



**Desarrollo de sistema SCADA y de
Comunicación inalámbrica aplicable a
prototipo de limpieza de Paneles Fotovoltaicos
“CLEAN OPTIMIZER”**

**Memoria para optar al título de
Ingeniero Civil Mecatrónico**

Profesor Guía : Dr. Javier Muñoz Vidal

Christian Antonio Romero Villagra

Curicó – Chile

Octubre 2019

CONSTANCIA

La Dirección del Sistema de Bibliotecas a través de su encargado Biblioteca Campus Curicó certifica que el autor del siguiente trabajo de titulación ha firmado su autorización para la reproducción en forma total o parcial e ilimitada del mismo.



Two circular official stamps and handwritten signatures in blue ink. The left stamp is from the 'DIRECCIÓN SISTEMA DE BIBLIOTECAS UNIVERSIDAD DE TALCA' and the right stamp is from the 'SISTEMA DE BIBLIOTECAS CAMPUS CURICO'.

Curicó, 2022

**Desarrollo de sistema SCADA y de
Comunicación inalámbrica aplicable a
prototipo de limpieza de Paneles Fotovoltaicos
“CLEAN OPTIMIZER”**

Christian Antonio Romero Villagra

Octubre 2019

© Christian Antonio Romero Villagra

Resumen

El siguiente proyecto se realizó dentro del marco del proyecto Clean Optimizer y la facultad de ingeniería de la Universidad de Talca.

El proyecto se realizó debido a la necesidad de responder de manera óptima ante los problemas y desventajas de algunas alternativas chilenas utilizadas en limpieza de paneles fotovoltaicos. Especialmente en el prototipo Clean Optimizer que quiere convertirse en una alternativa para poder suplir problemas asociados a la limpieza de paneles fotovoltaicos como escases del recurso hídrico, poco resguardo de los paneles fotovoltaicos y personal a cargo. elevados costos asociados a cableado para cubrir largas distancias y detención de la producción para generar limpiezas, entre otros.

El desarrollo de esta memoria de título muestra cómo se investigó, diseñó e implemento un sistema de comunicación de tipo inalámbrica y además un sistema SCADA para monitorear la aplicación de un prototipo limpiador de paneles fotovoltaico que lleva como nombre Clean Optimizer.

En esta memoria de título se emularon elementos como sensores y actuadores utilizados por Clean Optimizer para acceder a información y manipulación de parámetros que inciden en el desempeño del prototipo. Todo esto con el objetivo de poder manipular elementos como el motor, bomba de agua y sensores de nivel desde un sistema en el cual un operador pueda monitorear a cierta dicho proceso, dando un sello de posibles mantenimientos a futuro para generar una ventaja competitiva

La primera parte consta de la implementación de comunicación inalámbrica, basada en una topología de red punto a punto, la cual se basa en comunicación de largo alcance basada en tecnología ZigBee regulada por el estándar IEEE 802.15.4 y parte de comunicación basada en nivel de capa física del Modelo OSI. La segunda parte consta de desarrollo de un sistema SCADA, el cual fue diseñado, programado y probado bajo el software Visual Studio. Finalmente, la última parte consta de interconectar distintas tecnologías, logrando comunicación entre ellas, la cual se basó sobre la arquitectura maestro/esclavo mediante el protocolo Modbus/RTU.

Agradecimientos

Se agradece a los profesores que me formaron cómo profesional, a los profesores que me ayudaron a resolver dudas para la elaboración y desarrollo de esta memoria de título cómo: Javier Muñoz, Daniel Diaz, Roberto Ramírez. Se agradece el aporte de los estudiantes y profesor del Laboratorio de Energías Renovables y Acondicionamiento Eléctrico (LERAE) cómo: Ignacio Torres, Christopher Soto, Daniel Herrera, Rodrigo Aliaga, Sebastián Rojas, entre otros. Finalmente se agradece a todos los que aportaron en la confección de este informe, a los profesores por la formación académica y las personas que aportaron ya sea con conocimientos y consejos.

Dedicatoria

En primer lugar, a mi madre María Teresa, mi padre Christian, mi hermana Valentina, mi hermano Nicolás, mi abuela Elsa, mi abuelo Antonio y a toda mi familia. En segundo lugar, a mis amigos de toda la vida, a mis amigos de la universidad, a Christopher y en especial a los que de alguna u otra forma hicieron todo esto posible.

Índice

Capítulo 1. Introducción.....	12
1.1.- Introducción General	12
1.1.2.-Solución.....	14
1.2.- Revisión Bibliográfica	15
1.2.1.-Trabajos Previos	15
1.2.2.-Discusión Bibliográfica	23
1.3.- Hipótesis de trabajo	25
1.4.- Objetivos.....	26
1.4.1.- Objetivo General.....	26
1.4.2.- Objetivos Específicos	26
1.5.- Alcances	27
1.6.- Metodología	27
Capítulo 2. Diseño del sistema de comunicación inalámbrica.	28
2.1.- Clasificación de las comunicaciones inalámbricas	28
2.2.- ZigBee.....	30
2.2.1.-Dispositivos de tecnología ZigBee	30
2.2.2.-Red ZigBee	32
2.2.3.-Topologías para redes de tecnología XBee.....	33
2.3.- Selección dispositivos de radio frecuencia	34
2.3.1.-Módulo XBee pro 900HP-S3B	35
2.4.- Selección dispositivos de radio frecuencia	36
2.4.1.-Funcionamiento red inalámbrica.....	36
2.5.- Software XCTU	37
2.5.1.-Configuración Software XCTU	38
2.5.2.-Configuración dispositivos de radio frecuencia	38
2.5.3.-Modo Transparente	40
Capítulo 3. Microcontrolador	41

3.1.-	Arduino	41
3.2.-	Comunicación Serial.....	42
3.3.-	Placa Microcontroladora R3	42
3.3.1.-	<i>Microcontrolador ATmega328</i>	44
3.4.-	XBee Shield	46
3.5.-	USB.....	47
Capítulo 4. Implementación sistema SCADA.....		48
4.1.-	Unidad Terminal Remota.....	49
4.2.-	Protocolo Modbus.....	49
4.2.1.-	<i>Transmisión de datos</i>	50
4.2.2.-	<i>Trama de datos</i>	51
4.3.-	SISTEMA HMI.....	52
4.3.1.-	<i>Implementación sistema HMI</i>	52
4.4.-	Unidad Orientada	53
4.5.-	Unidad Automática	54
4.6.-	Unidad Planificada.....	55
4.7.-	Almacenamiento de datos	56
4.8.-	Monitoreo de Suciedad	56
4.8.1.-	<i>Detección de suciedad</i>	56
4.8.2.-	<i>Grafico porcentaje de sombreado estimado por efecto de suciedad</i>	58
Capítulo 5. Implementación sistema comunicación inalámbrica		59
5.1.-	Sistema de Comunicación Inalámbrica.....	59
5.1.1.-	<i>Configuración red inalámbrica</i>	60
5.1.2.-	<i>Configuración nodos de red punto a punto</i>	60
5.2.-	Módulo Remoto	61
5.2.1.-	<i>Comunicación entre placa microcontroladora y Dispositivo XBee</i>	63
5.3.-	Módulo Central	63
5.3.1.-	<i>Comunicación entre módulo remoto y Módulo central</i>	64
Capítulo 6. Sistema de Alimentación.....		65
6.1.-	<i>Descripción del sistema de alimentación</i>	65
Capítulo 7. Resultados		67

7.1.- Emulación Sistema	67
7.2.- Clean Optimizer	70
7.2.1.- <i>Módulo Central</i>	70
7.3.- Sistema SCADA	71
7.4.- Sistema SCADA para movimiento de Clean Optimizer	72
7.4.1.- <i>Pruebas de movimiento para limpieza de paneles fotovoltaicos</i>	73
7.4.2.- <i>Unidad de Detención</i>	74
7.4.3.- <i>Mover Derecha</i>	77
7.4.4.- <i>Activar y desactivar bomba de agua</i>	78
7.4.4.- <i>Activar Bomba</i>	79
7.4.5.- <i>Detener Bomba</i>	79
7.5.- Nivel de Líquido	80
7.5.1.- <i>Nivel máximo</i>	81
7.5.2.- <i>Nivel Mínimo</i>	81
7.5.3.- <i>Unidad SCADA para sistema de agua</i>	82
7.6.- Monitoreo de porcentaje estimado de sombreado por suciedad	83
7.6.1.- <i>Panel Fotovoltaico sin suciedad</i>	83
7.6.2.- <i>Panel Fotovoltaico con suciedad</i>	84
7.6.3.- <i>Panel Fotovoltaico con varios tipos de suciedad</i>	85
Capítulo 8. Conclusiones	87
8.1.- Trabajos Futuros	88
Capítulo 9. Bibliografía	89
Capítulo 10. Anexos	92
10.1.- Código Nodos de Comunicación Inalámbrica	92
10.1.- Código Modbus Serial	103
10.2.- Latencia	104

Índice de Figuras

Fig 1.1: Efecto de suciedad en paneles fotovoltaicos.....	1
Fig 1.2: Problema/solución para prototipo “Clean Optimizer”	4
Fig 2.2: Logo tecnología Zigbee	19
Fig 2.3: Tipos de antenas dispositivos XBee	20
Fig 2.4: Topologías de red.....	22
Fig 2.5: XBee Pro 900 HP-S3B.....	25
Fig 2.5: Red punto a punto mediante dispositivos XBee	25
Fig 2.7: Dispositivo XBee Explorer.	26
Fig 2.8: Reconocimiento Módulos RF	27
Fig 2.9: Configuración parámetros dispositivos XBee.....	29
Fig 3.1: Tecnología Arduino	30
Fig 3.2: Placa microcontroladora Arduino R3	32
Fig 3.3: Microcontrolador ATmega 328.....	33
Fig 3.4: Esquema pines ATmega 328.....	33
Fig 3.5: Microcontrolador ATmega 328.....	35
Fig 4.1: Estructura sistema completo.	37
Fig 4.2: Trama maestro/esclavo de sistema de comunicación	40
Fig 4.3: Diseño sistema HMI de sistema SCADA	41
Fig 4.4: Diagrama limpieza Orientada	42
Fig 4.5: Diagrama Limpieza Automática	44
Fig 4.6: Diagrama Limpieza Planificada.....	44
Fig 4.7: Escaneo de suciedad en paneles fotovoltaicos	47
Fig 5.1: Sistema de comunicación inalámbrica	48
Fig 5.2: Esquema distribución de red inalámbrica	49
Fig 5.3: Parámetros red inalámbrica.....	50
Fig 5.4: Acoplamiento placa microcontrolador y nodo XBee.....	50

Fig 5.5: Módulo Central	52
Fig 6.1: Alimentación sistema emulado	54
Fig 6.2: Batería alimentación Módulo remoto.....	54
Fig 6.3: Alimentación Módulo central	55
Fig 7.1: Setup Experimental para emular elementos de Clean Optimizer.....	56
Fig 7.2: Acoplamiento XBee pro 900HP-S3B con XBee Explorer.	57
Fig 7.3: Acoplamiento extensor USB/UTP mediante conector RJ45.	58
Fig 7.4: Extensión de cable mediante conversor UTP/USB.....	58
Fig 7.5: Modo Online para Sistema Inalámbrico	60
Fig 7.6: SCADA conectado a “Clean Optimizer”	61
Fig 7.7: Posición Inicial en prueba de movimiento.....	64
Fig 7.8: Prueba de movimiento sentido izquierdo.....	65
Fig 7.9: Unidad de limpieza sistema SCADA	66
Fig 7.10: Bloque de Bomba de agua en sistema SCADA	67
Fig.7.11: Bloque nivel de líquido en sistema SCADA.....	69
Fig.7.12: Unidad de sistema de agua de sistema SCADA.....	71
Fig 7.13: Escaneo sobre panel fotovoltaico sin suciedad	72
Fig 7.14: Monitoreo porcentaje estimado de medición anterior.....	73
Fig 7.15: Monitoreo de escaneo sobre panel fotovoltaico sin suciedad	73
Fig 7.16: Escaneo sobre panel fotovoltaico con suciedad localizada.....	73
Fig.7.17: Escaneo sobre panel fotovoltaico variación de suciedad	75
Fig 7.18: Monitoreo de panel fotovoltaico variación de suciedad	77

Índice de Tablas

Tabla 1.- Comparación entre tecnología Zigbee versus Lora	23
Tabla 2.- Características Módulo XBee pro 900HP-S3B.....	24
Tabla 3.- Características microcontroladora arduino.....	32
Tabla 4.-Características microcontrolador ATmega328-PU	34
Tabla 5.- Pines de conexión USB	36
Tabla 6.- Acceso de esclavo en Modbus	39
Tabla 7- Control maestro/esclavo ‘Porcentaje estimado de suciedad’	46
Tabla 8- Conexión sensores con placa microcontroladora.....	51
Tabla 9- Conexión actuadores con placa microcontroladora	51
Tabla 10- Conexión dispositivo XBee con placa microcontroladora.....	52
Tabla 11- Configuración de parámetros para nodo remoto y nodo central	53
Tabla 12- Características batería para módulo remoto.....	55
Tabla 13- Comunicación nodo remoto y microcontrolador ATmega328	59
Tabla 14- Comunicación nodo remoto y nodo central	59
Tabla 15.- Control maestro/esclavo modo ‘mover izquierda’	62
Tabla 16.- Control maestro/esclavo modo ‘parada’	63
Tabla 17- Control maestro/esclavo modo ‘mover derecha’	66
Tabla 18- Control maestro/esclavo modo ‘Encender Bomba’	68
Tabla 19- Control maestro/esclavo modo ‘Detener Bomba’	68
Tabla 20- Control maestro/esclavo modo ‘Nivel Mínimo’	70

Abreviaciones

HMI	: Interfaz Hombre Máquina
SCADA	: Supervisión Control y Adquisición de Datos
DECT	: Telecomunicaciones Digitales Inalámbricas mejoradas
IEEE	: Institute of Electrical and Electronics Engineers
FFD	: Dispositivos de Función completa
IrDa	: Asociación de Función Completa (Full Function Device)
NC	: Nodo Central
NFC	: Comunicación de Campo Cercano
NR	: Nodo Remoto
PAN	: Red Área Personal (Personal Area Network).
PC	: Computador Personal (Personal Computer).
PCB	: Placa de Circuito Impreso (Printed Circuit Board).
RFD	: Dispositivo de Función Reducida (Reduced Function Device).
RSII	: Indicador de Intensidad de Señal
RTC	: Reloj de Tiempo Real (Real Time Clock).
SCL	: Línea Serial de Reloj (Serial Clock Line).
SDA	: Línea Serial de Datos (Serial Data Line).
SPI	: Interfaz Periférica Serial (Serial Peripheral Interfaz).
UART	: Transmisor-Receptor Asíncrono Universal.
WSN	: Redes Inalámbricas de Sensores (Wireless Sensor Networks).
WPAN	: Redes Inalámbricas de Área Personal (Wireless Personal Area Networks).
WLAN	: Redes Inalámbricas de Área Local (Wireless Local Area Networks).
XCTU	: Configuración XBee y Utilidad de Prueba

Capítulo 1. Introducción

1.1.- Introducción General

La limpieza de paneles fotovoltaicos se ha convertido en un tema de gran relevancia en el área de energías renovables. La suciedad en paneles fotovoltaicos puede ser provocada por el polvo, arena, partículas transportadas por el aire, diferentes tipos de partículas contaminantes presentes en vapor, humo, hollín, deposiciones de aves, entre otros factores. Este fenómeno ha generado gran preocupación debido a que puede afectar hasta un 40% su generación de energía.

Alternativas como “Eccopia E4”, “SolarBrush”, “WashPanel”, entre otras son alternativas que están solo enfocadas en remover la suciedad, pero el problema radica en que ninguna se enfoca en los requerimientos de la limpieza de paneles fotovoltaicos, tales como: realizar limpiezas en donde no se tenga que detener la producción, en donde las limpiezas no sean invasivas, en donde las limpiezas no expongan al operador a riesgos, en donde se deba cuidar la vida útil de los paneles fotovoltaicos, en donde se cuide el recurso hídrico, en donde se reduzcan costos, en donde se responda ante eventualidades paneles en lugares donde el operador tengo un difícil acceso al robot limpiador ya sea por la sinuosidad del terreno o distancia.

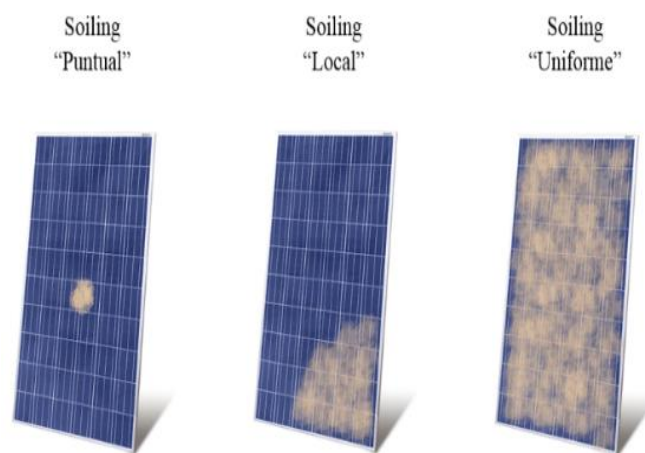


Fig. 1.1 Efecto de suciedad en Módulos fotovoltaicos

La alternativa de nombre “Clean Optimizer”, está enfocada en solo remover la suciedad, mediante un sistema rodillo y sistema de agua, utilizando una bomba y sensores de nivel de líquido. Debido a este enfoque es que se podrían generar los siguientes problemas en “Clean Optimizer” y en el proceso en sí:

Seguridad: si se quisiera realizar una limpieza sin detener la producción de energía, tendría que realizarse de noche. Pero se podría generar un gran problema para operar el robot de noche, debido a que no posee un modo para realizar limpiezas nocturnas, segundo cómo el robot necesita un botón para hacerlo funcionar necesita si o si de un operario, por lo que, si el si existe un difícil acceso por la sinuosidad del terreno o altura, esto generaría un gran problema. Otro problema se puede dar cuando el robot sufra alguna eventualidad cómo atacarse con algo pequeño si el robot se detiene y se encuentra en altura, existe un peligro latente para la persona encargada y para la vida útil de paneles fotovoltaicos.

Resguardar el recurso Hídrico: Es de vital importancia resguardar el sistema de agua, debido a que la escasez del recurso hídrico sobre todo en el norte del país. Debido a esto el gran problema que posee “Clean Optimizer”, es que no posee un sistema de alerta de los niveles de líquido, sobre todo saber cuándo se tiene un nivel mínimo de líquido para realizar una posible recarga, cómo también saber cuándo el estanque se encuentra en su máxima capacidad, esto ayudaría a realizar un uso inteligente del recurso hídrico.

Estrategias de limpieza: El problema de la suciedad es que es variable y puede depositarse en de manera puntual, local o uniforme en cualquier momento incluso después de haber realizado la limpieza, porque lo que “Clean Optimizer” no posee un sistema para responder de manera rápida ante ese problema.

Mantenimiento: El gran problema es que “Clean Optimizer”, no posee un sistema para ver si sus elementos están funcionando de manera correcta, para esto sí o sí se necesita monitorear tanto su parte mecánica cómo también la parte de los sensores.

1.1.2.-Solución

La solución ideal es que la que cumpliera con las exigencias y requerimientos que se dan en la limpieza de plantas fotovoltaicas, las cuales se basan en cubrir filas de paneles ubicadas a grandes distancias donde se deba acceder a la información del proceso de limpieza, en plantas ubicadas en lugares donde los factores geográficos afecten el acceso de personal e información del estado del panel fotovoltaico y donde el recurso hídrico sea un factor clave. Debido a esto nace la oportunidad de integrar un sistema el cual pueda cubrir con estas necesidades y se puede aplicar sobre “Clean Optimizer” para convertirse en el prototipo chileno con un elemento diferenciador y que en futuro genere un impacto en la limpieza rutinaria de paneles fotovoltaicos. Finalmente, la implementación de un sistema de monitoreo integrado en este prototipo podría generar una oportunidad para darle mayor robustez a la limpieza de paneles fotovoltaicos en donde puedan abrirse nuevas líneas de investigación y mejoras en el ámbito de las energías renovables.

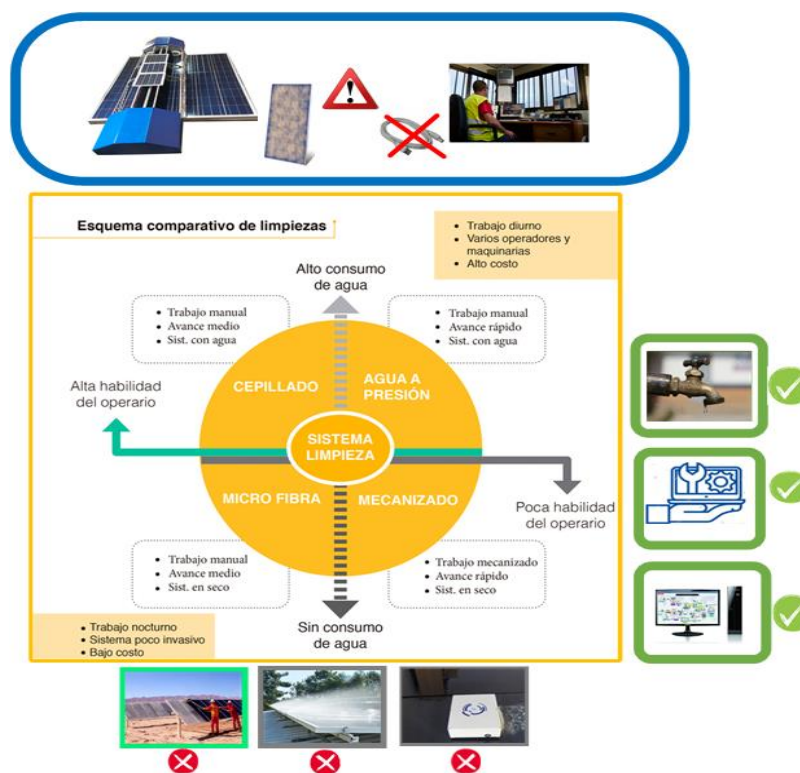


Fig. 1.2: Problema/solución para prototipo “Clean Optimizer”

1.2.- Revisión Bibliográfica

1.2.1.-Trabajos Previos

A. Aspectos Generales de las comunicaciones inalámbricas para robots

- L. Fields, “*Comparación de sistemas de comunicación inalámbrica para robots móviles*” 4, N.º 1, edición 2013 Universidad Tecnológica de Panamá.

