BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA.

FACULTAD DE CIENCIAS DE LA ELECTRÓNICA.

LIC. EN CIENCIAS DE LA ELECTRÓNICA.

DESARROLLO E IMPLEMENTACIÓN DE REDES INDUSTRIALES UTILIZANDO EL PROTOCOLO MODBUS PARA LA TRANSFERENCIA DE DATOS DE CONTROL DE UN VFD ENTRE DOS PLCS.

Tesis profesional presentada por:

Oscar Amado Ávila Tecocoatzi

Para obtener el título de:

Licenciado en Ciencias de la Electrónica.

Asesor:
M.C. Marco Antonio Sobrevilla González

Puebla, Pue., junio 2021

Dedicatoria.

Estoy agradecido con Dios padre celestial por darme fortaleza, la inteligencia y los ánimos todos los días para seguir adelante. Le agradezco que me haya dado la oportunidad de tener una profesión tan maravillosa y fascinante como lo es la electrónica.

Agradecimientos.

Hay una fuerza motriz más poderosa
que el vapor, la electricidad y
la energía atómica: la voluntad.
Albert Einstein.

Agradezco a mis papás, Isabel Díaz y Amado Ávila por apoyarme en todo momento, comprenderme en los tiempos difíciles durante mis años de estudio, por darme la motivación de salir adelante para ser mejor persona cada día. Así mismo, por darme consejos y tenerme mucha paciencia.

Gracias al maestro Marco quien aceptó ser mi asesor de tesis, así como; su tiempo, su conocimiento en el área de la automatización.

También quiero agradecer a las personar que estuvieron allí en algún momento para escucharme, darme consejos y ánimos.

Finalmente quiero dar las gracias a la universidad, particularmente a la facultad de electrónica por los recursos, las instalaciones y los profesores porque todos formaron parte de mi desarrollo profesional.

Resumen

La tesis presentada a continuación es el desarrollo, programación e implementación de redes industriales de comunicación utilizando el protocolo MODBUS particularmente utilizando dos de sus versiones TCP y RTU considerando el ethernet con la comunicación RS-485. Esto para intercomunicar dispositivos de control al mismo tiempo comandar un actuador o varios actuadores, bajo la metodología de comunicación, la cual se encuentra presente en todo sector industrial.

Más que una memoria, con este texto se pretende dar a conocer las maneras de comunicar las diferentes configuraciones del protocolo para la transferencia de datos de control, el cual es libre, eficiente por si fuera poco uno de los más utilizados en el sector industrial. Como se menciona antes, en este trabajo se plantea el protocolo MODBUS del cual se da a conocer su, configuración, aplicaciones y ventajas de comunicación industrial.

Al respecto conviene decir que, para implementar las redes industriales se requiere de los dispositivos de control industrial, tales como PLCs, HMIs, variadores de frecuencia con sus respectivos módulos de entradas y salidas, tarjetas de comunicación incluyendo el puerto de comunicación DB9, el cual sirve de medio para comunicarse con otros equipos. Así mismo, se requiere de un switch para conectar los diferentes dispositivos. los cuales, formaran parte de una misma red. Todos los dispositivos deberán tener integrado uno o dos puertos Ethernet de tipo RJ-45, esto, por un lado, por otro lado, se utiliza la plataforma TIA PORTAL V14 para la configuración y programación de los equipos.

Cabe mencionar que la plataforma antes mencionada, es una herramienta indiscutiblemente muy poderosa, también utilizada en cualquier área de la automatización, ya que integra toda su paquetería, dicho de otro modo, se puede programar diferentes familias de PLCs, HMIs, y otros dispositivos más de la marca Siemens.

Finalmente, con la intención de conocer la programación avanzada mediante el desarrollo de este trabajo, es adquirir las bases para el desarrollo profesional que se requiere en la 4ª revolución industrial, por lo cual, es importante como ingeniero tener conocimientos sobre redes industriales.

Palabras clave: *Modbus TCP, Modbus RTU, RS485, PLC, HMI, VFD, TIA PORTAl, switch TP LINK, Know How Protection.*

Índice general

Dedicatoria. III
Agradecimientos. IV
Resumen V

Índice general VI

Índice de figuras X

Índice de tablas XIII

Índice de ecuaciones XIV

1. Introducción 1
1.1. Introducción 1
1.2. Objetivo general 2
1.3. Objetivos particulares. 2
1.4. Alcances 3
1.5. Delimitación 3
1.5.1. Temporal 3
1.5.2. Espacial 3
1.5.3. Académica 3
1.6. Justificación 4
1.7. Metodología 5
1.7.1. Métodos 5
1.7.1.1. Método Inductivo 5

1.7.1.2. Método Teórico y sistemático 5
2. Marco teórico 6
2.1. Normas físicas 6
2.1.1. Norma física RS-232 6
2.1.2. Norma física RS-422 8
2.1.3. Norma física RS-485 8
2.1.4. TTL 11
2.2. Topología de redes 11
2.2.1. Punto a punto 11
2.2.2. Bus 12
2.2.3. Árbol. 13
2.2.4. Anillo 14
2.2.5 Estrella 15
2.4. Concepto de protocolo de comunicación 16
2.5. Concepto de bus 16
2.6. Formas de organización de nodos. 17
2.6.1. Maestro – Esclavo 17
2.6.2 Cliente - Servidor. 17
2.6.3. Productor – Consumidor 18
2.7. Modelo OSI 18
2.7.1. Capa de aplicación. 19
2.7.2. Capa de presentación. 20
2.7.3. Capa de sesión 20
2.7.4. Capa de transporte 20
2.7.5. Capa de red 20
2.7.6. Capa de enlace de datos 20
2.7.7. Capa física 20
2.8. MODBUS 22
2.8.1. Orígenes 22
2.8.2. TCP/IP 23
2.8.3. RTU 25
2.8.4. ASCII 25
2.8.5. Beneficios de utilizar el protocolo MODBUS RTU/TCP. 25

2.9. Variador de frecuencia. 26
2.9.1. Funcionamiento. 26
2.9.2. Tipos de variadores de frecuencia 27
2.9.3. Ventajas de utilizar un variador de frecuencia. 28
2.10. PLC 28
2.10.1. Concepto de PLC 28
2.10.2. Historia del PLC. 29
2.10.3. Clasificación de los PLCs. 30
2.10.3.1. Tipo nano 30
2.10.3.2. Tipo compacto 30
2.10.3.3. Tipo modular 32
2.10.3.4. Tipo montaje en rack 34
2.10.3.5. Tipo PLC con HMI integrado 35
2.10.4. Estructura del PLC 36
2.10.4.1. La CPU 36
2.10.4.2. Memorias: 36
2.10.4.3. Interface de entrada. 37
2.10.4.4. Interface de salida. 37
2.10.4.5. Interface de periféricos. 37
2.10.4.6. La fuente de alimentación. 37
2.10.4.7. Módulos de entradas y salidas. 37
2.10.5. Funciones de un PLC 38
2.10.6. Ventajas de los PLCs 39
2.10.7. Inconvenientes de los PLCs 40
2.11. Plataforma de programación 40
2.11.1. TIA PORTAL V14 40
2.11.2. Protección Know-How de bloques FC y FB 42
3. Metodología de programación y desarrollo. 43
3.1. Implementación de la red de comunicación utilizando el protocolo Modbus TCP. 43
3.1.1 Conexión de los equipos industriales. 44
3.1.2 Configuración software de los dispositivos. 45
3.1.3 Programación del PLC-CLIENTE. 46
3.1.4 Programación del PLC-SERVIDOR. 50

3.1.5 Programación del HMI. 54
3.2. Implementación de la red de comunicación utilizando el protocolo Modbus RTU. 58
3.2.1. Configuración software de los dispositivos. 59
3.2.2. Códigos de funciones MODBUS. 60
3.2.3. Comandos lógicos. 61
3.2.4. Referencia de velocidad. 62
3.2.5. Estado lógico 63
3.2.6. Lectura de frecuencia del VFD. 64
3.2.7. Programación del PLC-Master. 64
3.3. Integración de las redes de comunicación MODBUS TCP y RTU. 74
3.4. Programación del HMI. 76
4. Resultados y conclusiones 84
4.1. Conclusiones metodología. 84
4.1.1. MODBUS TCP 84
4.1.2. MODBUS RTU. 85
4.1.3. MODBUS TCP- RTU. 85
4.2. Programación PLC 86
4.3. Programación HMI 86
4.4. Programación VFD 87
4.5. Opciones de mejora 87
4.6. Conclusión general. 87

Tabla de costos. 89
Trabajos futuros. 90
Bibliografía. 91
Apéndice. 94

Índice de figuras

Figura 2.1. Conexión RS232. 7
Figura 2.2. Principio de funcionamiento de señales balanceadas. 8
Figura 2.3. Conexión multipunto. 9
Figura 2.4. Interferencia en señal diferencial. 11
Figura 2.5. Estructura de red en forma de punto a punto. 12
Figura 2.6. Estructura de red en forma de bus 13
Figura 2.7. Estructura de red en forma de árbol. 14
Figura 2.8. Estructura de red en forma de anillo. 15
Figura 2.9. Estructura de red en forma de estrella. 16
Figura 2.10. Modelo OSI. 19
Figura 2.11. Etapa de rectificación. 26
Figura 2.12. Etapa de filtrado. 27
Figura 2.13. Etapa de inversión. 27
Figura 2.14. PLC tipo nano. 30
Figura 2.16. PLC tipo compacto marca SIEMENS. 32
Figura 2.17. PLC tipo modular. 33
Figura 2.18. PLC tipo modular. 33
Figura 2.19. PLC tipo montaje en rack. 34
Figura 2.20. PLC tipo montaje en rack. 34
Figura 2.21. PLC + HMI. 35
Figura 2.22. TIA PORTAL V14 41
Figura 3.1. Diagrama a bloques de la conexión de los equipos para la red MODBUS TCP. 44

Figura 3.2. Conexión eléctrica de los dispositivos industriales. 45
Figura 3.3. Vista de redes. 45
Figura 3.4. Vista de redes/ IPs. 46
Figura 3.5. Instrucción MB_CLIENT. 47
Figura 3.6. Data block para configuración de la variable CONNECT. 49
Figura 3.7. Bloque de datos para escritura. 50
Figura 3.8. Instrucción MB_SERVER 51
Figura 3.9. Bloque de datos para lectura. 53
Figura 3.10. Activación de la protección utilizando KNOW HOW. 54
Figura 3.11. HMI Pantalla 1: Presentación. 55
Figura 3.12. HMI Pantalla 2: Activación de la Comunicación MODBUS TCP. 56
Figura 3.13. HMI Pantalla 3: Datos a transferir al PLC-SERVIDOR. 57
Figura 3.14. HMI Pantalla 4: Datos recibidos en el PLC-SERVIDOR 58
Figura 3.15. Diagrama a bloques, red MODBUS RTU. 59
Figura 3.16. Variador de frecuencia Powerflex 4. 63
Figura 3.17. Vista de redes. 65
Figura 3.18. TIA PORTAL: vista de dispositivos. 65
Figura 3.19. Bloque "MB_COMM_LOAD" 66
Figura 3.20. Bloque MB_MASTER 67
Figura 3.21. Instrucción para la orden de STOP 68
Figura 3.22. Instrucción para START. 68
Figura 3.23. Instrucción: cambio de giro a la derecha. 69
Figura 3.24. Instrucción: cambio de giro a la izquierda. 69
Figura 3.25. Lectura del código de error del variador. 70
Figura 3.26. Instrucción: tiempo de aceleración. 71
Figura 3.27. Instrucción: tiempo de desaceleración. 72
Figura 3.28. Instrucción: Lectura de la frecuencia de salida. 73
Figura 3.29. Instrucción: Clear Faults. 73
Figura 3.30. Instrucción: Logic Status. 74
Figura 3.31. Diagrama a bloques. 75

Figura 3.32. Conexión de los dispositivos en la misma red local. 76
Figura 3.33. Pantalla del HMI: inicio.....77
Figura 3.34. Pantalla del HMI: conectividad de la red. 78
Figura 3.35. Pantalla del HMI: selección de modo de operación Cliente-Servidor..... 78
Figura 3.36. Pantalla del HMI: Writting Logic Command Data. 79
Figura 3.37. Pantalla del HMI: Reading Logic Status Data. 80
Figura 3.38. Pantalla del HMI: lectura de frecuencia real..... 81
Figura 3.39. Inicio de sesión con usuario y contraseña única. 82
Figura 3.40. Ingresar usuario y contraseña.....82
Figura 3.41. Avisos de sistema pendientes.....83

Índice de tablas

Tabla 2.1. Comparativa de las diferentes normas físicas. 10
Tabla 2.2. Capas del modelo OSI. 21
Tabla 2.3. Modelo de datos 24
Tabla 3.1. Parámetros y tipo de datos de la instrucción MB_CLIENT. 48
Tabla 3.2. Parámetros y tipo de datos de la instrucción MB_SERVER. 52
Tabla 3.3. Parámetros de configuración del VFD. 60
Tabla 3.4. Códigos de funciones MODBUS. 61
Tabla 3.5. Escritura de datos de comandos lógicos. 62
Tabla 3.6. Datos de estatus lógico. 64

Índice de ecuaciones

Ecuación 1. Tiempo de aceleración. 70
Ecuación 2. Tiempo de desaceleración. 71

Capítulo 1

Introducción

1.1. Introducción

Actualmente en los sectores industriales se está viviendo un gran avance tecnológico que trae consigo nuevas tecnologías, con el objetivo de tener todos y cada uno de los procesos automatizados, líneas de producción automatizadas, celdas robotizadas, estaciones de trabajo, todo en comunicación y supervisión desde uno o varios sistemas centrales.

Es por ello, que las redes industriales tienen un papel importante en el proceso, así mismo, los protocolos de comunicación han ido evolucionando debido a la demanda que se tiene en la industria y cubrir las necesidades que requieren los procesos, precisamente para comunicar dispositivos y equipos de diferentes marcas con el fin de transferir información y obtener datos de control de interés, de esta forma se realiza una mejor gestión y control de flujo de datos de los procesos en tiempo real.

De lo anterior resulta que, se puede supervisar, modificar parámetros, y realizar la transferencia de información de actuadores, tales como variadores de frecuencia, motores trifásicos, bandas transportadoras, entre otros, siempre y cuando se acceda con autorización que solicite el sistema para no verse corrompido o haciendo mal uso de la información de los sistemas, por supuesto que, existen programas que sirven de cortafuegos, identificador de IPs, para detectar intrusos que intenten conectarse a la red, en realidad estás redes industriales son privadas, es decir, difícilmente se podrá tener acceso, ya que todos los sistemas se comunican a través de un módulo central, el cual, se interconectan con un cable ethernet, el cual se encuentra en un gabinete de control o gabeta, y que solo personal autorizado y calificado tiene acceso a estos.

En este proyecto de investigación contempla la implementación de dos redes de comunicación para la interconexión a nivel de campo de controladores y actuadores lo que comúnmente se ve en la industria que es la integración de tecnologías de fabricantes diferentes.

En el capítulo 1 se mencionan todos los hechos preliminares, tales como, alcances, delimitaciones, objetivos, justificación y metodología.

En el capítulo 2. Se especifica el marco teórico de los elementos más importantes de este proyecto sin adentrarse para dar a conocer al lector una mejor comprensión de los dispositivos industriales aplicados en la parte académica y reforzando varios conceptos que son útiles para el desarrollo e implementación del proyecto.

En el capítulo 3, se plantea la metodología que se siguió para configurar los parámetros, el proceso de conexión de los equipos y la programación de los dispositivos.

En el capítulo 4: se dan a conocer los resultados obtenidos.

1.2. Objetivo general

Programar e implementar una red de comunicación industrial utilizando el protocolo MODBUS entre dos PLCs S7-1200 con un variador de frecuencia para vigilar y controlar variables físicas de un sistema de llenado de fertilizante para la empresa IT Automatización S.A. De C.V.

1.3. Objetivos particulares.

1. Identificar las dos configuraciones TCP y RTU del protocolo MODBUS.
2. Ejecutar las herramientas de seguridad que ofrece TIA PORTAL V14 para la protección de información.
3. Configurar los PLCs S7-1200 para la comunicación Cliente-Servidor.
4. Configurar el VFD para trabajar a través de la red de comunicación MODBUS RTU.
5. Implementar la red de comunicación industrial entre dos PLCs S7-1212C AC/DC/RLY y S7-1215C DC/DC/RLY para utilizar el protocolo MODBUS TCP
6. Implementar la red de comunicación industrial MODBUS RTU para el PLC S7-1215C DC/DC/RLY y el variador de frecuencia PowerFlex4 utilizando el puerto RS-485 hembra del PLC.
7. Integrar las redes industriales MODBUS TCP y RTU.
8. Diseñar un sistema SCADA mediante el PLC S7-1212C AC/DC/RLY, S7-1215C DC/DC/RLY y el HMI TP177B 6” PN/DP para observar y controlar variables del proceso.

1.4. Alcances

El siguiente proyecto es la investigación del desarrollo e implementación de redes industriales para después implementarlo y adaptarlo al sistema de llenado de fertilizante.

Dentro del documento se especifican conceptos, definiciones y características más importantes sin adentrarse en temas, ya que este proyecto tiene como objetivo la implementación e integración de redes industriales. Sin embargo, se definen parámetros de carácter técnico los cuales son de interés, configuraciones de los dispositivos y procedimientos del desarrollo; para que se comprenda la manera en la cual se realizó cada una de las tareas del proceso.

1.5. Delimitación

1.5.1. Temporal

El presente proyecto se desarrolló considerando un periodo de tiempo de 6 meses, a partir de febrero a junio del año 2021.

1.5.2. Espacial

Ante todo, considerando que las instalaciones de la universidad se encuentran cerradas por los tiempos difíciles que se han vivido. Por lo tanto, el trabajo de grado para la obtención del título de “Licenciado en Electrónica”, fue implementado en la casa habitación del asesor, ubicada en la ciudad de Puebla.

1.5.3. Académica

Para este proyecto se consideran los recursos disponibles proporcionados por el asesor como lo son los PLCs, el variador de frecuencia, una pantalla HMI, un switch para comunicar los equipos industriales. Se considera el protocolo de comunicación MODBUS ya que es de acceso libre y compatible con diversas marcas de equipos industriales. Es importante mencionar que, no se utiliza un motor, sin embargo, se conecta directamente al VFD y se podrá tener el control del actuador.

En este proyecto un PLC S7-1215C DC/DC/RLY trabaja como maestro (cliente) y el otro PLC S7-1212C AC/DC/RLY como servidor (esclavo). Al mismo tiempo, los PLCs pueden trabajar bajo la arquitectura de Cliente-Servidor, esto es, que el PLC-Cliente puede trabajar como servidor y el PLC-Servidor puede trabajar como cliente dentro del protocolo MODBUS TCP. El PLC- servidor realiza una interfaz de comunicación utilizando el protocolo Modbus RTU con el variador de frecuencia Powerflex 4, el cual, se comporta como esclavo dentro de la red MODBUS RTU.

Para la transferencia de datos de control entre maestro y esclavo se enviarán a través del protocolo MODBUS. Se utiliza la librería de bloques de comunicación MODBUS TCP Cliente-Servidor y MODBUS RTU de SIEMENS con el fin de conocer las funciones de comunicación y enriquecer los conocimientos en el área de la automatización.

1.6. Justificación

Actualmente, en todos los sectores industriales, existe una demanda por la mejora en el control de procesos, reducción de costos al momento de comunicar dos o más dispositivos y reducción de tiempos para el mantenimiento, lo que involucra varios factores dentro de un proceso. Factores como: monitoreo y control de datos, implica la implementación de redes de comunicación para la transferencia de datos. Así mismo, la empresa buscará hacerlo de manera eficiente y a un costo accesible, es decir, que sea aceptable y un estándar compatible con los equipos industriales y que su implementación sea rápida, esto por un lado, por otro lado, el protocolo MODBUS TCP es un protocolo, el cual, utiliza el puerto ETHERNET RJ45 para su implementación y esta red es muy utilizada en todos los sectores industriales ya que se ha estandarizado y es de acceso libre esto quiere decir que, no se pagan licencias para utilizarlo, lo que hace más practica la comunicación entre equipos de diferentes marcas.

Cualquier dispositivo que tenga un puerto RJ-45 se podrá comunicar por el protocolo TCP, esto es un estándar, lo que hace posible que dispositivos de diferentes fabricantes puedan integrarse en una misma red, sin tener que adquirir e integrar algún convertidor de datos.

Por esta razón, es conveniente que el ingeniero que se inclinará hacía el área de la automatización y control industrial, tenga los principios y una noción, particularmente de dicho protocolo y sus configuraciones. No solo de la comunicación, si no también, de la programación lo que le dará opciones a escoger y proponerlas en la aplicación que mejor se adecue, tomando en cuenta el presupuesto disponible de la empresa para el proyecto, tiempos de ejecución y entrega.

En la tesis presentada a continuación se mencionan los principios básicos hasta las funciones avanzadas de programación para realizar una correcta implementación de una red utilizando el protocolo MODBUS para que sea adaptable en una aplicación en el mundo real. Por lo anterior, esta investigación tiene como justificación, proveer una idea general de la configuración y control de un sistema de llenado de fertilizante para la empresa IT Automatización S.A. De C.V.

