sistema con únicamente dos nodos significativos (un panel sensor y la base de datos en la Raspberry Pi). Al disponer de un canal exclusivo, se alcanza una velocidad de transmisión elevada y una latencia baja, dado que no se entra en competencia con el tráfico procedente de otros dispositivos. Adicionalmente, la comunicación directa se caracteriza por su mayor seguridad: no existen vías intermedias para la introducción de errores o interferencias, lo que asegura la integridad de los datos delicados de seguimiento.

Para establecer la red punto a punto se configura un XBee como coordinador ZigBee y el otro como router. Ambos deben compartir el mismo PAN ID. Con esta configuración, cualquier dato enviado por serie desde el nodo es transmitido automáticamente por RF al XBee receptor conectado a la Pi.

Esta topología satisface las metas de un monitoreo eficaz. El coordinador XBee, conectado a la Raspberry Pi, recibe de manera constante la información proveniente de los sensores de manera inalámbrica. La eliminación de saltos de ruta reduce la latencia y optimiza el ancho de banda disponible, facilitando el procesamiento y almacenamiento acelerado de los datos de cada panel. Esto cumple con la exigencia de transferir los registros hacia la Raspberry Pi de manera fiable y puntual para su análisis y visualización.

3.6.- Procesamiento en la Raspberry.

El módulo XBee configurado como Coordinador se conecta a uno de los puertos USB de la Raspberry Pi, la cual funge como nodo central encargado del procesamiento y gestión de los datos adquiridos. En su interior se ejecuta un programa desarrollado en Python que recibe de manera continua los paquetes provenientes del XBee. Dicho programa se encarga de interpretar y desempaquetar la información enviada por el nodo de sensado, transformándola a valores numéricos legibles para su posterior almacenamiento y visualización.

Una vez decodificados, los datos se procesan internamente para asegurar que cada variable conserve su formato y unidad correspondiente; de esta forma, la Raspberry Pi actúa como un concentrador que organiza los valores de voltaje, corriente, potencia, temperatura, humedad e irradiancia antes de enviarlos a las plataformas de presentación. Tal como se observa en la Figura 3.5, la información puede visualizarse directamente en una pantalla Nextion, la cual recibe desde la Raspberry Pi los valores procesados para mostrarlos en tiempo real mediante una interfaz HMI.

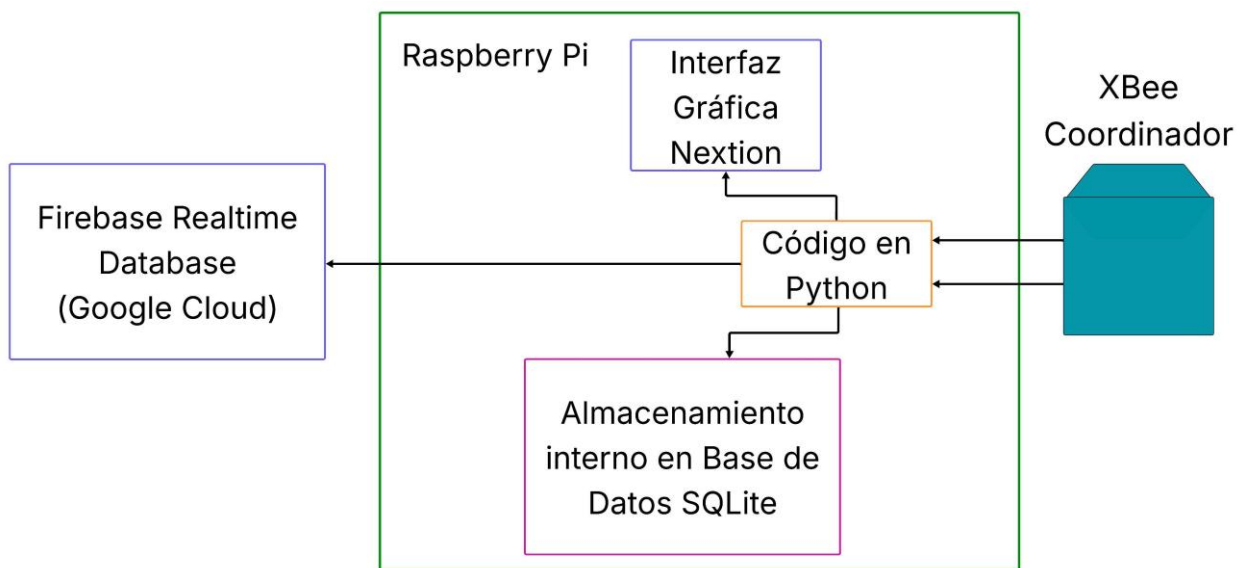


Figura 3. 5: Procesamiento y Gestión de Datos en la Raspberry Pi.

Fuente: Elaboración propia.

Además de la visualización local la pantalla Nextion, la Raspberry Pi permite consultar los datos de forma remota mediante su sincronización con Firebase Realtime Database. El

programa en Python replica cada muestra recibida desde el XBee en la base de datos en la nube, lo que posibilita acceder a la información desde cualquier dispositivo con conexión a Internet.

De manera simultánea, el sistema almacena todas las mediciones en una base de datos local SQLite, generada automáticamente durante la primera ejecución. Este archivo mantiene un registro histórico independiente de la disponibilidad de red, garantizando la conservación de los datos aun cuando no exista conexión a Internet.

La Raspberry Pi puede administrarse sin periféricos adicionales mediante acceso remoto dentro de la misma red, lo que facilita su operación y mantenimiento. Con esta arquitectura, el procesamiento en la Raspberry Pi asegura la organización, almacenamiento y disponibilidad continua de la información adquirida, cumpliendo con los objetivos de visualización y transmisión eficiente de los datos provenientes del panel fotovoltaico.

3.6.1.- Interfaz Gráfica Nextion.

La pantalla Nextion funciona como el medio principal de visualización local del sistema, permitiendo monitorear en tiempo real las variables eléctricas y ambientales procesadas por la Raspberry Pi. Su función metodológica consiste en presentar la información de manera clara y accesible, evitando que el usuario interactúe directamente con la consola o con los procesos internos del sistema.

La interfaz se compone de elementos HMI que reciben, mediante comunicación serial, los valores numéricos enviados por la Raspberry Pi. Estos valores corresponden a voltaje, corriente, potencia, temperatura, humedad e irradiancia, los cuales se actualizan dinámicamente conforme se procesan los paquetes provenientes del XBee Coordinador. De esta forma, la pantalla actúa como un dispositivo de monitoreo inmediato, proporcionando retroalimentación visual continua sin necesidad de periféricos adicionales.

La pantalla Nextion se incorpora como el elemento destinado a la visualización local de las variables procesadas, ofreciendo una lectura continua y accesible del comportamiento del panel fotovoltaico sin necesidad de intervención técnica.

3.6.2.- Base de Datos SQLite.

Para garantizar un almacenamiento local confiable, se implementa una base de datos embebida mediante SQLite dentro de la Raspberry Pi, la cual funciona como repositorio

principal de las mediciones adquiridas. Su selección responde a su eficiencia en la ejecución, autonomía y compatibilidad con sistemas embebidos, permitiendo conservar los datos incluso cuando no existe conexión a Internet.

La estructura de la base de datos contempla una tabla destinada a registrar las variables eléctricas y ambientales del panel fotovoltaico junto con su marca de tiempo, lo que asegura una trazabilidad ordenada y facilita el análisis posterior del comportamiento energético. Los registros se almacenan conforme a un intervalo de actualización definido, evitando redundancia y manteniendo un historial representativo de las condiciones monitoreadas.

En conclusión, el SQLite se integra como el mecanismo de almacenamiento local que garantiza la conservación estructurada y continua de las mediciones, fortaleciendo la confiabilidad del proceso metodológico de adquisición de datos.

3.6.3.- Firebase Realtime Database.

En el diseño del sistema se incorpora Firebase Realtime Database como plataforma de almacenamiento en la nube, para ampliar las capacidades de consulta del monitoreo. Su integración permite que los datos procesados por la Raspberry Pi puedan visualizarse desde cualquier dispositivo con conexión a Internet, sin depender del acceso físico al prototipo.

Firebase destaca por su capacidad de sincronización continua, lo que resulta adecuado para sistemas que requieren actualizaciones en tiempo real. Su estructura basada en objetos facilita organizar las variables eléctricas y ambientales del panel fotovoltaico en un formato coherente y accesible desde aplicaciones web o móviles. La Raspberry Pi envía cada muestra al servicio en un formato estructurado, asegurando la integridad de la información y permitiendo su lectura inmediata.

Mientras SQLite asegura la conservación local de los datos aun sin conexión a Internet, Firebase proporciona disponibilidad remota continua, permitiendo consultar el comportamiento del panel desde cualquier ubicación.

Capítulo 4

Implementación y validación experimental

Este capítulo se detalla las fases de puesta en marcha del sistema de seguimiento desarrollado, junto con los procedimientos empleados para corroborar su correcto funcionamiento en condiciones reales.

4.1.- Introducción al desarrollo experimental

Utilizando la metodología propuesta en el capítulo anterior, se llevó a cabo la integración física de los componentes del sistema, la configuración de los módulos de comunicación inalámbrica XBee Serie 2, la programación del procesamiento central en la Raspberry Pi y la implementación de los mecanismos de visualización y almacenamiento mediante la pantalla Nextion, la base de datos SQLite y la plataforma Firebase Realtime Database.

La implementación permitió corroborar la integración coherente de los elementos establecidos en la metodología en una arquitectura funcional, capaz de recibir, procesar y exhibir en tiempo real las variables eléctricas y ambientales del panel fotovoltaico. En consecuencia, se llevó a cabo una evaluación del rendimiento de la comunicación directa entre el nodo de censado y la Raspberry Pi, de la adecuada actualización de la interfaz gráfica, de la integridad del almacenamiento local y de la sincronización de registros en la nube.

A continuación, se realizó una validación experimental con el objetivo de asegurar que el sistema respondiera adecuadamente a las condiciones de funcionamiento prescritas. Este procedimiento involucró evaluaciones de la estabilidad de la transmisión de datos, la continuidad del registro durante el periodo solar y el análisis del comportamiento del sistema en situaciones de posibles interrupciones de comunicación. La evaluación de usabilidad se realizó mediante el instrumento UMUX-Lite, con el fin de cuantificar la percepción del usuario sobre la interfaz implementada en la pantalla Nextion.

Las conclusiones derivadas de esta fase permiten determinar el nivel de logro de los objetivos propuestos y la eficacia del sistema de supervisión del panel fotovoltaico, aportando pruebas cuantitativas y cualitativas sobre el rendimiento del prototipo final.

4.2.- Sistema de Adquisición de Datos.

El sistema de adquisición constituye la base del monitoreo desarrollado, ya que se encarga de medir las variables eléctricas y ambientales del panel fotovoltaico y de transmitirlos de forma inalámbrica al nodo coordinador. Para garantizar la confiabilidad de las mediciones, el proceso inició con la evaluación individual de cada sensor en una tabllita de pruebas. En la Figura 4.1 se muestra el montaje utilizado en esta etapa, en el que se conectaron el sensor de voltaje, el sensor de corriente, el sensor de irradiancia, así como el sensor de temperatura y humedad.

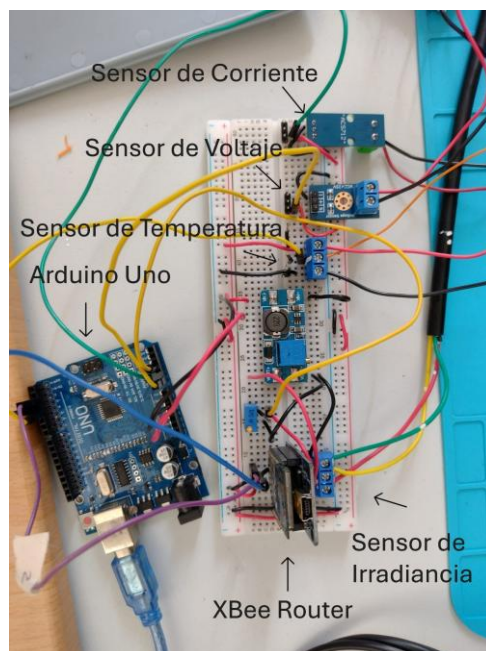


Figura 4. 1: Prototipo SAD.

Fuente: Elaboración propia.

Con el objetivo de validar el funcionamiento de cada dispositivo, se realizaron pruebas controladas. En el caso del sensor de voltaje y del sensor de corriente, se empleó una fuente de alimentación variable para observar la respuesta del sistema. Para el sensor de corriente se utilizó una carga de $150\ \Omega$ y un multímetro conectado en serie, lo que permitió comparar las mediciones reales con las obtenidas por el Arduino y ajustar los parámetros de calibración.

De acuerdo con las hojas de datos, ambos sensores requieren la corrección de un valor base (offset), por lo que el programa implementado calcula este valor automáticamente al inicio del sistema, eliminando el error introducido por el divisor resistivo en el caso de voltaje y por el punto de referencia interno en el sensor de corriente.

En cuanto al sensor DHT11, encargado de medir temperatura y humedad relativa, también se realizaron ajustes tomando como referencia instrumentos comerciales. Debido a que estos sensores presentan desviaciones notables, se aplicaron correcciones de offset tanto para la temperatura como para la humedad, obteniendo mediciones más precisas durante la fase de operación.

La integración del sensor de irradiancia RIKA 200-04 requirió un proceso adicional de acondicionamiento de señal. Este dispositivo tiene una salida de 4–20 mA, por lo que no puede ser leído directamente por el Arduino. Para resolverlo, se empleó una resistencia de 250 Ω , que convierte la corriente en un voltaje dentro del rango permitido por la entrada analógica del microcontrolador. Posteriormente, se realizó una calibración en oscuridad para determinar la irradiancia mínima real y corregirla mediante un offset calculado automáticamente por el programa. Con esto se obtuvo una medición estable y proporcional dentro del rango de 0 a 1500 W/m².

Una vez verificada la operación individual de cada sensor, se procedió al diseño del circuito impreso (PCB) utilizando KiCad. En esta etapa se trasladó el diagrama esquemático a un diseño físico que permitiera integrar los sensores, conectores, módulo XBee y etapa de acondicionamiento de manera compacta y organizada. Se definió la distribución de componentes, el ancho de pistas, la separación entre señales, las zonas de alimentación y las dimensiones finales de la placa. Posteriormente, se generaron los modelos 3D para verificar visualmente la orientación de los módulos y la compatibilidad mecánica de los conectores. La Figura 4.2 muestra el modelo tridimensional obtenido.