Este documento aporta información relevante para el desarrollo de esta memoria de título, debido a que, al analizar este documento, se obtiene cómo ayuda que la aplicación de los sistemas de comunicación inalámbrica se ha convertido en un desafío actual debido a las limitantes que presenta la comunicación cableada en cuanto a la movilidad del robot y los recursos necesarios. Por otra parte, el artículo revisado aporta información para realización de este proyecto debido a la comparación de algunos sistemas de comunicación inalámbrica tales como comunicación por radio frecuencia, ultrasónica o laser. Donde la comunicación ultrasónica tiene poco alcance en el aire y más aplicable en comunicaciones subacuáticas. Por otro lado, la comunicación por láser posee la gran desventaja requerir siempre condiciones atmosféricas favorables y línea de visión directa exacta entre emisor y receptor. Finalmente, este artículo sirve para entender que para desarrollar un sistema en base a comunicación por radiofrecuencia es necesario establecer algunas normas entre el emisor y receptor, una de ellas es definir la frecuencia de onda a utilizar y el sistema de codificación/decodificación empleado para la conversión analógica/digital. Por último, hace referencia a que se puede tener comunicación confiable debido a que es posible encriptar el emisor y receptor.

B. Redes Inalámbricas

- J. Prieto. “Introducción a los sistemas de comunicación inalámbricos”. Universidad Oberta de Cataluña.

[https://www.exabyteinformatica.com/uoc/Informatica/Tecnologia_y_desarrollo_en_dispositivos_moviles/Tecnologia_y_desarrollo_en_dispositivos_moviles_\(Intro\).pdf](https://www.exabyteinformatica.com/uoc/Informatica/Tecnologia_y_desarrollo_en_dispositivos_moviles/Tecnologia_y_desarrollo_en_dispositivos_moviles_(Intro).pdf)

Este artículo ayuda a dilucidar que es necesario evaluar y analizar las posibilidades en base a la comunicación inalámbrica, en donde se debe conocer los tipos, clasificación, característica, etc. Donde se investiga que dentro de los sistemas de comunicación inalámbrica se encuentran tres grandes grupos: las WPAN que son redes inalámbricas de área personal, las WLAN que son redes inalámbricas de área local y las WWAN que son redes inalámbricas de área global.

Estos sistemas de transmisión inalámbricos cuentan con dispositivos que permiten la generación de señales de transmisión para su respectiva comunicación. En el caso de redes WPAN, existe actualmente un protocolo de comunicación regido por el estándar IEEE 802.15.4 denominado ZigBee, este protocolo permite habilitar redes inalámbricas con capacidad de control, manipulación y monitoreo, además son redes seguras, de bajo coste de procesador y de muy bajo consumo de energía.

En la literatura existente hay diferentes documentos de sistemas de comunicación con dispositivos que integran el protocolo ZigBee, así como diferentes aplicaciones inalámbricas en varias áreas.

- J. Rodriguez. “*Diseño de un sistema inalámbrico para el monitoreo en tiempo real de temperatura y humedad relativa bajo invernaderos*”, Universidad de la Salle, Bogotá,2006.

Este documento es de gran ayuda para esta memoria de título debido a que en este artículo se obtiene información sobre cómo se podría diseñar, implementar y desarrollar una red inalámbrica para la transmisión y monitorización de datos. Debido al análisis de este documento se logra identificar la viabilidad técnica para incorporar tecnología inalámbrica al robot limpiador de paneles fotovoltaicos “Clean Optimizer” frente a las soluciones tradicionales de datos por cableado. También este documento ayuda en el desarrollo de esta memoria de título debido a que se realiza la comparación entre las posibles tecnologías existentes donde se describen tecnologías de comunicación inalámbrica cómo por ejemplo Wifi, Bluetooth y otros tipos de tecnología inalámbrica. Luego comparar estas tecnologías, es necesario saber los estándares o protocolos, en este caso se deduce que el protocolo IEEE 802.15.4 es un protocolo adecuado ya que en este documento se hace comparación entre los diferentes estándares y este posee características que destacan sobre otras en base a consumo de potencia, confiabilidad en la transmisión de datos, implementación y bajos costos.

- M. Flores, “*Desarrollo de un sistema inalámbrico para monitoreo de humedad de suelo en diversos puntos de un viñedo*” (Memoria para optar al Título de Ingeniería en Mecatrónica). Universidad de Talca, 2014. 4

Un aspecto importante en todo proyecto es el consumo de energía que generan los componentes, lo que se busca es minimizar el consumo que poseen las tarjetas electrónicas, en donde se acopla los sensores, el microcontrolador y los dispositivos XBee, considerando, además, solamente los componentes estrictamente necesarios para lograr reducir al mínimo el consumo energético.

Respecto a la integración entre un microcontrolador y dispositivos de radio frecuencia, es posible encontrar interacción directa entre ambos dispositivos, con el objetivo de crear redes inalámbricas que transmitan información captada por sensores conectados a la placa microcontroladora Arduino, generando un control sobre la red. En la literatura se encontró un ejemplo de una red de sensores inalámbricos de bajo costo, además de los dispositivos ya mencionados cuentan con dispositivo XBee Shield, el cual permite la conexión de los dispositivos de radiofrecuencia con la placa microcontroladora, simplificando la conexión de la red inalámbrica en donde se muestra claramente que se puede ver la factibilidad de integrar elementos que son considerados en esta memoria. Finalmente, con la ayuda de este artículo se podrá realizar la comunicación entre una unidad remota terminal y el robot limpiador de paneles fotovoltaicos, por medio de dispositivos XBee, implementando un Red del protocolo IEEE 802.15.4, permitiendo un confiable monitoreo y supervisión de los elementos del robot. Finalmente, este artículo entrega las herramientas para realizar la transmisión y recepción de información a través de una red Modbus cableado y una red inalámbrica Zigbee.

C. Desarrollo de red inalámbrica con sistema de monitoreo y transmisión de datos

- M. Tarhan, F. De la Cruz, M. Diaz-Granados "*Diseño de Interfaz Gráfica para Implementación de tecnología Zigbee en una Red Modbus para Aplicaciones Inalámbricas en Automatización Industrial*", Universidad Nacional Experimental Politécnica 'Antonio José de Sucre', febrero 2015.

Este documento es considerado un gran aporte para el desarrollo de esta memoria de título, debido a que su contenido se convierte en una pieza clave para entender el procedimiento para realizar el intercambio de datos desde el prototipo limpiador de paneles fotovoltaicos hasta una unidad de monitoreo. Este artículo entrega las herramientas para el desarrollo de la comunicación, en donde se puede aplicar dos protocolos de comunicación, por una parte, tenemos el protocolo de comunicaciones Modbus RTU y por otro lado tenemos el protocolo de comunicación inalámbrica Zigbee. En el caso de protocolo Modbus se especifica el procedimiento que el maestro y el esclavo utilizan para el intercambio de datos, el formato de estos datos y cómo se tratan los errores. Por lo tanto, con la ayuda obtenida gracias a este artículo se opta por la aplicación del protocolo Modbus en modo RTU, esto quiere decir que cuando un elemento de "Clean Optimizer" envíe información hacia el sistema SCADA, el mensaje será enviado mediante dos dígitos hexadecimales de 4 bits, donde cada mensaje debe ser enviado en un flujo continuo de datos. Esto se refiere a que posee una estructura de entramada de un campo de dirección (1byte), campo de función (1byte), luego campo de función (N bytes según la función a realizar) y el fin de la trama (2 bytes) donde por el campo de control de error CRC (Comprobación de redundancia cíclica), que controla el contenido del mensaje completo.

D. Aplicación de sistemas SCADA

- J. Somoza N. Galan N. Mejias E. Orozco "*Diseño de Interfaz Gráfica para Implementación de Sistema SCADA en la Supervisión y Adquisición de datos en Módulos Fotovoltaicos*" Universidad Politécnica de Sinaloa, septiembre.2017

En esta investigación se hace énfasis en el diseño de una interfaz gráfica para implementar un sistema SCADA que cumpla la función de supervisar y adquirir datos de Módulos fotovoltaicos. en donde se extraen ideas respecto la aplicación de este tipo de tecnología para paneles fotovoltaicos que se encuentran en exteriores bajo condiciones poco amigables, lejanas a las encontradas en un laboratorio. Por esta razón es necesario iniciar el proceso de selección de materiales basándose en información encontrada en esta literatura, dando prioridad a la robustez y confiabilidad, por lo que el hardware seleccionado debe cumplir con características adecuadas además de ser compatibles con el software a utilizar.

Los sistemas de monitoreo iniciaron con tecnologías que contaban con un mayor número de componentes y cableado para la realización de lecturas (Voltaje, Corriente, Etc.), que con el paso de los años han ido evolucionando hasta llegar al uso de procesadores. Estos sistemas se caracterizan porque utilizan interfaces amigables destinadas a los usuarios donde se presentan de forma instantánea las lecturas de datos de manera organizada, utilizando también elementos visuales y alarmas para facilitar la interpretación de la información recabada, generándose así los sistemas SCADA, los cuales controlan, supervisan y proporcionan toda la información de un sistema de monitoreo a un usuario final.

- J. Herrera, M. Barrios, S. Pérez, “*Desing and implementation of a wireless SCADA system by means of zigbee and arduino technology*”, IEEE, Vol 12, N° 2, pp.65-72, Dec .2014.

Este artículo es un gran aporte al desarrollo de esta memoria de título, debido a que acá se encuentra la base para implementar la interconexión del sistema de supervisión, control y adquisición de datos con microcontrolador AVR y tecnología XBee. Este documento científico deja cómo evidencia que se puede crear un sistema SCADA libre y de bajo costo. Por otro lado, gracias a esta investigación se pudo concretar el proceso de configuración de hardware de Arduino en reemplazo de una tarjeta de adquisición de datos. Por lo tanto, se deja en evidencia la compatibilidad entre las tecnologías nombradas anteriormente en donde se puede monitorear mediciones remotas con sensores para la toma de decisiones. Esta literatura es clave para entender cómo puede funcionar un dispositivo versátil el cual puede funcionar con varios sensores de forma remota entregando una señal limpia y útil para procesamientos posteriores.

En base a lo investigado en la literatura, se aplica la factibilidad del uso de diversos sensores que pueden ser aplicados para la limpieza de paneles fotovoltaicos mediante la integración de un AVR de tipo Arduino con el protocolo ZigBee.

En varios documentos se tiene evidencia de la creación y validación de tecnología en donde se puede realizar un sistema SCADA con comunicación inalámbrica, demostrando que estos sistemas poseen alta fiabilidad, teniendo cómo ventajas la comunicación sin una visión directa, bajos costos, mejoras en problemas de conexión y requieren poco mantenimiento.

Finalmente, la información obtenida de este artículo generara parte de metodología de este proyecto, para lo cual se tiene cómo esquema básico tres partes, por una parte, se tiene un sistema de para los sensores y actuadores, por otro lado, se tiene un sistema de interconexión, este sistema abarca lo que serán las comunicaciones entre el robot limpiador de paneles fotovoltaicos y una estación de monitoreo. Por último, se tiene un sistema que cuenta en software de supervisión, control y adquisición de datos. Dicho sistema puede ser desarrollado gracias a la lectura remota de los sensores a través del dispositivo remoto XBee, en donde la señal es enviada inalámbricamente hacia otro dispositivo XBee el cual está conectado mediante interfaz USB serial a un computador que contiene un sistema HMI.

- I. Ferreira "*Sistemas de Controlo e Supervisão de Sistemas Embebidos-Tipo SCADA*" Universidade do porto, Faculdade de Engenharia Porto, marzo 2008.

El artículo aporta en el desarrollo de un sistema de control, supervisión y adquisición de datos , el cual se basó en la aplicación de una arquitectura descentralizada, en este caso el algoritmo de control será ejecutado por el usuario en computador personal y la adquisición de los datos será mediante un red inalámbrica de topología punto a punto con la capacidad para transferir información de manera frecuente, soportar tráfico aperiódico(eventos), cumpliendo con los requisitos de tiempo requerido, inmunidad al ruido electromagnético y protección mecánica ante condiciones ambientales muy hostiles , alto nivel de confiabilidad, disponibilidad y seguridad, bajo costo global.

El gran aporte que entrega este artículo es optar por decidir cómo se va a implementar el sistema HMI. Donde se requería un sistema acorde a los requerimientos y compatibilidad de la red inalámbrica la cual se está desarrollando. Dentro de las alternativas se tenía crear una aplicación usando PHP y HTML, usando lenguajes de programación cómo C, C++, Visual Basic, Java, entre otros. Inicialmente fue analizar la opción de usar PHP lo cual gracias a este artículo se encontró que esta opción posee la desventaja de que se requiere configurar el puerto serie cada vez que se carga la aplicación y requiere siempre de internet. Debido al análisis y comparación de las diferentes aplicaciones para crear el sistema HMI se toma la decisión de utilizar Visual Basic ya que se utilizaría para interacción con el puerto serie de la API de Windows, pero también posee la ventaja de que se podría desarrollar fácilmente en diferentes plataformas. Finalmente, gracias a este artículo se obtiene la información y las herramientas para aplicarlas ante alguna posible implementación un sistema de supervisión, control y adquisición de datos en el que se necesite monitorear parámetros y en donde se tiene el control total sobre el sistema a través de la manipulación de botones y por otro manipular las entradas y salidas de los elementos del sistema.

1.2.2.-Discusión Bibliográfica

Los artículos recabados abren el debate para analizar las tecnologías de redes para robots, debido a esto se generó gran interés por saber cómo se estructuraban para luego analizar los desafíos en cuanto a arquitectura, protocolos y aplicaciones. Es por esto que se debe tener en cuenta que la concatenación de distintas tecnologías para la monitorización de procesos en el que se encuentren involucrados robots ayuda a obtener las herramientas para implementar sistemas en los cuales se pueda controlar y visualizar parte del proceso en el que se encuentra involucrado el robot mediante en un entorno gráfico. Debido a esto y a un análisis en las comunicaciones inalámbricas para robots es que la de radiofrecuencia es la que puede tener mayor aplicación debido a ventajas sobre las demás [1].

Hoy en día se busca evaluar y analizar las alternativas de comunicación inalámbrica, en donde es necesario saber sus características y requerimientos como los tipos, clasificación y su disponibilidad para trabajar con diferentes tecnologías. Dentro de la literatura al evaluar y analizar parte de estas alternativas, se acepta que las redes se enfocan mayoritariamente en tres grandes grupos que son las de área personal(WPAN), las redes de área local(WLAN) y las redes inalámbricas de área global(WWAN), para lo cual se puede inferir el uso de tecnología XBee se relaciona con el estándar IEEE 802.15.4 pero que no es lo mismo[2], en el cual se permite habilitar redes inalámbricas con capacidades de control y monitorización que sean seguras. que las alternativas de sistemas de comunicaciones para robots que sean cada vez más integra y versátiles con el objetivo de reducir costos y recursos materiales.

En los documentos estudiados hay artículos que verifican la aplicación de sistemas de comunicación inalámbrica utilizando tecnología XBee, como es el caso del proyecto realizado por CITRA de la universidad de Talca, el consta de del monitoreo de humedad de suelo con una red tipo malla [4]. En lo que se puede rescatar que se utiliza protocolo ZIGBEE y se integran sensores a distancia, en donde se logra dilucidar que es la potencia de transmisión es muy importante abarcar la distancia requerida entre nodos.

Hay artículos que entregan información más detallada respecto de la integración entre un microcontrolador y dispositivos de radio frecuencia, en donde es posible encontrar interacción directa entre ambos dispositivos, con el objetivo de crear redes inalámbricas que transmitan información captada por sensores conectados a un microcontrolador AVR, generando un control sobre la red [4].

En base a lo investigado se llega al acuerdo que lo analizado anteriormente se podría aplicar para un robot móvil como es el caso del prototipo limpiador de paneles fotovoltaicos “Clean Optimizer”, por lo que queda claro que se puede realizar el intercambio de datos, en donde cabe destacar que inicialmente es identificar los nodos, cuál será la unidad maestra y cuál será la esclava, después entender la manera y la capacidad de lo que se desea transmitir, segundo analizar si se va utilizar uno o más de un protocolo de comunicación que se va a utilizar.

También, existen trabajos en los cuales se analiza los posibles protocolos de comunicación, de los cuales en la mayoría de los casos el protocolo Modbus es el más aplicado debido a sus características y versatilidad para operar con la mayoría de las tecnologías posibles [5]. Dicho protocolo que se basa en las capas 1,2 y 7 del modelo OSI, posee gran versatilidad para trabajar en interfaces, en este caso, el enfoque será sobre Modbus rtu, para analizar cómo implementar un método envió secuencial de datos binarios mediante su estructura de trama, el contenido de cada encabezado, los mapeos de memoria y su método de verificación de errores.

Finalmente, después de integrar distintas tecnologías, se requiere llevar a cabo un sistema en lo que la información recopilada, sea centralizada en una computadora central, donde se pueda administrar y ser presentada a operadores. Esto permite una visualización y monitoreo del proceso de limpieza de paneles fotovoltaicos. Es en la literatura donde se presenta la estructura de sistemas HMI(SCADA), los posibles softwares que se puede pueden utilizar y como se pueden aplicar [6] con el objetivo de supervisar y controlar un proceso.

1.3.- Hipótesis de trabajo

- Una topología de red punto a punto es capaz de comunicar los elementos que se utilizan en la limpieza de paneles fotovoltaicos con un sistema de monitoreo de manera inalámbrica en tiempo real.
- Es posible probar un sistema SCADA solo tomando los sensores y actuadores utilizados para una limpieza de paneles fotovoltaicos por medio de simulación computacional y a través de la implementación experimental de un prototipo de laboratorio.
- Los sistemas SCADA enfocados en energías renovables buscan procesar señales y crear una base de datos con información que permita interpretar de manera correcta lo que se está monitoreando, además de detectar fallas técnicas en el sistema, teniendo así la seguridad de tomar decisiones correctas.
- Un sistema SCADA aplicado al monitoreo de paneles fotovoltaicos puede ayudar a realizar estudios sobre la eficiencia y desempeño de elementos utilizados para la limpieza y mantención de paneles fotovoltaicos en condiciones donde no es fácil tener acceso a la información en tiempo real, además de la creación de bases de datos para posterior análisis por parte de un usuario final.

1.4.- Objetivos

1.4.1.- Objetivo General

Desarrollar un sistema SCADA y comunicación inalámbrica, el cual se aplicará sobre el prototipo Clean Optimizer con el objetivo de acceder a los parámetros que inciden en el desempeño del prototipo en la limpieza de paneles fotovoltaicos.

1.4.2.- Objetivos Específicos

Los objetivos específicos llevan al desarrollo del objetivo general son los que mencionan a continuación:

- Establecer Topología de red punto a punto para la comunicación inalámbrica en base a los requerimientos del Proyecto.
- Diseñar, configurar y programar red inalámbrica para establecer comunicación entre “Clean Optimizer” y estación de monitoreo.
- Diseñar, Configurar y programar los dispositivos de radio frecuencia.
- Diseñar y establecer comunicación física en base los requerimientos.
- Diseñar, configurar y programar un sistema SCADA para monitorear y manipular parámetros de Clean Optimizer.
- Establecer interconexión entre sistema de comunicación inalámbrica y sistema SCADA.

1.5.- Alcances

Los alcances de esta memoria de título se resumen en los siguientes ítems:

- Se contempla la aplicación de comunicación inalámbrica para realizar pruebas experimentales sobre un panel fotovoltaicos de 1650mm x 990mm, ubicado en la facultad de ingeniería de la Universidad de Talca.
- Se considera una topología de red punto a punto para comunicación inalámbrica para el sistema de comunicación.
- Se establece solo el uso de tecnología XBee para el desarrollo de esta memoria de título.
- Se establece el uso de protocolo Modbus para esta memoria de título.
- Los parámetros por considerar en el sistema SCADA son: Movimientos, Nivel de Líquido y Mando de Bomba.

1.6.- Metodología

A continuación, se presenta la metodología para diseñar, construir e implementar un sistema de comunicación inalámbrica y un sistema SCADA:

- Desarrollo de un análisis y estudio teórico de las Problemática de comunicación existentes.
- Estudio y requerimiento del robot limpiador de paneles solares.
- Análisis del fenómeno de ensuciamiento de paneles fotovoltaicos en Chile y en el resto del mundo.
- Sistemas de limpieza de paneles fotovoltaicos en Chile y en el resto del mundo.
- Estudio y aplicación de sistemas de comunicaciones.
- Análisis y estudio teórico de las Problemática de red.
- Configuración y programación de módulos XBee.
- Diseño y construcción de tarjetas electrónicas para sistema de comunicación inalámbrico.
- Estudio, diseño e implementación de sistemas SCADA.

Capítulo 2. Diseño del sistema de comunicación inalámbrica.

En este capítulo se muestran los aspectos generales que avalan el desarrollo de esta solución en donde se dan a conocer las definiciones, clasificaciones y tecnologías actualmente existentes. por otro lado, se da a conocer la topología de red, configuración y elementos elegidos para el desarrollo del sistema de comunicación inalámbrica.

2.1.- Clasificación de las comunicaciones inalámbricas

Las comunicaciones inalámbricas se diferencian de las alámbricas en aspectos de diseño, costos, coberturas en puntos de difícil acceso, flexibilidad, ubicuidad, como también el modo en que se logra el intercambio de información. Este intercambio de información puede ser entre dos o más dispositivos sin el uso de un medio físico como un cable, sino que, por medio del espectro electromagnético, utilizando el aire para la transmisión de dicha información. Existen tres grandes grupos en las que se pueden clasificar como redes de cierta área de alcance tales como:

- Redes Inalámbricas de área extensa (WWAN)
- Redes Inalámbricas de área local (WLAN)
- Redes Inalámbricas de área Personal (WPAN)
 - a) Redes Inalámbricas de área extensa fijas.
 - b) Redes Inalámbricas de área extensa móviles.

Las redes inalámbricas permiten su desarrollo en distintas áreas y aplicaciones. Debido a esto es que para esta memoria de título se va a dar conocer las redes inalámbricas de área personal, esto hace referencia a las WPAN y se explican a continuación:

Bluetooth: sistema regulado por el grupo IEEE 802.15.1, trabaja bajo un enlace de radiofrecuencia en la banda de 2,4 GHz. Su alcance no va más allá de los 10 metros para la clase 2 y un alcance de hasta 100 metros para la clase 1. El uso de Bluetooth se masificó debido a la integración en teléfonos celulares.

DECT: opera bajo las bandas de frecuencias que van desde 1.880 a 1.900 MHz. Creado para reemplazar la telefonía analógica por la digital, permitiendo una mayor movilidad, menos interferencia y más seguridad.

IrDa: utiliza el espectro de frecuencia infrarrojo para la transmisión de datos, emplea un diodo led y es necesario que ambos dispositivos estén completamente alineados para la transmisión de información. Su aplicación ya es casi nula, principalmente por su corto alcance y la baja tasa de transmisión.

NFC: Trabaja bajo la banda de radiofrecuencia de 13,56 MHz. Su rango de alcance es muy bajo, hasta los 20 centímetros. Actualmente, el uso de esta tecnología se está incluyendo en los nuevos smartphones, para transmitir información por medio de contacto entre dos smartphones.

ZigBee: regulado por el grupo IEEE 802.15.4 creada en el año 2004 por ZigBee Alliance, permite crear redes inalámbricas con capacidad de control, monitoreo, bajo consumo energético y bajo costo. Especialmente, diseñado para establecer redes de sensores.