1.7. Metodología

1.7.1. Métodos

A continuación, se describen los métodos que serán aplicados para el desarrollo del proyecto.

1.7.1.1. Método Inductivo

Este método se aplica, puesto que, se hace uso de conocimientos de las materias de Automatización Industrial I y II, y programación para la implementación de las redes de comunicaciones industriales.

Esta memoria de trabajo podrá orientar y servir de guía para el lector, quien podrá conocer cuáles son los principios y entender cómo es que funciona el protocolo MODBUS.

Se aplicará el análisis y se verificará que la transferencia de datos de información sea correcta en cada red de comunicación para que no haya errores en la comunicación y se reciba la información que se está enviando.

1.7.1.2. Método Teórico y sistemático

Primero se recopiló la información de los equipos industriales con el fin de conocer sus características tales como:

- Información de los controladores programables
- Características del variador de frecuencia
- Principios de funcionamiento del VFD
- Hardware del TOUCH PANEL
- Funcionamiento de los bloques de instrucciones de comunicación MODBUS TCP
- Funcionamiento de los bloques de instrucciones de comunicación MODBUS RTU
- Características del conector RS-485

La información de los dispositivos a conectar y programar es importante, ya que se conoce que número parte se va a integrar en el proyecto, el firmware de cada dispositivo, la versión del equipo con la que se va a trabajar e integrar, puesto que, dentro de la plataforma existen diferentes tipos de PLCs. Una vez conociendo todo esto, es identificar y agregar los equipos necesarios.

Capítulo 2

Marco teórico

2.1. Normas físicas

Dentro de lo que es el nivel físico del modelo OSI, y al igual que otros aspectos de la comunicación, el enlace de datos también se encuentra normalizado.

Empleando las palabras de Penin (2008), las conexiones físicas en el entorno industrial se realizan mediante interfases serie, normalizador por la asociación de Industrias Electrónicas de los Estados Unidos (EIA). Estos estándares solo determinan las características del soporte de comunicación, y como debe ser la señal eléctrica.

Las normas principales que se utilizan en las comunicaciones industriales (Recommended Standard, RS), de los cuales, los más conocidos son:

- RS-232
- RS-422 A
- RS-485
- TTL

2.1.1. Norma física RS-232

Guerrero (2018) mencionó que, en 1969 fue definida por la EIA (Electronic Industries Association) en lo que es la definición del conector y del cable. Cabe mencionar que, esta norma fue creada para comunicar un ordenador con un modem a velocidades ≤ 20 kbps hoy en día este estándar se utiliza también a nivel industrial para comunicar punto a punto diferentes componentes y equipos como reguladores, PLCs, variadores de frecuencia, etc.

Además, la longitud del pulso se ve limitado, siendo inversamente proporcional a la velocidad, es decir, se podrá aumentar la velocidad si se disminuye la longitud del cable o bien disminuir la velocidad de transmisión si se aumenta la longitud del cable.

Esta norma define la interconexión serie entre un dispositivo transmisor de datos (DCE, Data Communication Equipment) y un receptor de datos (DTE, Data Terminal Equipment).

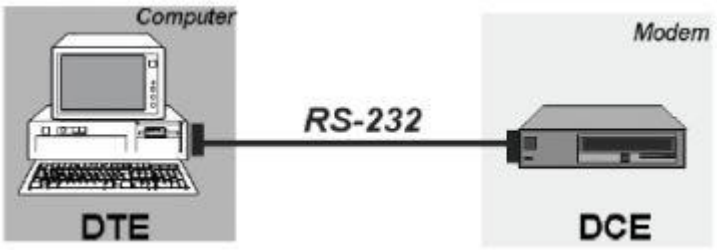


Figura 2.1. Conexión RS232.

Las especificaciones básicas de RS232:

- Permite velocidades de 300, 600, 1200, 2400, 4800, 9600, 19200, 38400bps.
- Longitud máxima del cable: 15 m (L mayor C menor de 2500 pF)

La conexión es punto a punto debido a su estructura (no dispone de un estado de alta impedancia que lo haga adecuado para la conexión en paralelo).

- Tiene limitaciones de velocidad y distancia de transmisión debido a la longitud del cable.
  A medida que aumenta la longitud, aumenta la capacitancia del cable, con lo cual van desapareciendo armónicos de la onda cuadrada al comportarse la línea como un filtro paso-bajo. Esto significa la perdida de flancos, bit de marcha, bit de paro, etc.

- Perturbaciones eléctricas.
  En esta tecnología, los niveles binarios de la señal se indican mediante niveles de tensión, positiva y negativa, respecto del punto de potencial común (+10V, -10V). Esto es punto desfavorable, pues este tipo de transmisión es susceptible de introducir fallo de transmisión frente a las perturbaciones eléctricas.

- Velocidad de transmisión.
  El aumento de la velocidad implica acortar la duración de bit, lo cual implica menor tolerancia a la perdida de sincronismo, a la deformación de la onda cuadrada y al ruido en la línea.

Como solución a las limitaciones de RS232, aparecen RS422 y RS485.

2.1.2. Norma física RS-422

La solución que se planteó ante las nuevas exigencias de la industria fue la de utilizar una línea diferencial mediante dos hilos que ofrece las siguientes características.

- Utiliza dos hilos respecto a una masa.
- El dato recibido se obtiene de la diferencia de tensiones entre dos hilos.
- Permite mayores distancias de transmisión que la transmisión simple.
- Es más inmune al ruido eléctrico.

Hace lo posible el hecho de poder transmitir datos a mayor velocidad por un mismo cable. Así mismo, otra ventaja es el hecho de poder interconectar equipos transmisores y receptores que utilicen la alimentación de 5 voltios disponible en los ordenadores, por lo que no es necesaria una alimentación auxiliar.

Con esta tecnología las perturbaciones se acoplan por igual a la tensión absoluta TD- y TD+. De esta manera el valor de tensión entre las dos líneas se mantiene constante durante la perturbación. Así, los unos y ceros se transmiten en forma de diferencia de tensión entre los dos conductores del circuito, presentando una gran inmunidad a los ruidos eléctricos y permitiendo una mayor distancia entre conexiones.

El estado de alta impedancia introducido en su diseño eléctrico permite la conexión en paralelo de varios elementos. RS422 solo permite aplicaciones de bus con 10 equipos.

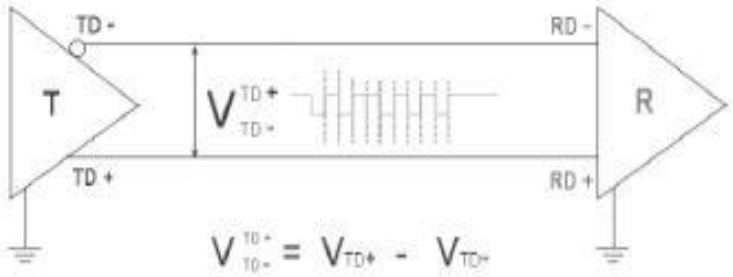


Figura 2.2. Principio de funcionamiento de señales balanceadas.

2.1.3. Norma física RS-485

En 1983, la EIA definió un nuevo estándar: la RS-485. para resolver algunos de los problemas que presentaban las anteriores normas. Se considera como interface multipunto y permite la comunicación de hasta 32 equipos emisores- receptores en un bus de datos común, por lo que se considera un tercer estado (habilitación) para que no haya colisiones en el canal de comunicación, satisfaciendo al mismo tiempo los requerimientos de la RS-422.

Se basa en un sistema diferencial que permite eliminar los posibles ruidos que se pueden incorporar en el canal de comunicación. En esta norma se usa el tipo de transmisión diferencial Half-Duplex que se caracteriza por:

- Utilizar dos hilos o cables, referenciado a masa.
- Conexión de hasta 32 estaciones.
- Rango de velocidades que abarca de 9.6kBit/s a 12Mbit/s.
- Buena inmunidad a las interferencias al tratarse de señales de diferencia de tensión.
- El dato se obtiene de la diferencia de la señal eléctrica entre los dos hilos que componen el canal de comunicación.
- El hecho de que el tipo de transmisión sea Half-Duplex indica que cada equipo puede enviar o recibir, pero no de forma simultánea.
- Permite velocidades de hasta 10 Mbps y distancias de hasta un máximo de 1,200 m.

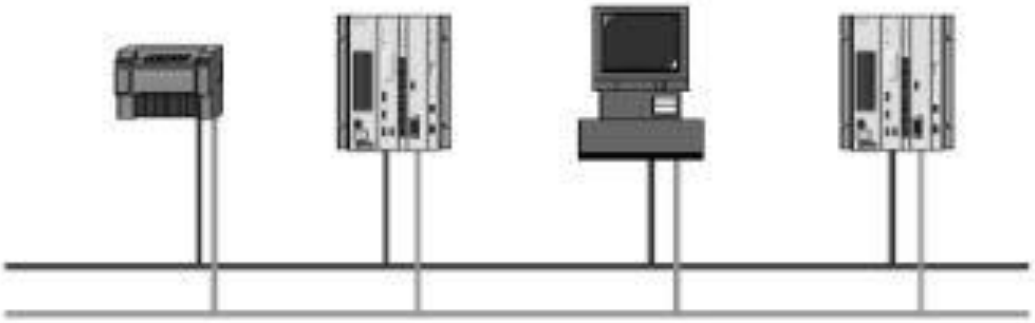


Figura 2.3. Conexión multipunto.

En la siguiente tabla se muestran las características de cada norma física. (tabla tomada del libro de comunicaciones industriales, Guerrero V.).

Parámetros		RS-232C	RS-422-A	RS-485
Modo de trabajo		Unipolar	Diferencial	Diferencial
Numero de emisores y receptores		1 emisor 1 receptor	1 emisor 32 receptores	32 emisores 32 receptores
Longitud máxima del cable		15 m	1,200 m	1,200 m
Velocidad de transmisión máxima		20 kbps	Hasta 10 Mbps	Hasta 10 Mbps
Número de líneas		Hasta 25 (datos y control)	Hasta 4 (datos y control por software)	2 (datos y control por software)
Tipo de cable		Cable especifico (hilos tantos como señales)	Par trenzado (2 pares)	Par trenzado (1 par)
Topología que admiten		Punto a punto	Multipunto (punto a punto desde principal a secundarias) Anillo	Punto a punto Multipunto Anillo Bus
Simultaneidad en la transmisión		Simplex Half duplex Full duplex	Full duplex	Half duplex
Tensión de salida del emisor	Sin cargar	+/- 5 V	+/- 2 V	+/- 1.5 V
	Cargado	+/- 15 V	+/- 6 V	+/- 6 V

Tabla 2.1. Comparativa de las diferentes normas físicas.

Principio de transmisión.

RS485 se basa en la transmisión de señales como diferencia de potencial, sin referencia de tierra. En el caso de haber interferencias, afectaran por igual a las dos señales.

En el destino, la señal definitiva se obtiene de restar las dos señales recibidas, con lo cual la interferencia se anula

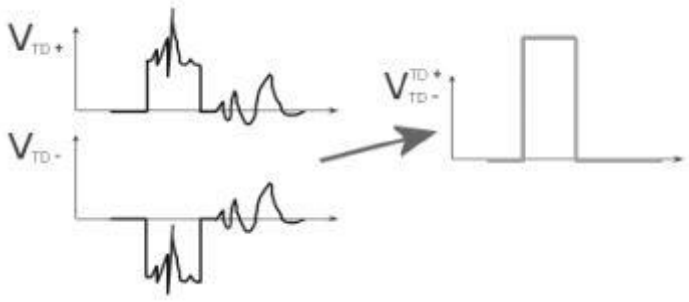


Figura 2.4. Interferencia en señal diferencial.

Para compensar las diferencias de potencial que aparecen en las grandes instalaciones suele tenderse un cable de igualación de potencial, para evitar corrientes que fluyan a través de la malla del cable de señal.

2.1.4. TTL

Se basan en la transmisión digital a niveles TTL (5V), y en la tecnología CMOS. Esta tecnología permite la conexión en paralelo de varios nodos a un mismo cable al permitir un estado de alta impedancia en la conexión del dispositivo. De esta manera se obtiene el efecto de un solo elemento conectado al cable, y que será el que controle la comunicación en ese momento.[15]

2.2. Topología de redes

Es el aspecto físico que forman los equipos y el cableado de estos. Se pueden encontrar sistemas industriales con las siguientes topologías [11]:

- Punto a punto
- Bus
- Árbol
- Anillo
- Estrella

2.2.1. Punto a punto

Se basa en la conexión directa de dos equipos. Sus principales características son:

- Dentro de la trama del mensaje ya no se agregan las direcciones, tanto de origen como la de destino.

- Se comunican por sistemas Half- Duplex (RS- 485) o Full- Duplex (RS- 422). No es necesario el acceso al medio, por lo que se pueden comunicar bidireccionalmente y de manera simultánea.

Ventajas:

- Instalación simple.
- Fácil control de acceso a la red.
- Si un nodo falla, no ocasiona problemas al resto de los equipos.

Inconvenientes:

- Cableado complejo.
- Múltiples tarjetas de comunicaciones.

Figura 2.5. Estructura de red en forma de punto a punto.

2.2.2. Bus

Se comparte una única línea para toda la red.
Es necesario averiguar si el bus se encuentra no ocupado para enviar un mensaje.
Puede circular por el bus un solo mensaje.
Se producirá una colisión, si un equipo envía un mensaje cuando haya otro mensaje en la red.

Ventajas:

- Bajo costo en la instalación.
- Si un nodo falla, no ocasiona problemas al resto de los equipos.
- Control del flujo sencillo.
- Se pueden comunicar entre sí todos los equipos conectados.
- Es sencilla la integración a la red de más equipos.

Inconvenientes:

- Alta probabilidad de colisiones en la red.
- La red podría caerse, si el canal falla, ya que todos los equipos conectados dependen del mismo.

Aplicaciones:

- Redes industriales.

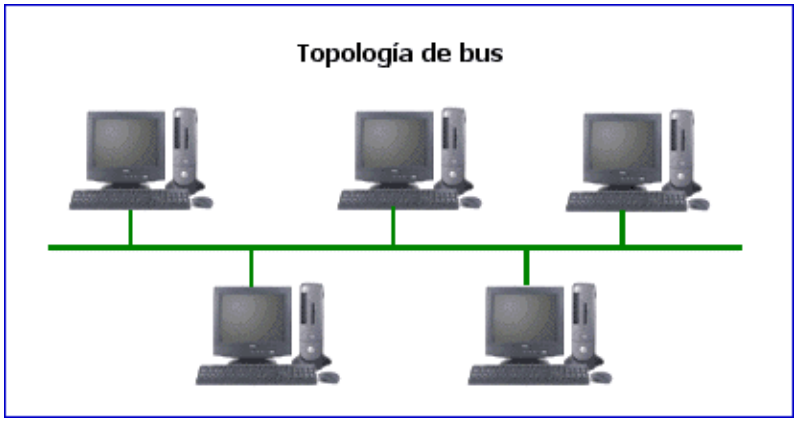


Figura 2.6. Estructura de red en forma de bus

2.2.3. Árbol.

Se conforma por grupos de buses, los cuales se encuentran conectados entre sí, de tal manera que toma la estructura arbórea.

Ventajas:

- Mayor alcance en comparación al bus simple.
- Fácil adaptación a departamentos o zonas independientes dentro de una empresa.

Inconvenientes:

- Se incremente la atenuación.

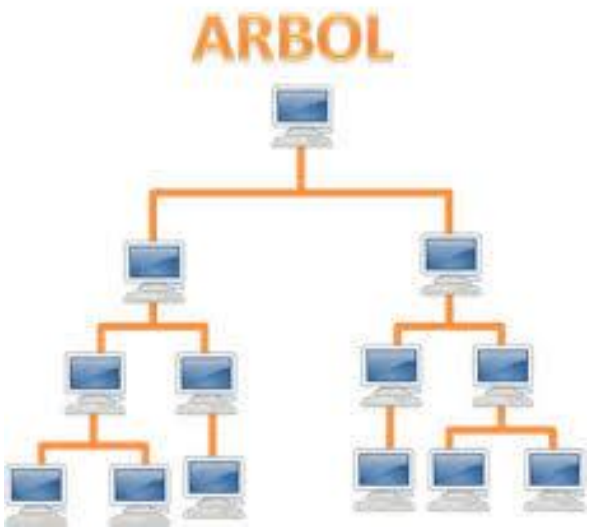


Figura 2.7. Estructura de red en forma de árbol.

2.2.4. Anillo

Caso especial de la conexión en bus, en el que los dos extremos se unen para formar un bus cerrado en forma de anillo. Sus características principales son:

- La información se transfiere en un único sentido.
- El mensaje viaja por el anillo hasta llegar al receptor indicado.
- Se podrá enviar más de un mensaje a través del anillo.
- Facilidad de integración de un nuevo equipo al anillo, solo basta con conectarlo físicamente al medio de transmisión.
- El rendimiento de la red puede ser muy elevado, la velocidad la marca el equipo más lento.
- Desde el punto de vista hardware y software, el control es simple.

Ventajas:

- Todos los mensajes recorren el mismo camino.
- Facilidad de integrar nuevos dispositivos.
- No hay fallas por colisiones.
- No hay atenuación, cada nodo actúa como repetidor.

Inconvenientes:

- Si falla un equipo, el anillo estará fuera de servicio, por ende, la red deja de operar.
- Funciona en distancias cortas.

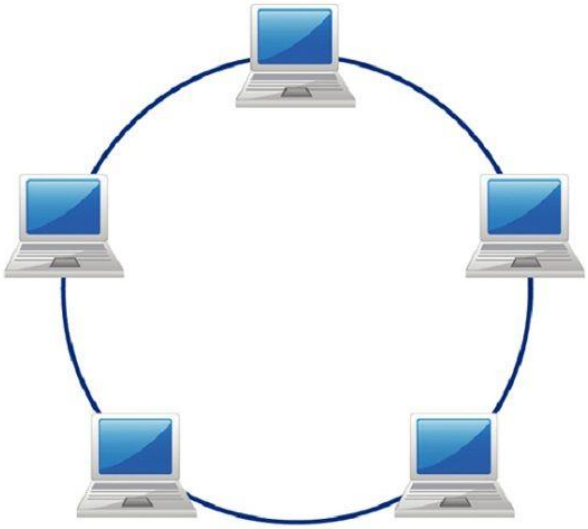


Figura 2.8. Estructura de red en forma de anillo.

2.2.5 Estrella

Todos los dispositivos se conectan a un mismo nodo, el cual puede ser un concentrador o repetidor de la información. Este nodo es el único que controla toda la de información, así, todos los dispositivos dependen de este elemento, debido a que, si falla dicho elemento, estará fuera de servicio toda la red.

Penin (2013), mencionó características importantes de esta topología, las cuales son las siguientes:

- Menor longitud de cable
- Las conexiones de alta impedancia permiten conectar y desconectar dispositivos de forma sencilla.
- Mayor velocidad de transmisión.
- La comunicación es multipunto.
- Ampliación sencilla.

Ventajas:

- Mayor rendimiento, puesto que la información va del emisor al receptor.
- Facilidad de agregar o quitar nodos
- Se facilita el mantenimiento.

Inconvenientes:

- Es indispensable el concentrador; la red no funcionará, si el dispositivo falla.
- Si el HUB no es eficiente, podrá verse afectada la red.

Figura 2.9. Estructura de red en forma de estrella.

2.4. Concepto de protocolo de comunicación

Rodríguez (2008) afirmó que, engloba todas las reglas y convenciones que deben seguir los equipos cualesquiera para poder intercambiar información. Los protocolos son conjuntos de condiciones o reglas que habilitan el intercambio de información entre dispositivos.

2.5. Concepto de bus

De acuerdo con Rodríguez (2008), la distribución básica se realiza alrededor de un segmento de cable al cual se conectan los equipos. El modo de transmisión es aleatorio, un equipo transmite cuando lo necesita. Si hay transmisiones simultaneas (colisiones), algunos algoritmos especiales solventan el problema.

Un bus de campo es generalmente “un sistema de dispositivos de campo (sensores y actuadores) y dispositivos de control, que comparten un bus digital serie bidireccional para transmitir informaciones sobre ellos, sustituyendo a la convencional transmisión analógica punto a punto” [12].

2.6. Formas de organización de nodos.

Rodríguez (2013), mencionó que, los nodos de una red pueden clasificarse también en torno a su forma de gestionar la información, en relación con los otros nodos.

- Maestro – Esclavo
- Cliente – Servidor
- Productor – Consumidor

2.6.1. Maestro – Esclavo

Generalmente se establece una jerarquía entre los equipos al momento de organizar las comunicaciones, en la que uno de ellos tiene el control de estas (de forma temporal o permanente). Es lo que se conoce como relación Maestro – Esclavo.

En el entorno industrial, el maestro es un autómata que puede leer o escribir sobre los esclavos de la red que controla, mientras que el esclavo recibe los mensajes enviados por el maestro y envía una respuesta cuando le es enviada una solicitud.

Penin argumenta también que dentro de los esclavos hay dos categorías.

- **Esclavos activos:** son equipos con un PLC que recibe órdenes y ejecuta un programa propio.

- **Esclavos pasivos**: se comportan como terminales, no ejecutan algún programa y realizan la función de entradas-salidas remotas del autómata maestro.