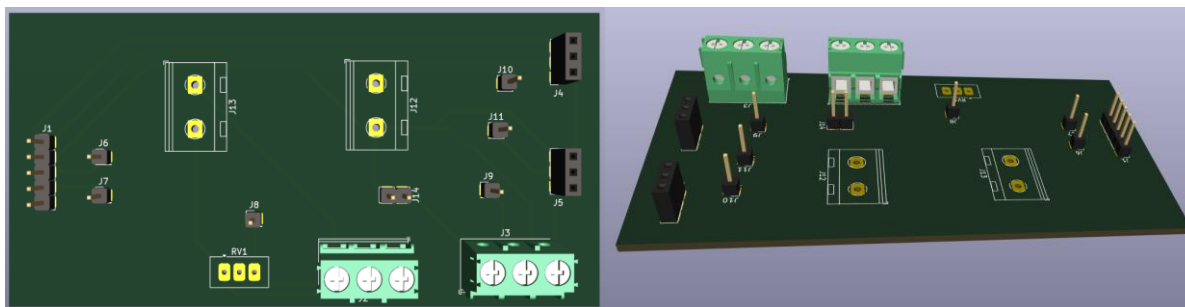


Figura 4. 2: Diseño del circuito (PCB) en formato 3D.

Fuente: Elaboración propia.

Una vez fabricada la PCB, se realizó el montaje manual de los componentes y la verificación del funcionamiento eléctrico de cada etapa. Para ello, se conectó el Arduino UNO cargado con el programa definitivo encargado de leer las señales, ejecutar los procedimientos de calibración, filtrar el ruido mediante promedios móviles y enviar las variables procesadas a

través del módulo XBee transmisor. Con esto quedó integrado el sistema de adquisición de datos encargado de alimentar el proceso de monitoreo implementado en la Raspberry Pi. En la Figura 4.3 se muestra el diseño final del SAD con las entradas de los sensores montados y conectados al Arduino.

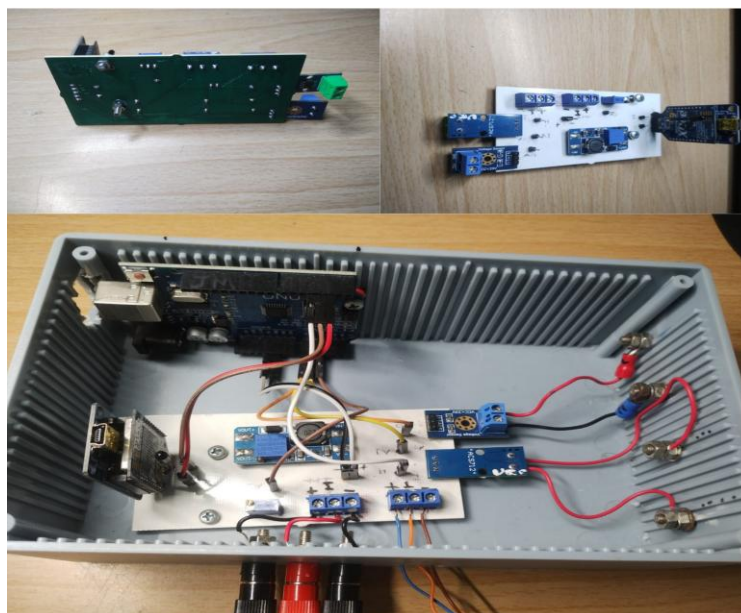


Figura 4. 3: Circuito SAD con sensores conectados.

Fuente: Elaboración propia.

4.3.- Red Inalámbrica XBee.

La comunicación inalámbrica entre el Sistema de Adquisición de Datos (SAD) y la Raspberry Pi se realiza mediante dos módulos XBee Serie 2, configurados para el protocolo ZigBee. Este enlace constituye una red punto a punto en la que el SAD actúa como router y la Raspberry Pi como coordinador, lo que permite la transmisión de datos censados en tiempo real.

Para establecer la red, fue necesario configurar cada módulo XBee de forma individual mediante el software XCTU, proporcionado por el fabricante. Para ello, cada dispositivo se conectó a la computadora mediante un adaptador USB-XBee, lo que permitió acceder a los parámetros de configuración y asignar las funciones correspondientes a cada nodo.

En el caso del módulo destinado al Sistema de Adquisición de Datos, se cargó el firmware ZigBee Router AT, que permite operar en modo transparente (AT), lo que facilita el envío continuo de datos sin requerir el uso de tramas de API. Se asignó el identificador

ROUTER, y se configuró el mismo PAN ID que utilizaría el coordinador para asegurar que ambos módulos pertenecieran a la misma red. Además, se estableció la dirección de destino (DH y DL) correspondiente a la dirección MAC del módulo coordinador, lo que permite que toda la información generada por el Arduino se envíe directamente a la Raspberry Pi. La Figura 4.4 muestra los parámetros configurados para este módulo.

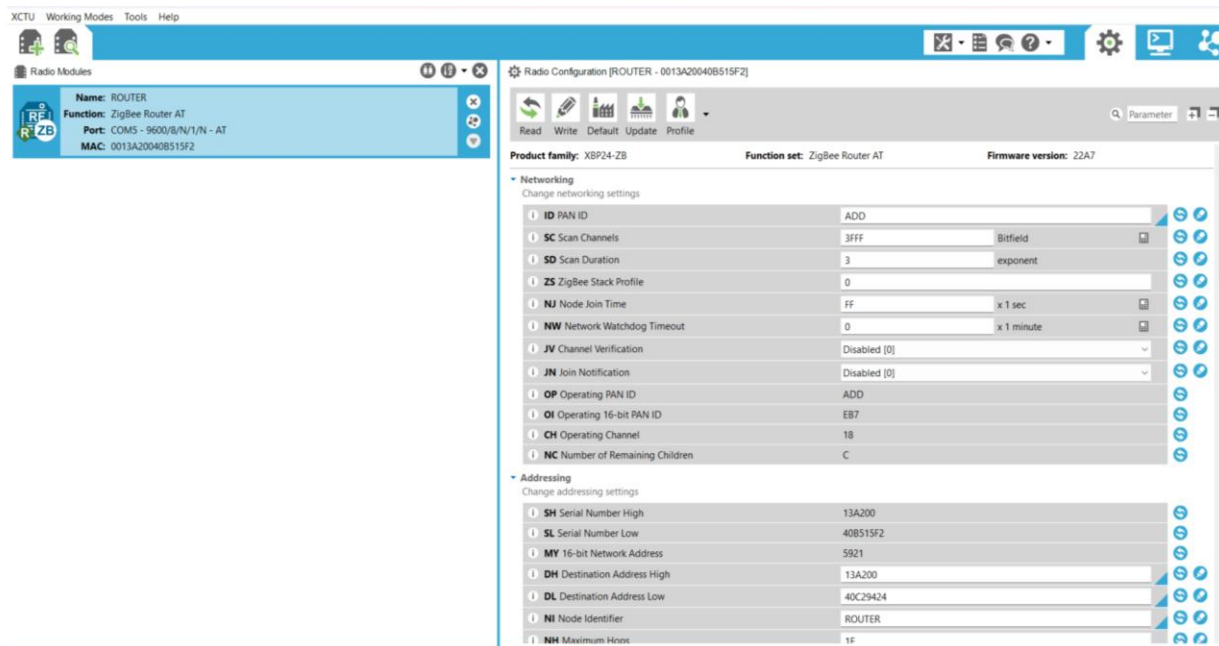


Figura 4. 4: Parámetros XBee Router.

Fuente: Elaboración propia.

Por su parte, el módulo conectado a la Raspberry Pi fue configurado con el firmware ZigBee Coordinator AT, permitiéndole operar como nodo principal de la red. A este módulo se le asignó el identificador COORDINADOR, y se configuró el mismo PAN ID y canal de operación que el router. Del mismo modo, se estableció como dirección de destino la MAC del router, lo que habilita la posibilidad de comunicación bidireccional en caso de requerirse. La Figura 4.5 se presentan los parámetros correspondientes al módulo coordinador.

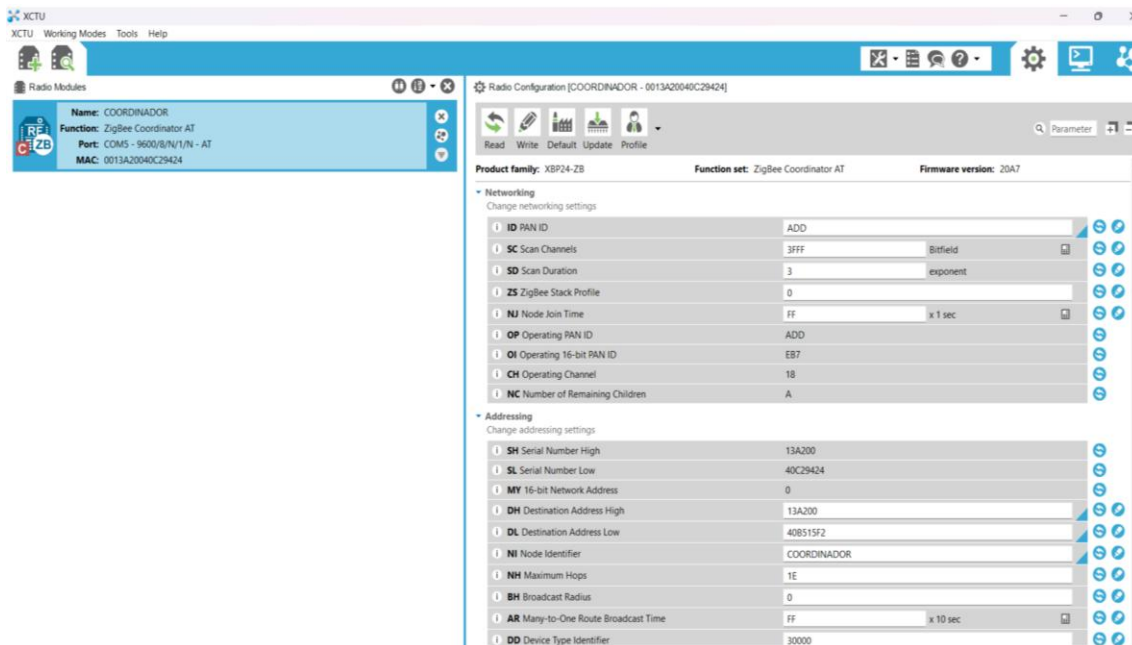


Figura 4. 5: Parámetros XBee Coordinador.

Fuente: Elaboración propia.

Al operar ambos dispositivos en modo AT, la comunicación se realiza de manera transparente: el Arduino envía datos en formato de texto mediante el puerto serial y el XBee Router los transmite inalámbricamente sin modificación alguna. La Raspberry Pi recibe estas cadenas a través del XBee Coordinador conectado por USB, permitiendo su lectura e interpretación mediante el programa en Python desarrollado para este proyecto.

Esta configuración asegura una comunicación estable y de baja latencia, adecuada para aplicaciones de monitoreo donde se requiere simplicidad en el envío de datos y confiabilidad en la transmisión punto a punto.

4.4.- Configuración e Implementación en la Raspberry Pi.

La Raspberry Pi actúa como núcleo del sistema, recibiendo las variables del módulo XBee coordinador, implementando el programa en Python, almacenando los datos localmente y enviándolos a la nube por medio de Firebase. Para hacerlo funcionar de manera autónoma, se tuvo que configurar primero el sistema operativo y el procesador.

El primer paso fue instalar el sistema operativo Raspberry Pi OS, el cual proporcionado por la Fundación Raspberry Pi. Se usó el software Raspberry Pi Imager para seleccionar la imagen del sistema operativo y escribirla en una tarjeta microSD, que se insertó en la Raspberry Pi. Para la configuración inicial se conectó un monitor, teclado y ratón para completar la instalación inicial, la conexión a la red WiFi y la actualización de paquetes del sistema operativo.

Una vez configurado el entorno gráfico del sistema, se procedió a instalar las librerías necesarias para la ejecución del programa en Python, incluyendo los paquetes para comunicación serial, manejo de SQLite y conexión con Firebase. Posteriormente, se conectó físicamente el módulo XBee coordinador al puerto USB de la Raspberry Pi, permitiendo que esta recibiera las tramas enviadas desde el Sistema de Adquisición de Datos.

Con el objetivo de automatizar el funcionamiento del sistema, se configuró la Raspberry Pi para ejecutar el programa principal inmediatamente después de encenderse. Esto evita la necesidad de intervención manual y permite un monitoreo continuo del panel fotovoltaico. Para ello, se editó el archivo `/etc/rc.local`, que se ejecuta automáticamente al final del arranque del sistema. En el archivo se añadió la instrucción mostrada a continuación, la cual inicia el programa encargado de procesar los datos recibidos:

```
sudo python3 /home/juanmanuel/Desktop/Proyecto/comunicacion2.py &
```

La utilización del símbolo “&” posibilita que el programa se ejecute en segundo plano, garantizando un reinicio normal del sistema operativo. Así, cada vez que se le suministra electricidad a la Raspberry Pi, arranca automáticamente el servicio de monitoreo, recibe los datos del módulo coordinador XBee y lleva a cabo las tareas de almacenamiento y visualización.

Esta configuración proporciona un funcionamiento completamente autónomo y continuo, elemento esencial en sistemas de monitoreo de energía renovable donde se requiere registrar datos de manera ininterrumpida y sin depender de supervisión manual. La validación

del arranque automático se comprobó mediante diversas pruebas, desconectando y reconectando la alimentación de la Raspberry Pi y verificando que el programa iniciara correctamente y que las variables se mostraran en la pantalla Nextion y se almacenaran en la base de datos interna.

En la Figura 4.6 se muestra la pantalla inicial de Raspberry Pi OS tras completar la configuración del sistema y habilitar el arranque automático del programa de monitoreo.

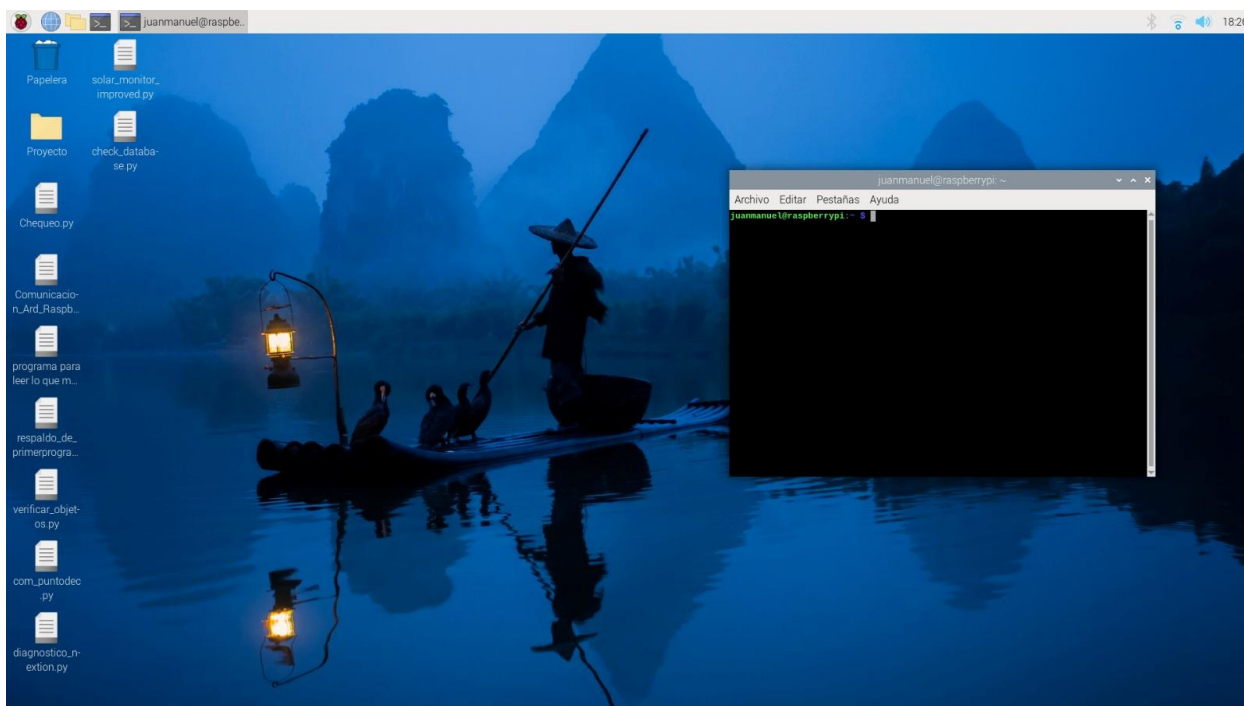


Figura 4. 6: Monitor conectado a la Raspberry Pi.