En base a las redes inalámbricas explicadas anteriormente, la de protocolo ZigBee, es la elegida para aplicar en la comunicación inalámbrica debido a su características y tiempo que lleva liderando en dispositivos de bajo consumo energético y largo alcance. Como requerimientos se podrían definir como alcance mínimo que el robot recorra un panel fotovoltaico de medidas 1.65 x 1 metros, por lo tanto, la distancia mínima requerida es de 1 metro, por lo general las filas de paneles fotovoltaicos se componen de 20 paneles.

2.2.- ZigBee

El protocolo ZigBee está basado en el estándar global abierto IEEE 802.15.4, creada por ZigBee Alliance, organización compuesta por más de 200 empresas entre las que destacan Motorola, Philips, Samsung, Siemens, ente otras prestigiosas empresas en la fabricación de semiconductores y equipos electrónicos. ZigBee opera bajo las bandas de 2,4 GHz, 900MHz y 868 MHz.



Fig. 2.2: Logo tecnología ZigBee Alliance

Las antenas o dispositivos inalámbricos encargados de realizar la comunicación reciben el nombre de XBee y operan bajo el protocolo de ZigBee. Las empresas encargadas de su fabricación son: Digi y MaxStream. Existen diversos modelos de XBee que varían según las características de su hardware y se identifican según el tipo de series correspondiente.

2.2.1.-Dispositivos de tecnología ZigBee

Toda serie cuenta con dos tipos de Módulos: los básicos y Pro; los Módulos XBee básicos permiten un alcance en interiores de hasta 30 metros y en exteriores un alcance máximo de 100 metros; mientras que, los Módulos XBee Pro en interiores el alcance es de hasta 100 metros y en exteriores hasta los 1.500 metros. Actualmente, existen tres series diferentes entre sí, las cuales se definen en orden cronológico de fabricación:

- XBee Serie 1: es el primer dispositivo creado, también llamados XBee 802.15.4. Son los dispositivos más básicos, no necesitan configuración para crear una red punto a punto.
- XBee Znet 2.5: corresponden a la Serie 2, actualmente retirados del mercado, siendo reemplazados por la serie ZB. Necesitan ser configurados para trabajar en modo AT y API.

- XBee ZB: corresponden a la actual Serie 2. Para utilizar dicha serie es necesario actualizar el firmware de los dispositivos Znet.
- XBee 2B: son los modelos más modernos de la serie 2, ya que poseen considerables mejoras con relación al hardware de la serie 2.
- XBee 900 MHz: tal cómo lo menciona su nombre opera bajo la frecuencia de 900 MHz. Este módulo posee la ventaja de tener mayor alcance que los demás y de seguir siendo un módulo de bajo consumo. Funciona con dos tipos de firmware: el firmware DigiMesh y el firmware Point-to-Multipoint.
- XBee XSC S3B: son similares a los Módulos de 900 MHz, pero están configurados de fábrica para trabajar para grandes distancias de transmisión y posee una tasa de transferencia de datos de 10 Kbps. No necesitan configuración, sólo trabaja mediante comandos. Corresponden a la actual Serie 3. Las principales ventajas que posee esta tecnología son:
 - Bajo costo de los dispositivos.
 - Bajo consumo de energía.
 - Soporta múltiples Problemática de red.
 - Encriptación AES, bloquea la red y evita que otros nodos se conecten.
 - Programación, control y configuración sencilla.

Todos los modelos de XBee cuentan con un tipo de antena para realizar la respectiva conexión con los demás dispositivos que componen la red. Existen cuatro tipos de antenas diferentes. La de chipe como se nombre lo dice actúa como un chip, la antena cable es una antena integrada que sobresale, la antena U-FL es un pequeño conector usado en microondas, se utiliza con frecuencia en tarjetas de radio mini-PCI a una antena y la RP-SMA es la que, es hecha con las mismas pistas, pero también permite conectar antenas de altas ganancias a diferencia de las otras.



Fig. 2.3: Tipos de antenas dispositivos XBee

2.2.2.-Red ZigBee

Toda red ZigBee permite conectar hasta 65535 nodos, una red de esta tecnología podría alcanzar una velocidad de transferencia de hasta 256 Kbps, dependiendo de su potencia de transmisión. Un aspecto importante que, al momento de configurar una red inalámbrica, es conocer cuáles son los tres tipos de nodos existentes en una arquitectura de una red ZigBee:

- **Coordinador:** es el nodo que crea y controla toda la red inalámbrica, debe existir solo uno por red. Su principal función es formar la red, establecer el canal de comunicaciones y el PAN ID para toda la red. Tiene la capacidad de almacenar información, cómo, por ejemplo, la clave de seguridad.
- **Router:** su principal función es conectar, enlazar y elegir la mejor ruta para transmitir la información entre los nodos hasta llegar al coordinador.
- **Dispositivo Final:** solamente transmite información a un router o un coordinador, no tiene la capacidad de enrutar paquetes de datos. Generalmente, se utiliza para conectar los sensores. Además, existe otra forma de clasificación para las XBee según su funcionalidad en la red, para esta forma solo se conocen dos tipos:
- **Dispositivo de funcionalidad completa (FFD):** son los dispositivos que trabajan en cualquier topología de conexión. Son considerados FFD los coordinadores y los routers. También son llamados nodos activos, ya que es necesario que siempre al menos uno de ellos se encuentre en funcionamiento para mantener la red.
- **Dispositivo de funcionalidad reducida (RFD):** son los dispositivos que construyen las redes con topología estrella. Solamente son considerados RFD los dispositivos finales, ya que sólo se comunican con el coordinador de red. Estos son llamados nodos pasivos.

2.2.3.-Topologías para redes de tecnología XBee

El protocolo ZigBee permite el desarrollo de configuraciones o topologías de redes punto a punto, multipunto, peer-to-peer (todos los nodos conectados entre sí) o redes complejas de sensores que interconectan dos o más redes en malla. Las topologías más básicas y utilizadas son las siguientes:

- Topología estrella (Star): todos los nodos de la red pueden conectarse sólo con el coordinador, no existe enrutamiento de datos entre los nodos, por lo tanto, existe una baja latencia.
- Topología árbol (Tree): el coordinador establece la red inicial, mientras que los denominados routers se encargan de formar las ramas con los demás routers o dispositivos finales y transmitir los mensajes al coordinador. El mayor problema es que si falla un router cae gran parte de la red.
- Topología malla (Mesh): existe más de una ruta entre los nodos y el coordinador, en caso de que una ruta falle. Es la red más compleja de formar y posee alta latencia de los datos al pasar por varios nodos.

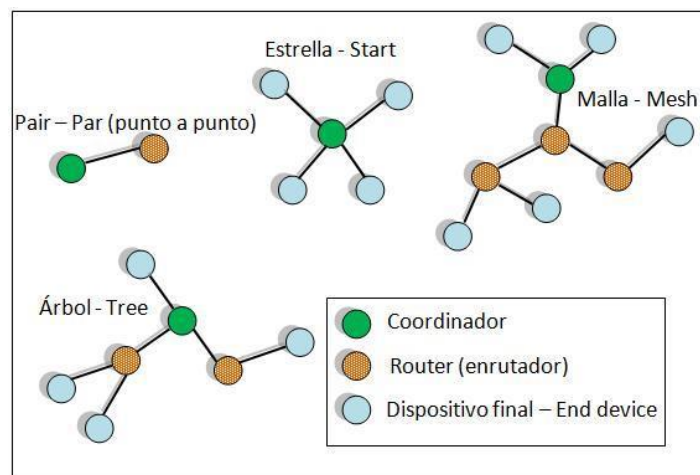


Fig. 2.4: Problemática de Red

2.3.- Selección dispositivos de radio frecuencia

Para la selección de los dispositivos, los dispositivos deben cumplir con los requerimientos en donde se necesita el envío de información sin el uso de cableado y cubriendo una distancia estimada entre el prototipo “Clean Optimizer” y un sistema HMI. Debido a esto se consideraron módulos XBee para el envío de información mediante ondas electromagnéticas, los cuales pueden ser aplicados en redes de sensores en donde se quiera comunicar un robot y una estación de monitoreo en el proceso de limpieza de los paneles fotovoltaicos.

Las ventajas de los módulos XBee sobre otros módulos, es que estos módulos poseen una mayor tasa de datos, permite conectar unos módulos con otros para realizar redes escalables, no poseen controlador central y las cargas se distribuyen de manera uniforme en toda la red y poseen más autonomía que otros módulos existentes. Por ejemplo, si lo comparamos con tecnología Lora, se investigó que esta tecnología funciona solo para las bandas de frecuencia utilizadas en Europa, se basa en tipos de modulación para redes 3G, por otro lado, sus dispositivos están fabricados para redes IoT, también solo puede trabajar con su propio Gateway, lo que no hace sea versátil, posee menos prestaciones en base a transferencia de datos y mayor tiempo de latencia que XBee.

TABLA 1 Comparación entre tecnología XBee versus Lora

XBee	Lora
Tasa de Transferencia Alta.	Tasa de Transferencia Baja
Tamaño de Red Grande.	Tamaño de Red Limitada.
Recomendada para aplicaciones en Tiempo Real.	Trabaja solo para redes 3G Mayor tiempo de latencia Fluctuación de fase acotada

2.3.1.-Módulo XBee pro 900HP-S3B

El módulo XBee Pro 900HP-S3B, pertenece a la serie 3 que corresponde a la última serie de módulos XBee, poseen la ventaja de permitir configurar el firmware en función de la conectividad que se necesite ya sea BLE, Mesh, ZigBee, 802.15.4 se caracteriza por ser un Módulo pequeño y contiene una antena externa de tipo RP-SMA Duck de 900 MHz, la cual no posee problemas para ubicarla al interior o intemperie.



Fig. 2.5: XBee Pro 900HP RP-SMA

Los módulos XBee Pro 900HP-S3B poseen características que los destacan sobre otros Módulos las cuales se muestran en la siguiente tabla:

TABLA 2 Características Módulo XBee pro 900HP-S3B

Características	Especificación
Alimentación	3.3 V
Rango de Alcance	14 KM
Potencia de salida	250 mW
Tasa de Transferencia	20 Kbps
Conector	RP-SMA
Interfaz de datos serial	UART
Métodos de configuración	AT
Banda de Frecuencia	900 Mhz
Digital I/O	14
Temperatura de operación	-40+85°C

Obtenido de <https://www.digi.com/products/embedded-systems/rf-modules/sub-1-ghz-modules/XBee-pro-900hp>

2.4.- Selección dispositivos de radio frecuencia

La configuración de la red inalámbrica mediante uso de módulos XBee de radio frecuencia se basa en una topología punto a punto. La red está configurada de manera que se genere una comunicación bidireccional con el objetivo de enviar y recibir información cuando se está realizando la limpieza con “Clean Optimizer”



Fig. 2.6: Red punto a punto mediante dispositivos XBee

2.4.1.-Funcionamiento red inalámbrica

El funcionamiento correcto de la red inalámbrica depende de establecer un adecuado modo de comunicación entre los dispositivos, configurando los puertos y direcciones de los dispositivos cómo también realizar un correcto acoplamiento entre los módulos de radio frecuencia, placa microcontroladora, sensores y actuadores.

Para la interconexión entre los Módulos XBee con el microcontrolador se utilizará la interfaz de datos serial UART. Por otro lado, la programación de los Módulos y la configuración de sus parámetros requiere aprender a utilizar dispositivos y software relacionados con ZigBee, los cuales se nombran a continuación:

- Computador
- Software XCTU
- XBee Explorer USB
- Driver XBee Explorer USB
- Cable USB

El dispositivo XBee Explorer es el encargado de realizar la configuración entre el módulo XBee y el computador, se conecta mediante un cable tipo USB y para el caso de reconocimiento del computador se necesita instalar el controlador para XBee Explorer. Este componente se encarga de la regulación de 5V a 3.3V, acondicionamiento de señales e indicadores de alimentación, intensidad de señal recibida y los pines de transmisión y recepción de los Módulos XBee.



Fig. 2.7: Dispositivo XBee Explorer USB

El software para configurar los parámetros de los Módulos XBee es el software XCTU y puede operar en todos los sistemas operativos, en este caso se trabajará con Windows.

2.5.- Software XCTU

El primer paso trata de la descarga del software XCTU (Configuración XBee y utilidad de prueba) y el controlador, luego de instalar el software XCTU y controlador, se debe conectar el dispositivo XBee Explorer con el Módulo XBee PRO 900HP-Series 3B y luego mediante cable USB conectamos al computador.

2.5.1.-Configuración Software XCTU

Al iniciar el software aparece un menú en el cual el usuario puede agregar y reconocer los módulos de radio frecuencia. Cómo también configurar el software y elegir en qué modo trabajar.



Fig.2.8: Menú Reconocimiento de Módulo

2.5.2.-Configuración dispositivos de radio frecuencia

En la opción de agregar dispositivo buscamos el Módulo conectado al computador a través de un puerto COM. Luego al detectar módulo se deben configurar los siguientes parámetros de la velocidad de los datos en baudios (Baud Rate), el número de bits (Data Bits), la paridad de los datos (Parity), los bits de parada (Stop Bits) y el control de flujo de los datos (Flow Control)

- Baud Rate: 9600.
- Data Bits: 8.
- Parity: NONE.
- Stop Bits: 1
- Flow Control: NONE.

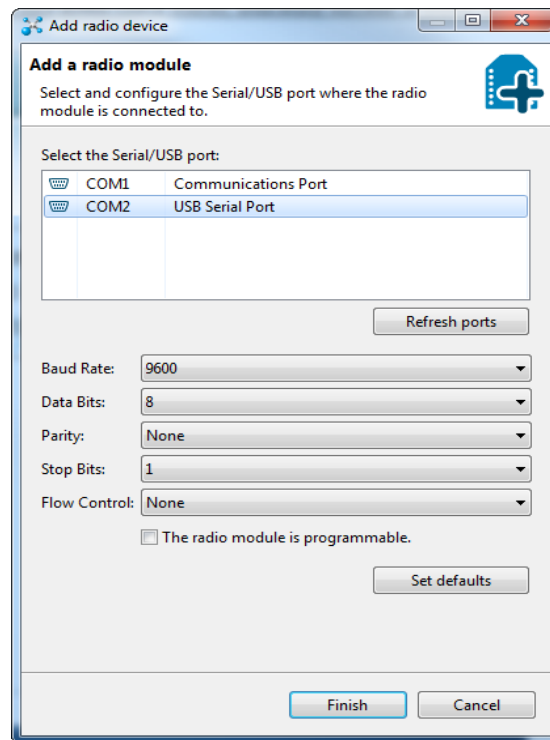


Fig.2.9: Configuración de parámetros

Es importante saber los parámetros que deben ser modificados en el software antes de configurar por completo los Módulos XBee Pro 900HP- S3B, los cuales son:

- PAN ID: Encargado de identificar para la red de área personal para los Módulos XBee se comuniquen entre sí a través de un mismo ID de red.
- ZS: Ajusta la potencia de salida de los Módulos XBee a programar, configurando el perfil ZigBee, existiendo tres posibles rangos:
 - 0: red específica, básica.
 - 1: ZigBee-2006.
 - 2: ZigBee-PRO.

DH: Se refiere a la dirección de destino en alto. Esto hace referencia a los últimos 32 bits más significativos de los 64 bits de la dirección de destino.

DL: Se refiere a la dirección de destino en bajo. Esto hace referencia a los últimos 32 bits menos significativos de los 64 bits de la dirección de destino.

NI: corresponde a un identificador o nombre asignado a cada XBee, designado de acuerdo a su función.

2.5.3.-Modo Transparente

El modo transparente es el modo en que se van a comunicar los dispositivos XBee Pro 900HP-S3B seleccionados. Este modo tiene la ventaja que actúa como un reemplazo de una línea de cable cuando uno de los Módulos XBee recibe información. Por otro lado, cuando el nodo ZigBee Coordinador envía información hacia el nodo ZigBee Router, lo hace sobre el mismo canal y el nodo que envía el mensaje necesita la dirección del Módulo destinatario. Por ende, los Módulos pueden almacenar la dirección de 64 bits de este Módulo destinatario. Puede ser una dirección de destino alta (DH) y dirección de destino baja (DL).

La ventaja de trabajar en este modo es que es ideal para redes punto a punto, debido a que un Módulo recibe datos del otro módulo XBee debido a los Módulos se configuran para trabajar en el mismo canal (CH) y valor de identificador (ID) para que no existan interferencias. Además, si los módulos poseen el mismo PAN ID, nadie podrá intervenir en la conexión punto a punto, por otro lado existen direccionamientos únicos en el mundo para estos módulos, por lo que no se repiten las direcciones DH y DL, así se tiene que mensajes serán destinados de manera segura hacia el otro módulo y este se le suma que poseen un método que entrega seguridad en el envío de datos debido a que el módulo del receptor envía un paquete llamado ACK, al módulo de origen lo cual indica que el mensaje se recibió correctamente.

Capítulo 3. Microcontrolador

En esta memoria de título el microcontrolador es el encargado de procesar la información obtenida de los sensores para luego mediante comunicación serial, transmitir información a uno de los módulos XBee Pro 900HP-S3B, luego este módulo de radio frecuencia se comunica de manera inalámbrica con otro módulo otro módulo de radio frecuencia.

3.1.- Arduino

Consiste en una plataforma de hardware y software de código abierto. Fundado por Massimo Banzi con el principal objetivo de desarrollar proyectos electrónicos a bajo costo. Su origen nace en el año 2006 donde fue creado el primero de ellos, al paso del tiempo y ante nuevos avances en la electrónica se han creado diferentes modelos mejorando las características y capacidad de cada uno. Principalmente, Arduino es una placa electrónica la cual posee un microcontrolador de tipo Atmel AVR y un entorno de programación para configurar puertos de entrada y salida. Su lenguaje de programación es denominado Arduino Programming Language (basado en wiring) y el entorno de programación se llama Arduino Development Environment (basado en processing). Por otro lado, Arduino tiene la capacidad de trabajar con todos los sistemas operativos, posee compatibilidad con diferentes shields y sensores para agregar funciones extras.



Fig.3.1: Logo Tecnología Arduino

3.2.- Comunicación Serial

La comunicación serial se logra a través del puerto serial, también conocido como UART, que significa Transmisor y Receptor asíncrono universal, esto quiere decir que se encarga de manejar las interrupciones de los dispositivos conectados al puerto serie y convertir los datos transmitidos al bus de sistema para luego ser transmitidos a través de los puertos. Por otro lado, se encarga de controlar los puertos y dispositivos serie. Para el caso de la transmisión de los datos, todas las placas Arduino tienen al menos un puerto serial, el cual consiste en los pines RX (Lectura) y TX (Escritura). Este puerto se utiliza principalmente para la comunicación con una computadora, y en ese caso no se debe utilizar para lectura o escritura de datos digitales.

3.3.- Placa Microcontroladora R3

Para este proyecto el AVR seleccionado corresponde de la placa microcontroladora R3 de Arduino, el cerebro principal ATmega328P y un ATmega16u2 el cual cumple la función de convertidor USB-Serie para la interconexión con el computador y programación.

Para los requerimientos de este proyecto, la aplicación correspondiente a esta placa microcontroladora abarcara lo siguiente:

- ❖ Entradas y Salidas análogas
- ❖ Entradas y Salidas digitales
- ❖ Puerto Serial para la comunicación con los Módulos XBee
- ❖ Plataforma para Software y Hardware
- ❖ Compatibilidad con la mayoría de las Shields

Para una mayor visualización de la placa microcontroladora Arduino R3, se presenta la estructura y partes de esta:

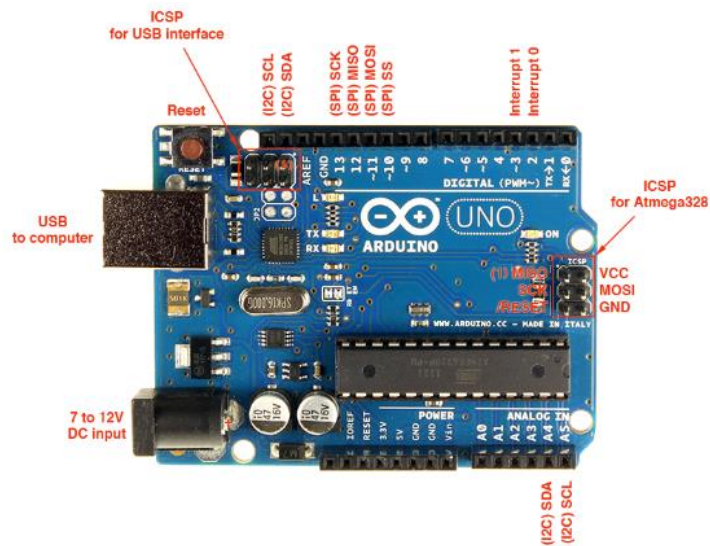


Fig.3.2: Placa Microcontrolador Arduino R3

Es importante saber las características técnicas de la placa microcontroladora Arduino R3, debido a que gracias a esta información específica podemos saber las prestaciones de esta placa aplicadas en este proyecto:

TABLA 3 Características placa microcontroladora Arduino R3

Características	Especificación
Microcontrolador	ATmega328-PU
Voltaje Operación	5V
Pines Digitales I/O	14 (6 salidas PWM)
Pines Entradas Análogas	6
Corriente DC Pines I/O	20 mA
Memoria Flash	32 KB
SRAM	2 KB
EEPROM	1 KB
Velocidad Reloj	16 MHz

3.3.1.- Microcontrolador ATmega328

El chip corresponde a un microcontrolador fabricado por la empresa Atmel, el cual es un microcontrolador de 8 bits basado en arquitectura RISC, por lo que su empaquetado o package que se utiliza en la placa microcontroladora R3 es compatible con nodo remoto de tipo DIP de 28 pines, cómo el que se muestra a continuación

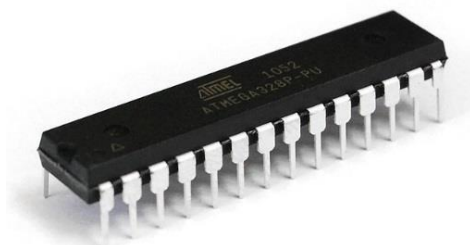


Fig.3.3: Microcontrolador ATmega 328-PU

El microcontrolador ATmega328-PU posee 28 pines los cuales poseen distintas funciones como por ejemplo las de alimentación, entradas digitales, salidas digitales, entradas análogas, pines UART (RX, TX), I2C, etc. A continuación, se presentan las características técnicas y físicas del microcontrolador ATmega 328-PU:

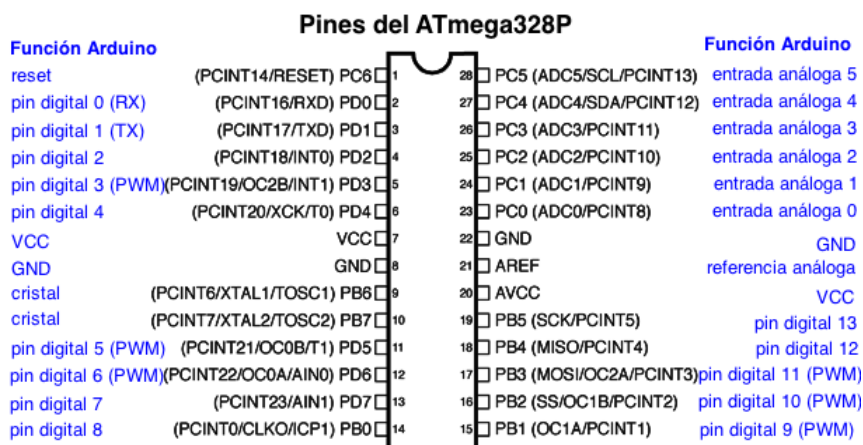


Fig.3.4: Pines Microcontrolador ATmega 328P

TABLA 4 Características microcontrolador ATmega328P

Características	Especificación
CPU	8-bit AVIR
Arquitectura	RISC
Voltaje de Operación	1,8V-5,5V
Pines Digitales I/O	26
Interrupciones Internas	16
Interfaz Integrada	I2C, SPI, UART
Memoria Flash	32 KB
SRAM	2 KB
Máxima Frecuencia Operación	20 MHz

En la placa microcontroladora Arduino Uno R3 se encuentran los pines que alimentan las entradas y salidas de los siguientes elementos utilizados por el prototipo de limpieza de paneles fotovoltaicos los cuales son:

- ❖ Sensores de luminosidad (TSL2560).
- ❖ Sensores Nivel de Líquido (YKC-Y25-PNP)
- ❖ Módulos XBee + Antena RPSMA (900HP-S3B)
- ❖ Motor DC (WG5539)
- ❖ Bomba Agua (SZY-4155)

Existe gran demanda de pines al uso de los sensores TSL2561, los sensores para nivel de líquido de tipo YKC-Y25-PNP, para los pines utilizados para el motor GW5395, La bomba de agua SZY-4155. Finalmente, la programación de los elementos nombrados se realiza en el software IDE de Arduino de tipo open source, programado en lenguaje orientado a C y C++.