2.6.2 Cliente - Servidor.

Tiene como base la prestación de servicios por parte de algunos interlocutores de una red, y el aprovechamiento de estos servicios por parte del resto. Se denominan relaciones de tipo Cliente-Servidor.

Un cliente de la red es un equipo que solicita los servicios a una estación. El servidor es una estación que proporciona esos servicios solicitados.

Un servidor puede ser, a su vez, un esclavo de la red y una estación dicho de otra manera puede ser a la vez, cliente y servidor.

2.6.3. Productor – Consumidor

Se basa en el concepto de comunicación broadcast (para todos). Un nodo productor emite un mensaje global a la red cuando lo necesita. Los nodos consumidores reciben la información y determinan si son los destinatarios del mensaje.

Con base en este método le permite a todos los dispositivos de la red el acceso de manera simultánea a un dispositivo para leer sus datos, incrementando la eficiencia del sistema sin importar el número de solicitantes y proporcionando una sincronización automática al llegar los datos a todos los destinatarios de forma simultánea [14]

2.7. Modelo OSI

De acuerdo con Peciña (2018), en la actualidad los sistemas de comunicaciones industriales son abiertos, en otras palabras, no tienen un propietario, por tanto, cualquiera podría fabricar apegándose a una norma en específico.

Guerrero (2018), mencionó que, el Modelo OSI fue definido por la ISO en el año 1983. Este modelo dispone de siete capas o niveles y, en cada uno de ellos, se define como debe comportarse la comunicación. Pueden verse los niveles en la figura 2.10. Se aprecian los siete niveles para establecer la comunicación entre dos interlocutores A y B.

A continuación, se explica el modelo OSI, que es el acrónimo de Open System Interconection (Interconexión de Sistemas Abiertos).

Como señaló Penin (2013), básicamente, cada nivel se comunica con su nivel homólogo de otro sistema, haciendo servir de intermediarias las capas que tiene por debajo creando una serie de canales que codifican (envío) y decodifican (recepción) la información.

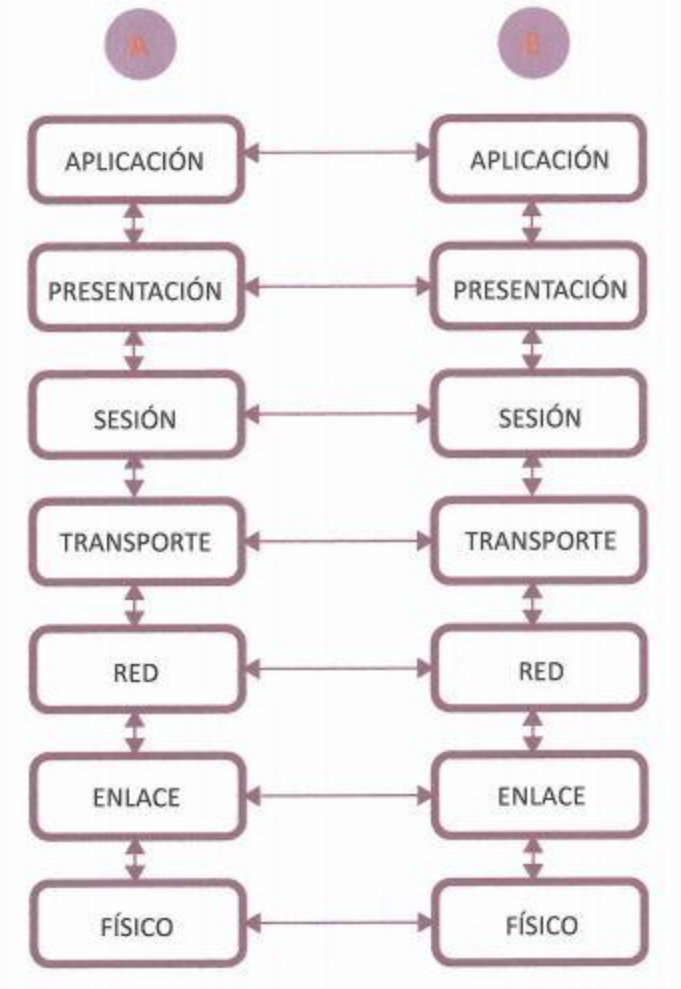


Figura 2.10. Modelo OSI.

Dentro de cada nivel se establecerá el protocolo, esto es, el conjunto de reglas que se deben cumplir para un tipo de comunicación determinada. No resulta obligatorio que se cumplan los siete niveles. De hecho, en comunicaciones industriales lo normal es que se cumpla tan solo con tres niveles; siempre dependerá de la complejidad de la comunicación. A estos niveles el usuario no accede, ya que se hace mediante programas, los cuales ejecutan las reglas de cada nivel.[13]

2.7.1. Capa de aplicación.

Peciña (2018), mencionó que, la capa define los protocolos de las aplicaciones para intercambiar datos con el correo electrónico, los gestores de bases de datos y los servidores de ficheros. El usuario conecta con esta capa a través de los programas que lo permiten.

2.7.2. Capa de presentación.

Se encarga de poner en orden toda la información que va a llegar. Los dos equipos pueden ser diferentes y presenten diferentes representaciones internas de caracteres, los datos deben llegar de manera reconocible

2.7.3. Capa de sesión

Peciña (2018), señaló que, no es una capa importante. Esta capa se encarga de mantener y controlar el enlace establecido entre dos computadoras que están transmitiendo datos, esta se pueda efectuar para las operaciones definidas de principio a fin y reanudarla en caso de interrupción.

2.7.4. Capa de transporte

Esta capa es la encargada de efectuar el transporte de los datos de la máquina de origen a la de destino independientemente del tipo de red física que esté utilizando.

Todas las comunicaciones industriales son del tipo serial, con lo que evitamos transportar grandes cantidades de cables por las fábricas, de modo que se ahorra materiales, en mantenimiento y en puestas en marcha de las instalaciones.

2.7.5. Capa de red

Peciña (2018), argumentó que, se encarga de identificar el camino existente entre una o más redes.

El objetivo de la capa de red es hacer que los datos lleguen desde el origen al destino, aun cuando ambos no se encuentren conectados directamente. *Los firewalls (cortafuegos) actúan sobre esta capa principalmente para descartar direcciones de máquinas.*

2.7.6. Capa de enlace de datos

Esta capa se ocupa del direccionamiento físico, del acceso al medio, detección de errores, distribución ordenada de tramas y del control del flujo.

2.7.7. Capa física

Peciña (2018), afirmó que, de todas las capas, la física es la más fácil de comprender ya que reúne todos los aspectos materiales de la conexión, como el tipo de cableado, los niveles eléctricos de transmisión.

Peciña (2018), destacó que, se puede resumir cada capa del siguiente modo:

- Capa de aplicación: programas de usuario
- Capa de presentación: estandarización de la presentación a las aplicaciones
- Capa de sesión: gestión de las conexiones entre aplicaciones.
- Capa de transporte: conexión física y proporciona el envió de datos.
- Capa de red: gestión de las conexiones hacia las capas superiores.
- Capa de enlace de datos: detección y corrección de errores.
- Capa física: red material.

Comparando con Penin (2013), resume el modelo OSI en la siguiente tabla, la cual fue tomada del libro de Comunicaciones Industriales:

Nivel	Nombre	Función	Objetivo
7	Aplicación (Application Layer)	Funciones de usuario y servicios de comunicación.	Intercambio de información.
6	Presentación (Presentation Layer)	Conversión de los datos a un formato común entendible por todos los equipos	Lenguaje propio del equipo para transmitir y recibir.
5	Sesión (Session Layer)	Control de las sesiones de comunicación (inicio, transcurso y final)	Coordinación y fiabilidad de la transmisión (comunicación por turnos).
4	Transporte (Transport Layer)	Garantizar un enlace fiable entre terminales. Formación y gention de los paquetes de información.	Transmisión segura.
3	Red (Network Layer)	Direccionamiento a través de sistemas (routing) y control de flujo.	Comunicación entre redes.
2	Enlace de datos (Data Link Layer)	Método de acceso. Control de la transmisión de bits y detección y corrección de los errores.	Utilización del medio.
1	Físico (Physical Layer)	Características mecánicas y eléctricas del sistema físico de transporte.	Adaptación al medio.

Tabla 2.2. Capas del modelo OSI.

De las siete capas, son indispensables los niveles 1, 2 y 7.
Todas las redes de comunicación hacen servir, los niveles OSI situados en los extremos de la estructura de capas:

- **Capa física (1)**: Define las topologías aceptadas, tipo de cable, el modo de emisión y el soporte de transmisión.

- **Capa de enlace (2)**: Acceso a la red, el método de acceso al bus y define los mecanismos del intercambio.

- **Capa de aplicación (7)**: Se proponen diferentes servicios a los usuarios.

2.8. MODBUS

2.8.1. Orígenes

La historia del protocolo MODBUS va ligada a la historia de los PLCs. El PLC apareció con el propósito de eliminar el enorme costo que significaba el mantenimiento de un sistema de control electromecánico (basado en relés) a finales de los años 60s.

La posibilidad de comunicar equipos apareció por el año 1973. El primer sistema que lo hizo servir fue Modbus de MODICON. Permitía establecer comunicaciones Maestro-Esclavo y/o Cliente- Servidor entre dispositivos inteligentes y con dispositivos de campo (transmitir señales digitales, analógicas, registros e información de estado entre ellos) [1].

Conectar los autómatas entre sí, permitió eliminar casi todo el cableado de control entre máquinas, quedando solamente una línea de comunicación entre ellas. A través de esta se podía coordinar el funcionamiento de todos los componentes de un sistema, programar a distancia, supervisar remotamente, diagnosticar todos los elementos conectados.

Debido a las dificultades que siempre han presentado los sistemas cerrados en 1979, MODICON público de forma abierta y estandarizado el protocolo.

- El código era totalmente abierto y libre.
- Fácil y rápido de implementar
- Intercambio sencillo de información entre equipos

MODBUS define una estructura de mensaje que los controladores podrán reconocer y utilizar sin tener en cuenta el tipo de red que estos harán servir para comunicarse. [1]

2.8.2. TCP/IP

De acuerdo con diversos autores, el protocolo de control de transmisión es un protocolo de comunicación (en inglés Transmission Control Protocol o TCP), diseñado que permite a equipos industriales tales como PLCs, PCs, drivers para motores y otros tipos de dispositivos electrónicos físicos de entrada/salida, comunicarse sobre una red Ethernet. Fue introducido por Schneider Automation como una variante de la familia de protocolos MODBUS, ampliamente usada para la supervisión y el control de equipo de automatización.

TCP es el protocolo que proporciona un transporte fiable de flujo de bits entre diversas aplicaciones.

Específicamente el protocolo define el uso de mensajes MODBUS en un entorno internet usando los protocolos TCP/IP.

La especificación MODBUS/TCP define un estándar interoperable en el campo de la automatización industrial, el cual es simple de implementar para cualquier dispositivo que soporte sockets TCP/IP. Todas las solicitudes son enviadas vía TCP sobre el puerto registrado 502 y normalmente usando comunicación Half-Duplex sobre una conexión dada, es decir, no hay beneficio en enviar solicitudes adicionales sobre una conexión única mientras una respuesta está pendiente.

Ventajas:

- Permite el monitoreo del flujo de datos, con lo cual se evitan colisiones en la red.
- Permite multiplexar los datos, es decir, dar una secuencia cuando proviene información de varias aplicaciones.
- Alta fiabilidad.
- Adecuado para aplicaciones y empresas grandes y medianas.
- Compatible con múltiples tecnologías.
- Permite iniciar y terminar la comunicación amablemente.

Inconvenientes:

- Complejo para configurar
- No distingue entre servicios, interfaces, lo cual afecta el diseño de nuevas tecnologías.

Interpretación de números de referencias.

MODBUS basa su modelo de datos en una serie de tablas que tienen características distintivas. Tabla tomada del artículo que se encuentra en línea [6].

Entrada discreta	Bit único, proporcionado por un sistema de E/S, solo lectura
Salida discreta	Bit único, alterable por un programa de aplicación, lectura-escritura.
Registros de entrada	Cantidad de 16 bits, proporcionada por un sistema de E/S, solo lectura.
Registros de salida	Cantidad de 16 bits, modificable por un programa de aplicación, lectura-escritura.

Tabla 2.3. Modelo de datos

2.8.3. RTU

Es un protocolo de comunicaciones, basado en la arquitectura maestro/esclavo o cliente/servidor.

Debido a que este protocolo fue público, de fácil uso y que requiere poco desarrollo (maneja bloques de datos sin suponer restricciones) se convirtió en un protocolo de comunicaciones estándar en la industria. Tiene mayor disponibilidad para la conexión de dispositivos electrónicos industriales. El protocolo MODBUS permite el control de una red de dispositivos. Por ejemplo, un equipo de medición de temperatura y humedad puede comunicar los resultados a una PC.

Protocolo abierto con arquitectura Cliente-Servidor, es un formato de transmisión en serie de datos utilizando extensamente en las comunicaciones de los PLCs.
Es un protocolo ampliamente aceptable dentro del ambiente industrial, debido a su facilidad de uso y fiabilidad [7]

Ventajas

- Es público.
- Fácil de implementar y rápido desarrollo.
- Maneja bloques de datos sin restricciones.

2.8.4. ASCII

Este protocolo opera en la capa de red en serie. Con la ayuda de un cliente, se puede configurar y recuperar los datos ASCII de MODBUS desde su dispositivo y ponerlos a disposición de otros dispositivos, o se puede emular un servidor MODBUS y enviar datos a su dispositivo MODBUS desde otros dispositivos que no sean MODBUS.[7]

2.8.5. Beneficios de utilizar el protocolo MODBUS RTU/TCP.

- De código abierto, no se requiere pagar por licencia
- Ampliamente soportador por HMIs o softwares SCADA
- Fácil de usar
- Se pueden integrar varios equipos en la red
- Bajo costo de desarrollo
- Reconocido ampliamente en la industria

2.9. Variador de frecuencia.

De acuerdo con diversos autores los variadores o convertidores de frecuencia son sistemas que se encuentran entre la fuente de alimentación y los motores eléctricos. Sirven para regular la velocidad de giro de los motores de corriente alterna (AC).

Por sus siglas en inglés, se hace referencia al variador de frecuencia como VFD, que viene de Variable Frequency drive, que se traduce como regulador / variador de frecuencia variable. A pesar de ello, también están presentes en el mercado otras aceptaciones como puede ser VSD (variable speed drive o regulador de velocidad variable) o ASD (adjustable speed drive, que en castellano seria "accionamiento de velocidad variable")

Regulando la frecuencia de la electricidad que recibe el motor, el variador de frecuencia consigue ofrecer a este motor la electricidad demandada, evitando así la perdida de energía, o lo que es lo mismo, optimizando el consumo.

2.9.1. Funcionamiento.

Se trata de controlar la velocidad del motor. Los variadores o convertidores de frecuencia convierten la corriente alterna (CA) de la red eléctrica en corriente continua (CC) por medio de un puente de diodos o SCRs rectificador. Este es el primer paso del proceso y se lleva a cabo por una parte esencial del variador, llamada **rectificador**, (véase figura 2.11).

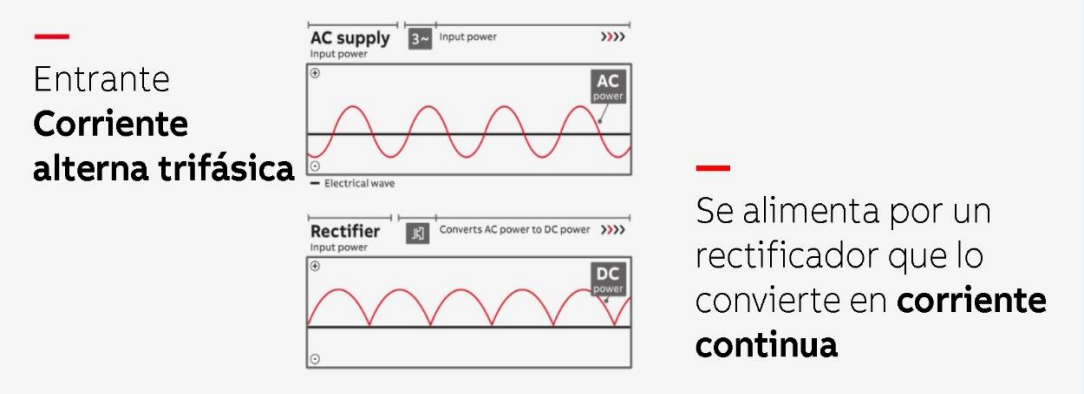


Figura 2.11. Etapa de rectificación.

En la siguiente fase, de la que se encargan **los condensadores** del variador. Estos se cargan con la corriente continua transformada por el rectificador, con el fin de suavizar la forma de onda de la corriente eléctrica resultante y reducir la emisión de variaciones en la señal, (véase figura 2.12).

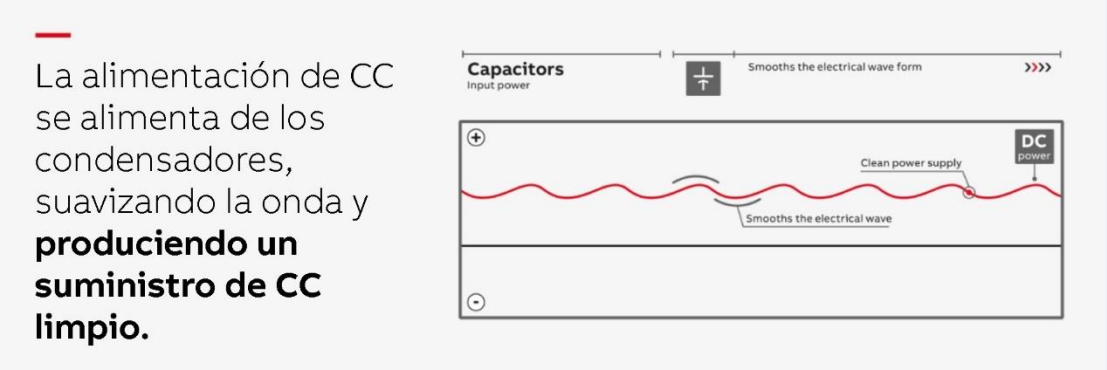


Figura 2.12. Etapa de filtrado.

Finalmente, la última etapa es la del **inversor**, la cual está compuesta por un sistema de transistores bipolares de puerta aislada, (IGBT, Isolated Gate Bipolar Transistor), que encienden y apagan en determinada secuencia (enviando pulsos) para generar una forma de onda cuadrada de voltaje de CD a una frecuencia constante y su valor promedio tiene la forma de onda senoidal de la frecuencia que se aplica al motor (véase figura 2.13).

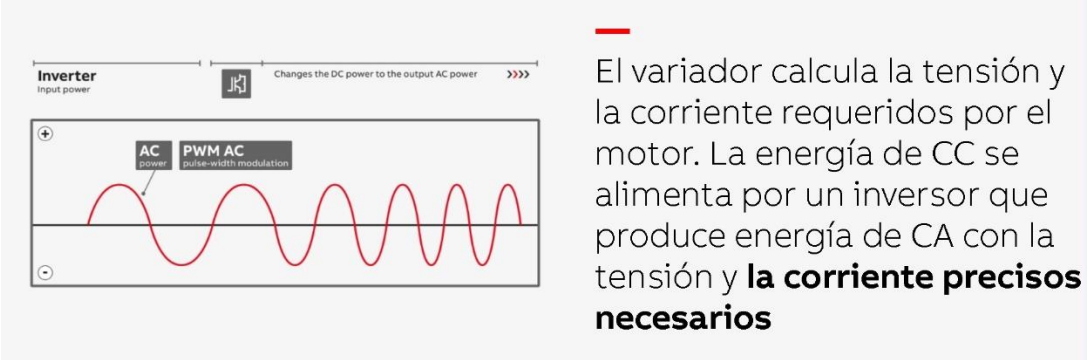


Figura 2.13. Etapa de inversión.

2.9.2. Tipos de variadores de frecuencia

Para seleccionar un variador de frecuencia que más se ajuste a los fines deseados, se debe de conocer de antemano dos factores: Cuál es el voltaje de operación y qué tipo de motor se conectará al variador [20].

Tipos de variadores de frecuencia

- **Variadores de frecuencia de corriente alterna:** generalmente, es el que se utiliza en este proyecto.
- **Variadores de frecuencia de corriente directa:** destinados a motores alimentados por corriente continua.
- **Variadores de frecuencia de voltaje de entrada:** se encargan de generar una nueva onda sinusoidal de tensión, introduciendo una serie de onda cuadrada y mediante la variación del voltaje.
- **Variadores de frecuencia de fuentes de entrada:** en este caso es el caudal de onda cuadrada el que se recibe de entrada. Requieres grandes inversores para mantener una corriente constante.
- **Variadores de frecuencia de ancho pulso modulado:** llamado PWM (por sus siglas en ingles pulse-width modulation), mantiene el par motor constante por medio de una serie de pulsos de voltaje constantes realizados por unos transistores.
- **Variadores de frecuencia de vector de flujo de ancho de pulso modulado:** cuentan con un microprocesador que gestiona el proceso de la regulación o variación de la corriente al motor.