Fuente: Elaboración propia.

4.5.- Programa en Python para la gestión del sistema de monitoreo.

El núcleo lógico del sistema fue desarrollado en Python y ejecutado dentro de la Raspberry Pi, actuando como puente de comunicaciones y como procesador central de los datos enviados desde el módulo Arduino vía XBee. Este programa integra cuatro funciones críticas para el funcionamiento completo del sistema:

1. Recepción y decodificación de datos provenientes del XBee.
2. Visualización de los valores en tiempo real en la pantalla Nextion.
3. Almacenamiento local en SQLite.
4. Sincronización remota con Firebase Realtime Database.

4.5.1.- Recepción de datos desde el módulo inalámbrico XBee.

Cuando se ejecuta, el programa identifica automáticamente el puerto serial al que está conectado el módulo XBee USB. Una vez realizada la conexión, el sistema empieza a recibir tramas del tipo:

V:XX.XX,I:XX.XXX,P:XX.XXX,H:XX.XX,T:XX.XX,R:XXX.XX

Cada paquete es decodificado y separado en variables numéricas de voltaje, corriente, potencia, temperatura, humedad e irradiancia.

Para verificar el correcto funcionamiento de esta etapa, se realizaron pruebas de transmisión continua desde el Arduino, comprobando que la Raspberry Pi recibiera sin interrupciones las tramas completas, sin errores de codificación y sin pérdidas de paquetes.

4.5.2.- Actualización en tiempo real de la pantalla Nextion.

Antes de llegar a la pantalla principal donde se observan en tiempo real las variables medidas del panel fotovoltaico, el sistema muestra una pantalla de inicio con autenticación de usuario. Esta interfaz se desarrolló para prevenir accesos no autorizados y garantizar que solo el operador autorizado pueda interactuar con el sistema.

La pantalla de inicio pide dos campos: usuario y contraseña. Si el usuario ingresado no coincide con el registro establecido, la pantalla arroja el mensaje “Usuario incorrecto”, como se muestra en la Figura 4.7. Si el nombre de usuario es correcto, pero la contraseña es incorrecta, el sistema visualiza el mensaje "Contraseña incorrecta" como se muestra en la Figura 4.8.

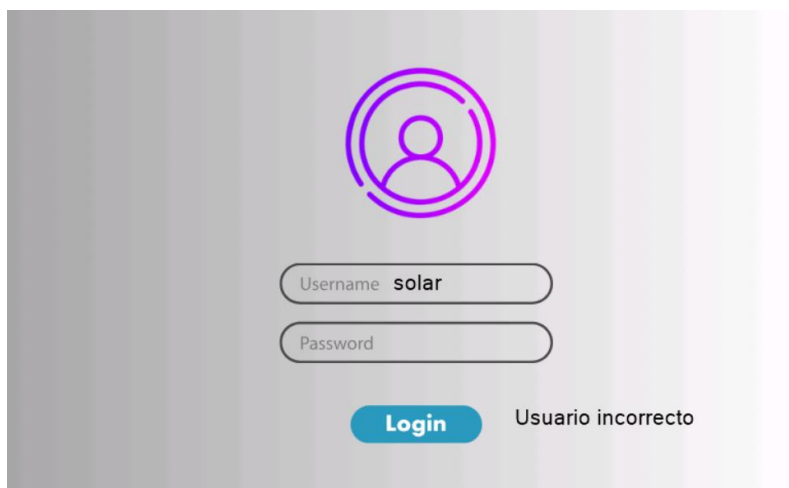


Figura 4. 7: Mensaje mostrado al ingresar un usuario incorrecto.

Fuente: Elaboración propia.

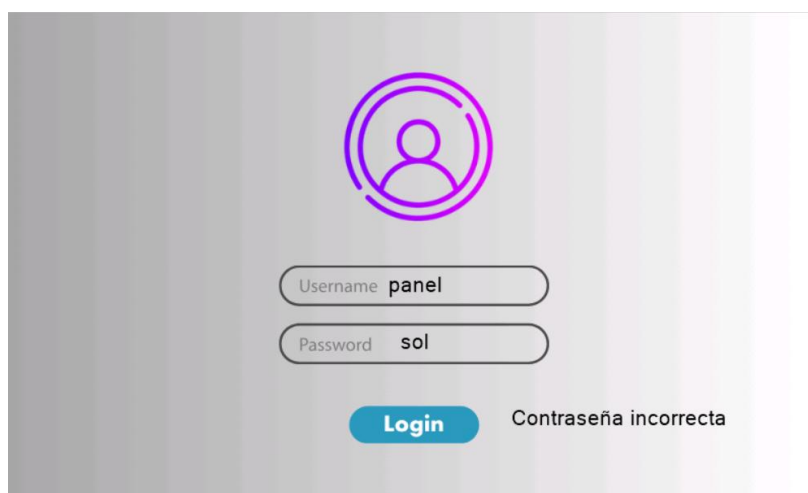


Figura 4. 8: Mensaje mostrado al ingresar una contraseña incorrecta.

Fuente: Elaboración propia.

Solo cuando se introduce el par correcto —usuario: panel, contraseña: MIE— se abre la puerta. MIE— La pantalla da paso a la pantalla principal de monitoreo, donde se visualizan las variables en tiempo real. La migración exitosa a esta pantalla se muestra en la Figura 4.9.

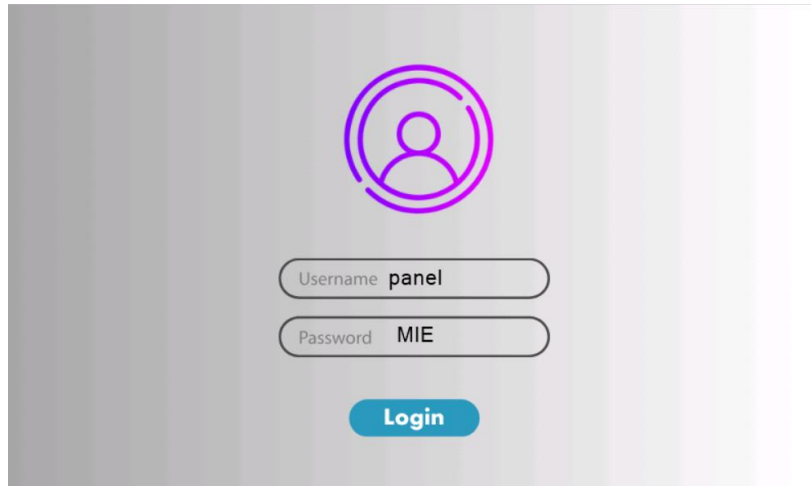


Figura 4. 9: Acceso exitoso después de ingresar el usuario y la contraseña correctas.

Fuente: Elaboración propia.

Este sistema de verificación se codificó directamente en el Nextion Editor a través de condicionales en los botones de inicio de sesión. Su objetivo es garantizar que el sistema no revele información sensible ni permita manipulación hasta que la autenticación sea exitosa, mejorando la confiabilidad del sistema de monitoreo.

Los valores recibidos se envían de inmediato a la pantalla Nextion mediante el puerto serial interno de la Raspberry Pi.

El programa construye comandos del tipo:

t6.txt="12.54 V" // Voltaje

t7.txt="0.082 A" // Corriente

t8.txt="1.025 W" // Potencia

t9.txt="45.20 %" // Humedad

t10.txt="28.35 °C" // Temperatura

t11.txt="820.50 W/m²" // Irradiancia

y los transmite utilizando la codificación requerida por Nextion (ISO-8859-1 y terminadores 0xFF 0xFF 0xFF).

Este proceso permite que la pantalla muestre datos actualizados cada medio segundo, funcionando como indicador visual del desempeño del panel solar.

La validación consistió en comparar los valores mostrados en Nxtion contra los registros del Arduino en tiempo real, confirmando coincidencias en voltaje, corriente e irradiancia. En la Figura 4.10 se muestra la interfaz con los valores obtenidos en una medición.

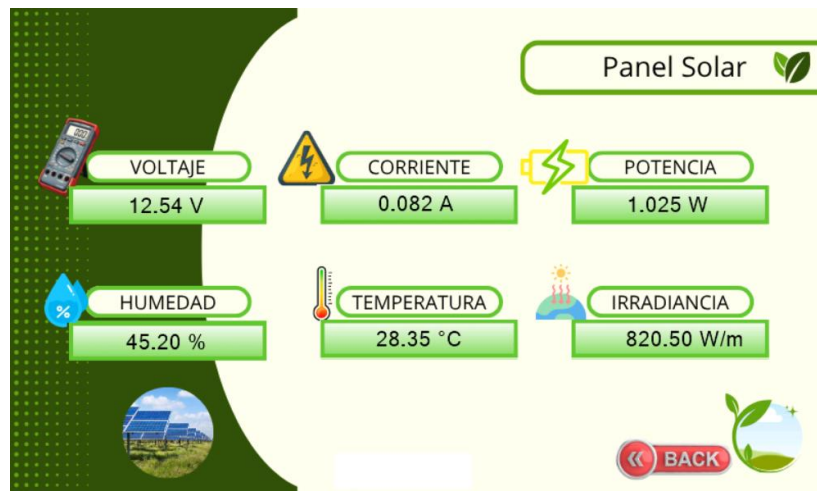


Figura 4. 10: Interfaz principal del sistema mostrando las variables del panel fotovoltaico en tiempo real.

Fuente: Elaboración propia.

4.5.3.- Almacenamiento local mediante SQLite.

Para la persistencia de datos y la operación autónoma del sistema aún sin conexión a Internet, se instaló una base de datos SQLite directamente en el almacenamiento interno de la Raspberry Pi. Se eligió SQLite por ser ligera, de configuración cero, muy fiable y compatible de forma nativa con Python, lo que la hace perfecta para aplicaciones embebidas y sistemas de monitorización que necesitan almacenamiento local fiable.

La primera vez que se ejecuta el programa en Python, se genera automáticamente el archivo `sensor_data.db` en la ruta `/home/juanmanuel/Desktop/Proyecto/`. La base de datos contiene una tabla principal con el siguiente esquema:

```
CREATE TABLE IF NOT EXISTS sensor_data (  
    timestamp TEXT NOT NULL,  
    voltage REAL NOT NULL,
```

current REAL NOT NULL,
power REAL NOT NULL,
temperature REAL NOT NULL,
humidity REAL NOT NULL,
irradiance REAL NOT NULL,
status TEXT NOT NULL
)

Para mejorar el almacenamiento y disminuir la redundancia, se desarrolló un sistema de promediado temporal. En vez de guardar cada muestra individual (que llega cada 500 ms), el sistema guarda las muestras durante 1 minuto y saca el promedio de cada variable. Esta estrategia se activa solamente en el horario de sol efectivo (6:00-20:00), para no tomar datos en la noche cuando el panel no genera. En la Figura 4.11 se muestra la tabla de registro de las variables.

Database Structure

Browse Data

Execute SQL

Edit Pragmas

Table: sensor_data

Filter in any column

	timestamp	voltage	current	power	temperature	humidity	irradiance	status
	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
19	2025-11-29 16:30:37	18.946	0.396	7.504	80.35	20.573	197.859	active
20	2025-11-29 16:31:37	18.838	0.394	7.415	80.767	20.393	177.504	active
21	2025-11-29 16:32:37	18.678	0.383	7.154	80.15	20.653	150.022	active
22	2025-11-29 16:33:37	18.522	0.377	6.988	81.0	20.3	124.851	active
23	2025-11-29 16:34:37	18.255	0.384	7.006	81.0	20.3	117.916	active
24	2025-11-29 16:35:37	18.059	0.38	6.859	81.0	20.3	132.232	active
25	2025-11-29 16:36:37	17.952	0.37	6.642	81.0	20.3	165.164	active
26	2025-11-29 16:37:37	17.774	0.372	6.605	82.0	20.3	186.892	active
27	2025-11-29 16:38:37	17.727	0.368	6.532	82.0	20.26	216.106	active
28	2025-11-29 16:39:37	17.716	0.372	6.599	82.533	19.8	207.094	active
29	2025-11-29 16:40:38	17.748	0.453	8.298	82.333	19.8	147.995	active
30	2025-11-29 16:41:38	17.712	0.363	6.425	82.0	19.8	182.798	active
31	2025-11-29 16:42:38	17.812	0.359	6.388	81.817	19.8	232.194	active
32	2025-11-29 16:43:38	18.078	0.383	6.927	82.4	19.8	238.029	active
33	2025-11-29 16:44:38	18.208	0.375	6.826	83.0	19.8	220.617	active
34	2025-11-29 16:45:38	18.372	0.383	7.036	83.0	19.8	271.946	active
35	2025-11-29 16:46:38	18.368	0.368	6.767	84.0	19.8	340.353	active
36	2025-11-29 16:47:38	18.51	0.386	7.15	83.0	19.838	372.173	active
37	2025-11-29 16:48:38	18.659	0.387	7.224	83.1	20.3	365.91	active
38	2025-11-29 16:49:38	18.81	0.383	7.204	83.0	20.3	361.959	active
39	2025-11-29 16:50:38	18.887	0.387	7.311	83.0	20.3	351.827	active
40	2025-11-29 16:51:38	19.042	0.385	7.325	83.983	19.8	344.67	active
41	2025-11-29 16:52:38	19.114	0.386	7.385	84.0	19.8	335.801	active
42	2025-11-29 16:53:38	19.154	0.395	7.57	84.0	19.8	330.73	active
43	2025-11-29 16:54:38	19.093	0.405	7.733	83.083	19.82	321.169	active
44	2025-11-29 16:55:38	18.912	0.377	7.135	83.0	19.8	291.721	active

Figura 4. 11: Datos recopilados.

Fuente: Elaboración propia.