3.4.- XBee Shield

El dispositivo XBee Shield realiza la interconexión entre el Módulo XBee con la placa microcontroladora Arduino Mega. Para este dispositivo se debe realizar la conexión de los pines de manera ordenada respecto de la placa microcontroladora Arduino R3 y se puede configurar de dos modos de trabajo: Modo USB y Modo XBee.



Fig.3.5: Dispositivo XBee Shield

El módulo XBee shield puede ser conectado y configurado de diferentes modos, dependiendo el objetivo a cumplir. Para la transmisión y recepción de información el módulo XBee shield se conectad a los pines DIN Y DOUT de los Módulos XBee Pro 900HP. Por otro lado, el Módulo Shield posee un switch SPDT, lo cual permite seleccionar dos posiciones de conexión:

Posición UART: Esta posición crea un modo de interconexión entre los pines D0 Y D1 de la placa microcontroladora Arduino Mega 2560 con los pines DIN Y DOUT del Módulo XBee, logrando comunicar los Módulos XBee.

Posición DLINE: Esta posición crea un modo de interconexión entre los pines digitales D2 y D3 de la placa microcontroladora Arduino R3, de esta manera podemos realizar la comunicación entre la placa microcontrolador Arduino R3 con el PC.

El módulo shield XBee alimenta a los dispositivos XBee mediante un regulador de voltaje que posee este módulo, mientras que se alimenta de los 5V que entrega la placa microcontroladora Arduino Uno R3.

3.5.- USB

Parte la capa física de la implementación de esta memoria consta de USB 2.0 debido a que las conexiones con este bus son de tipo plug & play, lo que genera una ventaja cuando se conecte el nodo central con el PC, debido a que será detectado automáticamente, sin que vuelva a configurar de nuevo. Otra ventaja es que no es necesario desconectar ni reiniciar el equipo para instalar o desinstalar el nodo central y además permite alimentar hardware debido a que el propio bus nos proporciona una alimentación de 5V. Este bus posee una extensión de hasta 5 metros y una tasa de transmisión de datos de 480Mbps(60MB/s). A continuación, se muestra una tabla donde se explica los pines de la interfaz física(eléctrica) del bus USB 2.0.

TABLA 5 Pines de conexión USB

PIN	NOMBRE	SEÑAL	COLOR
1	VCC	+5V	ROJO
2	D-	DATOS -	BLANCO
3	D+	DATOS +	VERDE
4	GND	MASA	NEGRO

Capítulo 4. Implementación sistema SCADA

La estructura este proyecto contempla tres unidades las cuales constan en primer lugar con la unidad que se encuentra en el robot encargado de la limpieza de paneles fotovoltaicos de nombre “Clean Optimizer”. Esta unidad posee cómo controlador principal una placa microcontroladora Arduino R3, en la cual se procesa la información de los sensores y actuadores del robot. Es en esta unidad donde se implementa un módulo remoto el cual se comunica con el controlador mediante protocolo UART para luego enviar información de manera inalámbrica a un módulo central.

La segunda unidad es llamado módulo central, el cual cumple el objetivo de ser una unidad para recibir y enviar información. Finalmente tenemos una unidad llamada unidad PC, la cual consta de un sistema SCADA, en la cual se desarrollará un sistema para supervisar y controlar la adquisición de la información de Clean Optimizer.

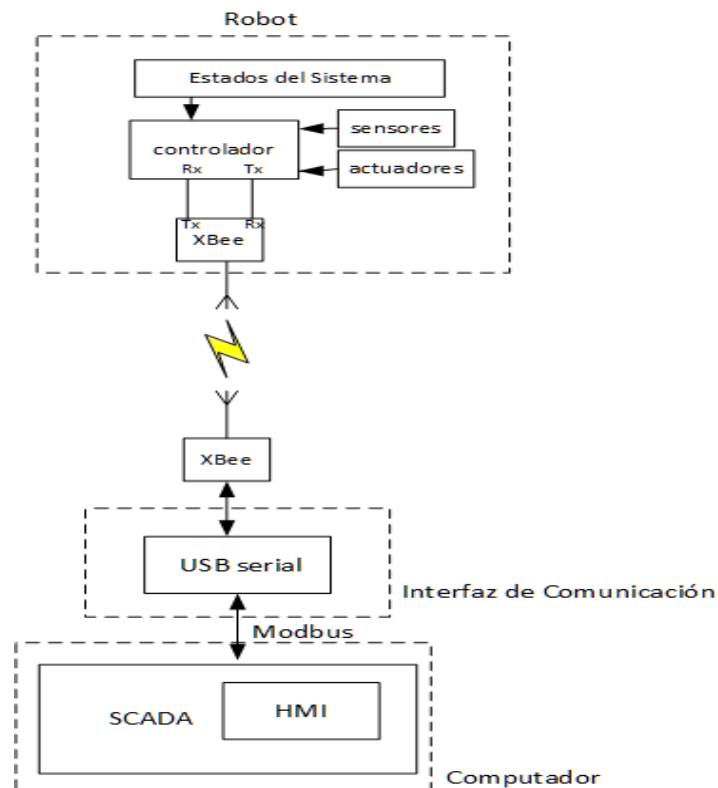


Fig.4.1: Estructura de Sistema Completo

En este proyecto el sistema SCADA se encarga de supervisar y tomar acciones cuando se requiera realizar el proceso de limpieza de paneles fotovoltaicos, esto es debido a que realiza un sistema de interfaz máquina-humano para visualizar y manipular parámetros que inciden en el desempeño del “Clean Optimizer”. La topología de red elegida para la comunicación inalámbrica es una topología de red punto a punto y el protocolo de comunicación para el sistema SCADA será mediante protocolo Modbus/RTU basado en arquitectura maestro/esclavo.

4.1.- Unidad Terminal Remota

Para este proyecto la unidad terminal remota cumple la necesidad de recoger los datos y enviarlos a la unidad central (Computador) para poder realizar acciones y visualizar los datos de un sistema HMI. La unidad remota puede ser desde un pequeño sistema embebido hasta un PLC. En este proyecto el sistema se basará en la placa microcontroladora Arduino Uno R3 cómo unidad RTU.

4.2.- Protocolo Modbus

Se implementará este protocolo de comunicación debido a que la información de los elementos utilizados en la limpieza de paneles fotovoltaicos se enviará y recibirá mediante comunicación serie y esta información será representada de forma binaria compacta para el protocolo de comunicación. El modo RTU tiene la capacidad de seguir a los comandados/datos con una suma de comprobación de redundancia cíclica cómo mecanismo de comprobación de errores para la fiabilidad de los datos. Los mensajes Modbus son entramados(separados) por periodos inactivos(silenciosos). El protocolo Modbus a utilizar cumple con los requerimientos de este proyecto y posee las siguientes ventajas:

- Modo Seguro y Fiable
- Maneja Bloques de datos sin restricciones
- No requiere licencias
- Soportado por todos los tipos de HMI y software SCADA

4.2.1.- Transmisión de datos

La manera de transmitir datos es tipo punto a punto, la transferencia de información enviados corresponden a datos binarios de 0 a 255, lo que hace que se envíe un bit a la vez y de manera secuencial, en modo semi-dúplex. Debido a esto los mensajes poseen una estructura de 8 bits, desde el bit 0 al 7, para lo cual el primer bit es el de inicio y los dos últimos son bits de parada. Los mensajes Modbus son entramados (separados) por períodos inactivos (silenciosos).

La aplicación de Modbus será para manejar los datos de manera similar a los PLC, de manera que la estructura de la aplicación de Modbus establece a que esclavo va dirigido la solicitud, el tipo de solicitud, el número de bytes del dato solicitado y el método para verificar la integridad de los datos. Se establece la forma de acceder al tipo de dato de cada elemento mediante prefijos en este caso se utilizara el prefijo 0 para salidas discretas, 1 para entradas discretas, 3 para entradas análogas y 4 para salidas análogas. Por otro lado, se tiene que el rango de direcciones va de 0 a 65.535. Debido a esto se utiliza el prefijo y el número de elemento en una misma dirección y el esclavo puede entenderlo de manera correcta.

TABLA 6 Acceso de esclavo en Modbus

BLOQUE DE MEMORIA	RANGO	ACCESO	TAMAÑO
SALIDAS DIGITALES	0-09999	SOLO LEER	1 BIT
ENTRADAS DIGITALES	10000-19999	LEER/ESCRIBIR	1 BIT
ENTRADAS ANÁLOGAS	30000-39999	SOLO LEER	16 BITS
SALIDAS ANÁLOGAS	40000-49999	LEER/ESCRIBIR	16 BITS

4.2.2.- Trama de datos

El maestro que en este caso corresponde al PC, solicita al esclavo que en este caso corresponde a la placa microcontroladora Arduino R3, acceder a los datos de los elementos del motor, bomba de agua o nivel de líquido. Los accesos a los datos de movimiento del robot son de tipo booleano con una capacidad de 1 bit, el acceso a los datos de activación de la bomba agua también es de tipo booleano con una capacidad de 1 bit, por otro lado, los accesos a los datos de nivel de líquido son de 1 bit. Debido a esto si el operario desea acceder a manipular los datos desde el sistema HMI, la dirección total de comunicación maestro esclavo sería el prefijo y el número de elemento al cual se quisiera acceder, un ejemplo de esto es que si se quisiera mover el robot a la izquierda y la variable de esa acción se guarda en el espacio 1, el direccionamiento quedaría de la siguiente manera “00001”, esto quiere decir que el primera número indica que se acceder a leer o escribir sobre una salida discreta , sobre el elemento número 1. Finalmente, el encabezado de la trama de Modbus basado en la arquitectura maestro/esclavo se muestra en la siguiente figura:

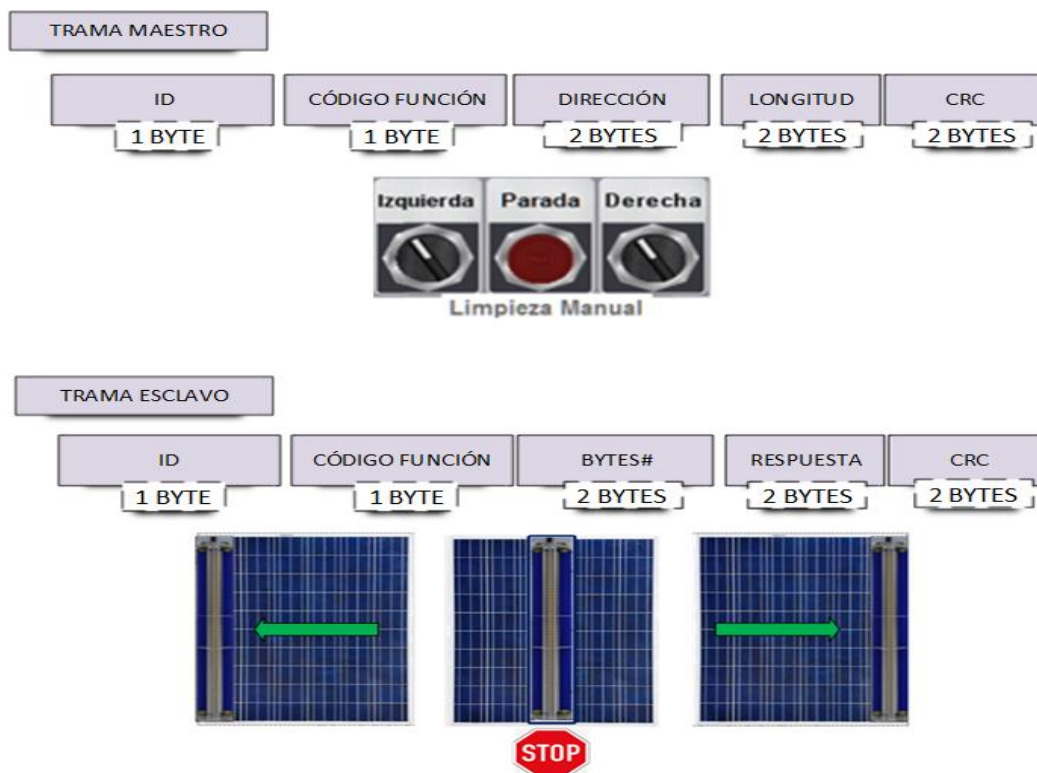


Fig.4.2: Trama maestro/esclavo de sistema de comunicación

4.3.- SISTEMA HMI

El software elegido es Microsoft Visual Studio, el cual es un software nativo, que utiliza clases publicas basadas en eventos, programados en lenguaje “Visual Basic”. El cual trae un componente que ayuda al diseño de una interfaz gráfica, en donde la funcionalidad de programación es en base a eventos, utilizando para este caso lenguaje de programación Visual. El complemento de visual ayudó a la integración de elementos como botones y gráficos. También para los datos se integró un Dattalogger virtual con el objetivo de guardar la fecha y hora de algún evento como hora en que se inició alguna limpieza. El Dattalogger virtual utiliza la memoria del PC como base de datos. Finalmente, parte de la programación se muestra en la sección “Anexos”.

4.3.1.- Implementación sistema HMI

El diseño ergonómico se basó en parte sobre la normativa ISO 9241, la cual abarca el diseño de programas para equipos con pantallas de visualización de datos, enumerando una serie de sugerencias que sirvan de guía a la hora de realizar su planteamiento y desarrollo. Por lo tanto, la implementación se realizó en base a un diseño industrial, en donde para la manipulación y monitoreo se implementó una unidad de limpieza manual, automática y programada, cómo también unidad de monitoreo de nivel de suciedad para sombreado de panel fotovoltaico.

Un bosquejo de cómo podrían estar distribuidos los elementos del sistema HMI, se definió de manera de hacerlo lo más parecido a los que ya existen industrialmente. se definió el color blanco cómo fondo color base de la interfaz gráfica con una resolución de 1378 x 780 pixeles, para el tamaño de fuente se definió la de tipo Arial de tamaño 12.

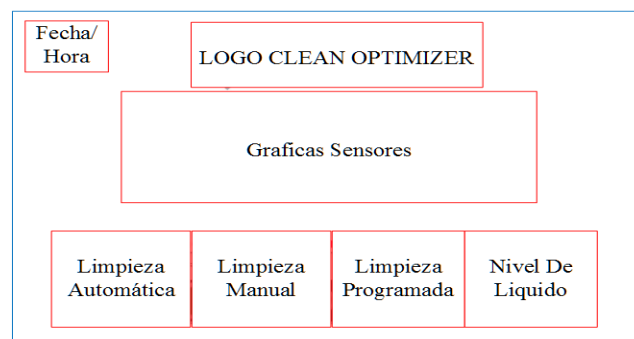


Fig.4.3: Diseño HMI de sistema SCADA

4.4.- Unidad Orientada

Se configuro una unidad en el HMI para realizar una limpieza orientada hacia uno de los sentidos de un panel fotovoltaicos, como también una función de parada. Se establecieron estos modos con el objetivo que el operario no tenga que desplazar con sus manos el robot si quisiera fijarlo en un punto en particular, debido a que puede estar expuesto a peligros como altas temperaturas, caídas, electrocución, entre otros peligros. También para realizar una limpieza ante alguna eventualidad de suciedad, teniendo en cuenta que la deposición de suciedad es variable. Por otro lado, se pueden hacer realizar pruebas de movimiento para ver el estado de la parte mecánica de “Clean Optimizer” o soporte de la estructura de paneles fotovoltaicos, poder realizar mantenimientos. La dinámica de esta unidad consta primeramente en configurar los parámetros, luego de que se configuran los parámetros, se realiza la petición hacia el esclavo, en donde se pregunta así la variable izquierda, derecha o parada toma el valor de 1 se ejecuta la solicitud, de lo contrario el prototipo no ejecuta dicha acción. Por parte de la programación la función que recibe los datos de comunicación inalámbrica recibe un carácter ‘L’, ‘R’ o ‘S’ dependiendo de la función que se ejecute.

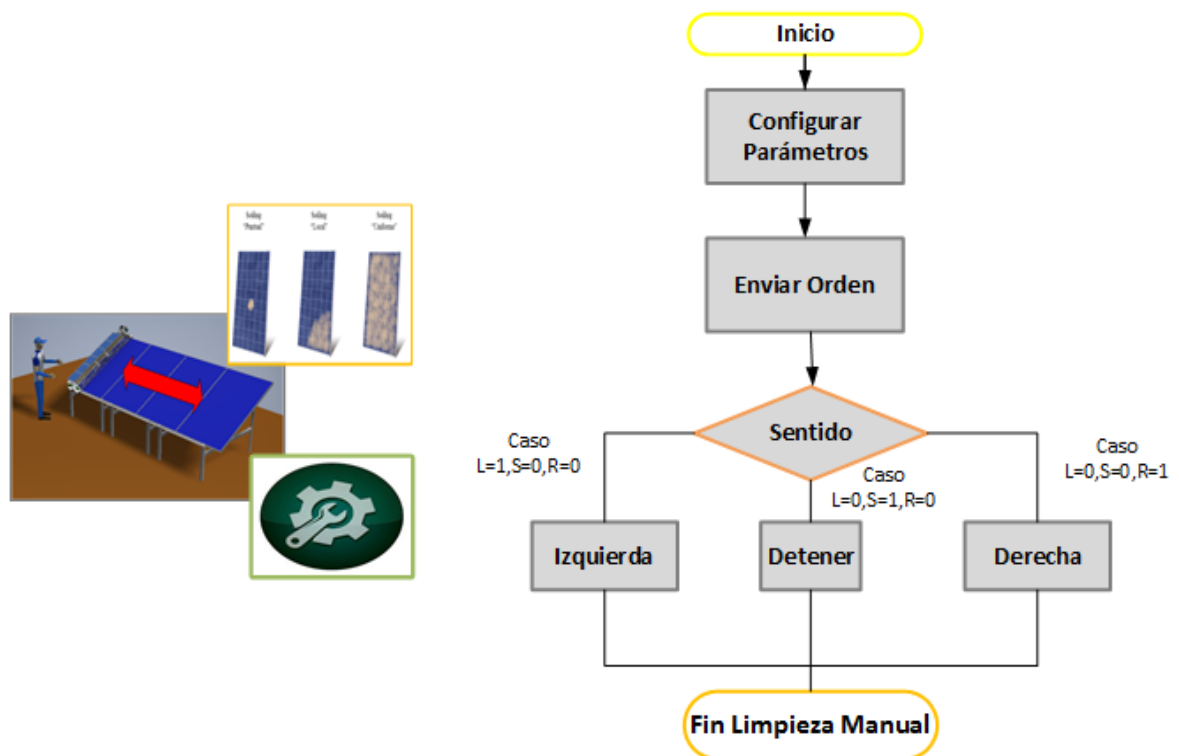


Fig.4.4: Diagrama bloques limpieza orientada

4.5.- Unidad Automática

Se configuro una unidad en el HMI para modo de limpieza automática, la cual consta en que el robot haga un barrido de ida y vuelta sobre el panel fotovoltaico. A esto se le agrega una unidad de temporizador si el usuario quisiera activar la limpieza automática mientras esta necesita realizar otra acción. El proceso consta en primer en la configuran los parámetros, luego configurar el temporizador, luego en el software de control se verifica si el tiempo configurado corresponde al tiempo transcurrido. En la cual se verifica que la variable “Temp” al transcurrir el tiempo toma el valor del tiempo ingresado, se ejecutará una limpieza automática. Para que se cumpla la función de limpieza automática el esclavo recibe el parámetro ‘A’.

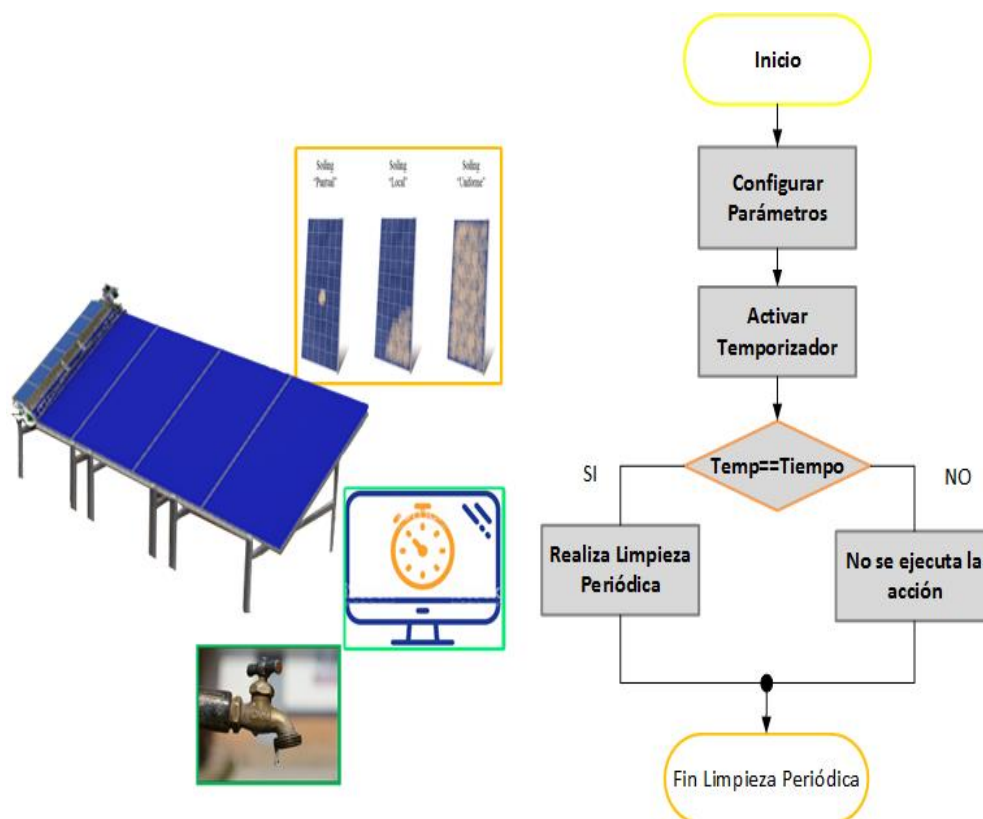


Fig.4.5. Diagrama de flujo de limpieza automática

4.6.- Unidad Planificada

Se diseñó en el sistema HMI una unidad para planificar la hora, día y fecha para realizar un limpieza orientada o automática, en la cual la unidad maestra se puedan enviar solicitudes hacia el esclavo con el fin de realizar una limpieza en una hora y fecha determinada. Esta unidad está enfocada realizar limpiezas nocturnas no invasivas, con el objetivo de no detener la producción de energía. Por otro lado, esta estrategia de limpieza permite hacer mantenimientos preventivos para lo cual se utilizó un complemento de “Datalogger” virtual asociado a la variable de limpiezas para guardar la información detallada de cuando se realizó la limpieza. Esto se lleva a cabo primeramente configurando los parámetros En este caso el para el software de control se configuran los parámetros, acá se verifica el registro de dirección del elemento de la unidad esclava, luego se envía una solicitud por parte de la unidad terminal maestra, la solicitud trata de comparar la fecha y hora ingresada por el usuario. si la fecha y hora ingresada es igual a la fecha actual se ejecuta la limpieza establecida.