2.9.3. Ventajas de utilizar un variador de frecuencia.

- Ahorro energético.
- Reducción de mantenimiento técnico.
- Prolongación de la vida útil de los actuadores a los que va dirigido.
- Reducción de costos.
- Reducción de contaminación auditiva.

2.10. PLC

2.10.1. Concepto de PLC

De acuerdo con la definición de la NEMA (National Electrical Manufacturers Association), un PLC (Controlador Lógico Programable) es “un dispositivo electrónico operado digitalmente, que usa una memoria programable para el almacenamiento interno de instrucciones con el fin de implementar funciones específicas, tales como lógica, secuencia, registro y control de tiempos, conteo y operaciones aritméticas; con el objetivo de controlar, a través de módulos de entrada/ salida digitales o analógicos, varios tipos de máquinas o procesos”

SIEMENS (2015) ha mencionado que un PLC o Controlador Lógico Programable, es un computador especialmente diseñado para la automatización industrial para el control de una maquina o proceso industrial. El PLC nace como una solución al control de circuitos completos de automatización, a el se conectan a las entradas dispositivos tales como: sensores; por ejemplo: finales de carrera, botones, pulsadores y en las salidas se conectan los actuadores, por ejemplo: contactores, relevadores, motores, lámparas, cilindros, válvulas, etc.

2.10.2. Historia del PLC.

En el siglo XX específicamente en el año 1960, nació el PLC originalmente para las empresas que se dedicaban únicamente al ramo de la industria automotriz.

De esta forma las nuevas tecnologías electrónicas de esa época impactaron directamente en el PLC, y este se vuelve una solución real eficiente para reemplazar los sistemas de control basado en circuitos eléctricos con relevadores, interruptores y otros componentes utilizados para el control de los sistemas de lógica combinacional de esa época [3].

Los PLCs fueron inventados en respuesta a las necesidades de automatización de la industria automotriz norteamericana por el ingeniero del mismo país Dick Morley.

Antes de los PLCs el control, las secuenciaciones, y a la lógica para la manufactura de automóviles era por medio de relevadores y contactores. El proceso para realizar dichas instalaciones en la industria año tras año era muy costosas, complejas y consumía mucho tiempo en rediseñar los sistemas basados en relevadores, ya que el recableado tenía que ser por especialistas profesionales [3].

A partir de 1968, la compañía General Motors encabezó un remplazo de componentes de relevadores e interruptores por el PLC. Así, la división de transmisiones automáticas de General Motors ofertó un concurso para una propuesta de reemplazo electrónico de los sistemas cableados. La propuesta ganadora vino de Bedford Associates de Boston Massachussets.

El primer PLC, fue designado 084, debido a que fue el proyecto ochenta y cuatro de Bedford Associates. Creó una nueva compañía dedicada al desarrollo, manufactura, renta y servicio para este nuevo producto: MODICON (Modular Digital Controller o controlador digital modular). Una de las personas que trabajó en ese proyecto fue Dick Morley, el cual es considerado como padre del PLC. La marca MODICON fue vendida en 1977, y posteriormente adquirida por la compañía alemana AEG y más tarde por Schneider Electric, el actual dueño [3].

2.10.3. Clasificación de los PLCs.

Dentro del mundo de los PLCs existe una gran variedad para las aplicaciones y procesos que las industrias demandan. Para ello, existen diferentes tipos de PLCs de acuerdo con sus prestaciones, como lo son sus funciones integradas, su capacidad, el número de I/O (entradas y salidas), tamaño, memoria, capacidades, entre otras cualidades.

2.10.3.1. Tipo nano

Es un tipo de PLC de tamaño reducido, contiene una fuente de poder, CPU y entradas y salidas integradas, puede manejar un conjunto reducido de I/O (véase figura 2.14). [8]

Figura 2.14. PLC tipo nano.

2.10.3.2. Tipo compacto

Es aquel que se ajusta a las especificaciones genéricas de los PLCs: tiene incorporada la fuente de alimentación, la CPU y diferentes módulos (véase figura 2.15).

Existen PLCs compactos en tamaño similar nano, que cuentan con un tamaño reducido, aunque permiten manejar desde un mínimo de 10 I/O hasta 20 I/O. este tipo de PLCs soportan gran variedad de módulos especiales (véase figura 2.16), tales como: [2]

- Módulos I/O analógicas
- Módulos I/O digitales

- Contadores rápidos
- Módulos de comunicaciones
- Interfaces de operador
- Expansiones de I/Os

Figura 2.15. PLC tipo compacto marca Allen Bradley.

Figura 2.16. PLC tipo compacto marca SIEMENS.

2.10.3.3. Tipo modular

Esta clase de PLC se caracteriza por estar compuesto por módulos aplicables, lo que le aporta características más potentes que el modelo compacto. Trabaja con programas más complejos, tiene mayor capacidad de memoria y operatividad (véase figura 2.17).
A continuación, se enlistan algunas cualidades:

- Rack
- Fuente de alimentación
- CPU
- Módulos de I/O de expansión
- Módulos de comunicaciones
- Contadores rápidos
- Robustos

Este tipo de PLCs pueden utilizar un gran número de I/O, se utilizan para procesos de mayor complejidad, manejan datos de gran tamaño e intercambiar información a gran velocidad, ya que operan bajo el modo de multitarea [2].

Figura 2.17. PLC tipo modular.

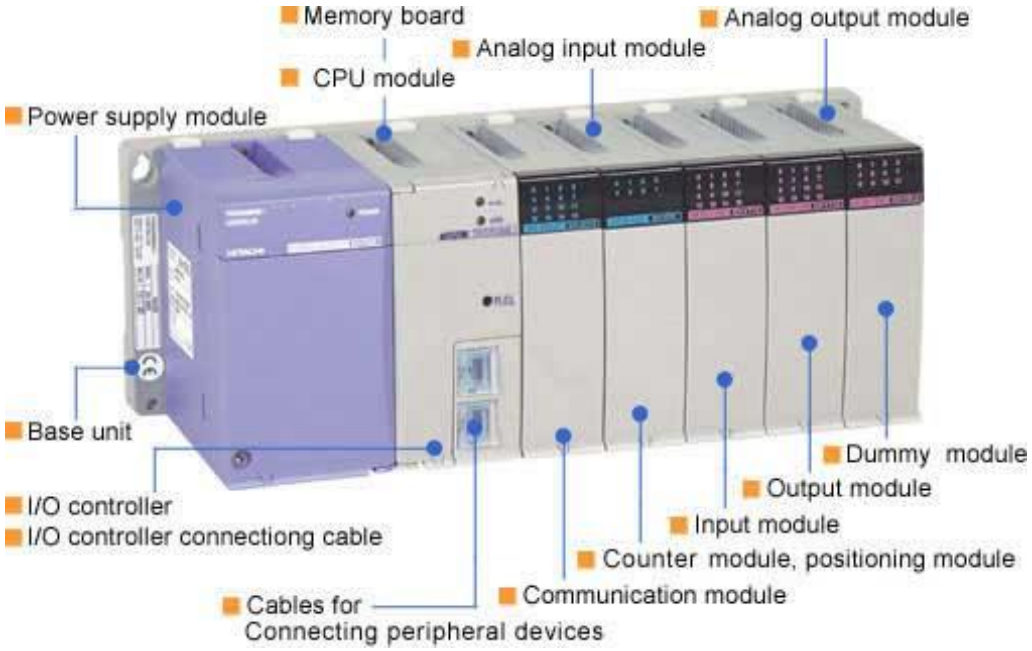


Figura 2.18. PLC tipo modular.

2.10.3.4. Tipo montaje en rack

Prácticamente tienen las mismas características que los PLCs modulares. No obstante, existen algunas diferencias en el bus o en el rack donde van colocados las tarjetas del PLC. Se permite el intercambio de información entre los diferentes módulos, esto permite una velocidad mayor de transmisión de la información y, por tanto, optimiza el funcionamiento del PLC. (véase figuras 2.19 y 2.20), [2].

Figura 2.19. PLC tipo montaje en rack.

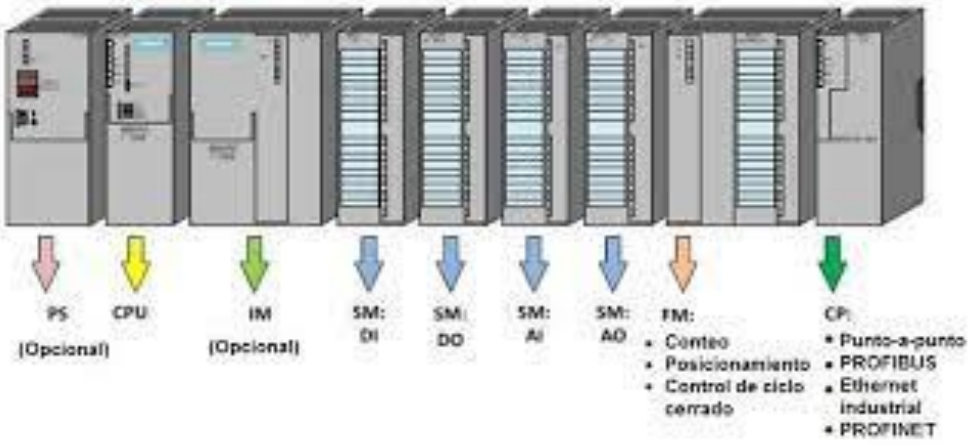


Figura 2.20. PLC tipo montaje en rack.

2.10.3.5. Tipo PLC con HMI integrado

El HMI (Human Machine Interface) es un dispositivo programable con una interfaz gráfica, que combinado con el PLC permite optimizar la experiencia de programación y uso de la máquina. Reduce el cableado de los elementos del PLC y los muestra en una pantalla grafica según se programe. Así mismo, trae un teclado o una pantalla táctil para navegar en el monitor, el cual puede ser tipo grafico o de texto. Como si fuera poco, este tipo de PLC ya no requiere que se programe el panel de forma separada, dado que toda la programación se realiza a través del mismo software, por ende, permite dar una mejor opción presupuestal (véase figura 2.21), [22]

Figura 2.21. PLC + HMI.

2.10.4. Estructura del PLC

A nivel interno un autómata se compone de las siguientes partes:

2.10.4.1. La CPU

Es la parte más importante del PLC, cumpliendo varias funciones [18]:

- Procesar las instrucciones del programa, se encarga de ejecutar las instrucciones del programa de manera secuencial y de forma cíclica. La velocidad de procesamiento es muy alta.
- Leer el estado de las entradas.
- Activar las salidas.
- Comprobar todo el sistema. Se cuenta con una serie de alarmas que el autómata vigila.

2.10.4.2. Memorias:

Las hay de dos tipos: tipo ROM, EPROM y EEPROM que son memorias de solo lectura, estas no pierden su información ante un corte de energía; y las memorias RAM, que son unas memorias de lectura/escritura y ante un corte de energía pierden su información [18].

Las memorias se emplean de tres formas:

- **Memoria de usuario**: es donde se almacena el programa del usuario. Es una memoria RAM que cuenta con el respaldo de una pequeña batería para evitar la pérdida de su información ante un corte de energía. Algunos autómatas cuentan con una memoria tipo EPROM y EEPROM en las que el fabricante ha almacenado previamente el programa.

- **Memoria de tabla de datos**: es una memoria tipo RAM. Aquí se almacena la imagen de las tablas de los estados de entrada y salida junto con las variables de los programas y datos internos (contadores, temporizadores, etc.).

- **Memoria y programa del sistema**: se encuentra dividida en dos áreas: por un lado, está el programa del sistema o firmware (que es grabado por el fabricante en la memoria tipo ROM) y la memoria que junto con el procesador componen la CPU.

2.10.4.3. Interface de entrada.

Adapta las señales de entrada para que las entienda la CPU.

2.10.4.4. Interface de salida.

Es el encargado de preparar las ordenes de la CPU en valores de salida.

2.10.4.5. Interface de periféricos.

Son el resto de los elementos que pueden conectarse al autómata, como, por ejemplo: una consola de programación, un panel de operación, otros PLCs, impresoras, etc.

2.10.4.6. La fuente de alimentación.

Los autómatas programables funcionan internamente a una tensión de 24 y 5 volts en corriente continua, por ello necesitan de una fuente de alimentación. Algunos autómatas integran internamente dicha fuente, pero otros emplean un módulo externo.

Si la fuente de alimentación es interna, el autómata se conecta directamente a la red eléctrica entre fase y neutro. Por el contrario, si el autómata necesita de una fuente externa, su alimentación es de 24 V. La fuente de alimentación, además de suministrar energía al propio autómata se emplea para alimentar a los sensores o dispositivos que lo necesiten. Algunos autómatas con fuente interna cuentan con unos terminales para esta función [18].

2.10.4.7. Módulos de entradas y salidas.

Los módulos de entradas y salidas se clasifican en función del tipo de datos que emplean, así se tienen [18]:

- Módulos digitales o binarios. Utilizan datos a nivel de bit, es decir todo o nada. Detectan tensión en una entrada o no la detectan, activan un bit de salida o no lo activan.

- Módulos analógicos. Poseen cualquier valor dentro de un rango. Utilizan datos a nivel de byte (8 bits) o Word (16 bits). Los módulos de entradas analógicas se emplean para leer magnitudes que no se pueden expresar en valores todo o nada como, por ejemplo: temperatura, presión, distancia, etc.

Estos módulos son de entradas o son de salidas, aunque también los hay mixtos, donde combinan tanto las entradas como las salidas, sin embargo, no mezclan el tipo de datos (analógico o discreto).
A las salidas del autómata se conectan los actuadores.

Existen varios tipos de salidas digitales [18]:

- Salidas a relé. Tienen dos terminales y actúan como un contacto. En sus bornes no existe tensión eléctrica, por ello se llaman libres de potencial. Pueden trabajar con cualquier tipo de tensión, alterna o continua y de cualquier valor.

- Salidas a transistor. La salida es con tensión, por ello los actuadores deben ser del mismo valor de tensión eléctrica. Son adecuados para actuadores de corriente continua. Pueden ser de dos tipos PNP o NPN.

- Salida a TRIAC. Se emplean donde se requieran altas capacidades de conmutación.

2.10.5. Funciones de un PLC

Dentro de las funciones básicas podemos encontrar las siguientes [2]:

- **Detección**: lectura de la señal de los captadores distribuidos por el sistema de fabricación.

- **Mando**: elaborar y enviar las acciones al sistema mediante los accionamientos y pre-accionamientos.

- **Dialogo hombre-maquina**: mantener un dialogo con los operarios de producción, obedeciendo sus consignas e informando del estado del proceso.

- **Programación**: para introducir, elaborar y cambiar el programa de aplicación del autómata. El dialogo de programación debe permitir modificar el programa incluso con el autómata controlando la máquina.

- **Redes de comunicación**: permiten establecer comunicación con otras partes de control. Las redes industriales permiten la comunicación y el intercambio de datos entre autómatas a tiempo real. en unos cuantos milisegundos pueden enviarse telegramas e intercambiar tablas de memoria compartida.

- **Sistemas de supervisión**: también los autómatas permiten comunicarse con ordenadores provistos de programas de supervisión industrial. Esta comunicación

se realiza por una red industrial o por medio de una simple conexión por el puerto serie del ordenador.

- **Control de procesos continuos**: además de dedicarse al control de sistemas de eventos discretos los autómatas llevan incorporadas funciones que permiten el control de procesos continuos. Disponen de módulos de entrada y salida analógicas y la posibilidad de ejecutar reguladores PID que están programados en el autómata.

- **Entradas- Salidas distribuidas:** los módulos de entrada-salida no tienen por qué estar en el armario del autómata. Pueden estar distribuidos por la instalación, se comunican con la unidad central del autómata mediante un cable de red.

- **Buses de campo:** mediante un solo cable de comunicación se pueden conectar al bus captadores y accionadores, reemplazando al cableado tradicional. El autómata consulta cíclicamente el estado de los captadores y actualiza el estado de los accionadores.

2.10.6. Ventajas de los PLCs

Entre las ventajas de utilizar PLCs dentro de un proceso están:

- Reducción de costos
- Reducción de tiempo en mantenimiento
- Reducción de tiempos muertos
- Seguridad en el proceso
- Control más preciso
- Aprovechar dimensiones y espacios.
- Flexibilidad.
- Confiabilidad
- Sustitución de sistemas mecánicos, lógica cableada
- Reducción de cables/ conexiones.
- Mayores prestaciones en programación
- Mayor adaptación a los cambios
- Mayor comodidad
- Mayor inmunidad a las señales de ruido
- Detección rápida de fallas

Cabe mencionar que, cada tipo de PLC puede adaptarse a las necesidades según la aplicación, por lo que existen diferentes gamas de PLCs.

2.10.7. Inconvenientes de los PLCs

Al adquirir un dispositivo de este tipo se puede encontrar algunos inconvenientes, algunos de ellos se enlistan a continuación:

- Personal calificado y especializado
- Mayor presupuesto para automatizar procesos
- Escalabilidad

2.11. Plataforma de programación

2.11.1. TIA PORTAL V14

De acuerdo con varios autores, poco después de 1996, Siemens presentó el concepto Automatización totalmente integrada (TIA, Totally Integrated Automation), la cual se concibió para resolver la mayoría de las tareas de automatización industrial, sea en plantas generadoras, tratadoras de agua, ingeniería para el transito o instalaciones de producción industrial.

Una nueva generación de PLC se lanzó al mercado estableciendo nuevos estándares. Los dispositivos *Simatic S7* estaban disponibles en tres clases de rendimiento: *S7-200, S7-300 y S7-400.* El bus de control en campo se universalizó con profibus y entonces, el Ethernet Industrial permitió una comunicación homogénea desde la planta hasta la oficina de manera económica y fiable. La era de las redes industriales había comenzado.

Los *S7* ya venían con funcionalidades de comunicación abiertas, integradas y sumamente sencillas de implementar en protocolos para el control determinístico en profibus DP.

En 2013 la visión de TIA se hizo realidad: con *TIA Portal,* Siemens introdujo una plataforma de uso intuitivo, para planificación de proyectos, programación y puesta en marcha de controles, unidades de operador, redes y accionamientos. Brindando asistencia al usuario en las tareas de automatización, que entonces eran más rápidas y estandarizadas. El *TIA Portal* fue el hito más importante en el camino hacia la fábrica digital en la historia de Simatic.

Se presentó una nueva generación de controladores. El *Simatic S7-1200 Basic Controller* estableció nuevos estándares de flexibilidad con amplias opciones de expansión, y se podía adaptar perfectamente a una gran variedad de tareas de automatización. La comunicación, ahora abierta sobre redes Ethernet tanto para control en Profinet como para transmisión de datos en TCP/IP se incluyó en todas las líneas de

S7-1200. En combinación con los *Simatic HMI Basic Panels*, ya casi no había límite para una amplia gama de aplicaciones en el rango medio.

La fábrica digital ya es posible con las nuevas formas de ingeniería que exigen al sistema de automatización estár preparado para integrarse con herramientas de simulación de planta (como *Teamcenter y NX*) o para la vinculación con desarrollos académicos con herramientas de simulación como *Simulink*. *Simatic* asume el formato de controlador dedicado en software u Open Controller o en formato híbridos como el S7 1507S. en el futuro cercano, las nuevas tecnologías como la puesta en marcha virtual, los gemelos digitales, la nube y el Edge computing permitirán tiempos de desarrollo de producto aún más rápidos, aplicaciones más complejas y procesos de optimización más eficientes. Es el próximo hito de la historia de la automatización.

El Portal de Automatización Totalmente Integrada (TIA PORTAL por siglas en inglés) proporciona acceso sin restricciones a la gama completa de servicios de automatización digitalizada, desde la planificación digital y la ingeniería integrada hasta la operación transparente.

TIA Portal es la puerta de acceso perfecta a la automatización de la empresa digital.

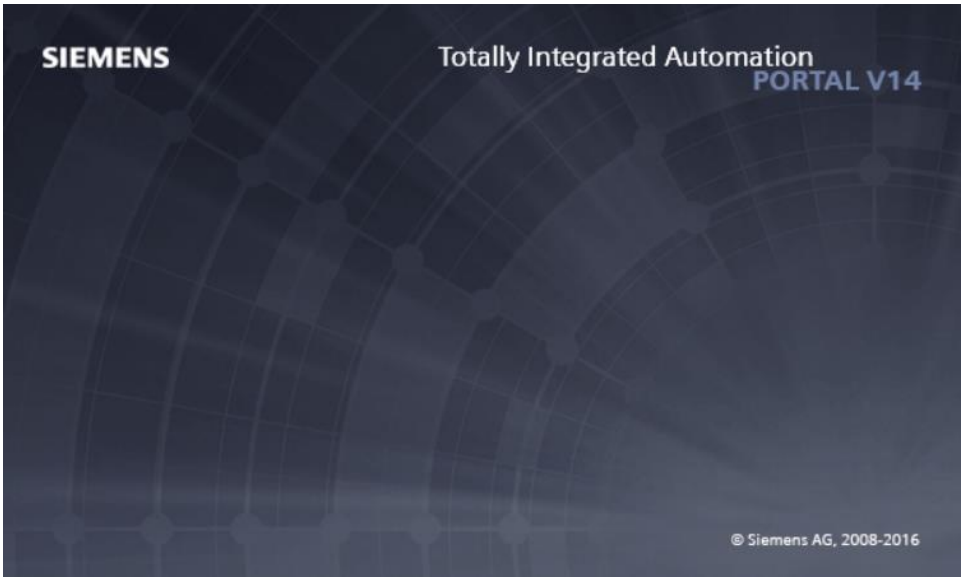


Figura 2.22. TIA PORTAL V14

2.11.2. Protección Know-How de bloques FC y FB

Con esta protección el usuario solo podrá visualizar los bloques protegidos tanto FC como FB de tal manera que solo podrá tener acceso para utilizarlos, pero no para modificarlos ni visualizar el contenido [9].