El programa hace uso de la biblioteca sqlite3 de Python para manipular la conexión y la base de datos. La forma de promediado se implementa con colas (deque) donde se van guardando temporalmente las muestras:

#Acumulador para promedios (1 minuto)

```
data_accumulator = {
    'voltage': deque(maxlen=60),
    'current': deque(maxlen=60),
    'power': deque(maxlen=60),
    'temperature': deque(maxlen=60),
    'humidity': deque(maxlen=60),
```

```

        'irradiance': deque(maxlen=60)

    }

```

Cada minuto, durante el horario diurno, se calculan los promedios y se ejecuta la inserción en la base de datos:

```

# Cálculo de promedios

avg_voltage = round(sum(data_accumulator['voltage']) / len(data_accumulator['voltage']), 3)
avg_current = round(sum(data_accumulator['current']) / len(data_accumulator['current']), 3)
# ... (cálculos similares para otras variables)

# Inserción en SQLite

cursor.execute("""

INSERT INTO sensor_data

(timestamp, voltage, current, power, temperature, humidity, irradiance, status)

VALUES (?, ?, ?, ?, ?, ?, ?, ?)

""", (timestamp, avg_voltage, avg_current, avg_power,
      avg_temperature, avg_humidity, avg_irradiance, "active"))

conn.commit()

```

La integridad y correcto funcionamiento del sistema de almacenamiento se validó mediante la verificación de la creación automática de nuevos registros cada minuto durante el horario solar, confirmación del formato correcto de timestamps y valores numéricos, validación de la consistencia de datos entre las muestras en tiempo real y los promedios almacenados.

Esta implementación garantiza un registro histórico confiable y eficiente, con aproximadamente 840 registros diarios (14 horas $\times$ 60 minutos), proporcionando una base sólida para el análisis posterior del rendimiento del panel fotovoltaico y facilitando la detección de tendencias y anomalías en el sistema.

4.5.4.- Sincronización con Firebase Realtime Database.

De manera paralela al almacenamiento local, el programa sincroniza con Firebase Realtime Database, posibilitando el acceso remoto a los datos en tiempo real y ofreciendo una capa adicional de respaldo. Esta arquitectura híbrida garantiza la disponibilidad de los datos tanto local como remotamente, proporcionando flexibilidad en el acceso y análisis de la información.

Para permitir la conexión desde la Raspberry Pi, se añadió un archivo de credenciales en formato JSON (sensor_data_project.json) con las claves de servicio para autenticar la aplicación con los servidores de Firebase. La inicialización se hace con el siguiente código:

```
import firebase_admin

from firebase_admin import credentials, db

cred = credentials.Certificate('/home/juanmanuel/Desktop/Proyecto/sensor_data_project.json')
firebase_admin.initialize_app(cred, {
    'databaseURL': 'https://panel-solar-79662-default-rtdb.firebaseio.com/'
})
```

Cada minuto, tras guardar los promedios en la base de datos local SQLite, el sistema intenta enviarlos al servidor en la nube a través de la API de Firebase. La forma de organizar los datos en Firebase es en nodos individuales por cada registro, usando push() para crear identificadores únicos automáticamente:

```
ref = db.reference('sensor_data')

ref.push({
    'timestamp': timestamp,
    'voltage': avg_voltage,
    'current': avg_current,
```

```
'power': avg_power,  
  
'temperature': avg_temperature,  
  
'humidity': avg_humidity,  
  
'irradiance': avg_irradiance,  
  
'status': "active"  
  
})
```

El sistema está preparado para funcionar en condiciones de conectividad interrumpida. Si la red está disponible, los datos se sincronizan con Firebase. Si se pierde la conexión, el programa sigue funcionando en local y muestra un mensaje por consola. No se detiene el almacenamiento local. Esta característica se probó desconectando la red Wi-Fi en medio de una prueba y se confirmó:

- La continuidad del almacenamiento local sin interrupciones.
- La generación de mensajes de alerta en consola ante la falta de conectividad
- La ausencia de pérdida de datos durante los periodos de desconexión
- La correcta resincronización una vez restaurada la conectividad

Esta implementación proporciona un sistema de almacenamiento redundante que combina la confiabilidad del almacenamiento local con la accesibilidad y respaldo en la nube, asegurando la integridad y disponibilidad de los datos en todos los escenarios operativos.

4.6.- Interfaz Gráfica Nextion.

La interfaz gráfica del sistema se diseñó en el software Nextion Editor, el cual permite crear pantallas interactivas con ayuda de componentes gráficos y eventos programables. La interfaz se desarrolló en dos páginas principales: una página de inicio con autenticación y una página de visualización para las variables adquiridas en tiempo real.

En la Página 0 (pantalla de inicio) se agregaron dos campos de texto, uno para el nombre de usuario y otro para la contraseña. Cada uno fue equipado con un Touch Press Event, para detectar al instante el toque del usuario. El botón de login llama a una rutina de autenticación que compara los datos ingresados con las credenciales configuradas para el sistema. Si los datos

son correctos, la pantalla se redirige automáticamente a la página de monitoreo; en caso contrario, se muestra un mensaje de error a través de un componente de texto adicional, ya sea "Usuario incorrecto" o "Contraseña incorrecta". Esta característica garantiza que solo usuarios autorizados puedan acceder a las mediciones del sistema.

Posteriormente, en la Página 1 (pantalla de monitoreo) se muestran los valores de voltaje, corriente, potencia, humedad relativa, temperatura e irradiancia que envía la Raspberry Pi. Cada información es actualizada en tiempo real a través de comandos seriales, con lo cual el usuario puede visualizar el comportamiento del panel solar de manera continua y sin latencia. Esta pantalla también cuenta con un botón de regreso (Back), el cual permite regresar a la pantalla de inicio sin tener que reiniciar el sistema.

Para montar físicamente la pantalla en un módulo pequeño y protegido, se diseñó y se imprimió en 3D un marco y una carcasa en Tinkercad, un software CAD gratuito y fácil de usar. El diseño se pensó para montar tanto la pantalla Nextion como una carcasa extra para la Raspberry Pi, creando así un conjunto único de fácil instalación en exteriores o en ambientes de prueba. La carcasa impresa protege la electrónica, organiza físicamente y refuerza la estructura del sistema en funcionamiento. En la Figura 4.12 se observa la interfaz gráfica instalada en la carcasa junto la Raspberry Pi.



Figura 4. 12: Carcasa impresa en 3D para interfaz gráfica y Raspberry Pi 4.

Fuente: Elaboración propia.

La interfaz interactiva, el diseño físico a medida y la comunicación con la Raspberry Pi dieron como resultado un módulo visual robusto, pequeño y apropiado para las pruebas de validación del sistema de monitoreo fotovoltaico.

4.7.- Test de usabilidad de la interfaz gráfica.

Para medir la facilidad de uso y la experiencia de los usuarios con el sistema se utilizó el cuestionario UMUX-Lite (Usability Metric for User Experience – Lite), un instrumento ampliamente utilizado para evaluar la usabilidad percibida de interfaces interactivas. Esta forma, basada en la norma ISO 9241-11, mide la efectividad, eficiencia y satisfacción del usuario con tan solo dos ítems, siendo una medida más corta, rápida y fiable que otras herramientas como el SUS (System Usability Scale).

El UMUX-Lite resulta adecuado para este proyecto debido a que permite evaluar prototipos funcionales sin requerir largas sesiones de retroalimentación, ofreciendo una medida precisa del nivel de usabilidad en etapas de validación experimental. Además, su estructura reducida facilita su aplicación en entornos donde los participantes cuentan con tiempos limitados o en los que se requiere evaluar la interfaz de manera ágil durante las pruebas de campo.

El cuestionario aplicado estuvo compuesto por los siguientes dos ítems, en una escala Likert de 1 a 7: "El sistema satisface mis necesidades", "El sistema es fácil de usar".

A cada participante se le solicitó interactuar con la interfaz gráfica Nextion, explorando sus páginas, iniciando sesión, observando la actualización en tiempo real de las variables medidas y manipulando los controles.

Posteriormente, se registraron sus respuestas de forma individual y anónima. La puntuación global del UMUX-Lite se calculó utilizando su fórmula estandarizada, que transforma los resultados en una escala de 0 a 100 para poder compararlos con otras métricas de usabilidad.

Tras la aplicación del instrumento a una muestra representativa de usuarios, se obtuvo un promedio de 90 puntos, lo cual indica un nivel de usabilidad altamente favorable y demuestra

que el prototipo ofrece una interacción sencilla, comprensible y satisfactoria para los usuarios evaluados.

Con el propósito de evaluar la facilidad de uso y la experiencia percibida por los usuarios del sistema, se aplicó el cuestionario UMUX-Lite (Usability Metric for User Experience – Lite), un instrumento ampliamente utilizado para medir la usabilidad percibida de interfaces interactivas. Este método, derivado del estándar internacional ISO 9241-11, evalúa la efectividad, eficiencia y satisfacción del usuario a partir de únicamente dos ítems, lo que lo convierte en una alternativa más breve, rápida y confiable en comparación con otros instrumentos como el SUS (System Usability Scale).

El UMUX-Lite resulta adecuado para este proyecto debido a que permite evaluar prototipos funcionales sin requerir largas sesiones de retroalimentación, ofreciendo una medida precisa del nivel de usabilidad en etapas de validación experimental. Además, su estructura reducida facilita su aplicación en entornos donde los participantes cuentan con tiempos limitados o en los que se requiere evaluar la interfaz de manera ágil durante las pruebas de campo.

El cuestionario aplicado estuvo conformado por los siguientes dos ítems, valorados en una escala Likert de 1 a 7:

"El sistema cumple con mis necesidades."

"El sistema es fácil de usar."

A cada participante se le solicitó interactuar con la interfaz gráfica Nextion, navegar entre las páginas, iniciar sesión, observar la actualización en tiempo real de las variables medidas y utilizar los controles disponibles. Posteriormente, se registraron sus respuestas de forma individual y anónima.

La puntuación global del UMUX-Lite se obtuvo mediante su fórmula estandarizada, que convierte los resultados en una escala de 0 a 100 para facilitar su interpretación en comparación con otras métricas de usabilidad. Tras la aplicación del instrumento a una muestra representativa de usuarios, se obtuvo un promedio de 90 puntos, lo cual indica un nivel de usabilidad altamente favorable y demuestra que el prototipo ofrece una interacción sencilla, comprensible y satisfactoria para los usuarios evaluados.

Estos resultados corroboran que la interfaz gráfica desarrollada cumple adecuadamente su función dentro del sistema de monitoreo fotovoltaico, proporcionando una comunicación clara de las variables medidas y facilitando la operación del dispositivo incluso para usuarios sin experiencia previa con equipos de instrumentación.

4.8.- Recolección de datos en campo.

La recolección de datos en campo es un paso esencial para confirmar la operatividad completa del Sistema de Adquisición de Datos (SAD) y comprobar que las mediciones obtenidas, los valores mostrados en la interfaz Nextion y los registros guardados en la base de datos interna (SQLite) y en la plataforma Firebase Realtime Database coincidan.

Las pruebas finales fueron realizadas en el Laboratorio de Eléctrica del Instituto Tecnológico de Ciudad Madero, Campus II, la última semana de noviembre del año 2025. Se llevaron a cabo entre las 09:00 y las 18:00 h, lo que permitió evaluar el funcionamiento del sistema en condiciones ambientales e irradiancias reales.

Para la toma de datos se instaló el prototipo completo, conformado por el panel solar conectado al Sensor de Adquisición de Datos (SAD), el módulo XBee encargado del envío inalámbrico, la Raspberry Pi con la pantalla táctil Nextion, responsable del procesamiento, visualización y almacenamiento.

Durante las sesiones de campo, el sistema registró automáticamente las variables eléctricas y ambientales —voltaje, corriente, potencia, temperatura, humedad e irradiancia— mediante el procesamiento realizado en la Raspberry Pi, que almacena promedios cada un minuto en la base de datos SQLite. En la Figura 4.13 se muestra el SAD instalado en panel solar.



Figura 4. 13: Monitoreo de Panel Fotovoltaico.

Fuente: Elaboración propia.

Los datos obtenidos en esta etapa fueron utilizados posteriormente para realizar el análisis estadístico (promedios, variaciones, histogramas y comportamiento diario) así como las pruebas de minería de datos presentadas en el Capítulo 5, lo que permitió evaluar el desempeño del prototipo bajo condiciones reales de operación.

4.9.- Conclusiones.

La validación experimental del sistema de monitoreo de variables eléctricas y ambientales del panel fotovoltaico permitió comprobar el funcionamiento integral del prototipo desarrollado. Durante las pruebas realizadas en el Laboratorio de la Maestría en Ingeniería Eléctrica del Instituto Tecnológico de Ciudad Madero, se verificó que los módulos XBee establecieron una comunicación inalámbrica estable, permitiendo la transmisión continua de las mediciones adquiridas por el Sistema de Adquisición de Datos (SAD) hacia la Raspberry Pi.

La configuración de la Raspberry Pi, junto con la ejecución automática del programa en Python, garantizó un procesamiento ininterrumpido de los datos recibidos, incluso después de interrupciones eléctricas o reinicios forzados. Asimismo, se corroboró el almacenamiento local en la base de datos SQLite, el cual ofreció un respaldo confiable para la conservación de la información aun cuando no existiera conexión a Internet.

Por su parte, la interfaz gráfica desarrollada en la pantalla Nextion demostró ser una herramienta eficaz para la visualización en tiempo real de las variables medidas. La incorporación del sistema de autenticación (usuario y contraseña) permitió un acceso controlado, mientras que la organización de las páginas y los elementos gráficos facilitó una interpretación clara y rápida del estado del panel.

La conexión con Firebase Realtime Database se verificó durante las pruebas iniciales, evidenciando la capacidad del sistema para enviar información hacia la nube. Esto brinda la posibilidad de consultar los registros desde cualquier ubicación con acceso a Internet, permitiendo futuras ampliaciones hacia plataformas web o aplicaciones móviles.

Capítulo 5

Análisis de datos

En este capítulo se presentan los prototipos contruidos, las pruebas de funcionalidad del sistema completo y la verificación del cumplimiento de los objetivos específicos mediante experimentación controlada.

5.1.- Introducción al análisis de datos.

Después de comprobar la transmisión inalámbrica a través de XBee e integrar totalmente el sistema de adquisición de datos (SAD) al panel fotovoltaico, se comenzó a recolectar información ambiental y eléctrica de manera constante. El sistema guardó automáticamente mediciones de corriente, voltaje, temperatura, irradiancia, humedad y potencia durante varios días. Como resultado, se creó una base de datos sólida en SQLite y Firebase.

Este capítulo tiene como objetivo principal realizar un análisis de los datos recopilados con el fin de entender el comportamiento real del panel solar bajo distintas condiciones ambientales y, de esta manera, distinguir patrones que posibiliten una mejor interpretación de las mediciones, optimizar la operación del sistema y establecer criterios de evaluación que puedan ser incorporados a la interfaz gráfica Nextion. Para dicho análisis se utilizó Google Colab como entorno computacional, empleando bibliotecas de procesamiento y análisis de datos como Pandas, NumPy, Matplotlib, Scikit-learn y SQLite3. Estas herramientas permitieron:

- Realizar análisis estadístico descriptivo (promedios, máximos, mínimos, desviaciones estándar).
- Visualizar la distribución de las variables mediante histogramas.
- Graficar relaciones eléctricas clave del panel, como las curvas I-V y P-V, utilizando los datos reales obtenidos por el sistema.
- Aplicar técnicas de minería de datos, incluyendo K-means para clasificación de condiciones solares y árboles de decisión para generar reglas de operación basadas en las mediciones reales.