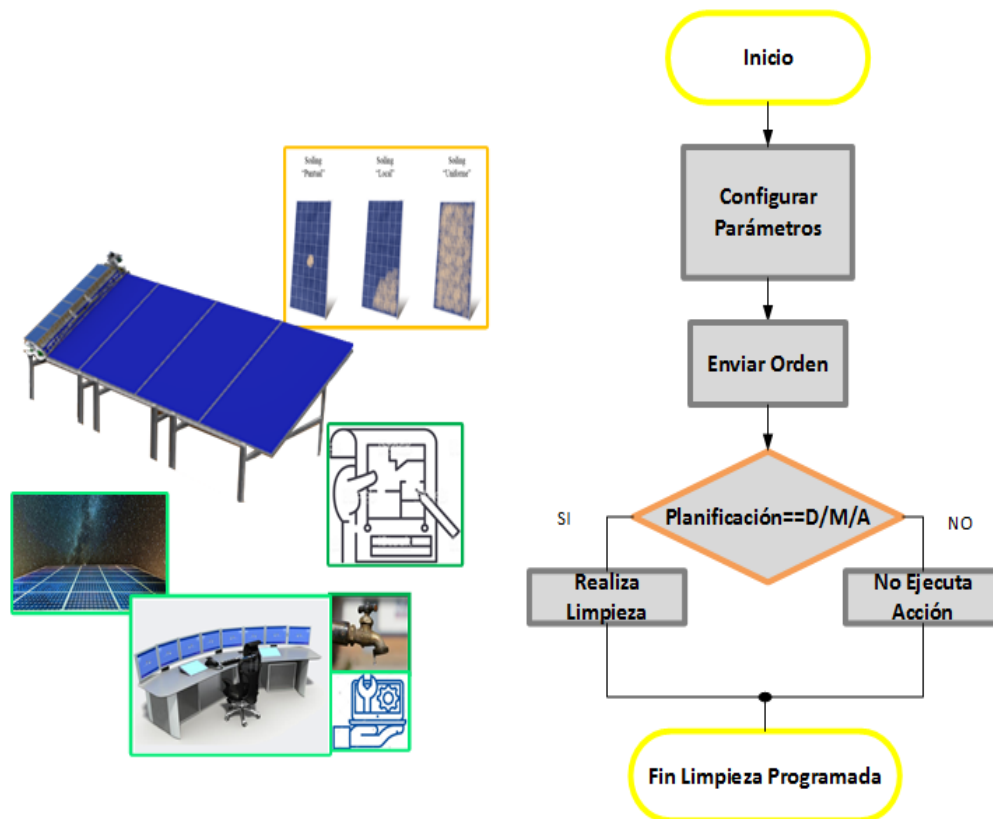


Fig.4.6: Diagrama de flujo de limpieza planificada

4.7.- Almacenamiento de datos

Se integró un Data Logger virtual que es un complemento de Visual Studio para poder asignar variables y guardar datos en formato de día, mes, año, hora, minutos y segundos, los cuales van a ser exportados en formato “.txt” y se guardan en el ordenador. los datos se almacenarán en un formato de tipo.

4.8.- Monitoreo de Suciedad

Se estableció una unidad en el sistema HMI una unidad para monitorear la estimación del porcentaje de sombreado provocado por efecto de suciedad. La medición obtenida es la cantidad de luz reflejada al sensor de la fuente lumínica que es un diodo led de luz blanca en proporción a la densidad de polvo acumulada sobre la superficie del panel fotovoltaico, **pero para efectos de esta memoria, es despreciable obtener de forma exacta la densidad de partículas de polvo sobre la superficie**, pero si es importante saber que tan sombreado se encuentran las celdas fotovoltaicas por causa de la acumulación de polvo. También la persona encargada de los algoritmos de los sensores para estimar el porcentaje de suciedad realizo un filtrado de señal para poder escalar los valores tomados debe escalar de 0% a 100%.

Se realizaron pruebas de laboratorio en un ambiente oscuro, debido a que el monitoreo de suciedad se piensa realizar de noche, por lo tanto, se realiza una estimación del porcentaje de sombreado provocado por efecto de suciedad. Para esto se implementa un gráfico en el cual se puede monitorear la estimación de porcentaje de sombreado por efecto de suciedad con 20 sensores tipo TSL2561 que serán parte de “Clean Optimizer”.

4.8.1.- Detección de suciedad

Los sensores TSL2561 utilizan la unidad de medida “Lux”, en donde cada uno de los 20 sensores cubren totalidad de las zonas de un panel fotovoltaico instalado en un laboratorio. El direccionamiento de los 20 sensores para medir el porcentaje de sombreado por efecto de suciedad. En la siguiente tabla se muestra el direccionamiento de los sensores utilizados para detectar polvo:

TABLA 7 Control maestro/esclavo ‘Porcentaje Estimado de Suciedad’

SENSOR	FUNCIÓN	VARIABLE	MEMORIA	VARIABLE	DIRECCIÓN
TSL2561		GRAFICO		MEMORIA	
1	MEDIR POLVO	SS1	11	SS_1	40012
2	MEDIR POLVO	SS2	13	SS_2	40014
3	MEDIR POLVO	SS3	15	SS_3	40016
4	MEDIR POLVO	SS4	17	SS_4	40018
5	MEDIR POLVO	SS5	19	SS_5	40020
6	MEDIR POLVO	SS6	21	SS_6	40022
7	MEDIR POLVO	SS7	23	SS_7	40024
8	MEDIR POLVO	SS8	25	SS_8	40026
9	MEDIR POLVO	SS9	27	SS_9	40028
10	MEDIR POLVO	SS10	29	SS_10	40030
11	MEDIR POLVO	SS11	31	SS_11	40032
12	MEDIR POLVO	SS12	33	SS_12	40034
13	MEDIR POLVO	SS13	35	SS_13	40036
14	MEDIR POLVO	SS14	37	SS_14	40038
15	MEDIR POLVO	SS15	39	SS_15	40040
16	MEDIR POLVO	SS16	41	SS_16	40042
17	MEDIR POLVO	SS17	43	SS_17	40044
18	MEDIR POLVO	SS18	45	SS_18	40046
19	MEDIR POLVO	SS19	47	SS_19	40048
20	MEDIR POLVO	SS20	49	SS_20	40050

4.8.2.- Grafico porcentaje de sombreado estimado por efecto de suciedad

Se estableció una unidad en el sistema HMI para que un operador pueda monitorear la estimación de porcentaje de sombreado por efecto de suciedad y mostrar los datos gráficamente. Esto quiere decir que de los valores brutos que sean recibidos, se debe hacer un escalamiento para representarlo en una estimación de porcentaje. El proceso de la comunicación basada en la arquitectura maestro/esclavo consta en que el maestro solicita al esclavo acceder a leer el contenido de las direcciones de los bloques de memoria de SS1_ (Sensor suciedad 1) hasta SS_20.

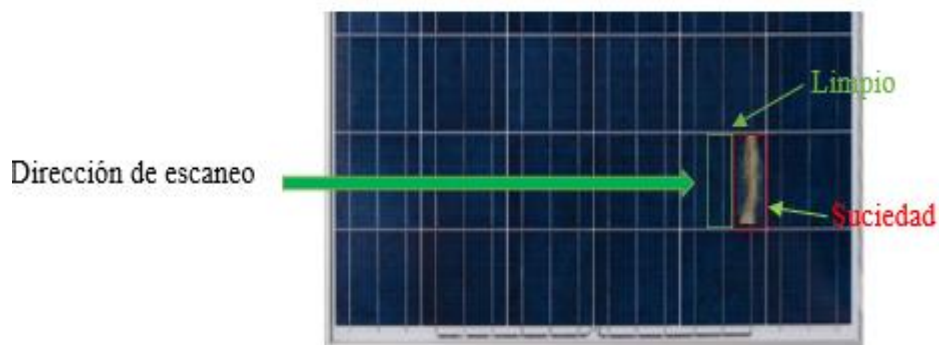


Fig.4.7: Escaneo de suciedad en paneles fotovoltaicos

El diseño en el sistema HMI consta de un gráfico en el cual cada variable se presenta en una barra vertical, representada con un color del gráfico, en donde primeramente se tomaran pruebas de sus valores brutos en lux y luego en la estimación de porcentaje. Esto quiere decir que, si no existen partículas de suciedad en el sector de medición del sensor, gráficamente se estimara un porcentaje de 0 a 20% y si se detecta partículas de suciedad que generen efecto de sombreado en el sector de medición del sensor ese porcentaje va a aumentar y si se detecta bastante suciedad se podría estimar un porcentaje cercano a 100%. Cabe decir que la unidad para detección de suciedad está en estudio debido a que se están investigando la posibilidad de probar nuevas alternativas como algoritmos de inteligencia artificial.

Capítulo 5. Implementación sistema comunicación inalámbrica

5.1.- Sistema de Comunicación Inalámbrica

Se implementará una topología de red punto a punto la cual consta de un Módulo remoto y un Módulo central. El Módulo remoto se encargará de enviar y recibir información hacia la unidad central. En dicha unidad central se implementará un Módulo coordinador que como dice su nombre, se encarga de coordinar la red inalámbrica para enviar y recibir información hacia el Módulo SCADA, en donde se implementara un sistema HMI el cual se va a implementar en un PC para que el usuario pueda acceder y a manipular información de los parámetros de limpieza de paneles fotovoltaicos del prototipo en donde los sensores deberían detectar si existen partículas de suciedad. A continuación, se muestra cómo se puede establecer dicho sistema, en donde describen en detalle cómo desarrollaron las unidades descritas anteriormente.

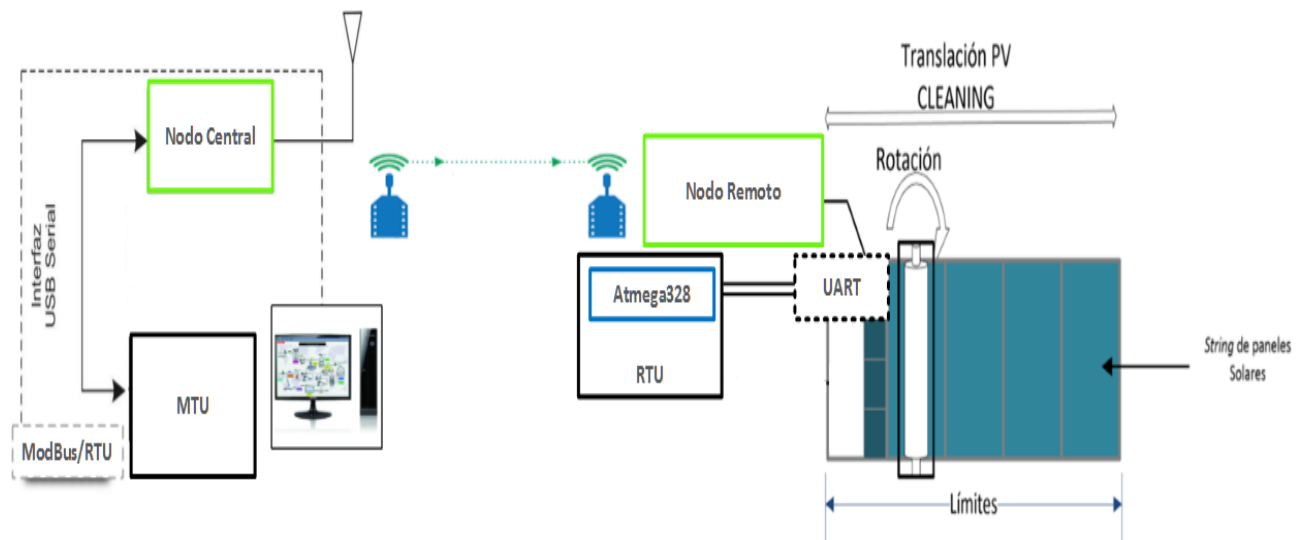


Fig.5.1: Sistema de comunicación inalámbrica

5.1.1.- Configuración red inalámbrica

La red inalámbrica es desarrollada en base a la topología punto a punto, en modo transparente o AT. Donde la red cuenta un nodo el cual es configurado y programado como ZIGBEE COORDINADOR, por otro lado, la red cuenta con un nodo configurado y programado como ZIGBEE ROUTER. en la siguiente figura se muestra el esquema de distribución y distancia para cada uno de los Módulos.



Fig. 5.2: Esquema distribución de red inalámbrica de sensores

La red está configurada de manera que se genere una comunicación bidireccional con el objetivo de enviar información de sensores y actuadores, los cuales se podrían utilizar para realizar una limpieza de paneles fotovoltaicos. La distancia mínima para recorrer un panel fotovoltaico es 1 metro, la mayoría de los arreglos de paneles fotovoltaicos se componen de 20, por lo tanto, se establece como distancia requerida apropiadamente 20 metros.

5.1.2.- Configuración nodos de red punto a punto

Antes de comunicar los Módulos, se debe configurar los parámetros del nodo remoto y nodo central. Esto se realiza conectando el dispositivo XBee Explorer al PC, mediante cable tipo USB. Luego se debe utilizar el software XCTU e instalar drivers de acuerdo con el sistema operativo, posteriormente se conecta el dispositivo XBee pro 900HP-S3B al dispositivo XBee Explorer, luego de verificar que estos dispositivos están conectados, se realiza la conexión de estos dispositivos al PC. Finalmente, Al realizar la conexión se puede verificar en el software XCTU que el dispositivo se encuentra conectado al puerto COM22:

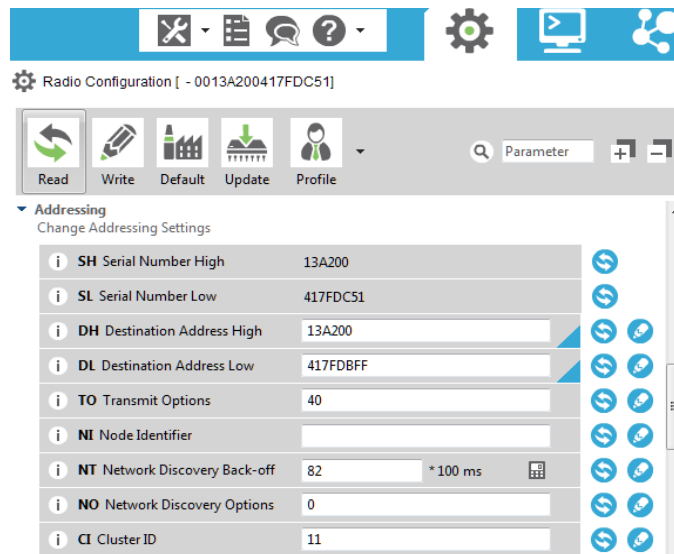


Figura 5.3: Parámetros red inalámbrica ZigBee

Al momento de cargar o escribir nuevos parámetros en los Módulos XBee PRO 900HP-S3B se debe presionar el botón “Write. El botón “Read” se debe usar cuando se requiera leer los datos parámetros cargados o los por defecto cargados anteriormente en el dispositivo.

5.2.- Módulo Remoto

Se desarrolla el Módulo remoto el cual consta del acoplamiento de sensores, actuadores, la placa microcontroladora Arduino R3, la placa Shield XBee y el dispositivo XBee Pro 900HP-S3B.

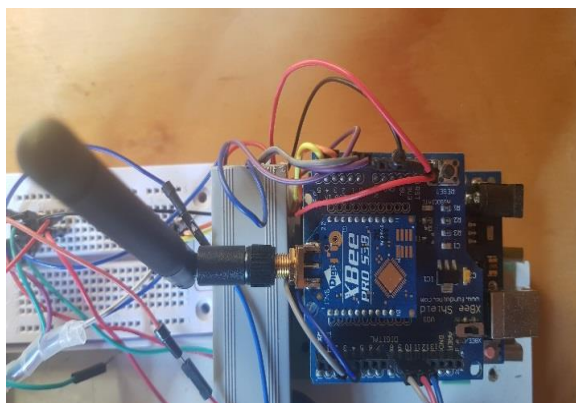


Fig. 5.4: Acoplamiento placa microcontroladora Arduino y Dispositivo XBee

Se realiza la conexión para la alimentación, envío y recepción de datos de los sensores cómo también actuadores, mediante los pines de la placa microcontroladora Arduino R3. A continuación, se muestra en las siguientes tablas cómo se realizó las conexiones para la transmisión y recepción de información.

TABLA 8 Conexión de sensores con placa microcontroladora

SENSORES	FUNCIÓN	VCC	GND	OUTPUT
TSL2561	SUCIEDAD	5V	GND	SDA =A4 SCL= A5 I2C
XKC-Y25	NIVEL LÍQUIDO	5V	GND	MIN =A1 MAX=A2

TABLA 9 Conexión de actuadores con placa microcontroladora

ACTUADORES	FUNCIÓN	VCC	GND	MICROCONTROLADOR
MOTOR	MOVIMIENTO	12 V	GND	ENTRADA 1= PIN 10 ENTRADA 2 = PIN 9 ENABLE A = PIN 8
BOMBA	LÍQUIDO	12 V	GND	ENTRADA 1 = PIN 4 ENTRADA 2 = PIN 3 ENABLE A = PIN 2

Finalmente Se realiza el acoplamiento entre el dispositivo XBee Pro 900HP-S3B y la placa microcontroladora Arduino R3, mediante el dispositivo Shield XBee.

5.2.1.- Comunicación entre placa microcontroladora y Dispositivo XBee

La comunicación se realiza conectando los pines DIN y DOUT del dispositivo XBee pro 900HP con los pines Tx y Rx de la placa microcontroladora Arduino R3. Cuando se conecta placa microcontroladora Arduino R3 y el dispositivo XBee Pro900HP-S3B, se utiliza una placa llamada “Shield XBee”, la cual permite trabajar con el XBee o el puerto serie de la placa microcontroladora R3. En la siguiente tabla se muestra la interconexión de los dos dispositivos.

TABLA 10 Conexión de Placa Microcontroladora R3 con dispositivo XBee

XBEE	ARDUINO R3
VCC	3.3 v
DOUT	PIN 0
DIN	PIN 1
GND	GND

5.3.- Módulo Central

Se realiza el acoplamiento de un dispositivo XBee pro 900HP-S3B con un dispositivo XBee Explorer, para lo cual conecta los 20 pines destacando que los pines más importantes son el pin 1 que corresponde al pin de alimentación, los pines 2 y 3 que corresponden a los pines de envío y recepción y el pin 10 correspondiente al pin GND. Luego esta interconexión entre dispositivos es llevada a un PC mediante el bus serial para ser programado.

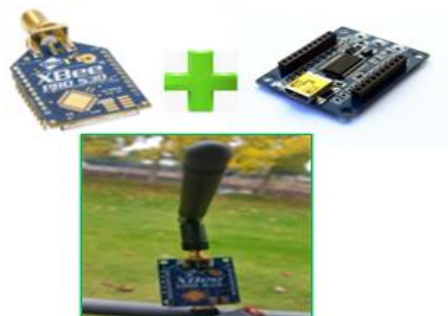


Fig.5.5: Módulo Central

5.3.1.- Comunicación entre módulo remoto y Módulo central

La comunicación entre el Módulo remoto y el Módulo central es implementada de manera que los Módulos se comuniquen de manera bidireccional. La transmisión de datos es desarrollada en modo transparente. Los nodos transmiten y reciben información a una velocidad de hasta 10 Kbps por segundo en la banda de 900MHz. La información que envía un nodo va dirigida a la dirección de 64 bits de destino del otro nodo, además él envió de información esta configurada de modo “unicast”, esto quiere decir que información solo le llega al nodo de destino de la red. En este modo se envía un ACK (Acuse de recibo), el cual consta en una respuesta por parte del nodo que recibe el paquete de datos. Debido a esto si el nodo no recibe el ACK, este modo permite reenviar el paquete tres veces es una respuesta entre dispositivos XBee, pero el usuario no puede verlo. La seguridad de la red es un tema importante al momento de la comunicación por ende estos nodos poseen parámetros que se configuran para proteger el envío de comunicación, los cuales se describen a continuación:

- Los nodos poseen direccionamiento único en el mundo, en este caso será de 64 bits, con una dirección de destino alta y dirección de destino bajo.
- Asignar PAN ID, la cual corresponde al nombre de la red esta puede ser de 16 bits o 64 bits, la condición de seguridad es que los nodos deben poseer el mismo pan ID para poder conectarse a la red. Se asignó 7FFF que era el PAN ID que venía por defecto.
- Posee parámetros como Habilitar Encriptación (EE), clave de red(NK), lo cuales podrían otorgarle más seguridad de la ya poseen los nodos XBee.
- La seguridad actúa encriptando la información, usando el mismo algoritmo por bancos y gobiernos para transacciones de seguridad.

TABLA 11 Configuración de parámetros para nodo remoto y nodo central

FUNCIÓN	PAN ID	NI	SH	SL	DH	DL
COORDINADOR	7FFF	NC	13A200	7FDC51	13A200	7FDBFF
ROUTER	7FFF	NR	13A200	7FDBFF	13A200	7FDC51

Capítulo 6. Sistema de Alimentación

Es importante saber cómo se va a alimentar este sistema, sobre todo el tema respecto a la fuente de alimentación de los elementos que componen la comunicación inalámbrica. En el siguiente capítulo se dan a conocer las partes y componentes para dicha etapa.

6.1.- Descripción del sistema de alimentación

Para el sistema que emula los elementos de Clean Optimizer se utilizó una batería de 12 V para la alimentación. Por un lado, la placa Arduino Uno R3 y el microcontrolador ATmega328 trabajan con un voltaje continuo de 5V, mientras que el Módulo XBee requiere 3.3V. Debido a esto la en la placa de circuitos impresos se realizó un circuito regulador de voltaje.