En un bloque con la función Know-How Protection activada solo se podrá leer únicamente los datos siguientes sin conocimiento de la contraseña:

- Parámetros definidos: Inputs, Outputs, InOut, Return, Static.
- Título del bloque.
- Comentario del bloque.
- Propiedades del bloque.
- Variables de bloques de datos globales sin indicación de la ubicación.

Se debe considerar lo siguiente:

- No se puede leer ni modificar el programa del bloque si no se accede con autorización.
- Sin la contraseña no es posible acceder al bloque.
- Las referencias cruzadas a variables, marcas, entradas y salidas utilizadas en bloques con protección Know-how no se muestran ni siquiera después de introducir la contraseña correcta.

Capítulo 3

Metodología de programación y desarrollo.

3.1. Implementación de la red de comunicación utilizando el protocolo Modbus TCP.

Para la implementación de esta red de comunicación se utilizaron los siguientes dispositivos:

- Una laptop con la plataforma de programación TIA PORTAL V14
- Un PLC SIEMENS S7-1215C DC/DC/RLY
- Un PLC SIEMENS S7-1212C AC/DC/RLY
- Un switch TP LINK
- Un SIMATIC PANEL TP177
- Tres cables UTP con conector RJ45 tipo macho

En esta primera etapa del proyecto, se realizó la conexión e implementación de la red para esto, se conectaron los dispositivos en la misma red, por tal motivo se siguió el diagrama como se muestra en la figura 3.1.

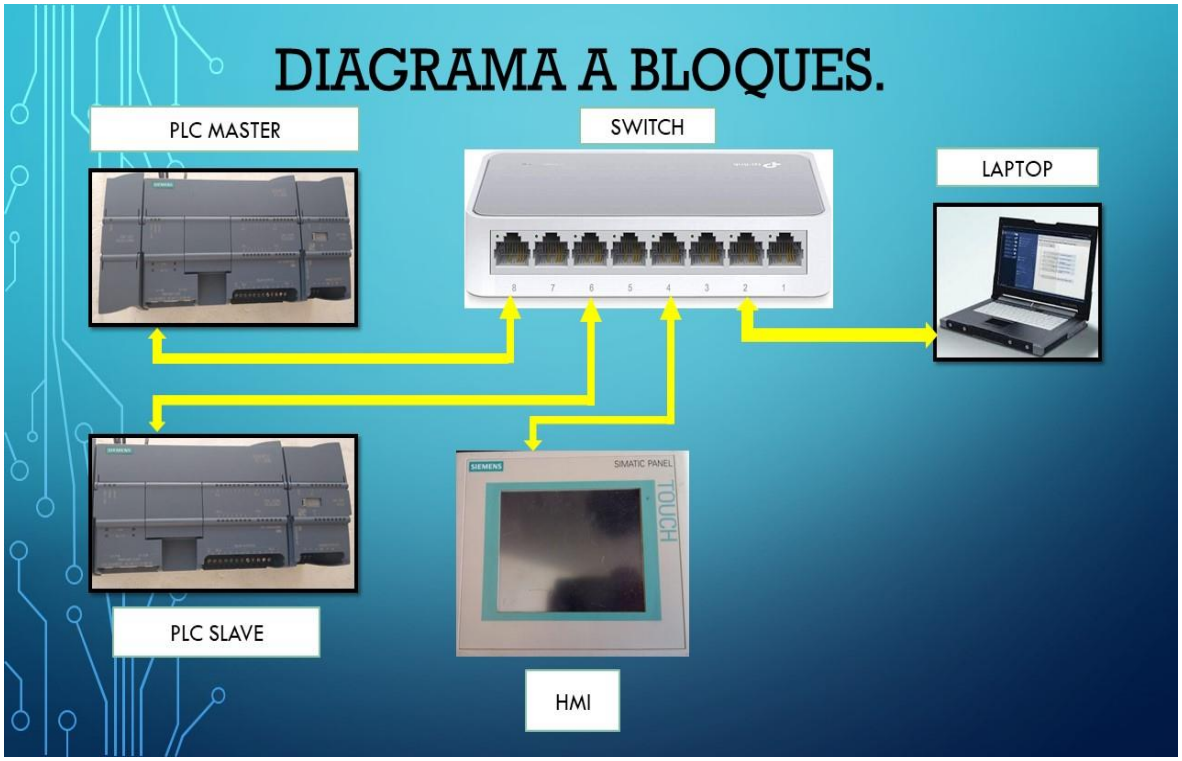


Figura 3.1. Diagrama a bloques de la conexión de los equipos para la red MODBUS TCP.

3.1.1 Conexión de los equipos industriales.

Como primer paso se realizaron las correspondientes conexiones eléctricas de los dispositivos. Un PLC se conectó a la red eléctrica que es de 127 Volts, este PLC tiene integrada una fuente de alimentación de 24 Volts DC, de modo que, se conecta el segundo PLC y el Touch Panel a 24 Volts DC. De lo contrario si no se tuviera esta fuente, se tendría que integrar una fuente de 127 Volts AC a 24 Volts DC (véase figura 3.2).

Una vez que los equipos se encuentran energizados deberá encenderse un led de color verde, esto nos indicará que todo está bien y se puede continuar con la configuración. En dado caso que, se ilumine un led de color amarillo o rojo, se tendría que revisar los voltajes de alimentación del equipo, si los leds siguen encendidos, se podría poner online con el equipo para hacer un diagnóstico.

Figura 3.2. Conexión eléctrica de los dispositivos industriales.

3.1.2 Configuración software de los dispositivos.

En segundo lugar, se ingresó a la plataforma TIA PORTAL, se creó un proyecto nuevo y se dan de alta los dispositivos con sus correspondientes módulos de entradas y salidas tomando de referencia el número de parte de cada equipo ya antes mencionados. Cabe mencionar que, todos los dispositivos que se vayan a conectar y comunicar entre sí, deben de estár conectados a la misma red PROFINET (véase figura 3.3).

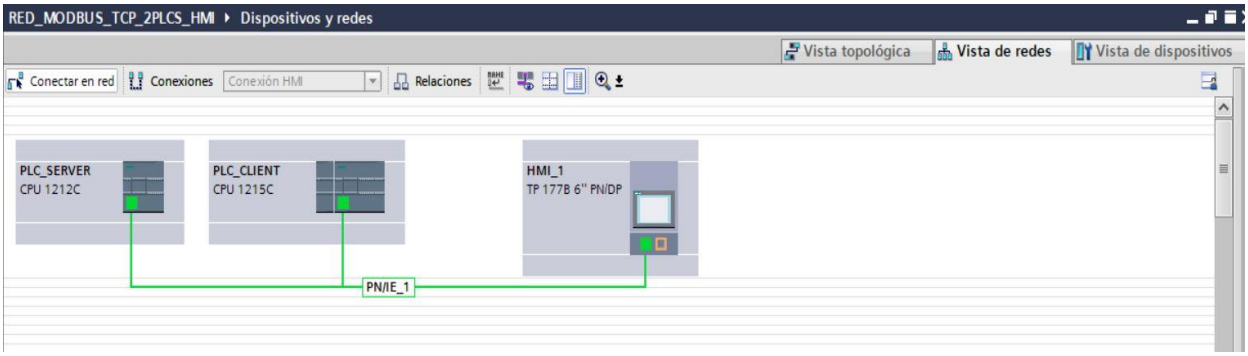


Figura 3.3. Vista de redes.

Para colocar los dispositivos en la misma red, basta con posicionarse en vista de redes, donde se podrá observar todos los dispositivos que se han dado de alta, en seguida, se selecciona el cuadro de color verde que se encuentra en la parte inferior izquierda de cada dispositivo y se une con los demás. Esto aplica para todos los dispositivos que vayan a comunicarse, (véase figura 3.4).

Lo más importante que debe resaltarse es cambiar las IPs, esto se realizó haciendo doble clic en el cuadro verde de cada equipo conectado a la conexión PN/IE_1, después apareció un recuadro con las propiedades de la IP, allí es donde se cambió la IP. Para esta red se configuraron las siguientes IPs.

- PLC_SERVER CPU 1212C: IP: 192.168.1.40
- PLC_CLIENT CPU 1215C: IP: 192.168.1.65
- HMI_1 TP 177B 6” PN/DP: IP: 192.168.1.15
- LAPTOP: IP: 192.168.1.30

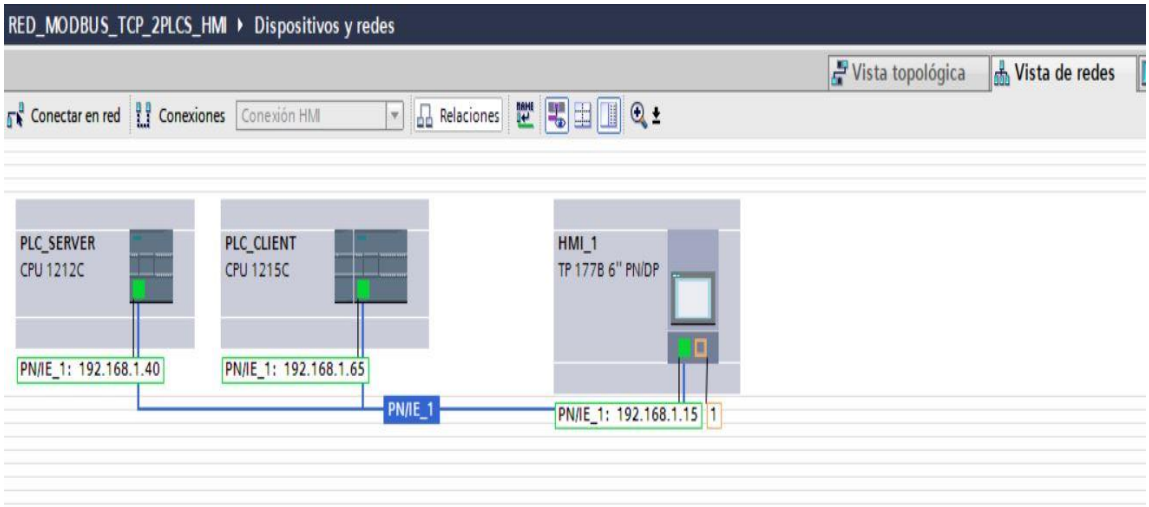


Figura 3.4. Vista de redes/ IPs.

3.1.3 Programación del PLC-CLIENTE.

Se programó el PLC 1215C como CLIENTE con la instrucción “MB_CLIENT_DB”. Lo que hace esta función, es indicarle al PLC, que tomará el control de la red de comunicación y le enviará peticiones al PLC- SERVER. De lo anterior resulta que, debe configurarse los parámetros en el bloque (véase figura 3.5).

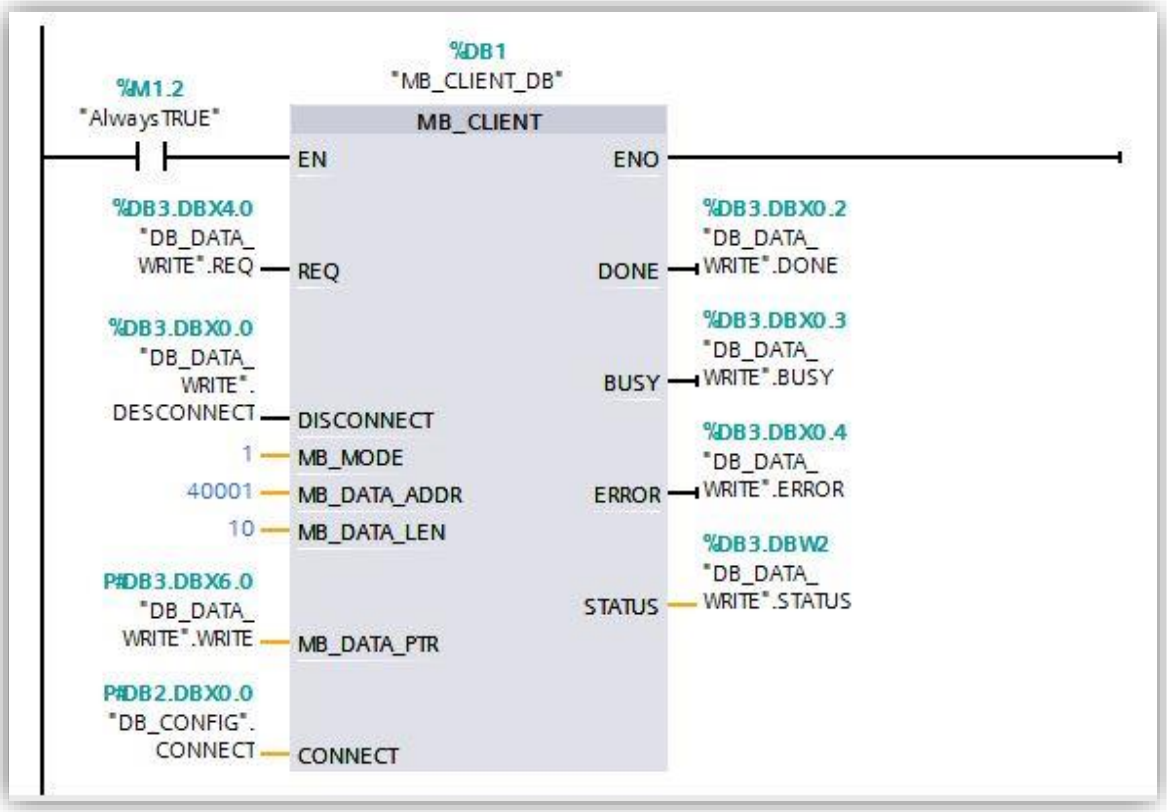


Figura 3.5. Instrucción MB_CLIENT.

El bloque “MB_CLIENT_DB”, inicializa y permite la comunicación entre dos PLCs utilizando el protocolo MODBUS TCP CLIENTE-SERVIDOR por medio de la conexión PROFINET de los CPUs, la cual se creó anteriormente. Para utilizar esta instrucción no se necesita algún módulo de hardware extra. Así mismo, recibir una respuesta, controlar la desconexión de un SERVIDOR MODBUS TCP y, además, podría indicar si hay algún error en la comunicación o datos a enviar.

En la tabla 3.1, de izquierda a derecha; en la primera columna se indica el tipo y el parámetro que debe configurarse, en la segunda columna se indica si es un parámetro de entrada o salida. En la tercera columna se indica el tipo de dato que se va a configurar y declarar para cada uno y por último en la cuarta columna se encuentra una descripción.

Parameter and type		Data type	Description
REQ	In	Bool	FALSE = No Modbus communication request TRUE = Request to communicate with a Modbus TCP server
DISCONNECT	IN	Bool	The DISCONNECT parameter allows your program to control connection and disconnection with a Modbus server device. If DISCONNECT = 0 and a connection does not exist, then MB_CLIENT attempts to make a connection to the assigned IP address and port number. If DISCONNECT = 1 and a connection exists, then a disconnect operation is attempted. Whenever this input is enabled, no other operation will be attempted.
CONNECT_ID	IN	UInt	The CONNECT_ID parameter must uniquely identify each connection within the PLC. Each unique instance of the MB_CLIENT or MB_SERVER instruction must contain a unique CONNECT_ID parameter.
IP_OCTET_1	IN	USInt	Modbus TCP server IP address: Octet 1 8-bit part of the 32-bit IPv4 IP address of the Modbus TCPserver to which the client will connect and communicate using the Modbus TCP protocol.
IP_OCTET_2	IN	USInt	Modbus TCP server IP address: Octet 2
IP_OCTET_3	IN	USInt	Modbus TCP server IP address: Octet 3
IP_OCTET_4	IN	USInt	Modbus TCP server IP address: Octet 4
IP_PORT	IN	UInt	Default value = 502: The IP port number of the server to which the client will attempt to connect and ultimately communicate using the TCP/IP protocol.

Parameter and type		Data type	Description
MB_MODE	IN	USInt	Mode Selection: Assigns the type of request (read, write, or diagnostic). See the Modbus functions table below for details.
MB_DATA_ADDR	IN	UDInt	Modbus starting Address: Assigns the starting address of the data to be accessed by MB_CLIENT. See the Modbus functions table below for valid addresses.
MB_DATA_LEN	IN	UInt	Modbus data Length: Assigns the number of bits or words to be accessed in this request. See the Modbus functions table below for valid lengths
MB_DATA_PTR	IN_OUT	Variant	Pointer to the Modbus data register: The register buffers data going to or coming from a Modbus server. The pointer must assign a standard global DB or a M memory address.
DONE	OUT	Bool	The DONE bit is TRUE for one scan, after the last request was completed with no error.
BUSY	OUT	Bool	• 0 - No MB_CLIENT operation in progress • 1 - MB_CLIENT operation in progress
ERROR	OUT	Bool	The ERROR bit is TRUE for one scan, after the MB_CLIENT execution was terminated with an error. The error code value at the STATUS parameter is valid only during the single cycle where ERROR = TRUE.
STATUS	OUT	Word	Execution condition code

Tabla 3.1. Parámetros y tipo de datos de la instrucción MB_CLIENT.

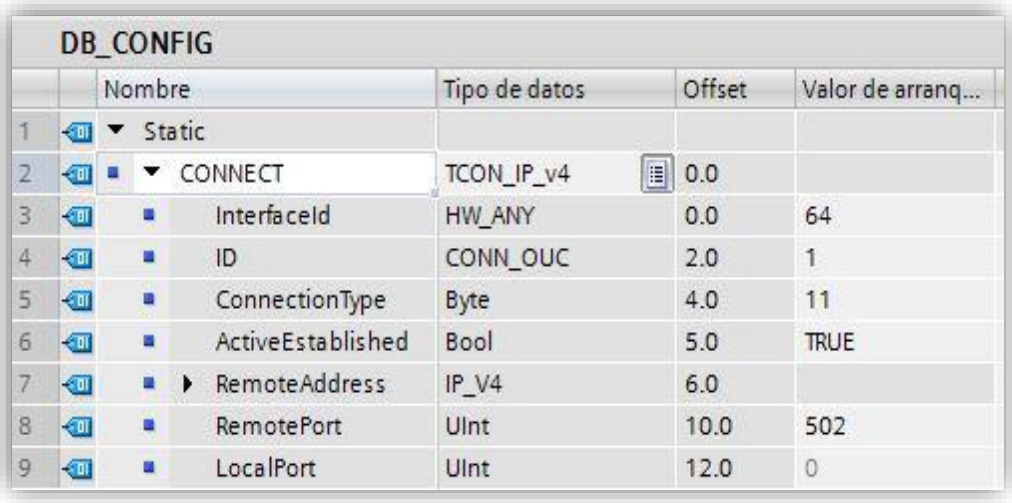
Cada conexión de cliente concurrente debe cumplir las siguientes condiciones:

- Cada conexión “MB_CLIENT” tendrá que utilizar un DB de instancia exclusivo.

- Cada conexión “MB_CLIENT” tendrá que establecerse con una dirección IP de servidor exclusiva. Se admite una sola conexión para cada puerto.
- Cada conexión “MB_CLIENT” tendrá que utilizar una ID de conexión exclusiva.
- Para cada DB de instancia de la instrucción debe utilizarse la correspondiente ID de conexión. Las ID de conexión y los DB de instancia se agrupan por pares y deben ser unívocos para cada conexión.
- Debe usarse una ID de conexión diferente con cada DB de instancia. Resumiendo, el DB de instancia y la ID de conexión están emparejadas y deben ser únicas para cada conexión.

El siguiente paso fue configurar los parámetros de esta instrucción, para ello también se creó un nuevo bloque de datos global. Seguidamente se accedió y se creó el parámetro CONNECT.
Esta variable asigna datos utilizados para establecer una conexión PROFINET.
Para este caso se nombró el bloque de datos como “DB_CONFIG” y una variable estática con un nombre significativo “CONNECT”, asignándole un tipo de datos del sistema “TCON_IP_v4”

Se seleccionó la variable para agregar los valores correspondientes a los parámetros de conexión, como se muestra en la siguiente figura 3.6.



DB_CONFIG

	Nombre	Tipo de datos	Offset	Valor de arranq...
1	Static			
2	CONNECT	TCON_IP_v4	0.0	
3	InterfaceId	HW_ANY	0.0	64
4	ID	CONN_OUC	2.0	1
5	ConnectionType	Byte	4.0	11
6	ActiveEstablished	Bool	5.0	TRUE
7	RemoteAddress	IP_V4	6.0	
8	RemotePort	UInt	10.0	502
9	LocalPort	UInt	12.0	0

Figura 3.6. Data block para configuración de la variable CONNECT.

En este PLC se escribieron los datos para el PLC 1212C AC/DC/RLY, el cual fue el servidor. Siendo así, se creó un bloque de datos global “DB_DATA_WRITE” y se dió de

alta otras variables que más adelante se utilizaron para programar la interfaz (véase figura 3.7).