La aplicación de estos métodos facilitó la correcta interpretación del rendimiento del panel fotovoltaico, el análisis de la consistencia de los datos proporcionados por el SAD y la definición de criterios automáticos que pueden utilizarse para mostrar en la pantalla Nextion mensajes pertinentes sobre las condiciones ambientales, la producción de energía y la irradiación.

Los análisis efectuados, los modelos adquiridos y las consecuencias prácticas para el sistema de monitoreo solar creado se expondrán en las secciones que siguen.

5.2.- Análisis en Google Colab.

Con el propósito de realizar un análisis estadístico preliminar de los datos adquiridos por el prototipo, se utilizó la plataforma Google Colab, ya que permite ejecutar código en Python sin necesidad de instalar software adicional y facilita la carga y procesamiento de archivos. En esta etapa se trabajó principalmente con la base de datos generada en formato SQLite, la cual contiene los registros de voltaje, corriente, potencia, irradiancia, temperatura y humedad capturados cada minuto por el sistema.

Una vez cargado el archivo en el entorno de Colab, se emplearon las librerías Pandas, NumPy y Matplotlib para realizar un análisis descriptivo de las variables. Se calcularon estadísticas básicas como:

- Promedio
- Máximo
- Mínimo
- Desviación estándar

Estas métricas permiten obtener una primera apreciación del comportamiento general del sistema fotovoltaico bajo condiciones reales de operación.

Asimismo, se generó un resumen gráfico de la variable irradiancia, que permite visualizar la distribución de los valores registrados durante el periodo de pruebas. Este tipo de gráfica facilita identificar si la mayor parte de los datos se concentran en niveles bajos, medios o altos de irradiancia, lo cual influye directamente en el nivel de potencia producida por el panel solar.

La Figura 5.1 muestra el histograma correspondiente a la irradiancia registrada durante las pruebas en campo.

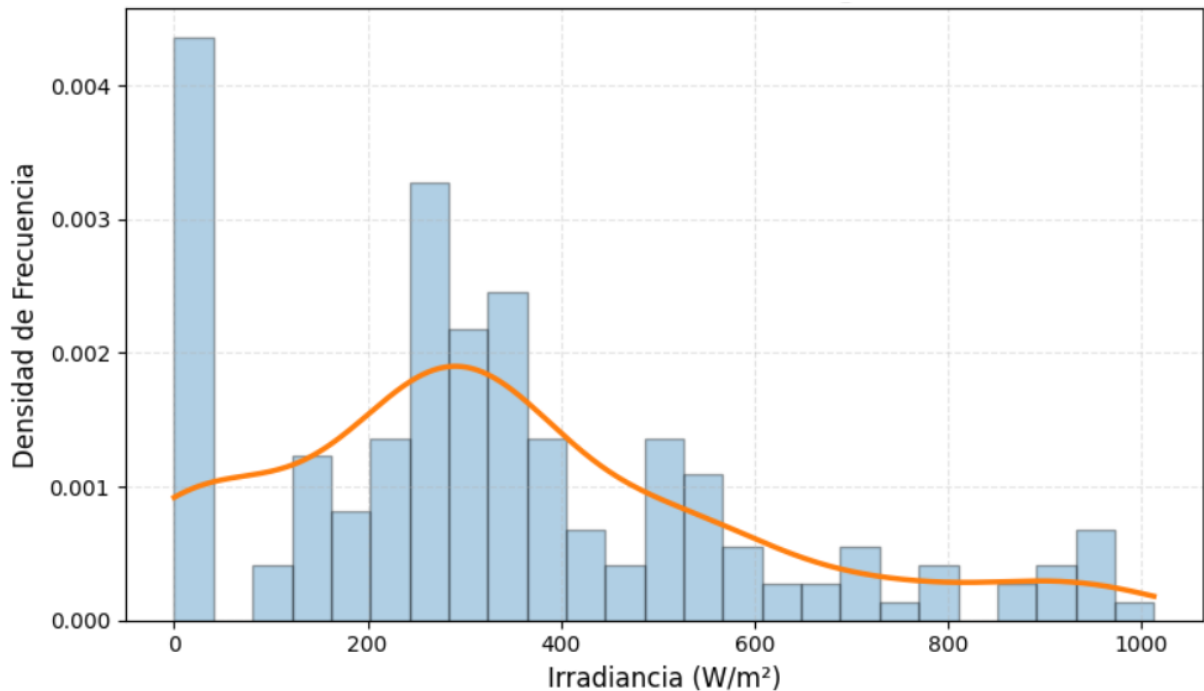


Figura 5. 1: Distribución de Irradiancia Registrada.

Fuente: Elaboración propia.

En general, el análisis realizado en Google Colab permitió validar que los datos fueron adquiridos correctamente y proporcionó una visión inicial del desempeño eléctrico y ambiental del sistema, sirviendo como base para los análisis más avanzados presentados en las siguientes secciones.

5.3.- Clasificación de condiciones solares mediante K-means.

Para identificar patrones de comportamiento en las variables medidas por el Sistema de Adquisición de Datos (SAD), aplicamos la técnica de minería de datos K-means. Esta técnica agrupa registros con características similares y nos ayuda a encontrar condiciones típicas de operación del panel solar. Es útil para segmentar el comportamiento del sistema bajo diferentes niveles de irradiancia y examinar su efecto en la energía generada.

Para este análisis, elegimos las variables clave que representan el rendimiento fotovoltaico: irradiancia, temperatura, humedad y potencia. Usando estas cuatro características, ejecutamos el algoritmo K-means con tres clústeres ($k = 3$) para clasificar las mediciones bajo condiciones de:

- Baja irradiancia.
- Irradiancia media.
- Irradiancia alta.

Una vez que se ejecutó el algoritmo, cada registro se colocó automáticamente en el clúster correcto. La Figura 5.2 muestra cómo se distribuyen los puntos en el plano Irradiancia-Potencia, junto con los centroides de cada grupo. Es evidente cómo los datos se organizan en tres zonas distintas. Esto confirma que la irradiancia es el factor principal en la producción de energía, mientras que los cambios en la humedad y la temperatura tienen un impacto menor.

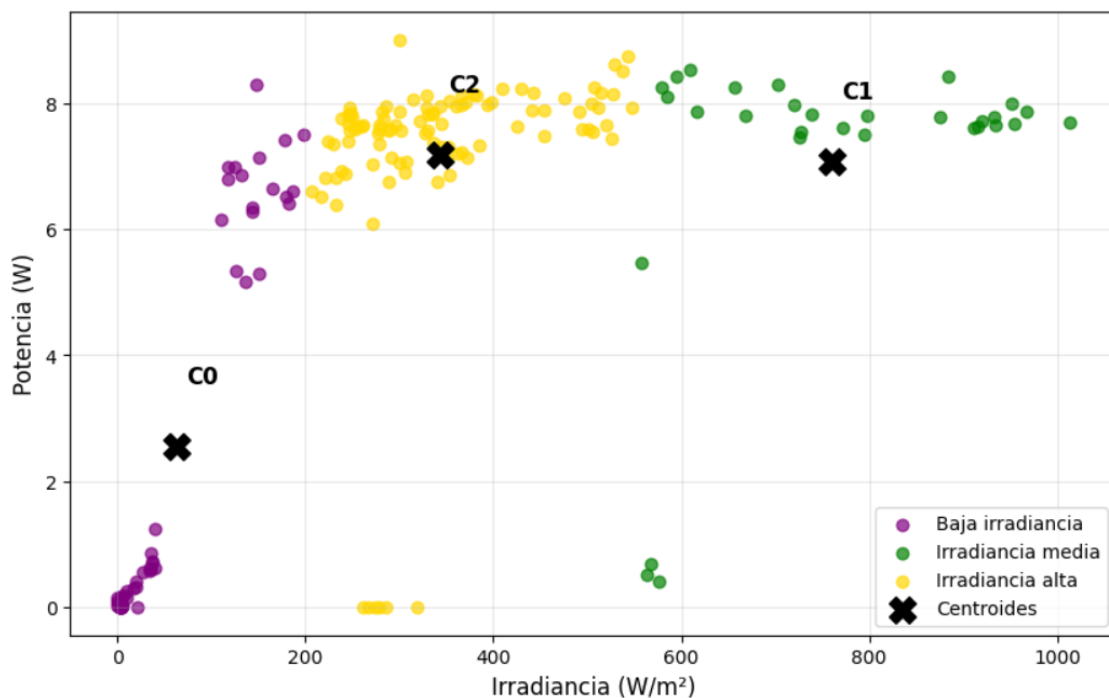


Figura 5. 2: Gráfica de las mediciones de irradiancia obtenidas.

Fuente: Elaboración propia.

Los resultados del análisis permitieron interpretar de manera cuantitativa el comportamiento del sistema:

- El clúster identificado como baja irradiancia (color morado), se concentra en valores menores de potencia, característicos de condiciones nubladas o primeras horas del día.
- El clúster de irradiancia media (color verde), agrupa puntos intermedios, típicos de condiciones parcialmente favorables.

- El clúster de irradiancia alta (color amarillo), corresponde a los niveles más elevados de potencia registrada, cercanos al desempeño ideal del panel dentro de las condiciones reales de prueba.

Esta clasificación ayuda al sistema al definir umbrales inteligentes. Estos umbrales se utilizaron posteriormente para crear reglas operativas con un árbol de decisión. Esto permite que el prototipo comprenda las condiciones solares actuales y muestre mensajes relevantes en la interfaz gráfica. Esto mejora la información y la eficiencia de la operación. La Figura 5.3 ilustra la distribución de los registros en los tres clústeres identificados por el algoritmo K-means, correspondientes a diferentes niveles de irradiancia.

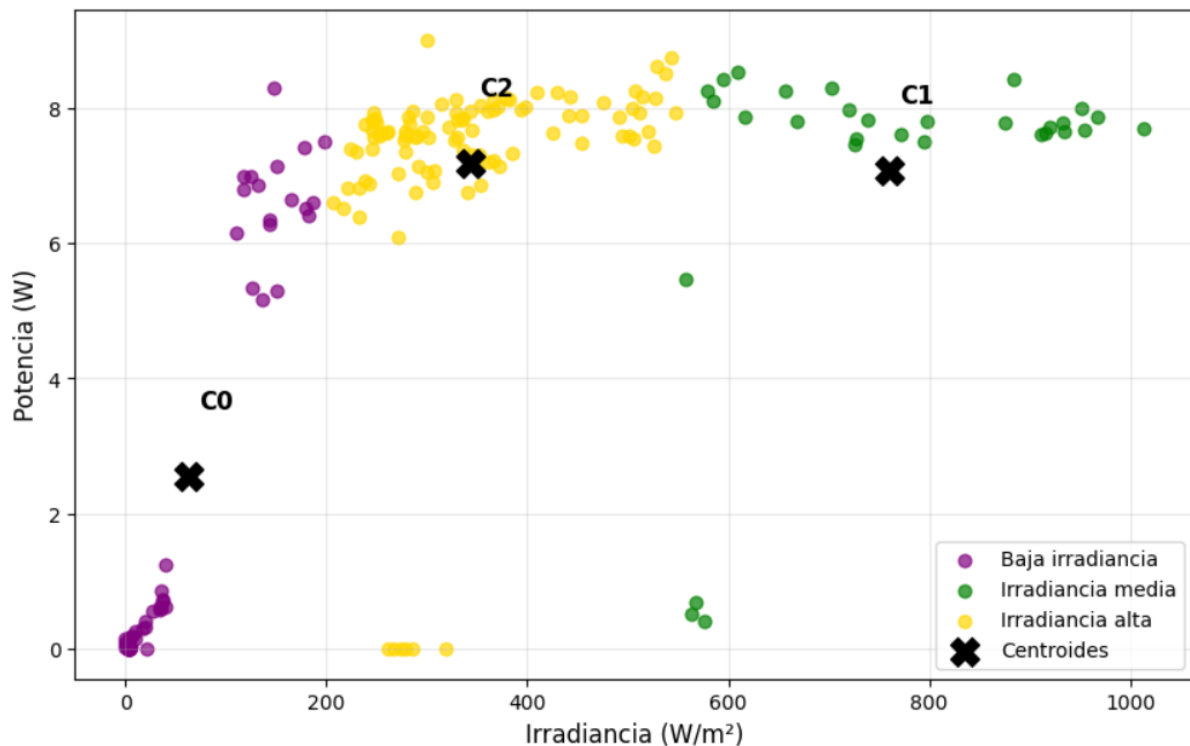


Figura 5. 3: Clasificación de condiciones solares mediante K-means.

Fuente: Elaboración propia.

5.4.- Generación de reglas mediante Árbol de Decisión.

Después de clasificar los datos con el algoritmo K-means, se utilizó un modelo de Árbol de Decisión para encontrar reglas sencillas para determinar el estado operativo del panel solar. Estas reglas se basan en las variables medidas por el prototipo. Este tipo de modelo es común en los sistemas de monitoreo porque genera reglas claras que son fáciles de entender y se pueden integrar en interfaces gráficas o sistemas embebidos.

Para el entrenamiento del modelo, se emplearon las variables:

- Irradiancia (W/m^2).
- Temperatura ($^{\circ}\text{C}$).
- Humedad relativa (%).
- Potencia (W).

Y como variable objetivo se utilizó el clúster generado por K-means (baja, media o alta irradiancia). De esta manera, el árbol aprende a reconocer en qué condiciones el panel presenta una producción baja, media o alta.

El modelo obtenido generó un conjunto de reglas que permiten clasificar el estado del panel mediante umbrales claros. Entre ellas destacan:

- Si la irradiancia es menor a un cierto valor, el sistema clasifica la condición como irradiancia baja, independientemente del resto de las variables.
- Si la irradiancia supera este valor, pero la potencia generada permanece por debajo de un umbral, la condición se considera irradiancia media.
- Si ambos parámetros se encuentran por encima de los límites establecidos, el sistema identifica irradiancia alta, correspondiente a la máxima generación del panel.

La Figura 5.4 muestra el árbol de decisiones resultante. La estructura indica que la irradiancia es el factor principal para clasificar el rendimiento solar. La potencia generada

sigue como el segundo factor más importante. Esto coincide con la teoría fotovoltaica, que establece que la irradiancia es el determinante clave del punto de operación del panel.