Fig.6.1: Alimentación sistema completo

El nodo retorno se necesita alimentar con voltaje de 3.3V, para lo cual se utilizó una batería recargable de litio con una capacidad de 2300mAh y un voltaje de 3,7V, para las pruebas del sistema que emula los elementos de “Clean Optimizer



Fig.6.2: Batería alimentación Módulo remoto

Las características más importantes de la batería elegida para alimentar el nodo remoto se presentan en la siguiente tabla:

TABLA 12 Características de batería de alimentación de Módulo Remoto

Características	Especificaciones
Voltaje	3.7V
Capacidad	2300mAh
Constante de Descarga	2V x Celda
Tasa de Auto-Descarga	< 8% por mes
Condiciones de Trabajo	-25°C a 60°C
Peso	33 gr
Dimensiones	4,2x55x87 mm

La alimentación del nodo central consta de la alimentación que viene directamente del bus USB y en donde no se tienen limitaciones de energía, siempre y cuando no se superen los 500mA que pueden abastecer cómo máximo el bus USB.



Fig.6.3: Alimentación de Nodo Central

Capítulo 7. Resultados

En el siguiente capítulo se muestran los resultados obtenidos en este proyecto, en donde se muestra los dos tipos de nodos que componen la red inalámbrica. Luego, se muestran las pruebas de funcionamiento de sensores y actuadores en el sistema SCADA, para ver la calidad de transferencia de los datos y cómo se visualizan en el sistema de monitoreo.

7.1.- Emulación Sistema

Se realizaron pruebas del funcionamiento del sistema inalámbrico y sistema SCADA, para lo cual se utilizó el un setup experimental implementado en este proyecto el cual posee elementos que utiliza Clean Optimizer los cuales se describen en el siguiente esquema.

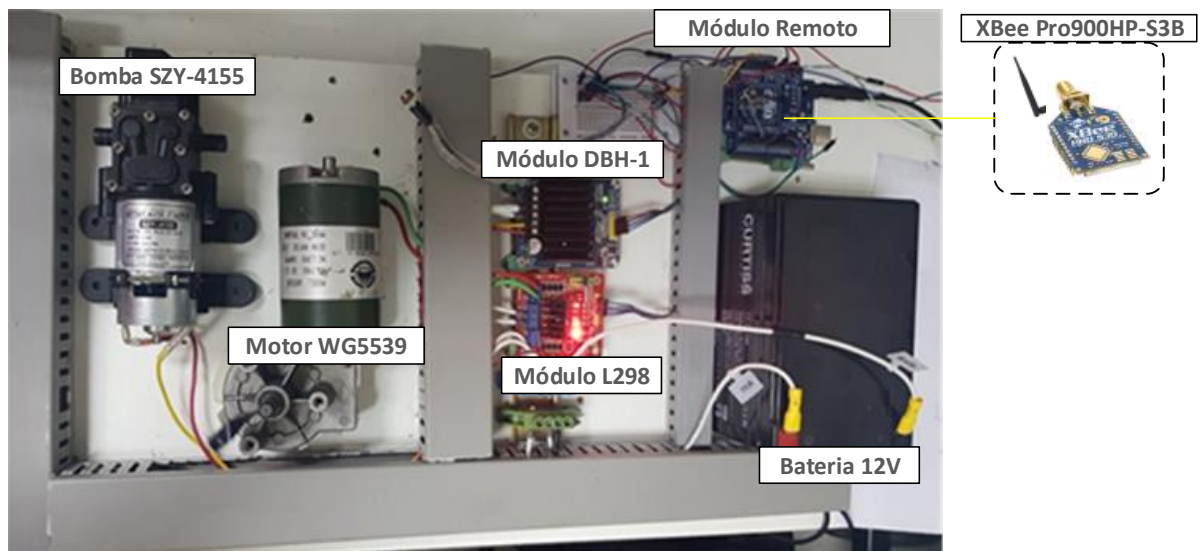


Fig.7.1: Setup Experimental para emular elementos de Clean Optimizer

En el Módulo remoto se encuentra acoplado el controlador que en este caso es el microcontrolador ATmega328P de la placa Arduino R3, el dispositivo Shield XBee y el nodo robot el cual consta de un dispositivo XBee Pro900HP-S3B el cual se comunica mediante protocolo UART con el microcontrolador ATmega328p.

El módulo central está ubicado a más de 100 metros del módulo remoto, el cual es el encargado de formar y controlar toda la red inalámbrica. por lo que solicita los datos al módulo remoto de manera inalámbrica en la banda de 900 MHz, en donde los datos se envían de un nodo a otro en una tasa de transmisión de 10kbps, mediante el protocolo de comunicación Modbus/RTU basada en la arquitectura maestro/esclavo. En el Módulo central se realiza el acoplamiento de un dispositivo Explorer XBee el cual posee conectores DIP de 20 pines para conectar el nodo coordinador el cual consta de un dispositivo XBee Pro900HP-S3B. A continuación, se describen los resultados de la implementación del Módulo central

- El desarrollo del Módulo central consta del acoplamiento entre el Nodo Central que tiene un dispositivo XBee Pro 900HP-S3B y una placa dispositiva XBee Explorer. Debido a esto se conectan los 20 pines del dispositivo XBee al socket de la placa XBee.

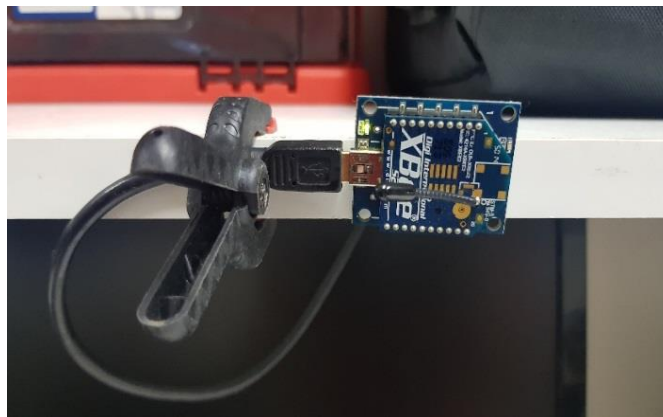


Fig.7.2 Acoplamiento XBee Pro S3B con XBee Explorer

- Mediante cableado USB conectamos la salida a un conversor de USB a UTP mediante conector RJ45, para extender el cableado debido a que la normativa USB permite trabajar hasta 5 metros.



Fig.7.3 Acoplamiento de extensor USB/UTP mediante conector rj45.

- Para el cableado UTP se utilizará cableado de categoría 5e el cual pasa por un conversor UTP a USB para poder conectar el sistema con un PC en el cual se encuentra el sistema de monitoreo.



Fig.7.4: Extensión de cable mediante conversor UTP/USB.

El módulo central se comunica con un sistema SCADA mediante interfaz USB serial, en el que el sistema SCADA posee un HMI en el cual se puede monitorear, acceder y manipular la información de los elementos emulados de Clean Optimizer.

7.2.- Clean Optimizer

Se realizó la interconexión entre el nodo remoto y la placa del microcontrolador ATmega328, la cual consiste en la unión de los pines de alimentación y comunicación del nodo y microcontrolador. La interconexión entre nodo remoto y placa del microcontrolador se describe en la siguiente tabla.

TABLA 13 Comunicación entre Nodo Remoto y Microcontrolador ATmega 328

MÓDULO REMOTO	ELEMENTOS	VCC	GND	COMUNICACIÓN
CLEAN OPTIMIZER	MICROCONTROLADOR ATMEGA 328P. NODO REMOTO.	3.3 V	GND	DIN → PIN 10 DOUT → PIN 11

7.2.1.- Módulo Central

El Nodo Central o nodo central es ubicado a la intemperie. El Nodo Central consiste en acoplar el dispositivo XBee con el dispositivo XBee Explorer, este nodo es alimentado mediante 5V desde el bus USB, pero el dispositivo XBee Explorer posee un regulador de 5V a 3.3V para alimentar el nodo central.

TABLA 14 Comunicación entre Nodo Remoto y Nodo Central

MÓDULO CENTRAL	ELEMENTOS	VCC	GND	COMUNICACIÓN
INTEMPERIE	XBEE PRO 900HP-S3B. XBEE EXPLORER	3.3 V	GND	PINES DE COMUNICACIÓN TX, RX.

7.3.- Sistema SCADA

El prototipo Clean Optimizer posee un modo para trabajar con el sistema de comunicación inalámbrica y el sistema SCADA, el cual mediante programación del microcontrolador ATmega328 recibe información del modo en cual está trabajando y hacia qué sentido se está realizando la limpieza de paneles fotovoltaicos. Finalmente, el sistema funcionara cuando el switch de Clean Optimizer se encuentre en la posición online, debido a que el robot puede trabajar en modo manual, automático y online.



Fig.7.5: Modo Online para Sistema Inalámbrico

Se realizaron pruebas del sistema SCADA para el prototipo limpiador de paneles fotovoltaicos Clean Optimizer, el cual muestra por una parte la unidad de nivel, en donde se desarrolló una unidad en la cual se representa gráficamente el nivel de líquido y un comando para activar y desactivar la bomba de Clean Optimizer. Por otro lado, se realizaron pruebas de los modos de limpieza en donde el usuario desde el sistema HMI envió órdenes para mover el robot sobre un panel fotovoltaico. Finalmente se realizaron pruebas para ver gráficamente el porcentaje de sombreado estimado por suciedad que detectaba cada sensor. A continuación, se muestra el resultado final de cómo quedó el sistema HMI y las respectivas representaciones gráficas de nivel de líquido, porcentaje de sombreado estimado por efecto de suciedad y modos de limpieza.

7.4.- Sistema SCADA para movimiento de Clean Optimizer

Se realizaron pruebas de Clean Optimizer funcionando mediante una red inalámbrica de topología punto a punto, en la cual se envía y recibe información desde el módulo remoto que esta implementado en el setup experimental, al módulo central. En donde el módulo central es el encargado de formar y controlar toda la red inalámbrica, por lo que solicita los datos al módulo remoto. Finalmente, se realizaron pruebas con el sistema SCADA de movimientos para las unidades de limpieza manual, automática y programada. A continuación, se presentan los resultados obtenidos de las pruebas de movimiento que se muestran ilustradas en los siguientes subcapítulos.

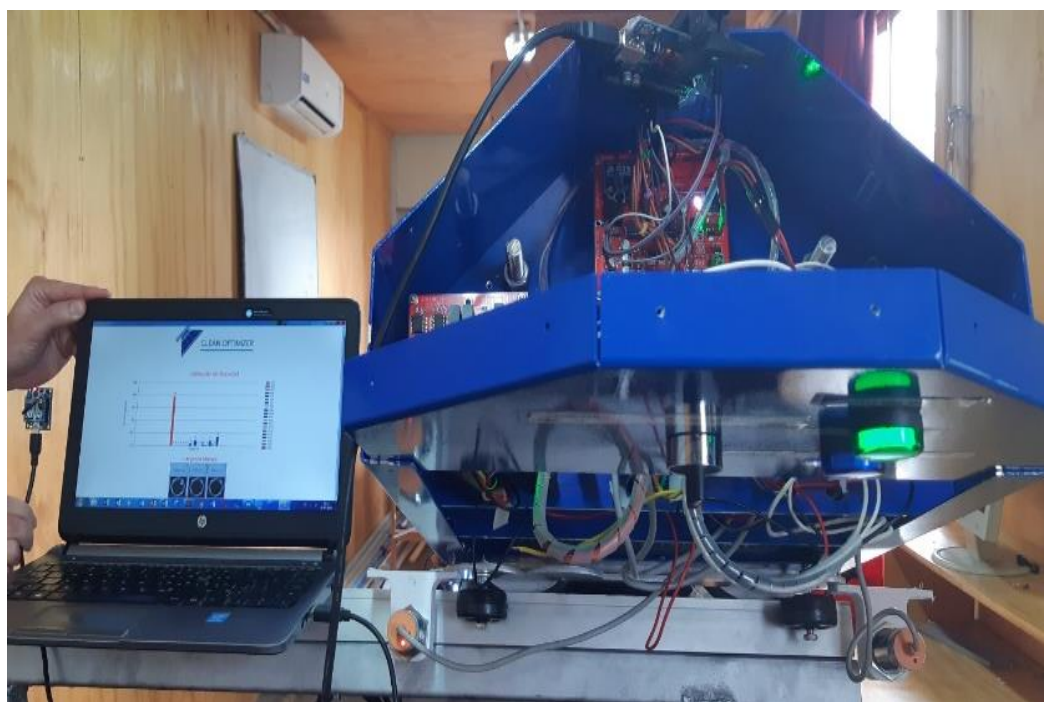


Fig.7.6: SCADA conectado a “Clean Optimizer”

7.4.1.- Pruebas de movimiento para limpieza de paneles fotovoltaicos

El operador desde el sistema HMI debe manipular el modo de limpieza orientada para lo cual el operador puede elegir la opción de mover izquierda. Acá se establecen botones para que el maestro solicite al esclavo una función para acceder al contenido del dato, ya sea leer o escribir sobre las salidas digitales correspondientes al motor de “Clean Optimizer”, utilizando 1 bit para cada función. La trama en este caso consta del envío del carácter ‘L’, la cual es recibida por el esclavo y se ejecuta la acción.

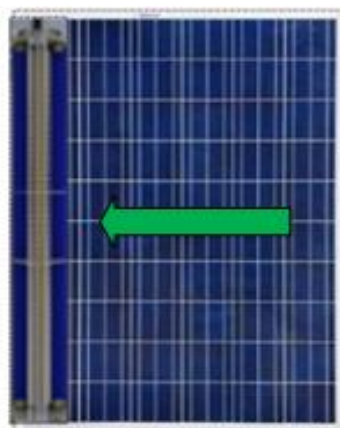


TABLA 15 Control maestro/esclavo ‘Mover Izquierda’

ELEMENTO	FUNCIÓN	BLOQUE DE MEMORIA	DIRECCIÓN
MOTOR	IZQUIERDA	54	00053

7.4.2.- Unidad de Detención

El operador desde el sistema HMI debe manipular el modo de limpieza orientada para lo cual el operador puede elegir la opción de parada. Acá se establecen botones para que el maestro solicite al esclavo una función para acceder al contenido del dato, ya sea leer o escribir sobre las salidas digitales correspondientes al motor de “Clean Optimizer”, utilizando 1 bit para cada función. La trama en este caso consta del envío del carácter ‘S’, la cual es recibida por el esclavo y se ejecuta la acción.

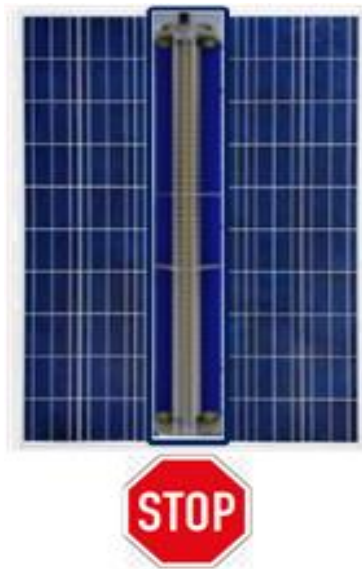


TABLA 16 Control maestro/esclavo ‘Mover Izquierda’

ELEMENTO	FUNCIÓN	BLOQUE MEMORIA	DIRECCIÓN
MOTOR	PARADA	56	00055

- Se asignó una posición inicial para el prototipo, posición en la cual se ubica en el lado derecho del panel fotovoltaico.



Fig.7.7: Posición Inicial en prueba de movimiento.

- Se realizó acoplamiento con setup experimental.
- Se conectó el sistema de comunicación inalámbrica
- Se habilitó el sistema SCADA
- Se inició el sistema HMI
- Se realizó una prueba de movimiento, pulsando la Botón del sistema SCADA de nombre “Izquierda”, luego de esto el robot se movió hacia la izquierda.
- Al momento de llegar al borde se tomó la decisión de presionar la Botón parada para el robot limpiador de paneles fotovoltaicos no se saliera del panel fotovoltaico, este paso y el anterior se muestran en la siguiente imagen.



Fig.7.8: Prueba de movimiento sentido izquierdo

Los resultados fueron los esperados debido a que se realizó una prueba mediante comunicación inalámbrica cuando se pulsaba un botón hacia un sentido, se enviaba una trama indicando el sentido, entonces el microcontrolador detectaba esa trama y si la trama era 'L' el robot se movía hacia la izquierda, si la trama era 'S' el robot se detenía.

		1, 0, M, L
Avanzando	→	1, 0, M, L
		1, 0, M, L
		0, 0, M, L
		1, 0, M, S
Detenido	→	1, 0, M, S
		1, 0, M, S

7.4.3.- Mover Derecha

El operador desde el sistema HMI debe manipular el modo de limpieza orientada para lo cual el operador puede elegir la opción de mover derecha. Acá se establecen botones para que el maestro solicite al esclavo una función para acceder al contenido del dato, ya sea leer o escribir sobre las salidas digitales correspondientes al motor de “Clean Optimizer”, utilizando 1 bit para cada función. La trama en este caso consta del envío del carácter ‘R’, la cual es recibida por el esclavo y se ejecuta la acción.

TABLA 17 Control maestro/esclavo ‘Mover Derecha’

ELEMENTO	FUNCIÓN	BLOQUE DE MEMORIA	DIRECCIÓN
MOTOR	DERECHA	58	00057

Finalmente, la unidad de limpiezas del sistema SCADA, en la cual el operador va a poder manipular o visualizar la unidad movimiento de “Clean Optimizer” quedaría de la siguiente manera:

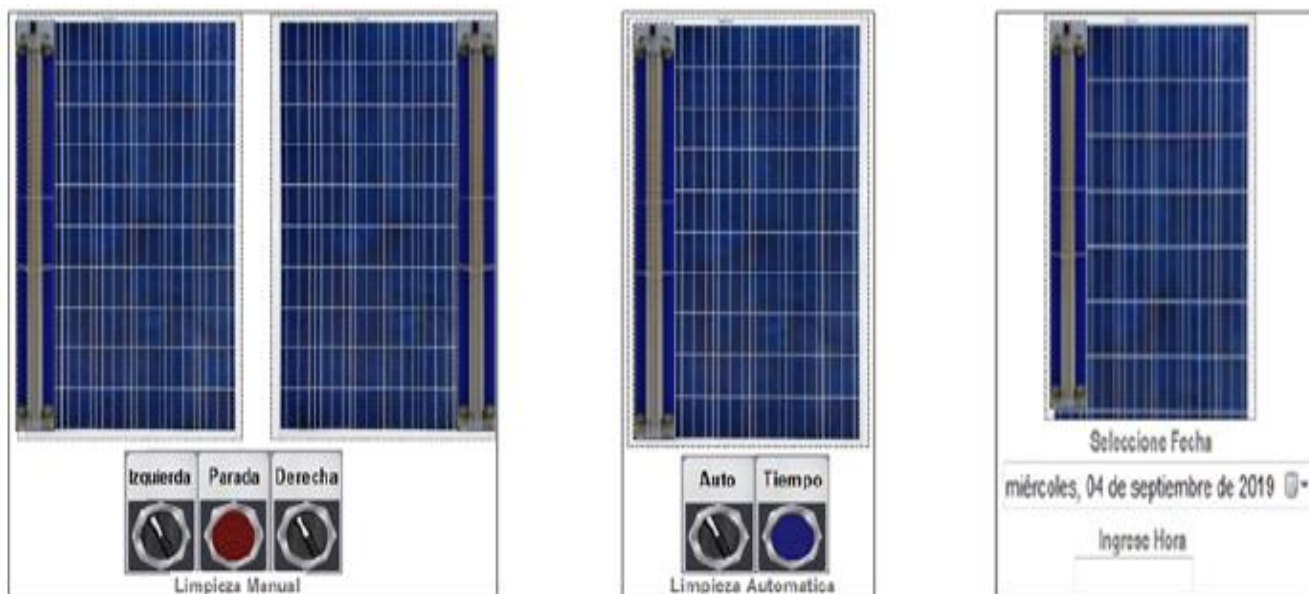


Fig.7.9: Unidad de limpieza sistema SCADA

7.4.4.- Activar y desactivar bomba de agua

Se diseñó una unidad en el sistema HMI en un caso que el operador requiera hacer uso de agua de manera localizada ante alguna deposición de suciedad o en un caso que quiera activar la bomba para vaciar el estanque donde se almacena el líquido el cual posee una capacidad de 8 litros. El proceso consta de la configuración de los parámetros de la función de la bomba de agua, acá se verifica el registro de dirección del elemento de la unidad esclava, luego se envía una solicitud por parte de la unidad terminal maestra, se verifica los casos para la variable “Bomba” y luego se ejecuta la acción de activar la bomba en caso que la variable de activar la bomba tome el valor de 1, de lo contrario no se ejecuta la acción.

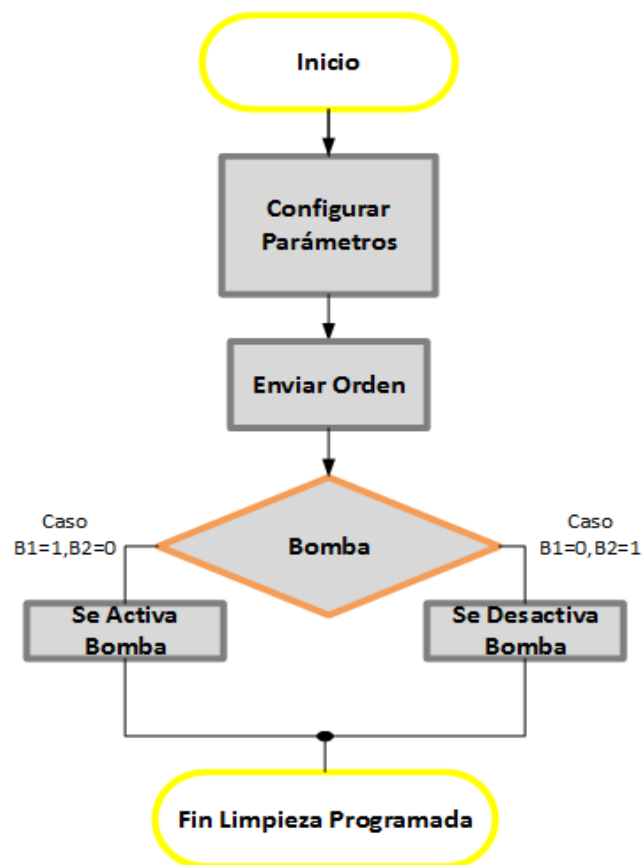


Fig.7.10: Bloque de Bomba de agua en sistema SCADA

7.4.4.- Activar Bomba

Se diseñó una unidad en sistema HMI para manipular la activación de la bomba de agua, la variable de tipo salida de digital que el esclavo debe acceder cuando el maestro se lo solicite. El elemento de la variable para activar la bomba que en este caso corresponde a “B1”, se utiliza asignando prefijo el 0 en el inicio de mapeo de dirección, lo cual indica que el maestro solicita al esclavo acceder a la salida digital del elemento mediante la dirección de la variable definida anteriormente, esa variable ocupa un espacio de memoria y una dirección. Debido a esto la respuesta del esclavo es 1 bit para activar la bomba, A continuación, en la siguiente tabla se muestra el mapeo de dirección de Modbus la función de encender la bomba.