DB_DATA_WRITE

	Nombre	Tipo de datos	Offset	Valor de arranq...
1	Static			
2	DESCONNECT	Bool	0.0	false
3	CONNECT	Bool	0.1	false
4	DONE	Bool	0.2	false
5	BUSY	Bool	0.3	false
6	ERROR	Bool	0.4	false
7	STATUS	Word	2.0	16#0
8	REQ	Bool	4.0	false
9	MB_DATA_PTR	Bool	4.1	false
10	WRITE_DATE	Bool	4.2	false
11	WRITE	Array[0..9] of Int	6.0	

Figura 3.7. Bloque de datos para escritura.

3.1.4 Programación del PLC-SERVIDOR.

Después se programó el PLC 1212 C, el cual fue el servidor, se le asignó el programa con la instrucción “MB_SERVER_DB”, (véase figura 3.8.), con esta instrucción se comunicó como servidor en la red MODBUS TCP a través del puerto PROFINET de la CPU. No se utilizaron módulos de hardware de comunicación extras. MB_SERVER puede aceptar una orden para conectarse a un cliente MODBUS TCP, recibir una orden de función MODBUS y enviar un mensaje de respuesta.

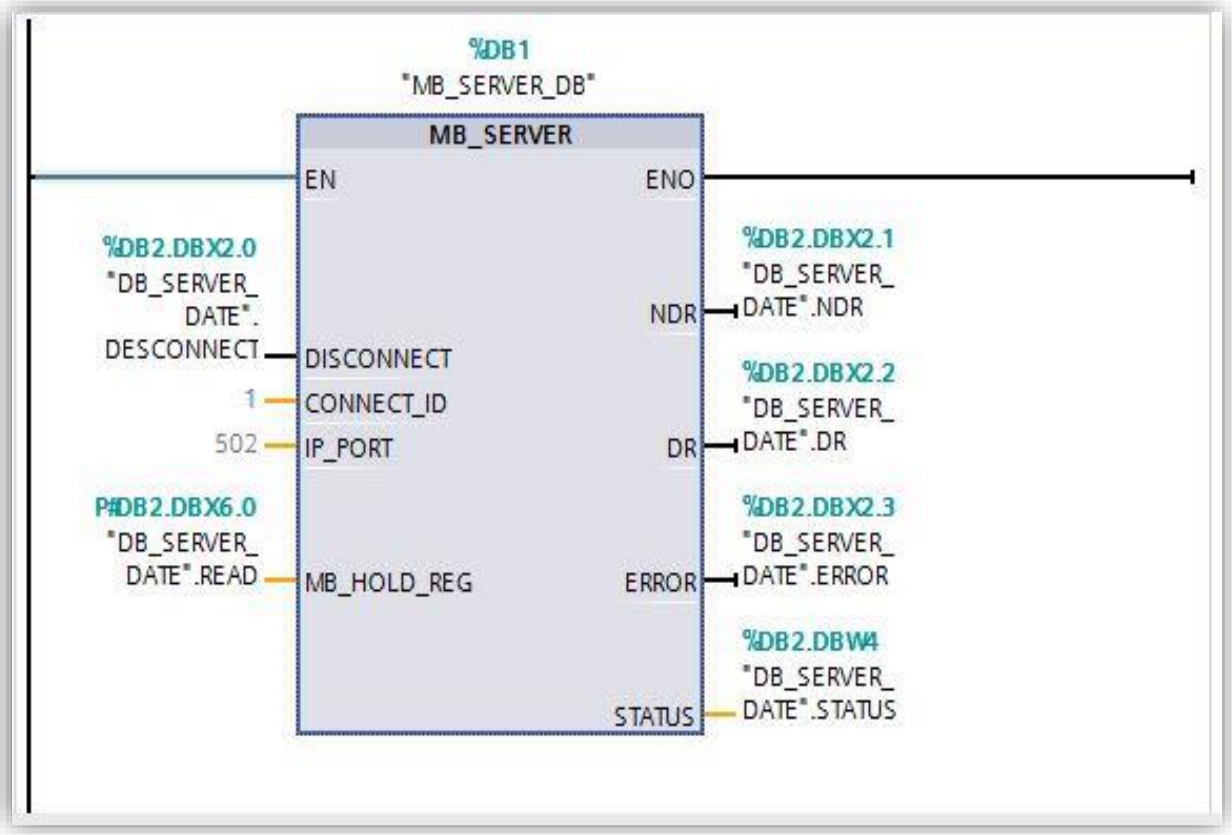


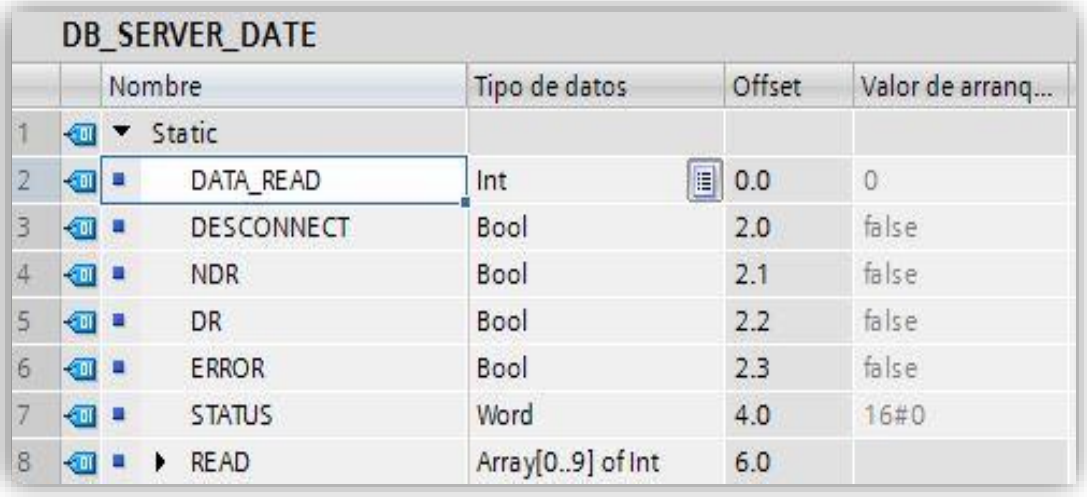
Figura 3.8. Instrucción MB_SERVER

En la siguiente tabla 3.2, de izquierda a derecha; en la primera columna se indica el tipo y el parámetro que debe configurarse, en la segunda columna se menciona si es un parámetro de entrada o salida. En la tercera columna muestra el tipo de dato que se va a configurar y declarar para cada uno y por último en la cuarta columna se encuentra una descripción.

Parameter and type		Data type	Description
DISCONNECT	IN	Bool	MB_SERVER attempts to make a "passive" connection with a partner device. This means that the server is passively listening for a TCP connection request from any requesting IP address. If DISCONNECT = 0 and a connection does not exist, then a passive connection can be initiated. If DISCONNECT = 1 and a connection exists, then a disconnect operation is initiated. This allows your program to control when a connection is accepted. Whenever this input is enabled, no other operation will be attempted.
CONNECT_ID	IN	UInt	CONNECT_ID uniquely identifies each connection within the PLC. Each unique instance of the MB_CLIENT or MB_SERVER instruction must contain a unique CONNECT_ID parameter.
IP_PORT	IN	UInt	Default value = 502: The IP port number that identifies the IP port that will be monitored for a connection request from a Modbus client. These TCP port numbers are not allowed for a MB_SERVER passive connection: 20, 21, 25, 80, 102, 123, 5001, 34962, 34963, and 34964.
MB_HOLD_REG	IN_OUT	Variant	Pointer to the MB_SERVER Modbus holding register: The holding register must either be a standard global DB or a M memory address. This memory area is used to hold the values a Modbus client is allowed to access using Modbus register functions 3 (read), 6 (write), and 16 (write).
NDR	OUT	Bool	New Data Ready: 0 = No new data, 1 = Indicates that new data has been written by a Modbus client
DR	OUT	Bool	Data Read: 0 = No data read, 1 = Indicates that data has been read by a Modbus client.
ERROR	OUT	Bool	The ERROR bit is TRUE for one scan, after MB_SERVER execution was terminated with an error. The error code value at the STATUS parameter is valid only during the single cycle where ERROR = TRUE.
STATUS	OUT	Word	Execution condition code

Tabla 3.2. Parámetros y tipo de datos de la instrucción MB_SERVER.

De la misma manera se agregó un bloque de datos para la configuración de los parámetros y recibir los datos del cliente (véase figura 3.9).



DB_SERVER_DATE

	Nombre	Tipo de datos	Offset	Valor de arranq...
1	Static			
2	DATA_READ	Int	0.0	0
3	DESCONNECT	Bool	2.0	false
4	NDR	Bool	2.1	false
5	DR	Bool	2.2	false
6	ERROR	Bool	2.3	false
7	STATUS	Word	4.0	16#0
8	READ	Array[0..9] of Int	6.0	

Figura 3.9. Bloque de datos para lectura.

Cada conexión de servidor concurrente debe cumplir las siguientes condiciones:

- Cada conexión MB_SERVER tendrá que usar un DB de instancia exclusivo.
- Cada conexión MB_SERVER tendrá que asignar un número de puerto IP exclusivo. Solo se admite una conexión por puerto.
- Cada conexión MB_SERVER tendrá que asignar un ID de conexión univoco.
- El MB_SERVER tendrá que ser llamado individualmente para cada conexión (con su DB de instancia respectivo).
- El ID de conexión tendrá que ser univoco para cada conexión individual. Debe usarse un ID de conexión univoco con cada DB de instancia. El DB de instancia y el ID de conexión están emparejados y deben ser unívocos para cada conexión.

A los bloques de datos y funciones se les puede definir una contraseña con el objetivo de proteger la información de personal no autorizado y ajeno al proceso (véase figura 3.10). Solo se podrá acceder a los programas de los controladores con la contraseña proporcionada. Esta función es conocida como *KNOW HOW* y se encuentra dentro de la plataforma de programación, esta es una herramienta de gran utilidad cuando se quiere proteger información de los procesos para no ser alterada.

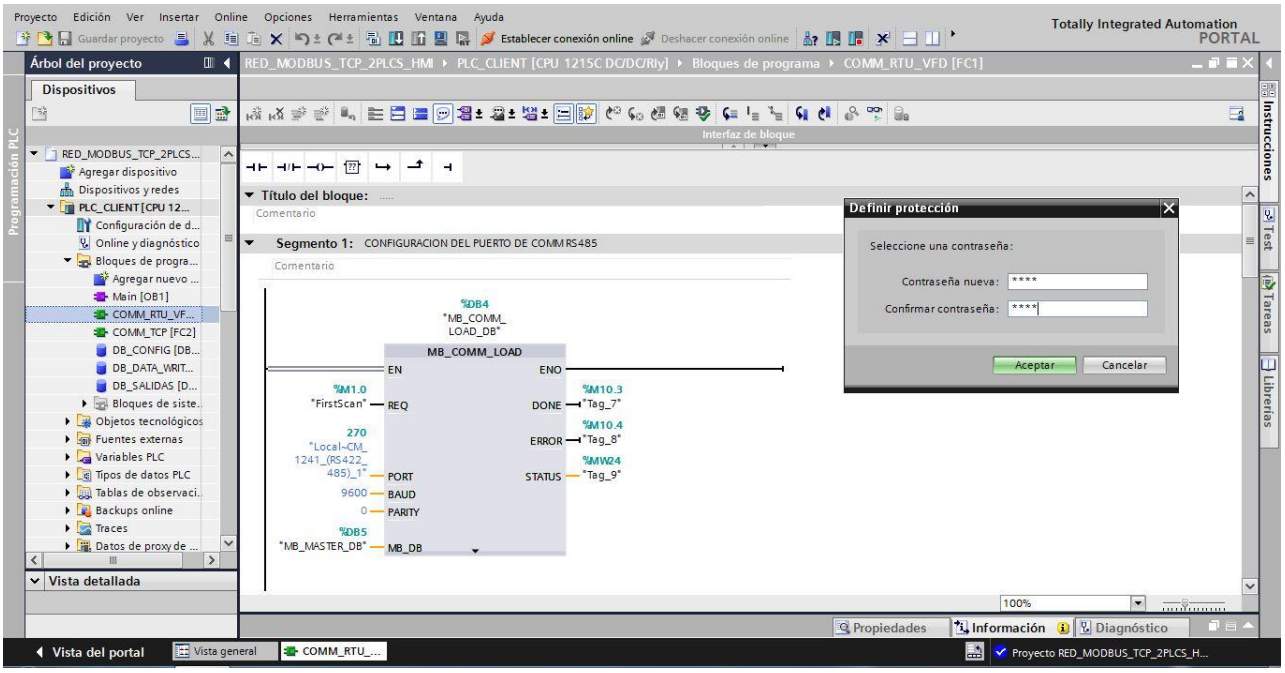


Figura 3.10. Activación de la protección utilizando KNOW HOW.

3.1.5 Programación del HMI.

Para la visualización de los datos enviados y recibidos, se realizó la programación del SIMATIC PANEL. Éste HMI se utilizó para ingresar los datos, los cuales, se escriben en el PLC-Cliente y se envían al PLC-Servidor, así mismo, los recibe y leerá. Se diseñaron y programaron 4 pantallas, las cuales se muestran a continuación:

En la programación del HMI se programó un menú, con el cual se pudo navegar por las diferentes ventanas.

- Ventana: PRESENTACIÓN
- Ventana: ACTIVAR
- Ventana: ENVIO DE DATOS
- Ventana: PROCESO

En la ventana principal (véase figura 3.11), muestra la presentación, aquí se visualiza la información del proyecto. En esta ventana aparte de la información proporcionada nos direcciona a la siguiente ventana, la cual es “ACTIVAR”. Además, tenemos una opción llamada RUNTIME, esta opción sirve para salir del programa, después se abrirá una ventana con opciones del sistema.

Figura 3.11. HMI Pantalla 1: Presentación.

En la segunda ventana, se encuentran las opciones para la activación tanto del cliente como del servidor. Para activar la red de comunicación del Cliente-Servidor, se activa el parámetro REQ. Esto permitirá iniciar la comunicación y realizar la escritura o lectura de datos, (véase figura 3.12).

Al finalizar la transferencia de datos, se deberá de desactivar la comunicación, para ello, se activan los parámetros “DESCONNECT- CLIENT” Y “DESCONNECT-SERVER”. Luego el parámetro REQ se desactivará y finalmente se desactivará las opciones “DESCONNECT- CLIENT” Y “DESCONNECT-SERVER” en dado caso que ya no se vaya a trasferir más datos.

Figura 3.12. HMI Pantalla 2: Activación de la Comunicación MODBUS TCP.

Para ingresar a la siguiente pantalla basta con dar clic en el botón “ENVIO DE DATOS”. En la tercera ventana, se encuentran los espacios para ingresar los datos a enviar, para este caso se pueden transferir hasta diez datos (véase figura 3.13). En el recuadro se da clic y se abrirá un teclado para ingresar el dato numérico que se requiera enviar. No es necesario ingresar 10 datos si no los necesarios.

Figura 3.13. HMI Pantalla 3: Datos a transferir al PLC-SERVIDOR.

Para visualizar los datos enviados se hace clic en el botón “PROCESO”. Esta es la cuarta ventana del HMI (véase figura 3.14), aquí se podrán ver los datos recibidos en el PLC-Server que el PLC-Client haya enviado.

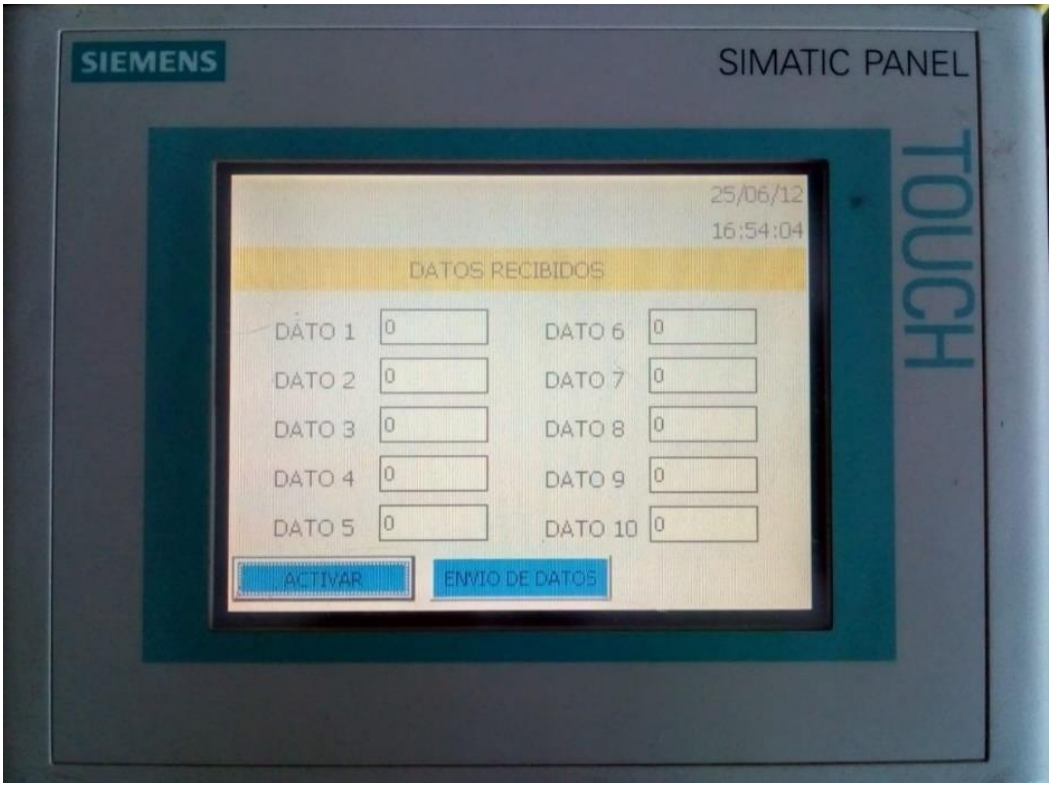


Figura 3.14. HMI Pantalla 4: Datos recibidos en el PLC-SERVIDOR

3.2. Implementación de la red de comunicación utilizando el protocolo Modbus RTU.

Ahora se presentará la segunda etapa del proyecto, en la cual, se programó un PLC y el variador de frecuencia para implementar la red MODBUS RTU. Para esto se utilizaron los siguientes dispositivos y materiales:

- Una laptop con la plataforma TIA PORTAL V14
- Un SIMATIC PANEL TP177
- Un PLC SIEMENS, S7-1215C DC/DC/RLY
- Una tarjeta de comunicación CM 1241 RS 422/ RS485
- Un VFD Powerflex 4 Allen Bradley
- Un switch TP LINK
- Un conector DB9 macho
- Un conector RJ45
- Cable UTP

3.2.1. Configuración software de los dispositivos.

Para la red MODBUS RTU, se conectó el variador de frecuencia al módulo de comunicación RS485 del PLC S7-1215 C, esto, por un lado, por otro lado, se desconectó el PLC S7-1212 C, ya que solo un PLC se utilizó como el maestro, el cual controló el VFD (esclavo) (véase figura 3.15). El HMI siguió estando disponible, porque se utilizó la interfaz. Las IPs no cambian por lo que siguen siendo las mismas.

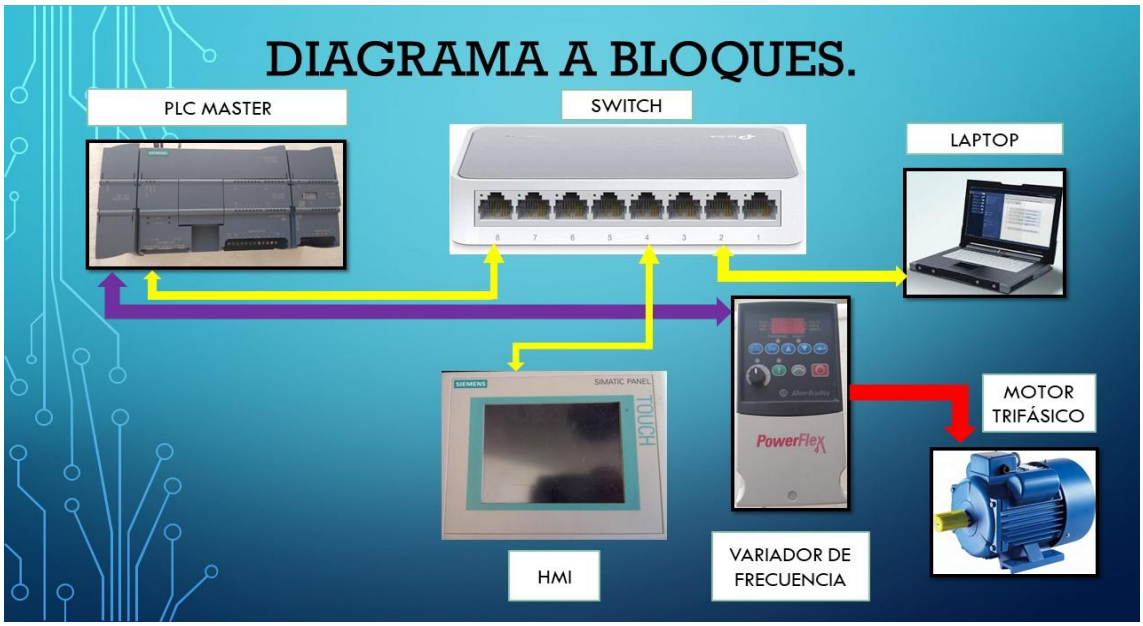


Figura 3.15. Diagrama a bloques, red MODBUS RTU.