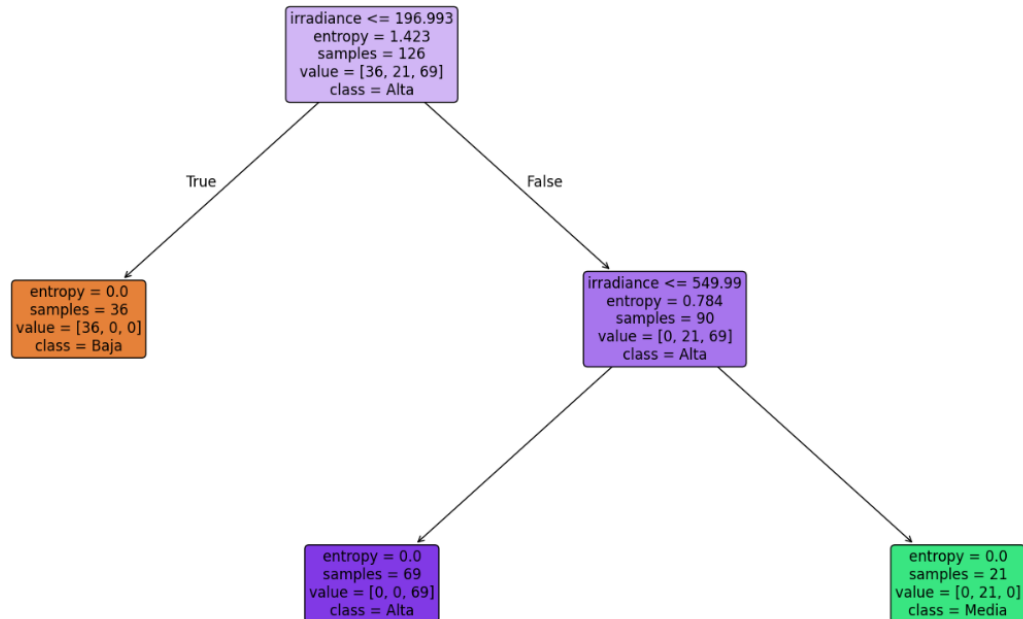


Figura 5. 4: Árbol de Decisión para Clasificación de condiciones solares.

Fuente: Elaboración propia.

Estas reglas facilitan la comprensión de cómo se comporta el sistema bajo diferentes condiciones. También forman la base para crear mensajes operativos automatizados en la interfaz gráfica. Estos mensajes ayudan al usuario a interpretar rápidamente el estado del panel, como se detalla en la siguiente sección.

5.5.- Acciones correctivas recomendadas.

Basándonos en el análisis estadístico realizado en Google Colab y en los resultados del K-means y del árbol de decisión, identificamos patrones claros en el comportamiento del panel solar y del sistema de monitoreo. Esto nos permitió establecer varias acciones correctivas que ayudan a mejorar la eficiencia del prototipo y a garantizar la fiabilidad de las mediciones.

Los análisis confirmaron que la irradiancia es el principal factor que afecta la energía generada por el panel. La temperatura y la humedad tienen efectos menores. Cuando la irradiancia registrada es baja, el sistema normalmente produce menos corriente y potencia. Es aconsejable verificar que el panel no tenga sombras parciales, suciedad en la superficie o una orientación incorrecta. Bajo condiciones de irradiancia media, inspeccione las conexiones eléctricas, compruebe la temperatura del panel y asegúrese de que la tendencia observada coincida con las condiciones ambientales normales. En escenarios de alta irradiancia, donde el panel funciona de manera más eficiente, asegúrese de que la energía generada se corresponda con el nivel de iluminación. Desviaciones significativas podrían indicar un sobrecalentamiento del módulo o problemas con los sensores.

Además de las recomendaciones sobre el comportamiento del panel solar, el análisis de datos también reveló formas de mejorar el sistema de monitoreo. Se sugiere revisar periódicamente la estabilidad de la comunicación inalámbrica a través de XBee, examinar el tamaño y la condición de la base de datos SQLite para evitar registros dañados y garantizar la sincronización adecuada con Firebase cuando haya cambios o retrasos en la carga de datos. La interfaz gráfica también puede usar los resultados del análisis para actualizar automáticamente los mensajes mostrados al usuario según las condiciones solares actuales, proporcionando una retroalimentación inmediata sobre el rendimiento del sistema.

En conjunto, estas acciones correctivas permiten mejorar la calidad del monitoreo, fortalecer la confiabilidad del prototipo y optimizar la interpretación de los datos adquiridos en condiciones reales de operación.

5.6.- Conclusiones.

El desarrollo del Sistema de Monitoreo Fotovoltaico basado en Raspberry Pi, comunicación inalámbrica XBee y visualización mediante una pantalla Nextion permitió validar satisfactoriamente la capacidad del prototipo para adquirir, procesar y almacenar información eléctrica y ambiental en tiempo real. A lo largo de las pruebas experimentales se comprobó que el sistema es capaz de registrar con precisión las variables fundamentales del módulo fotovoltaico —voltaje, corriente, potencia, temperatura, humedad e irradiancia— y mostrarlas en una interfaz gráfica intuitiva y de respuesta inmediata.

La implementación de una arquitectura híbrida de almacenamiento, utilizando simultáneamente SQLite para operación local y Firebase Realtime Database para respaldo en la nube, demostró ser adecuada para aplicaciones donde las condiciones de conectividad pueden variar. Esta combinación permitió asegurar la integridad de los datos, garantizando la continuidad del registro incluso ante interrupciones temporales de Internet. Asimismo, la comunicación entre la Raspberry Pi y la pantalla Nextion—incluyendo el sistema de autenticación y los mensajes contextuales basados en irradiancia—contribuyó a que el prototipo ofreciera una interacción más clara y funcional al usuario.

Por otra parte, el análisis estadístico realizado en Google Colab permitió obtener una caracterización inicial del comportamiento del panel bajo condiciones reales de operación, identificando tendencias coherentes en cada una de las variables medidas. El uso de técnicas de minería de datos, como K-means y árboles de decisión, permitió clasificar automáticamente los estados de operación del panel y generar reglas interpretables que pueden emplearse como base para futuras mejoras en la interfaz gráfica o en los algoritmos de supervisión. Estos resultados evidencian que el prototipo no solo es capaz de adquirir información, sino también de proporcionar insumos para análisis más profundos orientados a mejorar el desempeño del sistema.

Finalmente, las pruebas de campo confirmaron la confiabilidad general del prototipo. La comparación entre los valores mostrados en la pantalla Nextion, los registros locales de SQLite y los datos almacenados en Firebase evidenció coherencia, lo que valida tanto la arquitectura de

software como la adquisición inalámbrica mediante XBee. El sistema mostró estabilidad durante el funcionamiento continuo y capacidad para operar de manera autónoma desde el encendido, cumpliendo los objetivos planteados al inicio del proyecto. En conjunto, estos resultados demuestran que el prototipo desarrollado constituye una herramienta funcional para el monitoreo de sistemas fotovoltaicos en tiempo real y establece una base sólida para futuras ampliaciones, tales como predicción de energía, diagnóstico automático de fallas o integración con plataformas de gestión energética.

Capítulo 6

Conclusiones

En este capítulo se presentan las conclusiones, las contribuciones del proyecto, recomendaciones para implementaciones similares, y se proponen líneas de investigación futuras para escalar y mejorar el sistema desarrollado.

6.1.- Conclusiones Generales.

El desarrollo del sistema de adquisición de datos (SAD) para el monitoreo de un panel fotovoltaico permitió validar satisfactoriamente el funcionamiento del prototipo tanto en condiciones de laboratorio como en pruebas reales de campo. La integración entre sensores, el módulo SAD, la pantalla Nextion y la Raspberry Pi demostró ser funcional y estable, logrando la lectura continua de las variables eléctricas y ambientales del panel, así como su registro simultáneo en una base de datos local (SQLite) y en la plataforma en la nube Firebase.

Los datos recopilados durante la última semana de noviembre permitieron realizar un análisis estadístico y una evaluación del comportamiento real del sistema fotovoltaico. Las gráficas de voltaje, corriente, potencia, irradiancia, temperatura y humedad confirmaron que el prototipo responde de manera coherente a los cambios ambientales, sin presentar lecturas erráticas ni fuera de rango. Asimismo, la comparación entre las curvas I-V y P-V ideales contra los puntos reales mostró que el sistema de medición opera de manera correcta, aun cuando las pruebas no se realizaron bajo condiciones estándar (STC). La tendencia general observada en los datos coincide con el comportamiento esperado de un módulo fotovoltaico en operación real.

El análisis complementario realizado en Google Colab permitió clasificar los datos con K-means y generar reglas mediante un árbol de decisión. Estas técnicas confirmaron que la irradiancia es el factor principal que determina la producción de energía, mientras que la temperatura y la humedad presentan una influencia menor. El árbol derivado a partir de los datos permitió establecer criterios de operación que pueden incorporarse posteriormente en la interfaz gráfica para mostrar al usuario mensajes de diagnóstico o de advertencia.

En conjunto, los experimentos y análisis realizados permiten concluir que el prototipo cumple con los objetivos planteados: monitorear en tiempo real las variables relevantes de un panel solar, almacenar los datos para su análisis posterior y ofrecer una base confiable para implementar estrategias de evaluación del rendimiento. El sistema demuestra potencial para ampliarse hacia un monitoreo más completo, incorporar predicción energética y mejorar el soporte al usuario mediante acciones correctivas automatizadas.

6.2.- Contribuciones.

- Desarrollo de un sistema completo de adquisición de datos.
- Integración de una interfaz gráfica local en Nextion.

- Implementación de una arquitectura híbrida de almacenamiento (SQLite + Firebase).
- Generación de curvas características I–V y P–V a partir de datos reales.
- Análisis estadístico del comportamiento del panel en Google Colab.
- Aplicación de técnicas de minería de datos (K-means y Árbol de Decisión).
- Propuesta de acciones correctivas basadas en el análisis de datos.

6.3.- Recomendaciones para trabajos futuros.

- Integrar más sensores al sistema, como temperatura del módulo, velocidad del viento o nivel de radiación ultravioleta, para ampliar la caracterización ambiental del panel.
- Automatizar acciones correctivas, como implementar un sistema de riego automatizado para la limpieza del panel cuando la irradiancia disminuya por acumulación de polvo.
- Integrar visión artificial, utilizando una cámara con algoritmos de inteligencia artificial para detectar suciedad, sombras parciales o inclinación incorrecta del panel.

6.4.- Actividades realizadas.

- Participación en el evento “Detectives del Conocimiento: Ciencia y Cultura”, sede UPALT 2024.
- Participación en el 26° Certamen Estatal Creatividad e Innovación Tecnológica-Expociencias Tamaulipas 2024, con proyecto “Photech”.
- Participación en el “Foro Noreste Nearshoring y la participación de las Ingenierías en el crecimiento económico, de la Federación de egresados del Tecnológico Nacional de México.
- Participación como Ponente en el Encuentro de Jóvenes Investigadores, Tamaulipas 2024, con el tema: “Interfaz gráfica remota para el análisis de la señal eléctrica de panel solar”.
- Participación en la Cumbre Nacional de Desarrollo Tecnológico, Investigación e Innovación, “InnovaTecNM 2024”, con el proyecto “Photech”.
- Participación como Auxiliar de Tallerista en “Viernes de Detectives del Conocimiento, Ciencia y Cultura” STEAM Tamaulipas 2024.

6.5.- Publicaciones.

- Publicación en 17° Congreso Internacional de la Academia Mexicana Multidisciplinaria 2025, “Interfaz gráfica remota para el análisis de sistema solar fotovoltaico con IOT”. Autores: Juan Manuel Del Ángel Arredondo, Carolina Contreras Álvarez, Brenda Lizeth Reyes García, Ana Lidia Martínez Salazar, Pedro Martín García Vite.
- Publicación en Revista de la Academia Tamaulipeca de Investigación Científica y Tecnológica, Volumen 9, Número 1, enero – junio 2025, “Interfaz gráfica remota para el análisis de sistema solar fotovoltaico con IoT”. Autores: Del Angel Arredondo Juan Manuel, Hernández Ángel Francisca, Reyes García Brenda Lizeth, Martínez Salazar Ana Lidia, García Vite Pedro Martín.

Bibliografía.

- [1] Hauser, S. G., & Crandall, K. (2012). Smart Grid is a Lot More than Just “Technology”. En F. P. Sioshansi (Ed.), *Smart Grid* (pp. 3-28). Academic Press. <https://doi.org/10.1016/B978-0-12-386452-9.00001-2>.
- [2] Chang, L., Taghizadeh-Hesary, F., & Saydaliev, H. B. (2022). How do ICT and renewable energy impact sustainable development? *Renewable Energy*, 199, pp. 123-131. <https://doi.org/10.1016/j.renene.2022.08.082>.
- [3] Abdallah, F. S. M., Abdullah, M. N., Musirin, I., & Elshamy, A. M. (2023). Intelligent solar panel monitoring system and shading detection using artificial neural networks. *Energy Reports*, 9(10), 324-334. ISSN 2352-4847. DOI: 10.1016/j.egyr.2023.05.163.
- [4] National Geographic España. (2024, 7 marzo). National Geographic. <https://www.nationalgeographicla.com/medio-ambiente/2022/04/la-energia-solar-y-su-potencial-para-ayudar-a-reducir-el-calentamiento-global>.
- [5] United Nations. (s/f). Energías renovables: energías para un futuro más seguro | Naciones Unidas. Recuperado el 8 de marzo de 2024, de <https://www.un.org/es/climatechange/raising-ambition/renewable-energy>.
- [6] S. de Gobernación. (2016) Decreto promulgatorio del acuerdo de París, hecho en París el doce de diciembre de dos mil quince. [Online]. Available: http://dof.gob.mx/nota_detalle.php?codigo=5459825&fecha=04%2F11%2F2016.
- [7] A. Makkar, A. Raheja, R. Chawla, and S. Gupta, “IoT based framework: Mathematical modelling and analysis of dust impact on solar panels,” *3D Research*, vol. 10, 03 2019.
- [8] IPCC. (2022). *Climate Change 2022: Mitigation of Climate Change. Contribution of Working Group III to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change*. Cambridge University Press. <https://www.ipcc.ch/report/ar6/wg3/>
- [9] International Energy Agency. (2023). *World Energy Outlook 2023*. IEA Publications. <https://www.iea.org/reports/world-energy-outlook-2023>
- [10] International Renewable Energy Agency. (2024). *Renewable Power Generation Costs in 2023*. IRENA. <https://www.irena.org/Publications/2024/Aug/Renewable-Power-Generation-Costs-in-2023>
- [11] National Renewable Energy Laboratory. (2023). *Standard Scenarios Report: A U.S. Electricity Sector Outlook*. NREL/TP-6A20-89153.