TABLA 18 Control maestro/esclavo ‘Encender Bomba’

ELEMENTO	FUNCIÓN	MEMORIA	DIRECCIÓN
BOMBA	ACTIVAR	81	00080

7.4.5.- Detener Bomba

En el caso de que el operador desee detener la bomba, se estableció una unidad en sistema HMI para detener la bomba de agua cuando este en uso, la variable de tipo salida de digital que el esclavo debe acceder cuando el maestro se lo solicite. El elemento de la variable para activar la bomba que en este caso corresponde a “B2”, se utiliza asignando prefijo el 0 en el inicio de mapeo de dirección, lo cual indica que el maestro solicita al esclavo acceder a la salida digital del elemento mediante la dirección de la variable definida anteriormente, esa variable ocupa un espacio de memoria y una dirección. Debido a esto la respuesta del esclavo es 1 bit para activar la bomba, A continuación, en la siguiente tabla se muestra el mapeo de dirección de Modbus la función de detener la bomba.

TABLA 19 Control maestro/esclavo ‘Detener Bomba’

ELEMENTO	FUNCIÓN	MEMORIA	DIRECCIÓN
BOMBA	APAGAR	83	00082

7.5.- Nivel de Líquido

Se estableció una unidad en el sistema HMI para la detección de niveles de líquido del estanque de “Clean Optimizer”, los sensores utilizados son sensores capacitivos diseñados para detectar agua a través de superficies no metálicas, los cuales pueden detectar líquido sobre una pared de hasta 12mm de grosor. La unidad establecida tiene como hacer un uso adecuado desde el punto de vista de la eficiencia hídrica. El proceso consta realizar la configuración de los parámetros, luego el maestro hacia la solicitud para preguntar al esclavo sobre el nivel de agua, en el caso que la variable SL_MIN tome el valor de 1 y SL_MAX tome el valor de 0, se activara el indicador de nivel mínimo, en el caso que SL_MIN tome el valor de 0 y SL_MAX tome el valor de 1 esto quiere decir que se activara el indicador de nivel máximo.

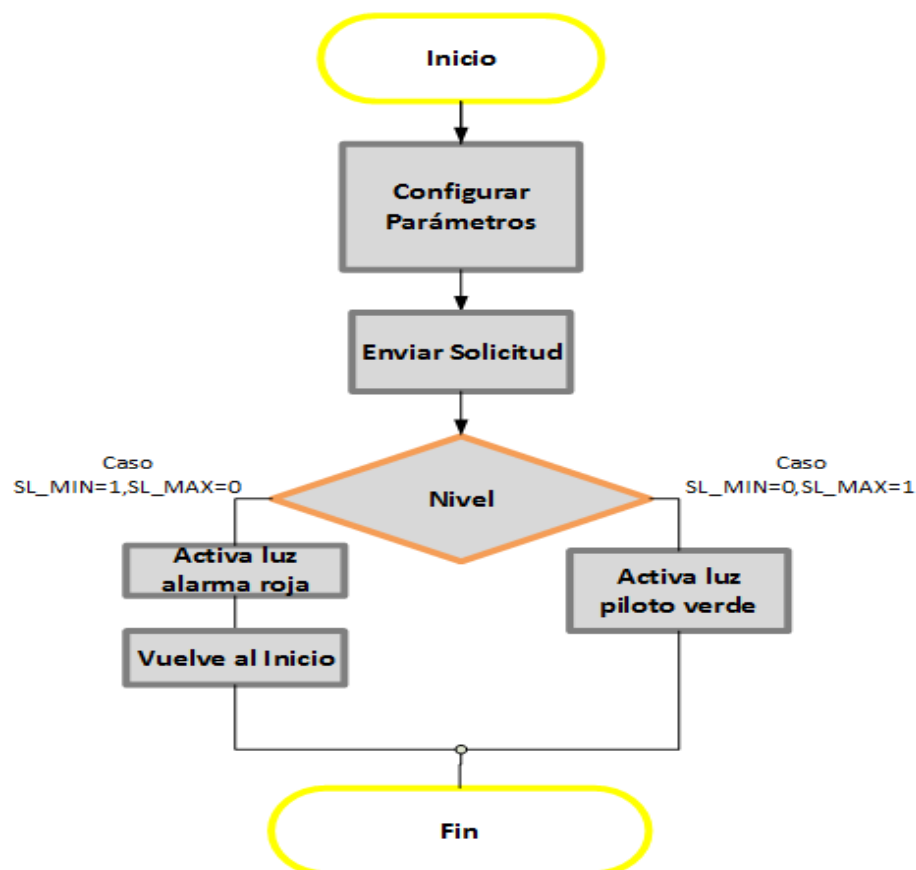
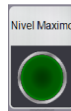


Fig.7.11: Bloque nivel de líquido en sistema SCADA

7.5.1.- Nivel máximo

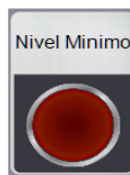
El operador desde el sistema HMI debe monitorear el nivel de líquido máximo, para lo cual el maestro solicita al esclavo acceder a la dirección 00092 del elemento de bloque de memoria de la posición 93 correspondiente a la variable SL_MAX y que luego el esclavo envíe la respuesta que en este caso será de 1 bit, para luego representar esa respuesta del esclavo en la unidad de nivel de líquido del sistema.



ELEMENTO	FUNCIÓN	MEMORIA	DIRECCIÓN
LÍQUIDO	NIVEL	93	00092

7.5.2.- Nivel Mínimo

El operador desde el sistema HMI debe monitorear el nivel de líquido mínimo, para lo cual el maestro solicita al esclavo acceder a la dirección 00090 del elemento de bloque de memoria de la posición 90 correspondiente a la variable SL_MAX y que luego el esclavo envíe la respuesta que en este caso será de 1 bit, para luego representar esa respuesta del esclavo en la unidad de nivel de líquido del sistema.



ELEMENTO	FUNCIÓN	MEMORIA	DIRECCIÓN
LÍQUIDO	NIVEL	91	00090

7.5.3.- Unidad SCADA para sistema de agua.

Se diseñó el modo en como el operador va a interactuar mediante la interfaz gráfica para monitorear los niveles de líquido, en donde se agregaron alarmas de nivel representadas como luz piloto para nivel máximo, la cual consta una luz de color verde, es este caso donde se indica el tope de recarga de agua. Por otro lado, se agregó una unidad de luz piloto para nivel mínimo de color rojo, lo cual indica que es una variable critica, debido a que, si el operador logra visualizar un nivel mínimo en el sistema HMI, se debería retornar una futura recarga. Un dato a mencionar es que la capacidad máxima de agua es de aproximadamente 8 litros distribuidos en dos de estanques con capacidad para 4 litros de agua.

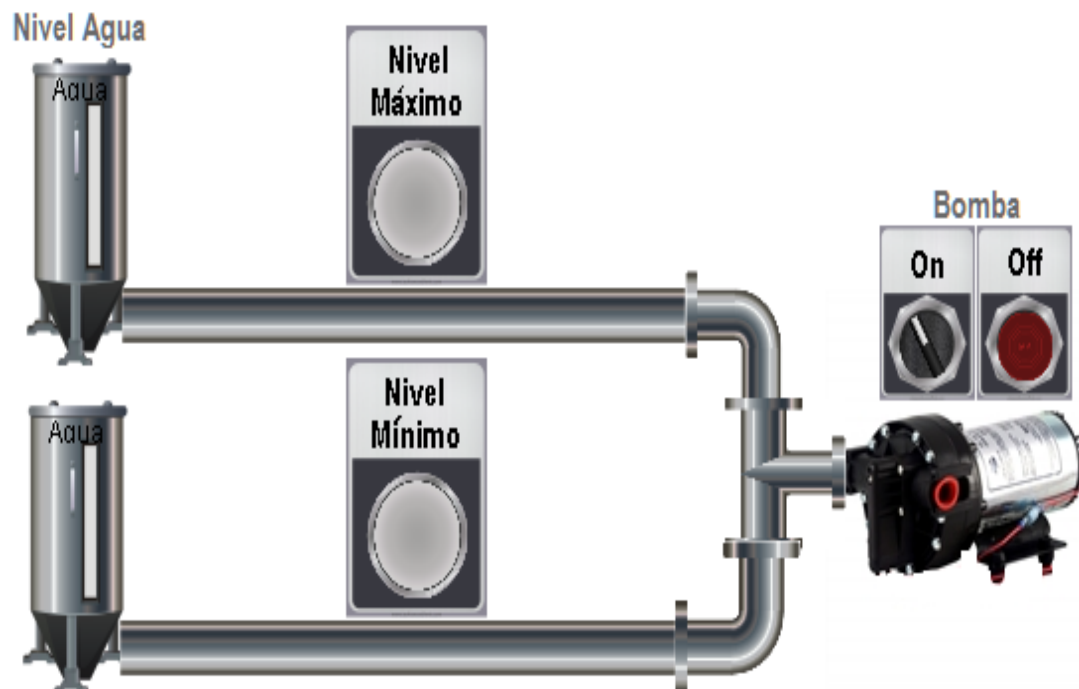


Fig.7.12: Unidad de sistema de agua de sistema SCADA

7.6.- Monitoreo de porcentaje estimado de sombreado por suciedad

Se realizaron pruebas de laboratorio en un ambiente oscuro, debido a que el monitoreo de suciedad se piensa realizar de noche, por lo tanto, se busca expresar el porcentaje de sombreado estimado provocado por efecto de suciedad. Los sensores encargados de las mediciones son de tipo TSL2561, los cuales entregan un valor en unidad de medida “LUX”, en donde luego se realizó de manera empírica la calibración entre los valores máximo y mínimos para lograr un offset que se presentara cómo porcentaje de sombreado estimado por efecto de suciedad.

7.6.1.- *Panel Fotovoltaico sin suciedad*

Se realizaron los algoritmos para cuando los sensores detectan partículas de suciedad en paneles fotovoltaicos y un algoritmo para estimar el porcentaje de sombreado estimado por efecto de partículas de suciedad. Se tomaron como referencia los datos entregados por cuatro sensores, los que luego se representaron como una estimación del porcentaje de sombreado por efecto de suciedad. Donde se obtuvieron resultados de una prueba que se basó en realizar una inspección manual con Clean Optimizer, luego se fijó el prototipo en un sector del panel fotovoltaicos el cual se muestra en la siguiente figura.



Fig.7.13: Escaneo sobre panel fotovoltaico sin suciedad.

En el momento de la medición se accedió a monitorear la información mediante el sistema SCADA para visualizar la información del porcentaje de sombreado estimado por efecto de suciedad en donde los resultados se muestran en la siguiente figura.

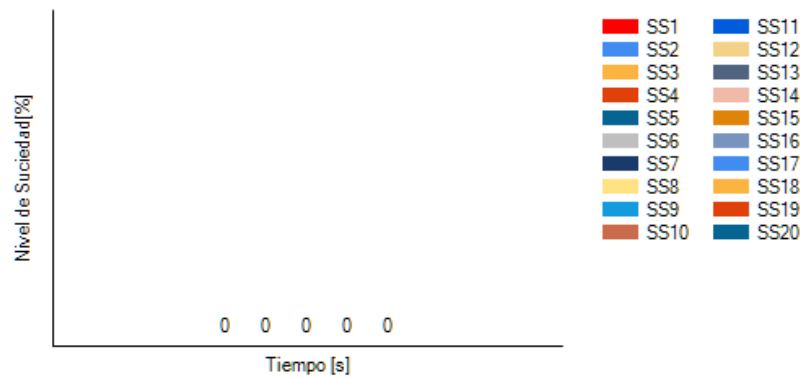


Fig.7.14 Monitoreo de escaneo sobre panel fotovoltaico sin suciedad

7.6.2.- Panel Fotovoltaico con suciedad

En esta unidad se realizó una inspección manual con Clean Optimizer dando la orden desde el sistema SCADA. En este caso se tomó como referencia el sensor SS1(Sensor de suciedad 1). Luego se aplicó bastante suciedad en el sector de medición del sensor SS1 y en los sectores de medición de los otros sensores no se aplicó nada de partículas de suciedad.



Fig.7.15: Escaneo sobre panel fotovoltaico con suciedad localizada

En la prueba se aplicó cantidad de polvo en el sector de medición del sensor SS1. En este caso el nodo remoto envía la información de la medición al nodo central para luego procesar la información del porcentaje de sombreado estimado por efecto de suciedad en un sistema SCADA. Finalmente, los resultados se muestran en la siguiente figura.

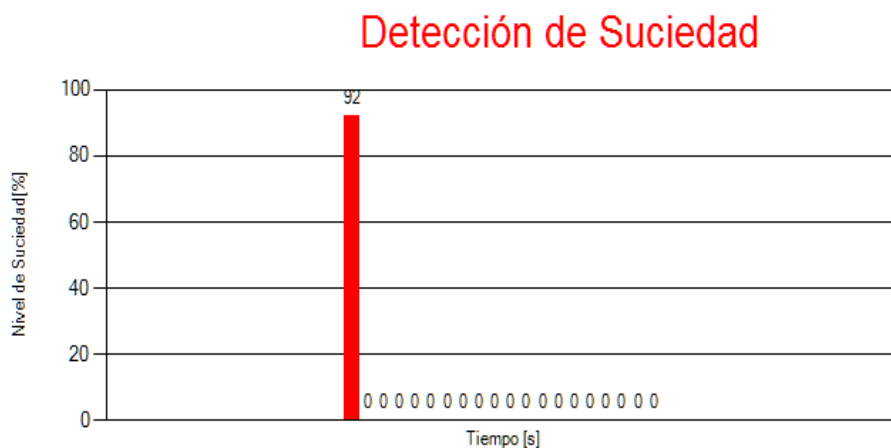


Fig.7.16: Monitoreo sobre panel fotovoltaico con suciedad localizada.

7.6.3.- Panel Fotovoltaico con varios tipos de suciedad

Se realizó una prueba de monitoreo tomando como referencia los tres primeros sensores para la estimación de porcentaje de sombreado por efecto de suciedad, los cuales se representan con las variables SS_1, SS_2 y SS_3. La medición del sensor 1 el cual se ubica a la izquierda de la imagen se tomó sobre una celda del panel en donde no había deposición de suciedad, la medición del sensor 2, se tomó sobre una celda en cual existía suciedad parcial y por último s la medición del sensor 3, la cual se tomó sobre una celda la cual estaba totalmente sucia distribuida de manera uniforme.



Fig.7.17: Escaneo sobre panel fotovoltaico variación de suciedad

Al realizar el monitoreo en el sistema SCADA se encontraron cambios significativos en cada medición y esto se debe al cambio en relación con cómo se depositaban las partículas de polvo. En este caso los resultados muestran que el sensor SS_1 detectó que existe una estimación de porcentaje de sombreado por efecto de suciedad de un valor mínimo de 8%, el sensor SS2 detectó un sobre un 60%, lo que era un valor esperado debido a que la medición de la celda tenía suciedad depositada de manera parcial y no uniforme y el sensor SS3 detectó que el porcentaje de sombreado estimado por efecto de suciedad cubría casi el 100% del sector de medición.

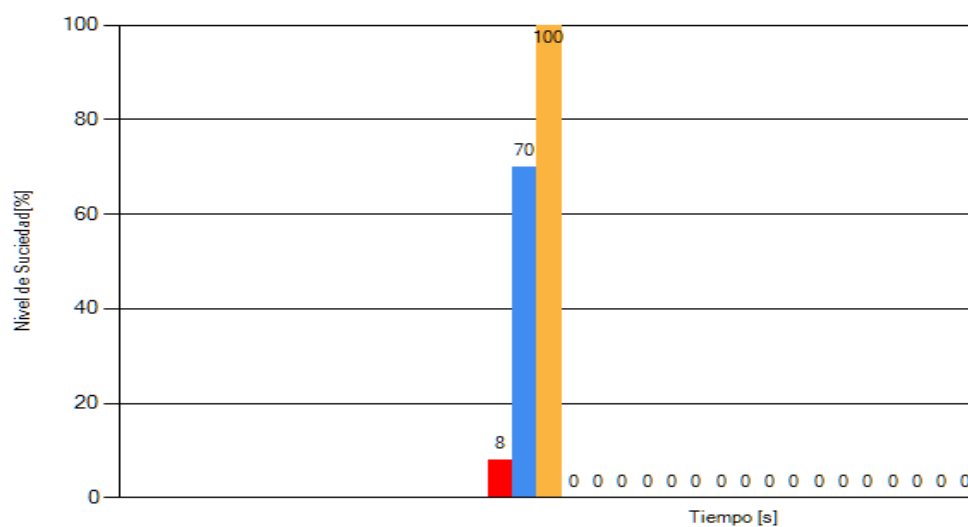


Fig.7.18: Monitoreo de panel fotovoltaico variación de suciedad.

Capítulo 8. Conclusiones

- El trabajo realizado se enfocó en entregar una herramienta para que un operador pueda interactuar con el proceso de limpieza de paneles fotovoltaicos en donde se pueda acceder a los modos de limpieza, bomba de agua y los sensores de nivel de Clean Optimizer, mediante un software de control en que se permite comandar y supervisar el proceso.
- Respecto del trabajo realizado se pueden identificar elementos claves que son críticos al emplear protocolo Modbus como asignar adecuadamente los ID en los equipos (Direcciones únicas), conocer el mapeo de memoria frente a las direcciones propias del protocolo Modbus, elegir la longitud de acuerdo a la información disponible habitualmente en el esclavo.
- Se comprobó la funcionalidad de placa microcontroladora Arduino R3 como tarjeta de adquisición de datos, a través de la supervisión de las señales del proceso que lleva a cabo Clean Optimizer las cuales fueron tomadas por los sensores.
- El Módulo XBee pro900HP-S3B se convirtió en un elemento primordial para el sistema de comunicación inalámbrica debido a que sus características de bajo consumo, rango alto de alcance y tasa de transferencia, las cuales cumplieron con los requerimientos de Clean Optimizer.
- La transmisión de datos de forma inalámbrica se puede ver afectado por varios factores cómo las sinuosidades del terreno o componentes que la conforman. Esto dificulta la visión directa entre el nodo implementado en el robo y nodo coordinador de la red inalámbrica.

- El trabajo realizado se enfocó en entregar una herramienta para que un operador pueda interactuar con el proceso de limpieza de paneles fotovoltaicos en donde se pueda acceder a los modos de limpieza, bomba de agua y los sensores de nivel de Clean Optimizer, mediante un software de control en que se permite comandar y supervisar el proceso.
- El desarrollo de esta memoria de título cumple a cabalidad el objetivo establecer un sistema de comunicación inalámbrica mediante una red punto a punto, se puede monitorear y manipular la información de los elementos utilizados en la limpieza de paneles fotovoltaicos en tiempo real, gracias al sistema SCADA.

8.1.- Trabajos Futuros

La memoria de título presentada correspondiente al desarrollo de sistema de comunicación inalámbrica y sistema SCADA, corresponde solo al prototipo inicial, por lo que lo presentado en esta memoria de título servirá como base para nuevos desarrollos e implementación para el prototipo inicial de Clean Optimizer. A continuación, se muestran los trabajos futuros que se podrían realizar para mejorar o complementar lo descrito en esta memoria de título.

- Desarrollar sistema SCADA para varios prototipos Clean Optimizer para manipular y acceder a los parámetros de cada robot, los cuales inciden en limpieza de paneles fotovoltaicos con el objetivo de lograr una estrategia de limpieza más eficaz y confiable.
- Implementación de sistema de base de datos para enviar datos almacenados a un nodo central, celular o subirlo a una web, mediante tecnología IoT o alguna tecnología que presenten integración con los microcontroladores utilizados.
- Integración de tecnología IoT para desarrollar un sistema que genere una ventaja competitiva sobre los prototipos existentes, crean un sistema con un mayor grado de robustez y confiabilidad.

Capítulo 9. Bibliografía

- [1] C. Santana, “Potencial Eólico, solar e hidroeléctrico de Arica a Chiloé”
<http://www.4eChile.cl/4eChile/wp-content/uploads/2017/03/Energias-Renovables-en-Chile-El-potencial-eolico-solar-e-hidroele%CC%81ctrico-de-Arica-a-Chiloe.pdf>
- [2] J. Prieto. (s.f). “Introducción a los sistemas de comunicación inalámbricos”.
Universidad Obterta de Caralunya.
[https://www.exabyteinformatica.com/uoc/Informatica/Tecnologia_y_desarrollo_en_dispositivos_moviles/Tecnologia_y_desarrollo_en_dispositivos_moviles_\(Intro\).pdf](https://www.exabyteinformatica.com/uoc/Informatica/Tecnologia_y_desarrollo_en_dispositivos_moviles/Tecnologia_y_desarrollo_en_dispositivos_moviles_(Intro).pdf)
- [3] Aravind, G., Gautham, V., Kumar, T., & Naresh, B. (2014). A Control Strategy for an Autonomous Robotic Vacuum Cleaner for Solar Panels.
- [4] M. Flores, “Desarrollo de un sistema inalámbrico para monitoreo de humedad de suelo en diversos puntos de un viñedo” (Memoria para optar al Título de Ingeniería en Mecatrónica).
Universidad de Talca, 2014.
- [5] M. Tarhan, F. De la Cruz, M. Diaz-Granados "*Diseño de Interfaz Gráfica para Implementación de tecnología Zigbee en una Red Modbus para Aplicaciones Inalámbricas en Automatización Industrial*", Universidad Nacional Experimental Politécnica 'Antonio José de Sucre', febrero 2015.
- [6] Ferreira " Sistemas de Control e Supervisão de Sistemas Embebidos-Tipo SCADA"
Universidade do porto, Faculdade de Engenharia Porto, marzo 2008

[7] M. Ramírez, “Desarrollo de una red inalámbrica auto configurable y de bajo consumo utilizando protocolo ZigBee” (Memoria para optar al Título de Ingeniería civil en computación). Universidad de Talca, 2007.

[8] A. Girod, “Desarrollo e implementación de una red de sensores ZigBee mediante el dispositivo XBee de Digi” (Memoria para optar al Título de Ingeniería en Automática y Electrónica Industrial), Universidad Rovira, mayo 2012.

[9] C. Jacome, “Diseño e implementación de un prototipo de red inalámbrica tipo mesh, para el monitoreo y control de riegos en una amplia gama de sectores y cultivos” (Memoria para optar al Título de Ingeniería en electrónica e instrumentación), Escuela Politécnica del Ejercito, Ecuador, Latacunga, mayo 2013.

[10] M. Contreras, “Diseño e implementación de un sistema de interconexión dinámica y monitoreo de sensores utilizando dispositivos compactos de arquitectura modular” (Memoria para optar al Título de Ingeniería de Ingeniero Civil Electricista), Universidad de Chile, Santiago, marzo 2011.