La comunicación MODBUS RTU (Remote Terminal Unit) es un protocolo de comunicaciones en red estándar, este protocolo utiliza conexiones RS232 o RS485 para transferencia serie de datos entre dispositivos de red MODBUS.
MODBUS RTU utiliza una red maestro-esclavo en la que un solo dispositivo maestro inicia todas las comunicaciones y los esclavos solo pueden responder a una petición del maestro. El maestro envía una petición a una dirección de esclavo y solo esa dirección de esclavo responde a la orden.

En principio, para el desarrollo e implementación de esta red de comunicación es necesario conocer el manual de usuario del VFD puesto que, se utilizó el documento para conocer que parámetros debían configurarse y que valores debiera tener cada uno de ellos.
A continuación, se muestra la tabla 3.3, la cual se tomó de referencia del manual de usuario de Powerflex 4 para la configuración del dispositivo.

Parameter Configuration

The following PowerFlex 4 parameters are used to configure the drive to operate on a network.

Parameter	Details	Reference
P036 [Start Source]	Set to 5 "RS485 (DSI) Port" if Start is controlled from the network.	Page 3-9
P038 [Speed Reference]	Set to 5 "RS485 (DSI) Port" if the Speed Reference is controlled from the network.	Page 3-11
A103 [Comm Data Rate]	Sets the data rate for the RS485 (DSI) Port. All nodes on the network must be set to the same data rate.	Page 3-23
A104 [Comm Node Addr]	Sets the node address for the drive on the network. Each device on the network requires a unique node address.	Page 3-23
A105 [Comm Loss Action]	Selects the drive's response to communication problems.	Page 3-24
A106 [Comm Loss Time]	Sets the time that the drive will remain in communication loss before the drive implements A105 [Comm Loss Action].	Page 3-24
A107 [Comm Format]	Sets the transmission mode, data bits, parity and stop bits for the RS485 (DSI) Port. All nodes on the network must be set to the same setting.	Page 3-24

Tabla 3.3. Parámetros de configuración del VFD.

Son 7 parámetros, los cuales se configuraron con base en las indicaciones para comunicar el variador de frecuencia por medio de una red de comunicación MODBUS RTU.

3.2.2. Códigos de funciones MODBUS.

La interfaz periférica utilizada en los variadores Powerflex 4 admite algunos de los códigos de función MODBUS. Para comunicarse a través del protocolo MODBUS RTU solo es necesario conocer dos funciones, que son: escritura y lectura. En la tabla 3.4, se puede observar que se utilizan dos registros. El registro 03 es para la función de lectura y el registro 06 es para la función de escritura.

MODBUS Function Code (Decimal)	Command
03	Read Holding Registers
06	Preset (Write) Single Register

Tabla 3.4. Códigos de funciones MODBUS.

3.2.3. Comandos lógicos.

En la tabla 3.5, se encuentran los datos para los comandos lógicos. La escritura se realiza en la dirección del registro 8192. (Esta tabla fue tomada del manual del variador de frecuencia). Una vez que se haya configurado el variador Powerflex 4 podrá aceptar los comandos.

Logic Command		
Address (Decimal)	Bit(s)	Description
8192	0	1 = Stop, 0 = Not Stop
	1	1 = Start, 0 = Not Start
	2	1 = Jog, 0 = No Jog
	3	1 = Clear Faults, 0 = Not Clear Faults
	5,4	00 = No Command
		01 = Forward Command
		10 = Reverse Command
		11 = No Command
	6	Not Used
	7	Not Used
	9,8	00 = No Command
		01 = Accel Rate 1 Enable
		10 = Accel Rate 2 Enable
		11 = Hold Accel Rate Selected
	11,10	00 = No Command
		01 = Decel Rate 1 Enable
		10 = Decel Rate 2 Enable
		11 = Hold Decel Rate Selected
	14,13,12	000 = No Command
		001 = Freq. Source = P036 [Start Source]
		010 = Freq. Source = A069 [Internal Freq]
		011 = Freq. Source = Comms (Addr 8193)
		100 = A070 [Preset Freq 0]
		101 = A071 [Preset Freq 1]
		110 = A072 [Preset Freq 2]
		111 = A073 [Preset Freq 3]
	15	Not Used

Tabla 3.5. Escritura de datos de comandos lógicos.

Para escribir en el registro 8192, se debió haber enviado el dato en formato decimal, es decir, se convirtió el dato binario a valor decimal. Como es un registro de 16 bits cada bit corresponde a un valor de una instrucción con base en la tabla de arriba.

3.2.4. Referencia de velocidad.

La referencia de velocidad para el dispositivo PowerFlex 4 se controló a través de la red de comunicación. Esto se pudo realizar enviando la función de escritura a la dirección de registro 8193 (Referencia). Es importante dejar claro que un valor decimal ingresado como xxx.x donde el punto decimal es fijo. Por ejemplo, un valor decimal “240” es igual a 24.0 Hz y “600” es igual a 60 Hz.

En la figura 3.16, se muestra el variador, este dispositivo tiene dos maneras de ser controlado y de configurarse. La primera es por medio de su panel que trae integrado, dentro del cual tiene un menú para acceder a los parámetros de programación. También tiene integrado un display, donde se muestra la información que contiene el dispositivo y la segunda manera es por medio del puerto de comunicación RS-485.

También trae un potenciómetro, con el cual se puede ir variando la velocidad de salida, un botón para hacer el cambio de giro. En la parte izquierda del display tiene tres indicadores, donde se muestra el sentido de giro (FORWARD o REVERSE), y si está en modo RUN. Así mismo, tiene otros indicadores como lo son: PROGRAM Y FAULT. El primero indica si el variador se encuentra en modo de programación o fuera de este modo y el segundo indica si el variador se fue a falla.

Figura 3.16. Variador de frecuencia Powerflex 4.

3.2.5. Estado lógico

En la tabla 3.6, la cual, fue tomada del manual del variador de frecuencia PowerFlex 4, se muestra que para la función de lectura de datos del estado lógico debe de leerse en la dirección del registro 8448. Este también es un registro de dos Bytes, en los cuales, cada bit corresponde y muestra el estado lógico de cada función del dispositivo. Así mismo, cada instrucción tiene un valor lógico 1 ó 0, dependiendo de la función que realice el variador, solo basta con cambiar de estado el bit y convertir ese número de binario a decimal.

Logic Status		
Address (Decimal)	**Bit(s)**	**Description**
8448	0	1 = Ready, 0 = Not Ready
	1	1 = Active (Running), 0 = Not Active
	2	1 = Cmd Forward, 0 = Cmd Reverse
	3	1 = Rotating Forward, 0 = Rotating Reverse
	4	1 = Accelerating, 0 = Not Accelerating
	5	1 = Decelerating, 0 = Not Decelerating
	6	1 = Alarm, 0 = No Alarm
	7	1 = Faulted, 0 = Not Faulted
	8	1 = At Reference, 0 = Not At Reference
	9	1 = Reference Controlled by Comm
	10	1 = Operation Cmd Controlled by Comm
	11	1 = Parameters have been locked
	12	Digital Input 1 Status
	13	Digital Input 2 Status
	14	Not Used
	15	Not Used

Tabla 3.6. Datos de estatus lógico.

3.2.6. Lectura de frecuencia del VFD.

La lectura de la frecuencia de salida del dispositivo PowerFlex 4 se hizo a través de la red de comunicación mandado la función de Lectura a la dirección de registro 8451 (Feedback) Vale la pena aclarar que una lectura de un valor decimal, donde el punto decimal es fijo. Por ejemplo, si se hace la lectura de un decimal “347” es igual a 34.7 Hz y “330” es igual a 33.0 Hz.

3.2.7. Programación del PLC-Master.

Una vez que se haya configurado el variador de frecuencia y haber dado de alta los dispositivos que se comunicaron dentro de la red (véase figura 3.17), se hizo la programación del PLC.

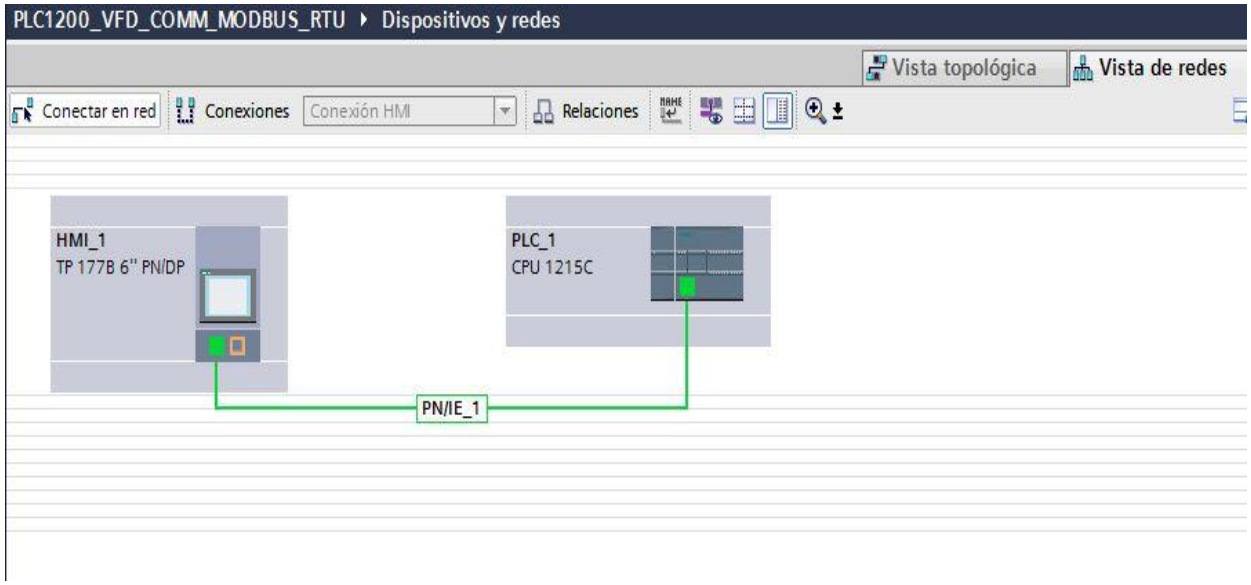


Figura 3.17. Vista de redes.

Se hizo la configuración del PLC, el cual fue el maestro y el variador de frecuencia fue el esclavo en la red de comunicación, desde la plataforma se integró la tarjeta de comunicación con puerto RS485 (véase figura 3.18). Dentro de las propiedades del sistema del PLC se activaron las marcas de ciclo ya que se utilizaron en la programación

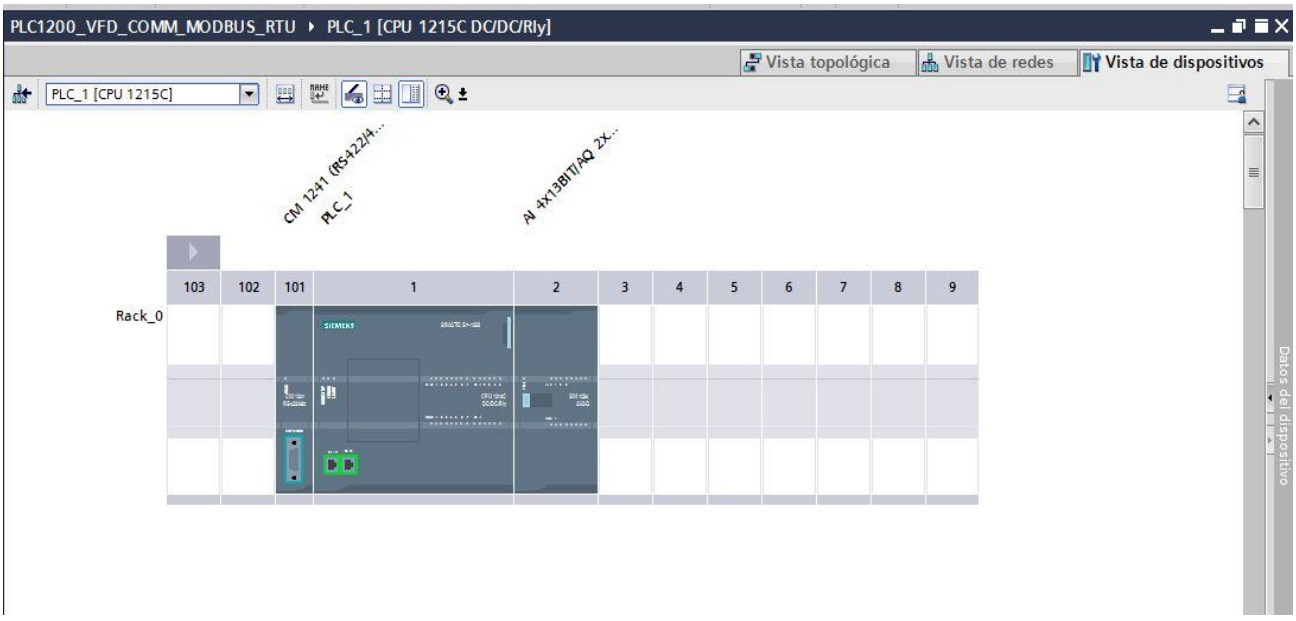


Figura 3.18. TIA PORTAL: vista de dispositivos.

Se agregó el bloque “MB_COMM_LOAD” (véase figura 3.19). Este bloque se utilizó para inicializar y configuración del puerto de comunicación RS485. En este bloque se ajustaron los parámetros de la comunicación. Es muy importante que el bloque “MB_COMM_LOAD” deba llamarse en el primer ciclo de programa mediante la activación de una marca del sistema que es FirstScan. Después se insertó el bloque de comunicación, se configuró el PORT para este parámetro se seleccionó el nombre del puerto que tiene la tarjeta de comunicación configurada, entonces se colocó la opción “Local-CM_1241_(RS422_485) _1”.

- **REQ**: un cambio de señal low a high (flanco ascendente).
- **BAUD**: selección de la velocidad de transferencia
- **PARITY**: selección de paridad
- El parámetro “**MB_DB**” referencia al bloque de datos de instancia que utilizan las instrucciones “MODBUS_MAESTRO” o “MODBUS_SLAVE”. Una vez insertadas las instrucciones “MODBUS_MAESTRO” o “MODBUS_SLAVE” en el programa, el identificador de DB aparece en la lista desplegable de parámetros disponibles en la conexión del cuadro MB_DB.

Los parámetros DONE, ERROR y STATUS son de salida.

- **DONE**: este bit es TRUE durante un ciclo tras haberse finalizado la última orden sin errores.
- **ERROR**: este bit es TRUE durante un ciclo tras haberse finalizado la última orden con un error. El valor del código de error en el parámetro STATUS solo es válido durante un único ciclo en que ERROR = TRUE.
- **STATUS**: muestra el código de condición de ejecución.

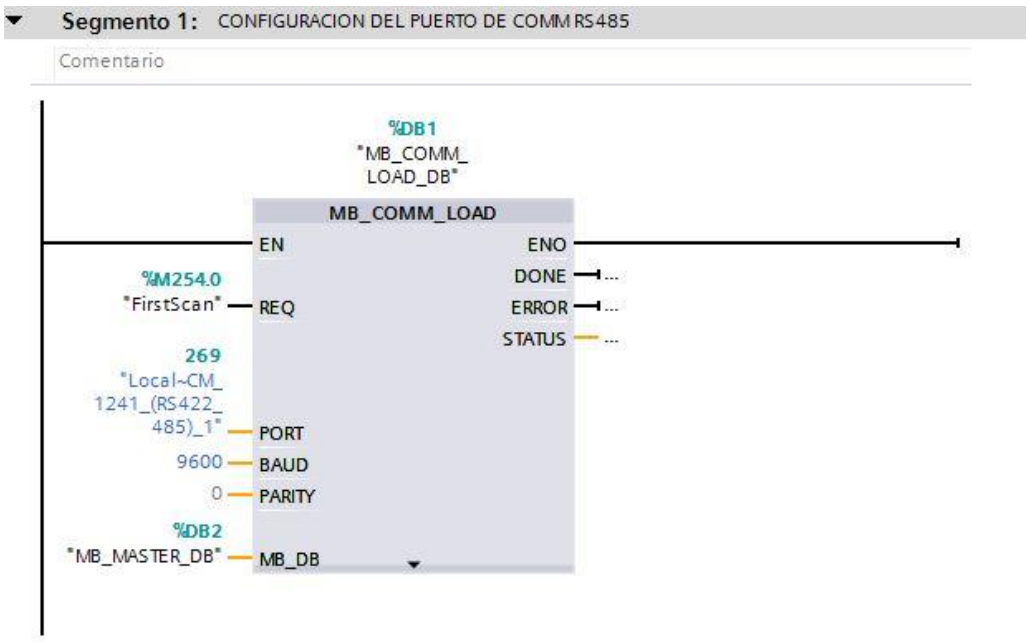


Figura 3.19. Bloque “MB_COMM_LOAD”

Después se agregó el bloque “MB_MASTER_DB” (véase figura 3.20), esta instrucción comunicó al PLC como el maestro MODBUS dentro de la red utilizando un puerto configurado por una ejecución anterior de la instrucción “MODBUS_COMM_LOAD”. Un bloque de datos de instancia se asignó automáticamente cuando la instrucción “MODBUS_MASTER” se insertó en el programa. El bloque de datos de instancia “MODBUS_MASTER” se utilizó cuando se especificó el parámetro MB_DB para la instrucción “MODBUS_COMM_LOAD”.
Los datos que se configuraron son los siguientes:

- REQ: este parámetro hace la solicitud para transmitir datos al esclavo MODBUS.
- MB_ADDR: dirección de estación MODBUS RTU.
- MODE: selección de modo
- DATA_ADDR: dirección inicial en el esclavo
- DATA_LEN: longitud de datos
- DATA_PTR: puntero a los datos
- DONE: este bit es TRUE durante un ciclo tras haberse finalizado la última orden sin errores.
- BUSY: muestra si alguna operación está en curso.
- ERROR*:* este bit es TRUE durante un ciclo tras haberse finalizado la última orden con un error.
- STATUS: código de condición de ejecución.

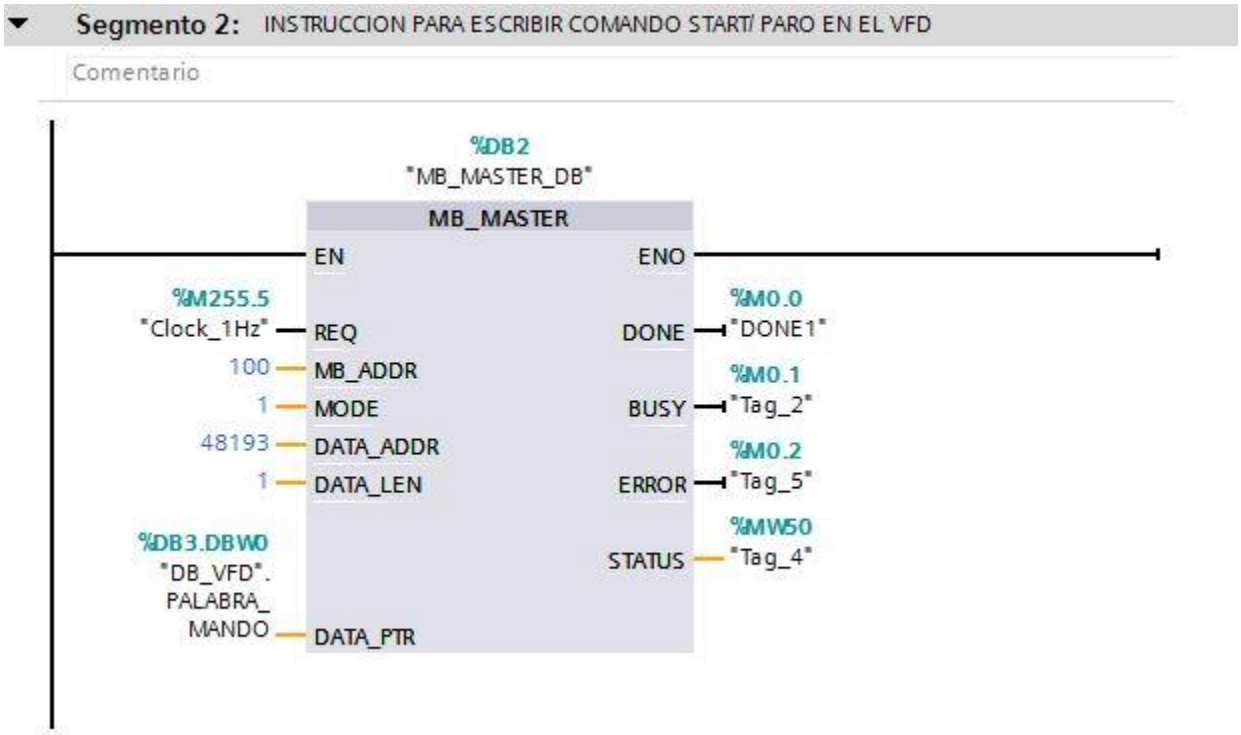


Figura 3.20. Bloque MB_MASTER

El siguiente paso fue agregar la instrucción MOVE (véase figura 3.21) para transferir el valor de 1 al registro de escritura. Con el valor de 1 en sistema decimal se mandó a STOP el actuador en dado caso que se encuentre en funcionamiento, basta con hacer clic en el botón STOP desde el HMI.