- [12] La energía solar y su potencial para ayudar a reducir el calentamiento global. (2022, abril 20). National Geographic. <https://www.nationalgeographicla.com/medio-ambiente/2022/04/la-energia-solar-y-su-potencial-para-ayudar-a-reducir-el-calentamiento-global>.
- [13] Kreith, F., & Kreider, J. F. (1985). Principles of Solar Engineering. Hemisphere Publishing Corporation.
- [14] A. E. Garcia Alonso, “Maximum power point tracking algorithms for solar photovoltaic systems”, 2017.
- [15] Enel Green Power. (s.f.). Célula fotovoltaica. Recuperado 2 de agosto de 2024, de <https://www.enelgreenpower.com/es/learning-hub/energias-renovables/energia-solar/celula-fotovoltaica>.
- [16] International Electrotechnical Commission. (2018). *IEC 60904-3: Photovoltaic devices - Part 3: Measurement principles for terrestrial photovoltaic (PV) solar devices with reference spectral irradiance data* (Ed. 3.1). <https://webstore.iec.ch/publication/26479>
- [17] Sánchez-Juárez, A., Martínez-Escobar, D., Santos-Magdaleno, R. de la L., Ortega-Cruz, J., & Sánchez-Pérez, P. A. (2017). Aplicaciones fotovoltaicas de la energía solar en los sectores residencial, servicio e industrial: Guía para el dimensionamiento y diseño de sistemas fotovoltaicos. Instituto de Energías Renovables, UNAM.
- [18] International Energy Agency (IEA). (2020). Trends in Photovoltaic Applications 2020. IEA PVPS Task 1.
- [19] Messenger, R. A., & Ventre, J. (2004). Photovoltaic systems engineering (2nd ed.). CRC Press.
- [20] García García, J. A. (2024). *Diseño y construcción de un regulador de carga para un sistema fotovoltaico autónomo* [Tesis de maestría, Instituto Tecnológico de Ciudad Madero]. Repositorio Institucional del TecNM. https://rinacional.tecnm.mx/bitstream/TecNM/7937/1/G21073008_donacion_tesis_bib%20%281%29.pdf
- [21] Footprint Hero. (s.f.). *Standard Test Conditions (STC) for solar panels* [Infographic]. Footprint Hero. <https://footprinthero.com/standard-test-conditions>
- [22] M. G. Hudedmani, V. Soppimath, and C. Jambotkar, “A study of materials for solar pv technology and challenges”, European Journal of Applied Engineering and Scientific Research, vol. 5, no. 1, pp. 1–13, 2017.

- [23] O. Perpiñan, *Energía Solar Fotovoltaica*, Madrid: Creative Commons, 2013.
- [24] National Instruments. (2020). *CompactDAQ user Manual*.
<https://www.ni.com/documentation/en/compactdaq-manual/>
- [25] Rashid, M. H. (2018). *Power electronics handbook: Devices, circuits, and applications* (5th ed.). Butterworth-Heinemann.
- [26] Bose, B. K. (2006). *Power electronics and motor drives: Advances and trends*. Academic Press.
- [27] Kjaer, S. B., Pedersen, J. K., & Blaabjerg, F. (2005). "A Review of Single-Phase Grid-Connected Inverters for Photovoltaic Modules". *IEEE Transactions on Industry Applications*, 41(5), 1292-1306.
- [28] Carrasco, J. M., Franquelo, L. G., Bialasiewicz, J. T., Galván, E., Guisado, R. C. P., Prats, M. A. M., León, J. I., & Moreno-Alfonso, N. (2006). Power-electronic systems for the grid integration of renewable energy sources: A survey. *IEEE Transactions on Industrial Electronics*, 53(4), 1002-1016. <https://doi.org/10.1109/TIE.2006.878356>
- [29] Yonghui Solar. (n.d.). *1000W 30V/36V 220VAC grid tie inverter MPPT function pure sine wave inverter*. Yonghui Solar.
- [30] Growatt. (2024, enero 30). *Growatt amplía su gama de soluciones Off-Grid*. Growatt. <https://es.growatt.com/media/news/growatt-amplia-su-gama-de-soluciones-off-grid>
- [31] Institute of Electrical and Electronics Engineers. (2018). IEEE standard for interconnection and interoperability of distributed energy resources with associated electric power systems interfaces (IEEE Std 1547-2018).
- [32] INELDEC. (s.f.). *Inversor Solar 3KVA Onda Pura POWEST*. INELDEC. <https://ineldec.com/producto/inversor-solar-3kva-onda-pura-powest/>
- [33] Esram, T., & Chapman, P. L. (2007). Comparison of photovoltaic array maximum power point tracking techniques. *IEEE Transactions on Energy Conversion*, 22(2), 439-449. <https://doi.org/10.1109/TEC.2006.874230>
- [34] Fraden, J. (2016). *Handbook of Modern Sensors: Physics, Designs, and Applications* (5th ed.). Springer.
- [35] Bentley, J. P. (2005). *Principles of Measurement Systems* (4th ed.). Pearson Education.
- [36] Northrop, R. B. (2014). *Introduction to Instrumentation and Measurements* (3rd ed.). CRC Press.

- [37] Sinclair, I. R. (2001). *Sensors and Transducers* (3rd ed.). Newnes.
- [38] Figliola, R. S., & Beasley, D. E. (2011). *Theory and Design for Mechanical Measurements* (5th ed.). John Wiley & Sons.
- [39] Universidad de Tarapacá. (s.f.). Acondicionamiento de las señales. Recuperado 2 de agosto de 2024, de http://www.eudim.uta.cl/files/8313/2215/7786/fm_Ch04_mfuentesm.pdf
- [40] Chan, J. A., Molina, A., Uribe, H. E., Hernández, A., & Galván, R. (2021). Zigbee y Xbee: Características, topologías y arquitectura.
- [41] Digi International. (2005). XBee/XBee-PRO OEM RF Modules Product Manual (v1.06). <https://www.digi.com/resources/library/product-manuals>
- [42] Álvarez Torrico, R. (2013). Introducción a las redes ZigBee XBee. Cursos TecBoJiva. <https://www.cursos.tecbojiva.com>
- [43] Pérez, F., Valdés, E., & Arias, O. (2013). Sistema de Adquisición de Datos con comunicación inalámbrica. RIELAC, *34*(3), 63–73.
- [44] Introducción a los módulos XBee y ZigBee. (2021). Documento de apoyo.
- [45] MCI Electronics. (s.f.). Raspberry Pi. Recuperado 2 de agosto de 2024, de <https://raspberrypi.cl/que-es-raspberry/>
- [46] Nextion. (2022). Nextion Intelligent HMI UART Displays: The Concept. <https://nextion.tech/nextion-intelligent-hmi-uart-displays-the-concept/>
- [47] Nextion. (2023b). Nextion Editor - Download & Introduction. <https://nextion.tech/nextion-editor/>
- [48] Nextion. (2023c). Nextion Instruction Set: The Comprehensive Guide. <https://nextion.tech/instruction-set/>
- [49] Ashton, K. (2009). That 'internet of things' thing. *RFID Journal*, 22(7), 97-114.
- [50] ITU (International Telecommunication Union). (2012). Overview of the Internet of things. Serie Y.2060.
- [51] Forero, M. A. (2023, 22 de agosto). *Internet de las cosas: 5 ejemplos para la empresa*. Inforolot. <https://www.inforolot.com/internet-de-las-cosas-5-ejemplos-para-la-empresa/>
- [52] Moroney, L. (2020). The Firebase Realtime Database. En *The Definitive Guide to Firebase* (pp. 61–89). Apress. https://doi.org/10.1007/978-1-4842-5173-5_3
- [53] Hashmi, G., Hasan, S., Efat, M., & Rahman, M. H. (2020). Portable solar panel efficiency measurement system. *SN Applied Sciences*, *2*, 1234. <https://doi.org/10.1007/s42452-020-03553-w>

- [54] López-Vargas, A., Fuentes, M., García, M. V., & Muñoz-Rodríguez, F. J. (2019). Low-cost datalogger intended for remote monitoring of solar photovoltaic standalone systems based on Arduino™. *IEEE Sensors Journal*, *19*(11), 4308-4320. <https://doi.org/10.1109/JSEN.2019.2902315>
- [55] Ahmad, M. W., Mourshed, M., & Rezgui, Y. (2025). Development of a smart cloud-based monitoring system for solar photovoltaic energy generation. *Energy Reports*, *11*, 150-165. <https://doi.org/10.1016/j.egy.2025.01.039>
- [56] Sharma, P., & Verma, A. (2023). Solar panel monitoring system using IOT. *International Journal of Engineering Research & Technology*, *12*(3), 45-50.
- [57] Khan, M. A., Khan, A., & ur Rehman, A. (2021). A review of monitoring technologies for solar PV systems using data processing modules and transmission protocols: Progress, challenges and prospects. *Sustainability*, *13*(15), 8120. <https://doi.org/10.3390/su13158120>
- [58] Sirichote, W., Wuttikornkanarak, C., Srathongkao, S., Suttiyan, S., Somdock, N., & Klongratog, B. (2021). IV tracer for photovoltaic panel. En *2021 9th International Electrical Engineering Congress (iEECON)* (pp. 54-57). IEEE.
- [59] Prastianto, A., Fadhil, M., Rahardjo, A., Jufri, F. H., & Husnayain, F. (2019). Design of solar irradiance measurement based on analytical data using microcontroller. En *2019 IEEE Conference on Energy Conversion (CENCON)* (pp. 137-141). IEEE.
- [60] Rana, M., Koprinska, I., & Agelidis, V. G. (2016). Univariate and multivariate methods for very short-term solar photovoltaic power forecasting. *Energy Conversion and Management*, *121*, 380-390. <https://doi.org/10.1016/j.enconman.2016.05.025>
- [61] Gao, X.-M., Yang, S.-F., & Pan, S.-B. (2017). Optimal parameter selection for support vector machine based on artificial bee colony algorithm: A case study of grid-connected PV system power prediction. *Computational Intelligence and Neuroscience*, *2017*, 1-14. <https://doi.org/10.1155/2017/9234613>
- [62] Colak, M., Yesilbudak, M., & Bayindir, R. (2018). Very short-term estimation of global horizontal irradiance using data mining methods. En *2018 7th International Conference on Renewable Energy Research and Applications (ICRERA)* (pp. 1472-1476). IEEE.
- [63] Zhou, B., Zhao, W., Su, X., Lu, S., Wang, T., Yao, W., Xie, P., Mao, T., Guan, L., & Lv, Y. (2018). PV power characteristic modeling based on multi-scale clustering and its

application in generation prediction. En *2018 IEEE Power & Energy Society General Meeting (PESGM)* (pp. 1-5). IEEE.

Anexos

Anexo A.

Código 1.

```
#include <SoftwareSerial.h>

#include <DHT.h>

#define DHTPIN 4

#define DHTTYPE DHT11

DHT dht(DHTPIN, DHTTYPE);

SoftwareSerial XBee(2, 3);

const int voltagePin = A0;
const int currentPin = A1;
const int irradianceSensorPin = A2;

// CORRECCIÓN PARA SENSOR FZ0430
const float R1 = 30000.0;
const float R2 = 7500.0;
const float voltageDividerRatio = R2 / (R1 + R2);
const float sensitivity = 0.100;

// FILTROS MÍNIMOS para UNO
const byte numReadings = 30;
float readings[30];
byte readIndex = 0;
float totalCurrent = 0;
float offset = 0;

const byte voltageReadings = 3;
float voltageValues[3];
byte voltageIndex = 0;
float totalVoltage = 0;
```

```

const byte irradianceReadings = 3;
float irradianceValues[3];
byte irradianceIndex = 0;
float totalIrradiance = 0;

// CALIBRACIONES (constantes no usan RAM dinámica)
const float resistorValue = 250.0;          // Cambiado a 250Ω (óptimo para
4-20mA)
const float tempOffset = -7.3;
const float humOffset = +12.0;

// NUEVAS CONSTANTES
float voltageOffset = 0.0;
const float currentNoiseThreshold = 0.02;
const float voltageNoiseThreshold = 0.05;

// NUEVA CALIBRACIÓN PARA RIKA 200-04
float irradianceOffset = 0.0;
const float minSensorCurrent = 4.0;
const float maxSensorCurrent = 20.0;
const float minIrradiance = 0.0;
const float maxIrradiance = 1500.0;

void setup() {
    Serial.begin(9600);
    XBee.begin(9600);
    dht.begin();

    // Inicializar arrays más eficientemente
    memset(readings, 0, sizeof(readings));
    memset(voltageValues, 0, sizeof(voltageValues));
    memset(irradianceValues, 0, sizeof(irradianceValues));

```

```

    calibrateCurrentSensor();
    calibrateVoltageSensor();
    calibrateIrradianceSensor(); // NUEVA CALIBRACIÓN

    Serial.println(F("=== SISTEMA UNO OPTIMIZADO ==="));
    Serial.print(F("Offset irradiancia: ")); Serial.print(irradianceOffset,
2); Serial.println(F(" W/m²"));
}

void loop() {
    // LEER SENSORES

    float instantVoltage = readVoltage();
    float instantCurrent = readCurrent();
    float instantPower = instantVoltage * instantCurrent;

    float temperature, humidity;
    readDHT11(temperature, humidity);

    float irradiance = readIrradiance(); // Ahora con calibración

    // ENVIAR DATOS (debug mínimo)
    static unsigned long lastDebug = 0;
    if (millis() - lastDebug > 5000) {
        printDebug(instantVoltage, instantCurrent, instantPower, temperature,
humidity, irradiance);
        lastDebug = millis();
    }

    sendToXBee(instantVoltage, instantCurrent, instantPower, temperature,
humidity, irradiance);
    delay(500);
}

```

```

// FUNCIONES OPTIMIZADAS:

float readVoltage() {
    int voltageRaw = analogRead(voltagePin);
    float arduinoVoltage = (voltageRaw * 5.0) / 1023.0;
    float instantVoltage = (arduinoVoltage / voltageDividerRatio) -
voltageOffset;

    if (instantVoltage < 0.05) instantVoltage = 0;

    totalVoltage -= voltageValues[voltageIndex];
    voltageValues[voltageIndex] = instantVoltage;
    totalVoltage += voltageValues[voltageIndex];
    voltageIndex = (voltageIndex + 1) % 3;

    return instantVoltage;
}

float readCurrent() {
    int currentRaw = analogRead(currentPin);
    float sensorVoltage = (currentRaw * 5.0) / 1023.0;
    float instantCurrent = (sensorVoltage - offset) / sensitivity;

    if (instantCurrent < 0.02) instantCurrent = 0;

    totalCurrent -= readings[readIndex];
    readings[readIndex] = instantCurrent;
    totalCurrent += readings[readIndex];
    readIndex = (readIndex + 1) % 30;

    return instantCurrent;
}

```

```

void readDHT11(float &temperature, float &humidity) {
    humidity = dht.readHumidity();
    temperature = dht.readTemperature();

    if (!isnan(humidity) && !isnan(temperature)) {
        temperature += tempOffset;
        humidity += humOffset;
        temperature = constrain(temperature, -40, 80);
        humidity = constrain(humidity, 0, 100);
    } else {
        humidity = temperature = 0;
    }
}

// FUNCIÓN MEJORADA PARA LEER IRRADIANCIA
float readIrradiance() {
    int sensorValue = analogRead(irradianceSensorPin);
    float voltageIrradiance = sensorValue * (5.0 / 1023.0);
    float currentIrradiance = voltageIrradiance / resistorValue * 1000.0;

    // NO forzar mínimo de 4mA - permitir lecturas reales
    if (currentIrradiance < 3.5) currentIrradiance = 0; // Ruido por debajo
    de 3.5mA = 0

    if (currentIrradiance > 21.0) currentIrradiance = 21.0; // Límite
    superior con margen

    // Conversión lineal CORREGIDA
    float irradiance = 0.0;
    if (currentIrradiance >= 4.0) {
        irradiance = ((currentIrradiance - 4.0) / 16.0) * 1500.0;
    }

    // Aplicar offset de calibración
    irradiance += irradianceOffset;
}