[11] J. Rodríguez F. Robles “Implementación de un sistema SCADA para el control y monitoreo de 7 ascensores y 8 escaleras eléctricas instaladas en el terminal terrestre” de (Memoria para optar al Título de Ingeniero en Electrónica), Universidad Politécnica Salesiana, Ecuador, febrero 2011.

[12] J. Prieto. (s.f). “Introducción a los sistemas de comunicación inalámbricos”. Universidad Oberta de Caralunya.

[https://www.exabyteinformatica.com/uoc/Informatica/Tecnologia_y_desarrollo_en_dispositivos_moviles/Tecnologia_y_desarrollo_en_dispositivos_moviles_\(Intro\).pdf](https://www.exabyteinformatica.com/uoc/Informatica/Tecnologia_y_desarrollo_en_dispositivos_moviles/Tecnologia_y_desarrollo_en_dispositivos_moviles_(Intro).pdf)

[13] J. Somoza N. Galan N. Mejias E. Orozco "Diseño de Interfaz Gráfica para Implementación de Sistema SCADA en la Supervisión y Adquisición de datos en Módulos Fotovoltaicos” Universidad Politécnica de Sinaloa, Septiembre.2017

- [14] C. Romero “Mando de control para Robot limpiador de paneles solares Clean Optimizer mediante aplicación móvil”, Universidad de Talca, diciembre 2017.
- [15] C. Santana, “Potencial Eólico, solar e hidroeléctrico de Arica a Chiloé”
<http://www.4eChile.cl/4eChile/wp-content/uploads/2017/03/Energias-Renovables-en-Chile-El-potencial-eolico-solar-e-hidroele%CC%81ctrico-de-Arica-a-Chiloe.pdf>
- [16] Comparación entre protocolos de comunicación inalámbrica.
<http://www.domoprac.com/protocolos-de-comunicacion-y-sistemas-domoticos/protocolos-de-red-tipos-y-utilidades.html>
- [17] E. Fuentealba, P. Ferrada, F. Araya, A. Marzo, C. Parrado, C. Portillo Photovoltaic performance and LCoE comparison at the coastal zone of the Atacama Desert, Chile Energy Convers Manag, 95 (2015), pp. 181–186
- [18] J. Herrera, M. Barrios, S. Pérez, “Desing and implementation of a wireless SCADA system by means of zigbee and arduino technology”, IEEE, Vol 12, N° 2, pp.65-72, Dec .2014.
- [19] L. Fields, “Comparación de sistemas de comunicación inalámbrica para robots móviles. 4, N.º 1, edición 2013 Universidad Tecnológica de Panamá.
http://www.utp.ac.pa/documentos/2014/pdf/08_Comunicacio_n_inala_mbrica_1.pdf
- [20] G. Echeverry, Y. Cabezas and N. Olarte, “Normatividad para las Frecuencias ISM Orientada hacia Aplicaciones de Dispositivos XBee-802.15.4-Pro,” *ING E CUC*, vol. 10, no. 2, pp. 81–90, 2014. Inge Cuc, Vol. 10, N° 2, pp 81-90, diciembre, 2014
- [21] Solar Panel Cleaning - Automatic Cleaning System: “Heliotex.”
<https://www.solarpanelcleaningsystems.com/>

Capítulo 10. Anexos

10.1.- Código Nodos de Comunicación Inalámbrica

```
void inicializarXBee () {  
  bool txpin = 3;  
  mb. configXBee (9600); //agregar  
  XBee.Begin (9600);  
  mb. setSlaveId(1);  
  modbus. Begin(9600);  
  Serial.begin(9600) ;//agregar  
  mb. addCoil(MBD); //Motor Bomba  
  mb. addCoil(MBP);  
  mb. addCoil(MBI);  
  mb. addCoil(MMI); //Motor Movimiento  
  mb.addCoil(MMP);  
  mb.addCoil(MMD);  
  mb. addCoil(SL_MIN); //Sensor Nivel de Líquido Min  
  mb. addCoil(SL_MAX); //Sensor Nivel de Líquido Max  
  mb. addHreg(SS_1); //Sensores para Suciedad Paneles Fotovoltaicos  
  mb. addHreg(SS_2);  
  mb. addHreg(SS_3);  
  mb. addHreg(SS_4);  
  mb. addHreg(SS_5);  
  mb. addHreg(SS_6);  
  mb. addHreg(SS_7);  
  mb. addHreg(SS_8);  
  mb. addHreg(SS_9);  
  mb. addHreg(SS_10);
```

```
mb. addHreg(SS_11);
mb. addHreg(SS_12);
mb. addHreg(SS_13);
mb. addHreg(SS_14);
mb. addHreg(SS_15);
mb. addHreg(SS_16);
mb. addHreg(SS_17);
mb. addHreg(SS_18);
mb. addHreg(SS_19);
mb. addHreg(SS_20);
}
void XBeeEnviar () {
  mb. Coil(SL_MIN,l_min);//(direccion,valor)
  mb. Coil(SL_MAX,l_max);// nivel de agua
  mb. Hreg(SS_1,sensoresPolvo[0]);
  mb. Hreg(SS_2,sensoresPolvo[1]);
  mb. Hreg(SS_3,sensoresPolvo[2]);
  mb. Hreg(SS_4,sensoresPolvo[3]);
  mb. Hreg(SS_5,sensoresPolvo[4]);
  mb. Hreg(SS_6,sensoresPolvo[5]);
  mb. Hreg(SS_7,sensoresPolvo[6]);
  mb. Hreg(SS_8,sensoresPolvo[7]);
  mb. Hreg(SS_9,sensoresPolvo[8]);
  mb. Hreg(SS_10,sensoresPolvo[9]);
  mb. Hreg(SS_11,sensoresPolvo[10]);
  mb. Hreg(SS_12,sensoresPolvo[11]);
  mb. Hreg(SS_13,sensoresPolvo[12]);
  mb. Hreg(SS_14,sensoresPolvo[13]);
  mb. Hreg(SS_15,sensoresPolvo[14]);
  mb. Hreg(SS_16,sensoresPolvo[15]);
  mb. Hreg(SS_17,sensoresPolvo[16]);
```

```
mb. Hreg(SS_18, sensoresPolvo [17]);
mb. Hreg(SS_19, sensoresPolvo [18]);
mb. Hreg(SS_20, sensoresPolvo [19]);

}

void XBeeRecibido () {

if (mb. Coil(MMI)) {
    Modo = 'M';
    Sentido = 'L';
}

else if (mb.Coil(MMP)) {
    Modo = 'M';
    Sentido = 'S';
}

else if (mb. Coil(MMD)) {
    Modo = 'M';
    Sentido = 'R';
}
```



```

void loop( ) {
  mb.task();
  XBeeEnviar();
  XBeeRecibido ();
  sensores ();
  enviar(true);
  while (Recibido ( ) {}
  // Serial.print("processing I2C channels...");
  switch (canal) {
    case 0:
      tca9546a.selectChannel(0);
      configureSensor();
      sensors_event_t event_29_0;
      tsl_29_0.getEvent(&event_29_0)
      sensors_event_t event_49_0;
      tsl_49_0.getEvent(&event_49_0);
      sensoresPolvo[0] = event_29_0.light - offset[0];
      sensoresPolvo [0] =constrain (sensoresPolvo [0], minSensores [0], maxSensores [0]);
      sensoresPolvo [0]=map (sensoresPolvo [0], minSensores [0], maxSensores [0],0,100);
      sensoresPolvo [0] =constrain (sensoresPolvo [0], 0, 100);
      sensoresPolvo [1] = event_49_0. light - offset [1];
      sensoresPolvo [1] =constrain (sensoresPolvo [1], minSensores [1], maxSensores [1]);
      sensoresPolvo [1]=map (sensoresPolvo [1], minSensores [1], maxSensores [1],0,100);
      sensoresPolvo [1]=constrain (sensoresPolvo [1], 0, 100);
      if (calibracion == 1) {
        if (contCal < ventana) {
          offset[0] += event_29_0.light * 1 / ventana;

```

```

    offset[1] += event_49_0.light * 1 / ventana;
  }
}
break;
case 1:
tca9546a.selectChannel(1);
configureSensor();
sensors_event_t event_39_1;
tsl_39_0.getEvent(&event_39_1);
sensors_event_t event_29_1;
tsl_29_0.getEvent(&event_29_1);
sensors_event_t event_49_1;
tsl_49_0.getEvent(&event_49_1);
sensoresPolvo[2] = event_29_1.light - offset[2];
sensoresPolvo [2] =constrain (sensoresPolvo [2], minSensores [2], maxSensores [2]);
sensoresPolvo [2]=map (sensoresPolvo [2], minSensores [2], maxSensores [2],0,100);
sensoresPolvo [2] =constrain (sensoresPolvo [2], 0, 100);
sensoresPolvo [3] = event_39_1. light - offset [3];
sensoresPolvo [3] =constrain (sensoresPolvo [3], minSensores [3], maxSensores [3]);
sensoresPolvo [3]=map (sensoresPolvo [3], minSensores [3], maxSensores [3],0,100);
sensoresPolvo [3] =constrain (sensoresPolvo [3], 0, 100);
sensoresPolvo [4] = event_49_1. light - offset [4];
sensoresPolvo [4] =constrain (sensoresPolvo [4], minSensores [4], maxSensores [4]);
sensoresPolvo [4]=map (sensoresPolvo [4], minSensores [4], maxSensores [4],0,100);
sensoresPolvo [4] =constrain (sensoresPolvo [4], 0, 100);
if (calibracion == 1) {
  if (contCal < ventana) {
    offset[2] += event_29_1.light * 1 / ventana;
    offset[3] += event_39_1.light * 1 / ventana;
    offset[4] += event_49_1.light * 1 / ventana;

```

```

    }
}
break;
case 2:
    tca9546a.selectChannel(2);
    configureSensor();
    sensors_event_t event_39_2;
    tsl_39_0.getEvent(&event_39_2);
    sensors_event_t event_29_2;
    tsl_29_0.getEvent(&event_29_2);
    sensors_event_t event_49_2;
    tsl_49_0.getEvent(&event_49_2);
    sensoresPolvo [5] = event_29_2. light - offset [5];
    sensoresPolvo [5] =constrain (sensoresPolvo [5], minSensores [5], maxSensores [5]);
    sensoresPolvo [5]=map (sensoresPolvo [5], minSensores [5], maxSensores [5],0,100);
    sensoresPolvo [5] =constrain (sensoresPolvo [5], 0, 100);
    sensoresPolvo [6] = event_39_2. light - offset [6];
    sensoresPolvo [6] =constrain (sensoresPolvo [6], minSensores [6], maxSensores [6]);
    sensoresPolvo [6]=map (sensoresPolvo [6], minSensores [6], maxSensores [6],0,100);
    sensoresPolvo [6] =constrain (sensoresPolvo [6], 0, 100);
    sensoresPolvo [7] = event_49_2. light - offset [7];
    sensoresPolvo [7] =constrain (sensoresPolvo [7], minSensores [7], maxSensores [7]);
    sensoresPolvo [7]=map (sensoresPolvo [7], minSensores [7], maxSensores [7],0,100);
    sensoresPolvo [7] =constrain (sensoresPolvo [7], 0, 100);
    if (calibracion == 1) {
        if (contCal < ventana) {
            offset[5] += event_29_2.light * 1 / ventana;
            offset[6] += event_39_2.light * 1 / ventana;
            offset[7] += event_49_2.light * 1 / ventana;
        }
    }
}

```

```

}
break;
case 3:
    tca9546a.selectChannel(3);
    configureSensor();
    sensors_event_t event_39_3;
    tsl_39_0.getEvent(&event_39_3);
    sensors_event_t event_29_3;
    tsl_29_0.getEvent(&event_29_3);
    sensors_event_t event_49_3;
    tsl_49_0.getEvent(&event_49_3);
    sensoresPolvo [8] = event_29_3. light - offset [8];
    sensoresPolvo [8] =constrain (sensoresPolvo [8], minSensores [8], maxSensores [8]);
    sensoresPolvo [8]=map (sensoresPolvo [8], minSensores [8], maxSensores [8],0,100);
    sensoresPolvo [8] =constrain (sensoresPolvo [8], 0, 100);
    sensoresPolvo [9] = event_39_3. light - offset [9];
    sensoresPolvo [9] =constrain (sensoresPolvo [9], minSensores [9], maxSensores [9]);
    sensoresPolvo [9]=map (sensoresPolvo [9], minSensores [9], maxSensores [9],0,100);
    sensoresPolvo [9] =constrain (sensoresPolvo [9], 0, 100);
    sensoresPolvo [10] = event_49_3. light - offset [10];
    sensoresPolvo [10] =constrain (sensoresPolvo [10], minSensores [10], maxSensores [10]);
    sensoresPolvo [10]=map (sensoresPolvo [10], minSensores [10], maxSensores
[10],0,100);
    sensoresPolvo [10] =constrain (sensoresPolvo [10], 0, 100);
    if (calibracion == 1) {
        if (contCal < ventana) {
            offset[8] += event_29_3.light * 1 / ventana;
            offset[9] += event_39_3.light * 1 / ventana;
            offset[10] += event_49_3.light * 1 / ventana;
        }
    }
    break;

```

case 4:

```

tca9546a.selectChannel(4);
configureSensor();
sensors_event_t event_39_4;
tsl_39_0.getEvent(&event_39_4);
sensors_event_t event_29_4;
tsl_29_0.getEvent(&event_29_4);
sensors_event_t event_49_4;
tsl_49_0.getEvent(&event_49_4);
sensoresPolvo [11] = event_29_4. light - offset [11];
sensoresPolvo [11] =constrain (sensoresPolvo [11], minSensores [11], maxSensores [11]);
sensoresPolvo [11]=map (sensoresPolvo [11], minSensores [11], maxSensores
[11],0,100);
sensoresPolvo [11] =constrain (sensoresPolvo [11], 0, 100);
sensoresPolvo [12] = event_39_4. light - offset [12];
sensoresPolvo [12] =constrain (sensoresPolvo [12], minSensores [12], maxSensores [12]);
sensoresPolvo [12]=map (sensoresPolvo [12], minSensores [12], maxSensores
[12],0,100);
sensoresPolvo [12] =constrain (sensoresPolvo [12], 0, 100);
sensoresPolvo [13] = event_49_4. light - offset [13];
sensoresPolvo [13] =constrain (sensoresPolvo [13], minSensores [13], maxSensores [13]);
sensoresPolvo [13]=map (sensoresPolvo [13], minSensores [13], maxSensores
[13],0,100);
sensoresPolvo [13] =constrain (sensoresPolvo [13], 0, 100);
if (calibracion == 1) {
    if (contCal < ventana) {
        offset[11] += event_29_4.light * 1 / ventana;
        offset[12] += event_39_4.light * 1 / ventana;
        offset[13] += event_49_4.light * 1 / ventana;
    }
}

```

```

    }
}
break;
case 5:
    tca9546a.selectChannel(5);
    configureSensor();
    sensors_event_t event_39_5;
    tsl_39_0.getEvent(&event_39_5);
    sensors_event_t event_29_5;
    tsl_29_0.getEvent(&event_29_5);
    sensors_event_t event_49_5;
    tsl_49_0.getEvent(&event_49_5);
    sensoresPolvo [14] = event_29_5. light - offset [14];
    sensoresPolvo [14] =constrain (sensoresPolvo [14], minSensores [14], maxSensores [14]);
    sensoresPolvo [14]=map (sensoresPolvo [14], minSensores [14], maxSensores
[14],0,100);
    sensoresPolvo [14] = constrain (sensoresPolvo [14], 0, 100);
    sensoresPolvo [15] = event_39_5. light - offset [15];
    sensoresPolvo [15] =constrain (sensoresPolvo [15], minSensores [15], maxSensores [15]);
    sensoresPolvo [15]=map (sensoresPolvo [15], minSensores [15], maxSensores
[15],0,100);
    sensoresPolvo [15] =constrain (sensoresPolvo [15], 0, 100);
    sensoresPolvo [16] = event_49_5. light - offset [16];
    sensoresPolvo [16] =constrain (sensoresPolvo [16], minSensores [16], maxSensores [16]);
    sensoresPolvo [16]=map (sensoresPolvo [16], minSensores [16], maxSensores
[16],0,100);
    sensoresPolvo [16] =constrain (sensoresPolvo [16], 0, 100);
    if (calibracion == 1) {
        if (contCal < ventana) {
            offset[14] += event_29_5.light * 1 / ventana;
            offset[15] += event_39_5.light * 1 / ventana;

```

```

    offset[16] += event_49_5.light * 1 / ventana;

    }
}
break;

case 6:
    tca9546a.selectChannel(6);
    configureSensor();
    sensors_event_t event_39_6;
    tsl_39_0.getEvent(&event_39_6);
    sensors_event_t event_29_6;
    tsl_29_0.getEvent(&event_29_6);
    sensors_event_t event_49_6;
    tsl_49_0.getEvent(&event_49_6);
    sensoresPolvo [17] = event_29_6. light - offset [17];
    sensoresPolvo [17] =constrain (sensoresPolvo [17], minSensores [17], maxSensores [17]);
    sensoresPolvo [17]=map (sensoresPolvo [17], minSensores [17], maxSensores
[17],0,100);
    sensoresPolvo [17] =constrain (sensoresPolvo [17], 0, 100);
    sensoresPolvo [18] = event_39_6. light - offset [18];
    sensoresPolvo [18] =constrain (sensoresPolvo [18], minSensores [18], maxSensores [18]);
    sensoresPolvo [18]=map (sensoresPolvo [18], minSensores [18], maxSensores
[18],0,100);
    sensoresPolvo [18] =constrain (sensoresPolvo [18], 0, 100);
    sensoresPolvo [19] = event_49_6. light - offset [19];
    sensoresPolvo [19] =constrain (sensoresPolvo [19], minSensores [19], maxSensores [19]);
    sensoresPolvo [19]=map (sensoresPolvo [19], minSensores [19], maxSensores
[19],0,100);
    sensoresPolvo[19] =constrain(sensoresPolvo[19], 0, 100);
    // Serial.print(tca9546a.getChannel());

```

```
if (calibracion == 1) {
  if (contCal < ventana) {
    offset[17] += event_29_6.light * 1 / ventana;
    offset[18] += event_39_6.light * 1 / ventana;
    offset[19] += event_49_6.light * 1 / ventana;
  }
}
break;
case 7:
  break;
}
if (canal >= 0 && canal <= 6) {
  canal++;
} else {
  if (calibracion == 1) {
    if (contCal < ventana) {
      contCal++;
    }
  }
}
canal = 0;
int i;
for(i=0;i<20;i++){
#ifdef Debug
  debug.print(sensoresPolvo[i]);
  debug.print("\t");
#endif
}
```


10.1.- Código Modbus Serial

```

#include "ModbusSerial.h"
SoftwareSerial XBee (10, 11);
ModbusSerial: ModbusSerial () {

}

bool ModbusSerial:setSlaveId(byte slaveId){
    _slaveId = slaveId;
    return true;
}

byte ModbusSerial:getSlaveId() {
    return _slaveId;
}

#ifdef USE_SOFTWARE_SERIAL
bool ModbusSerial:configXBee( long baud, int txPin) {
    this->_txPin = txPin;

    XBee.begin(baud);

    delay(2000);

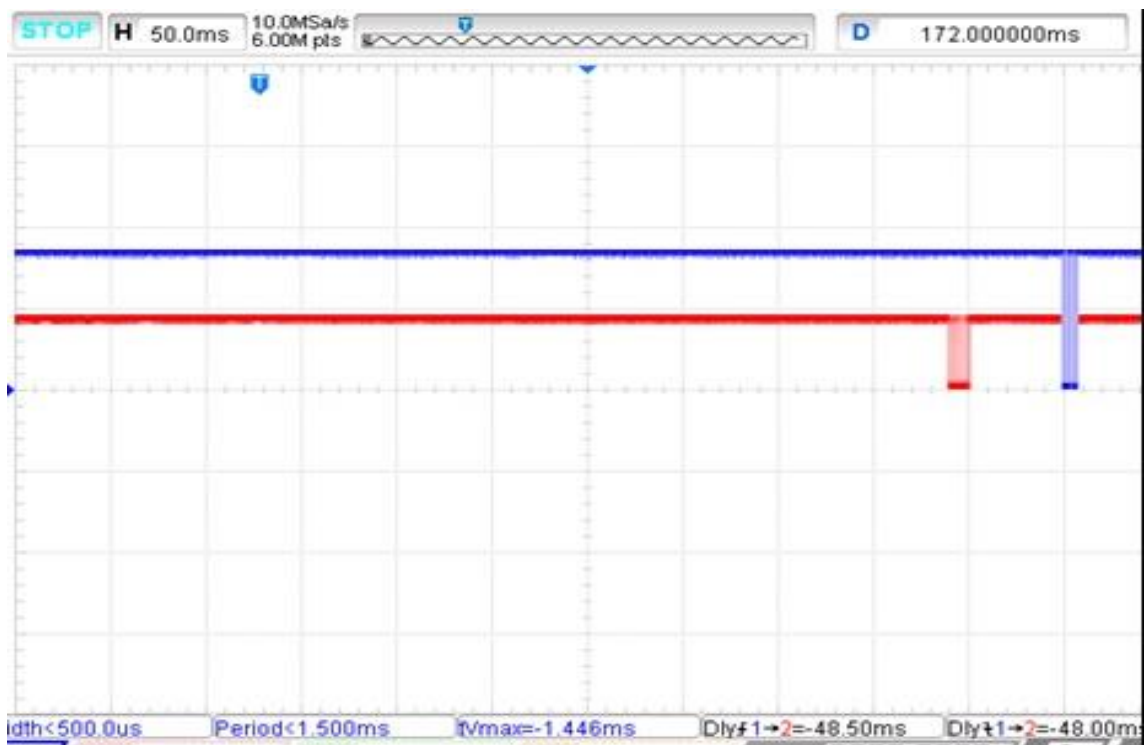
    if (txPin >= 0) {
        pinMode(txPin, OUTPUT);
        digitalWrite(txPin, LOW);
    }

    if (baud > 19200) {
        _t15 = 750;
        _t35 = 1750;
    } else {
        _t15 = 15000000/baud; // 1T * 1.5 = T1.5
        _t35 = 35000000/baud; // 1T * 3.5 = T3.5
    }
    return true;
}
#endif

```

10.2.- Latencia

Las pruebas de latencia se realizaron utilizando un Osciloscopio, en el cual se graficó la señal enviada desde el nodo central, en donde un terminal del osciloscopio se conectó a la salida 17(Cable Rojo) y GND(Debug) del microcontrolador ATmega 328p correspondiente al pin DOUT del nodo remoto del Robot “Clean Optimizer”. Después se utilizó otra terminal del osciloscopio, la cual se conectó al pin DIN(Rojo) y pin GND(Negro) del nodo central de la comunicación inalámbrica.



La grafica de color azul representa la señal que es enviada desde el sistema SCADA hacia “Clean Optimizer”. La grafica de color rojo representa la respuesta de “Clean Optimizer” en base a que recibió la señal desde el sistema SCADA. Finalmente, La latencia de comunicación del sistema entrego un resultado de aproximadamente 48 ms. Lo que se considera aceptable para los requerimientos de Clean Optimizer.