```

```

    if (irradiance < 0) irradiance = 0;

    // Filtro
    totalIrradiance -= irradianceValues[irradianceIndex];
    irradianceValues[irradianceIndex] = irradiance;
    totalIrradiance += irradianceValues[irradianceIndex];
    irradianceIndex = (irradianceIndex + 1) % 3;

    return totalIrradiance / 3;
}

void calibrateCurrentSensor() {
    Serial.println(F("Calibrando ACS712... (0A)"));
    float sum = 0;

    for (int i = 0; i < 200; i++) {
        int raw = analogRead(currentPin);
        sum += (raw * 5.0) / 1023.0;
        delay(5);
    }

    offset = sum / 200;
    Serial.print(F("Offset corriente: ")); Serial.println(offset, 3);
}

void calibrateVoltageSensor() {
    Serial.println(F("Calibrando voltaje... (0V)"));
    float sum = 0;

    for (int i = 0; i < 100; i++) {
        int voltageRaw = analogRead(voltagePin);
        float arduinoVoltage = (voltageRaw * 5.0) / 1023.0;
        sum += arduinoVoltage / voltageDividerRatio;
    }
}

```

```

        delay(5);
    }

    voltageOffset = sum / 100;
    Serial.print(F("Offset voltaje: ")); Serial.println(voltageOffset, 3);
}

// NUEVA FUNCIÓN PARA CALIBRAR SENSOR RIKA 200-04
void calibrateIrradianceSensor() {
    Serial.println(F("Calibrando sensor RIKA 200-04..."));
    Serial.println(F("Cubrir completamente el sensor (0 W/m² esperado)..."));
    Serial.println(F("Esperando 5 segundos..."));

    delay(5000); // Tiempo para cubrir el sensor

    float sum = 0;
    int calibrationReadings = 150;

    Serial.println(F("Midiendo irradiancia en oscuridad..."));

    for (int i = 0; i < calibrationReadings; i++) {
        int sensorValue = analogRead(irradianceSensorPin);
        float voltageIrradiance = sensorValue * (5.0 / 1023.0);
        float currentIrradiance = voltageIrradiance / resistorValue * 1000.0;

        // Calcular irradiancia actual
        float currentIrrad = 0.0;
        if (currentIrradiance >= 4.0) {
            currentIrrad = ((currentIrradiance - 4.0) / 16.0) * 1500.0;
        }

        sum += currentIrrad;
    }
}

```

```

// Mostrar progreso cada 30 lecturas
if (i % 30 == 0) {
    Serial.print(F("Lectura ")); Serial.print(i);

    Serial.print(F("/")); Serial.print(calibrationReadings);

    Serial.print(F(" - Corriente: ")); Serial.print(currentIrradiance,
2); Serial.print(F("mA"));

    Serial.print(F(" - Irrad: ")); Serial.print(currentIrrad, 2);
Serial.println(F(" W/m²"));
}

delay(50);
}

float averageIrradiance = sum / calibrationReadings;
irradianceOffset = -averageIrradiance; // Compensar el valor medido

Serial.println(F("=== CALIBRACIÓN IRRADIANCIA COMPLETADA ==="));

Serial.print(F("Irradiancia medida en oscuridad: "));
Serial.print(averageIrradiance, 2); Serial.println(F(" W/m²"));

Serial.print(F("Offset calculado: ")); Serial.print(irradianceOffset, 2);
Serial.println(F(" W/m²"));

if (abs(averageIrradiance) > 50.0) {
    Serial.println(F("⚠ ADVERTENCIA: Valor en oscuridad > 50 W/m²"));

    Serial.println(F(" Verificar que el sensor esté completamente
cubierto"));
} else {
    Serial.println(F("✓ Calibración dentro de rangos normales"));
}

Serial.println(F("====="));
}

void printDebug(float volt, float curr, float power, float temp, float hum,
float irrad) {
    Serial.print(F("V:")); Serial.print(volt, 2);

```

```

    Serial.print(F("V I:")); Serial.print(curr, 3);
    Serial.print(F("A P:")); Serial.print(power, 2);
    Serial.print(F("W T:")); Serial.print(temp, 1);
    Serial.print(F("C H:")); Serial.print(hum, 1);
    Serial.print(F("% R:")); Serial.print(irrad, 1);
    Serial.println(F("W/m2"));
}

void sendToXBee(float volt, float curr, float power, float temp, float hum,
float irrad) {
    XBee.print(F("V:")); XBee.print(volt, 2);
    XBee.print(F(",I:")); XBee.print(curr, 3);
    XBee.print(F(",P:")); XBee.print(power, 3);
    XBee.print(F(",H:")); XBee.print(hum, 2);
    XBee.print(F(",T:")); XBee.print(temp, 2);
    XBee.print(F(",R:")); XBee.println(irrad, 2);
}

```

Anexo B.

Código 2.

```
import serial
import time
import sqlite3
import glob
import os

from datetime import datetime, time as time_obj
from collections import deque

# -----
#   FUNCIONES DE PUERTOS
# -----

def encontrar_puerto_serial():
    puertos = glob.glob('/dev/ttyUSB*')
    if puertos:
        return puertos[0]
    else:
        raise Exception("No se encontró ningún puerto serial para XBee.")

# -----
#   CONEXIÓN A NEXTION
# -----

ser_nextion = serial.Serial('/dev/ttyS0', 9600)

# -----
#   CONEXIÓN A XBEE
# -----

try:
    puerto_xbee = encontrar_puerto_serial()
    ser_xbee = serial.Serial(puerto_xbee, 9600)
    print(f"Conectado a XBee en {puerto_xbee}")
```

```

except Exception as e:
    print("Error al conectar al XBee:", e)
    exit()

# -----
#   CONFIGURACIÓN SQLITE
# -----

db_path = '/home/juanmanuel/Desktop/Proyecto/sensor_data.db'
os.makedirs(os.path.dirname(db_path), exist_ok=True)

conn = sqlite3.connect(db_path)
cursor = conn.cursor()

cursor.execute('''
CREATE TABLE IF NOT EXISTS sensor_data (
    id INTEGER PRIMARY KEY AUTOINCREMENT,
    timestamp TEXT NOT NULL,
    voltage REAL NOT NULL,
    current REAL NOT NULL,
    power REAL NOT NULL,
    temperature REAL NOT NULL,
    humidity REAL NOT NULL,
    irradiance REAL NOT NULL,
    status TEXT NOT NULL
)
''')
conn.commit()

print("Base de datos SQLite lista:", db_path)

# -----
#   CONFIGURACIÓN FIREBASE

```

```

# -----
print("Inicializando Firebase...")

firebase_initialized = False

try:
    import firebase_admin
    from firebase_admin import credentials, db

    cred =
credentials.Certificate('/home/juanmanuel/Desktop/Proyecto/panel-solar-
final-firebase-adminsdk-fbsvc-5298240eaf.json')

    firebase_admin.initialize_app(cred, {
        'databaseURL': 'https://panel-solar-final-default-
rtadb.firebaseio.com/'
    })

    firebase_initialized = True
    print("Firebase conectado correctamente")

except Exception as e:
    print("Error al conectar Firebase:", e)
    print("Continuando solo con SQLite...")

# -----
#  FUNCIÓN PARA GUARDAR EN FIREBASE
# -----

def save_to_firebase(timestamp, voltage, current, power, temperature,
humidity, irradiance, status):
    if not firebase_initialized:
        return False

    try:
        ref = db.reference('sensor_data')

```

```

        ref.push({
            'timestamp': timestamp,
            'voltage': voltage,
            'current': current,
            'power': power,
            'temperature': temperature,
            'humidity': humidity,
            'irradiance': irradiance,
            'status': status
        })

    return True

except Exception as e:
    print("Error al guardar en Firebase:", e)
    return False


# -----
#  ENVÍO A NEXTION
# -----

def send_to_nextion_text(objname, value):
    command = f'{objname}.txt="{value}"'
    ser_nextion.write(command.encode('iso-8859-1'))
    ser_nextion.write(b'\xff\xff\xff')


def change_nextion_page(page_number):
    command = f'page {page_number}'
    ser_nextion.write(command.encode('iso-8859-1'))
    ser_nextion.write(b'\xff\xff\xff')


# -----
#  CLIMA SEGÚN IRRADIANCIA
# -----

```

```

def get_weather_text(irradiance):
    if irradiance >= 800:
        return "Cielo despejado"
    elif irradiance >= 500:
        return "Ligera bruma"
    elif irradiance >= 200:
        return "Producción baja"
    else:
        return "Muy baja"

# -----
#   RESET SI SE DESCONECTA EL ARDUINO
# -----

def reset_nextion_values():
    send_to_nextion_text("t6", "0.00 V")
    send_to_nextion_text("t7", "0.000 A")
    send_to_nextion_text("t8", "0.000 W")
    send_to_nextion_text("t9", "0.00 %")
    send_to_nextion_text("t10", "0.00 °C")
    send_to_nextion_text("t11", "0.00 W/m²")
    send_to_nextion_text("t12", "--")
    print("Nextion en ceros por desconexión")

# -----
#   RECORDS EN PÁGINA 1
# -----

current_record_index = 0
total_records = 0
last_nextion_page = 0 # 0: pantalla de tiempo real

def update_record_counter():
    cursor.execute("SELECT COUNT(*) FROM sensor_data")
    global total_records

```

```

total_records = cursor.fetchone()[0]

def load_current_record():
    global current_record_index, total_records

    if total_records == 0:
        send_to_nextion_text("t13", "--:--:--")
        send_to_nextion_text("t14", "0.00 V")
        send_to_nextion_text("t15", "0.000 A")
        send_to_nextion_text("t16", "0.000 W")
        send_to_nextion_text("t17", "0.00 W/m²")
        send_to_nextion_text("t18", "0.00 °C")
        send_to_nextion_text("t19", "0.00 %")
        return

    cursor.execute("SELECT * FROM sensor_data ORDER BY id DESC LIMIT 1
OFFSET ?",
                    (current_record_index,))
    record = cursor.fetchone()

    if record:
        timestamp = record[1]
        dt = datetime.strptime(timestamp, "%Y-%m-%d %H:%M:%S")
        hora_str = dt.strftime("%H:%M:%S")

        send_to_nextion_text("t13", hora_str)
        send_to_nextion_text("t14", f"{record[2]:.2f} V")
        send_to_nextion_text("t15", f"{record[3]:.3f} A")
        send_to_nextion_text("t16", f"{record[4]:.3f} W")
        send_to_nextion_text("t17", f"{record[7]:.2f} W/m²")
        send_to_nextion_text("t18", f"{record[5]:.2f} °C")
        send_to_nextion_text("t19", f"{record[6]:.2f} %")

```

```

# -----
#   HORARIO SOLAR
# -----

SUNRISE_TIME = time_obj(6, 0)
SUNSET_TIME = time_obj(20, 0)

def is_daytime():
    now = datetime.now().time()
    return SUNRISE_TIME <= now <= SUNSET_TIME

# -----
#   ACUMULADOR 1 MINUTO
# -----

data_accumulator = {
    'voltage': deque(maxlen=60),
    'current': deque(maxlen=60),
    'power': deque(maxlen=60),
    'temperature': deque(maxlen=60),
    'humidity': deque(maxlen=60),
    'irradiance': deque(maxlen=60)
}

last_save_time = time.time()

# -----
#   INICIO DEL SISTEMA
# -----

update_record_counter()

print("=====")

```

```

print("SISTEMA DE MONITOREO SOLAR")
print("SQLite + Firebase + Nextion")
print("=====")

last_data_received = time.time()
connection_lost = False

# -----
#   LOOP PRINCIPAL
# -----

try:
    while True:
        current_time = time.time()

        # -----
        #   LECTURA XBEE
        # -----

        if ser_xbee.in_waiting > 0:
            data = ser_xbee.readline().decode('utf-8').strip()
            values = data.split(',')

            if len(values) == 6:
                last_data_received = current_time
                if connection_lost:
                    print("Conexión restaurada")
                    connection_lost = False

                voltage = float(values[0].split(':')[1])
                current = float(values[1].split(':')[1])
                power = float(values[2].split(':')[1])
                temperature = float(values[3].split(':')[1])
                humidity = float(values[4].split(':')[1])
                irradiance = float(values[5].split(':')[1])

```

```

# Actualizar pantalla en tiempo real
if last_nextion_page == 0:
    send_to_nextion_text("t6", f"{voltage:.2f} V")
    send_to_nextion_text("t7", f"{current:.3f} A")
    send_to_nextion_text("t8", f"{power:.3f} W")
    send_to_nextion_text("t9", f"{humidity:.2f} %")
    send_to_nextion_text("t10", f"{temperature:.2f} °C")
    send_to_nextion_text("t11", f"{irradiance:.2f} W/m²")
    send_to_nextion_text("t12",
get_weather_text(irradiance))

# Acumular para guardado
if is_daytime():
    data_accumulator['voltage'].append(voltage)
    data_accumulator['current'].append(current)
    data_accumulator['power'].append(power)
    data_accumulator['temperature'].append(temperature)
    data_accumulator['humidity'].append(humidity)
    data_accumulator['irradiance'].append(irradiance)

# -----
# DETECCIÓN DE DESCONEXIÓN
# -----
if current_time - last_data_received > 10 and not connection_lost:
    print("Arduino no responde")
    connection_lost = True
    reset_nextion_values()

# -----
# GUARDADO DE PROMEDIOS (1 MIN)
# -----
if is_daytime() and (current_time - last_save_time >= 60):

```

```

if len(data_accumulator['voltage']) > 0:

    avg_voltage = round(sum(data_accumulator['voltage']) /
len(data_accumulator['voltage']), 3)

    avg_current = round(sum(data_accumulator['current']) /
len(data_accumulator['current']), 3)

    avg_power = round(sum(data_accumulator['power']) /
len(data_accumulator['power']), 3)

    avg_temperature =
round(sum(data_accumulator['temperature']) /
len(data_accumulator['temperature']), 3)

    avg_humidity = round(sum(data_accumulator['humidity']) /
len(data_accumulator['humidity']), 3)

    avg_irradiance = round(sum(data_accumulator['irradiance'])
/ len(data_accumulator['irradiance']), 3)

    timestamp = datetime.now().strftime("%Y-%m-%d %H:%M:%S")

    # Guardar en SQLite
    cursor.execute('''
INSERT INTO sensor_data
(timestamp, voltage, current, power, temperature, humidity,
irradiance, status)
VALUES (?, ?, ?, ?, ?, ?, ?, ?)
''', (timestamp, avg_voltage, avg_current, avg_power,
avg_temperature, avg_humidity, avg_irradiance,
"active"))

    conn.commit()

    # Guardar en Firebase
    if firebase_initialized:
        ok = save_to_firebase(timestamp, avg_voltage,
avg_current, avg_power,
avg_temperature, avg_humidity,
avg_irradiance, "active")

```

```

        print("Firebase:", "OK" if ok else "FALLÓ")

        print(f"Promedio guardado {timestamp}")

        data_accumulator = {key: deque(maxlen=60) for key in
data_accumulator}

        last_save_time = current_time

        time.sleep(0.1)

except KeyboardInterrupt:
    print("\nSistema detenido.")
    ser_nextion.close()
    ser_xbee.close()
    conn.close()

```