Figura 3.21. Instrucción para la orden de STOP.

Seguidamente se agregó la misma función (véase figura 3.22) para activar la instrucción START en el actuador basta con hacer clic en el botón START desde el HMI.

Figura 3.22. Instrucción para START.

Para hacer un cambio de giro a la derecha se debe de transferir el valor 18 al variador de frecuencia (véase figura 3.23) y solo basta con presionar el botón FORWARD desde el HMI automáticamente el variador se irá a su frecuencia mínima y enseguida volverá a su frecuencia programada, esto no causa ningún problema con el cambio de giro. Esto es una de las ventajas de utilizar variadores de frecuencia para los actuadores.

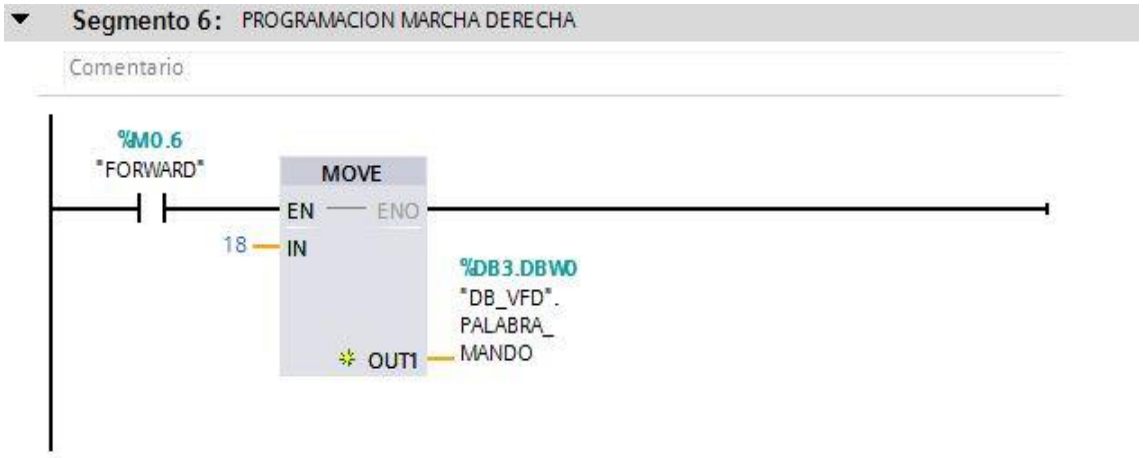


Figura 3.23. Instrucción: cambio de giro a la derecha.

De la misma manera sucedió para realizar el cambio de giro del motor a la izquierda se transfirió el valor 34, (véase figura 3.24).

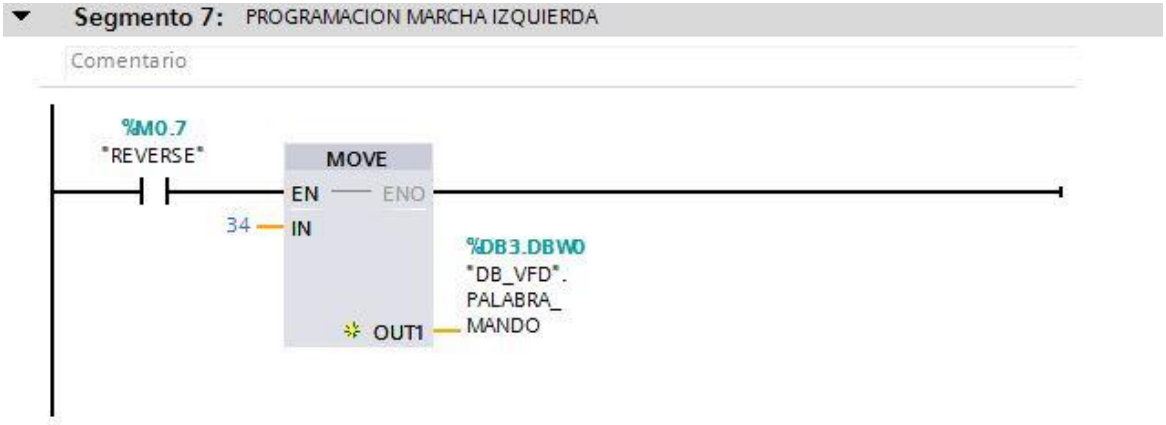


Figura 3.24. Instrucción: cambio de giro a la izquierda.

El código de error del variador Powerflex 4 se pudo leer por medio de la red mandando la función de lectura a la dirección del registro 8449 (véase figura 3.25).

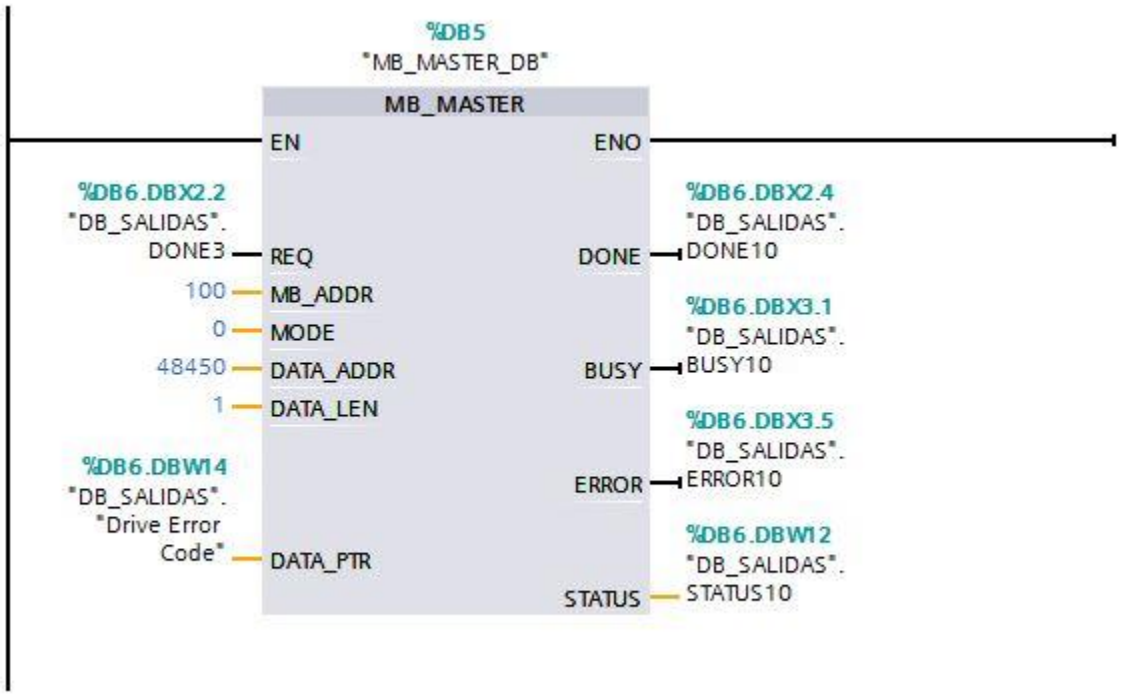


Figura 3.25. Lectura del código de error del variador.

El **tiempo de aceleración**: es el tiempo en milisegundos que requerirá el VFD para llevar al motor desde un estado inicial en reposo hasta una velocidad final.

Para determinar el tiempo de aceleración véase ecuación 1

$$\frac{Maximum\ Freq}{Accel\ Time} = Accel\ Rate$$

Ecuación 1. Tiempo de aceleración.

El valor se introduce en la variable correspondiente se escribirá en la dirección del registro 40040, (véase figura 3.26).

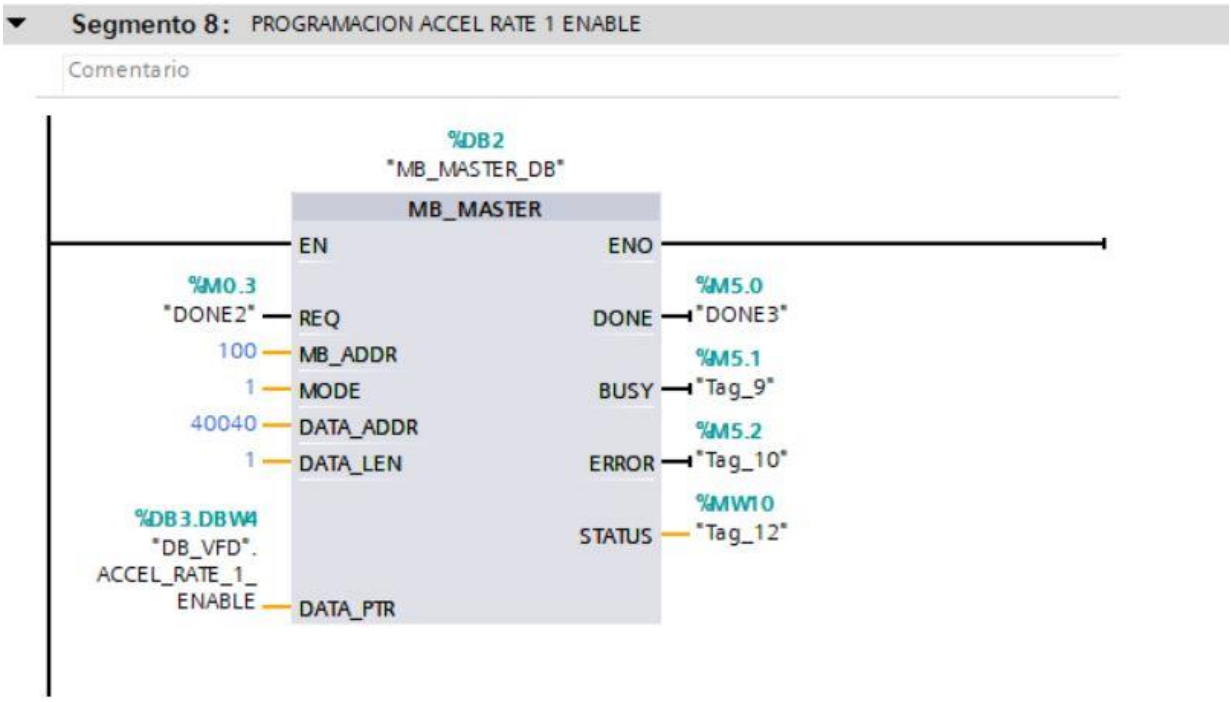


Figura 3.26. Instrucción: tiempo de aceleración.

El **tiempo de desaceleración**: es el intervalo de tiempo durante el cual el VFD es capaz de llevar al motor desde su velocidad máxima hasta su estado final en reposo Ahora bien, para el tiempo de desaceleración se ocupa la siguiente ecuación 2.

$$\frac{Maximum\ Freq}{Decel\ Time} = Decel\ Rate$$

Ecuación 2. Tiempo de desaceleración.

Para esto se deberá realizar el cálculo con el tiempo que deseamos que se detenga el VFD, así mismo, se envía el valor a la dirección de registro 40041, (véase figura 3.27).

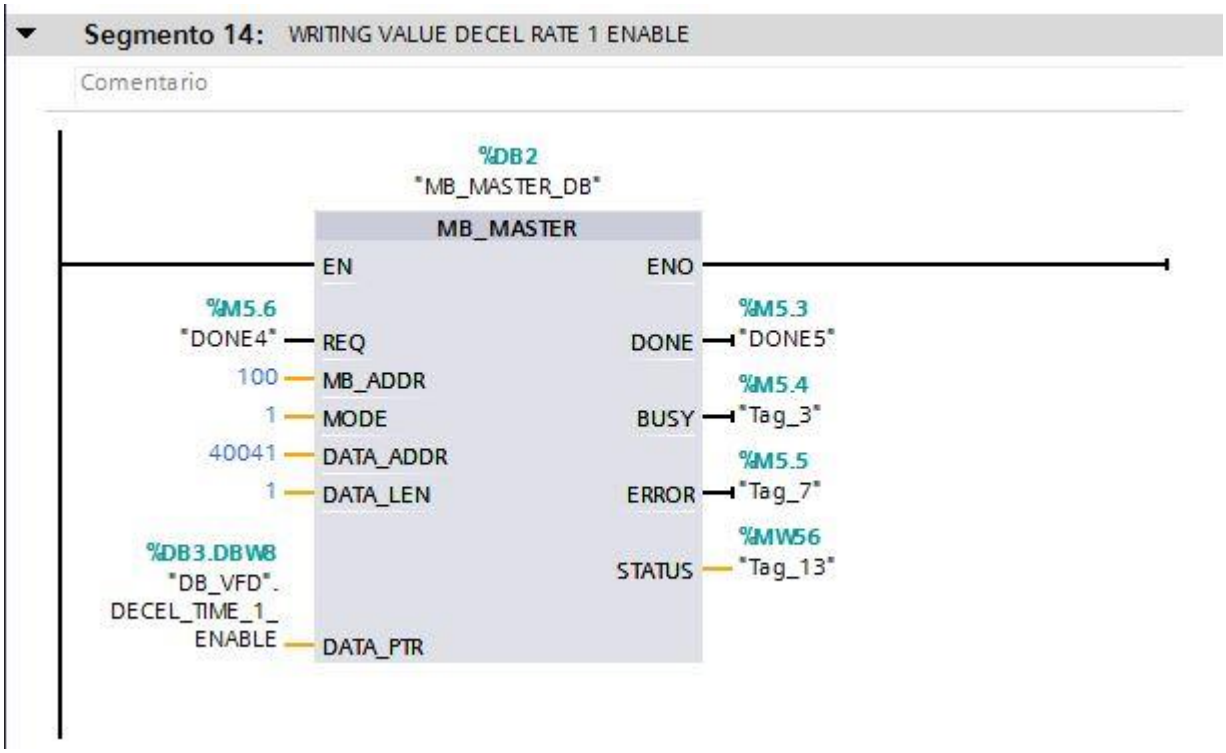


Figura 3.27. Instrucción: tiempo de desaceleración.

Para realizar la lectura de la frecuencia de salida, se modificó el parámetro MODE, en vez de escritura, se hizo una lectura, paro ello, se tuvo que agregar un bloque MB_MASTER y cambiar el valor de 1 a 0 para leer en la dirección del registro 8451 como se muestra en la figura 3.28.

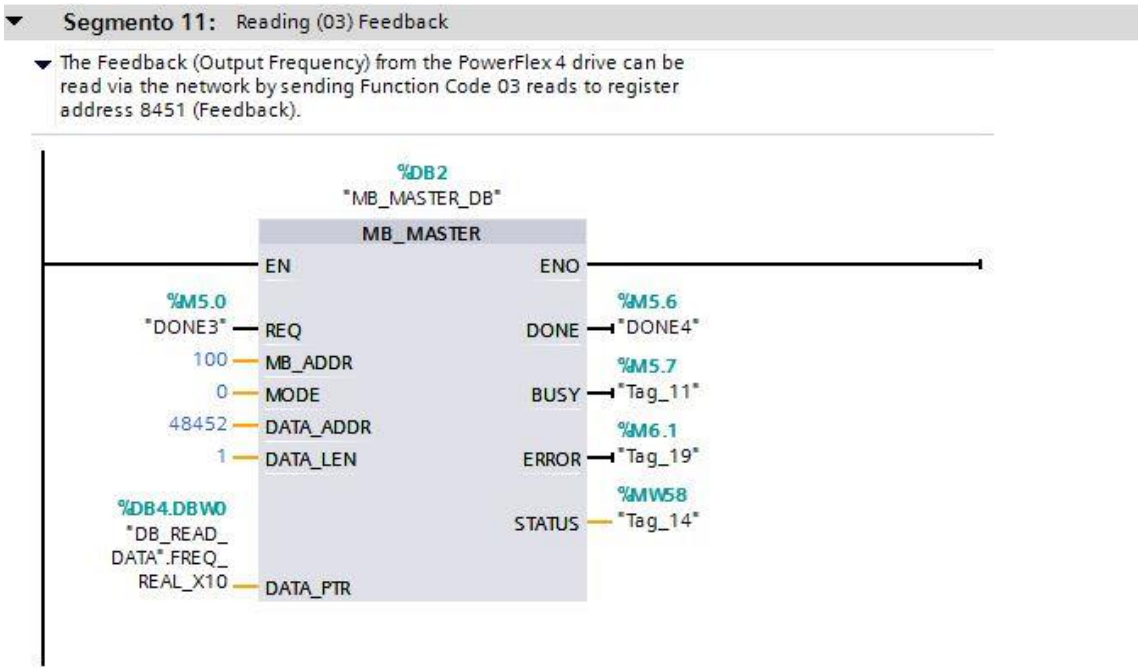


Figura 3.28. Instrucción: Lectura de la frecuencia de salida.

También se programó la función CLEAR FAULTS (véase figura 3.29), esta instrucción como su nombre lo indica se utilizó para limpiar las fallas presentadas durante el proceso en el variador de frecuencia. Por lo tanto, se debió enviar el valor de 130 y activar la instrucción desde el HMI.

Figura 3.29. Instrucción: Clear Faults.

Respecto al estado logico del variador (vease tabla 4.4), también es posible visualizarlo mediante el registro 8448, por lo que se leyó en esa dirección, (vease figura 3.30).

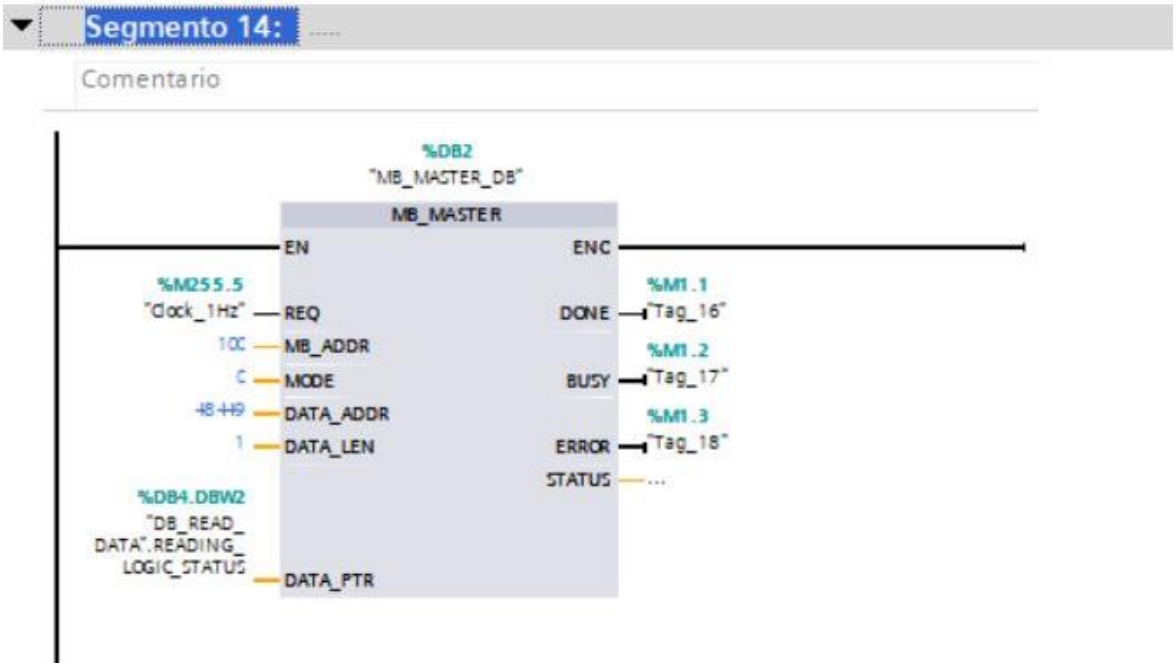


Figura 3.30. Instrucción: Logic Status.

3.3. Integración de las redes de comunicación MODBUS TCP y RTU.

En esta tercera etapa del proceso, se realizó la integración de las redes de comunicación anteriormente implementadas.

Para dicha integración se utilizaron los siguientes dispositivos y materiales:

- Una laptop con la plataforma TIA PORTAL V14
- Un SIMATIC PANEL TP177
- Un PLC SIEMENS S7-1215C DC/DC/RLY
- Un PLC SIEMENS S7-1212C AC/DC/RLY
- Una tarjeta de comunicación CM 1241 RS 422/ RS485
- Un VFD Powerflex 4 Allen Bradley
- Un switch TP LINK
- Un cable con conector DB9 macho y conector RJ45

Para comenzar con la integración, se creó un nuevo proyecto dentro de la plataforma TIA PORTAL V14, por lo que se siguió la misma metodología de programación. Una vez que se han agregado todos los módulos de control, los cuales se intercomunicaron, se transfirieron los programas ya antes realizados. Al respecto conviene decir que, aquí se ocupó un switch, con el cual se pudo conectar todos los dispositivos en la misma red. Así mismo, desde la computadora se conectó a un puerto del switch directamente con cada dispositivo y bajar el programa para cada uno sin la necesidad de desconectar los demás dispositivos.

La conexión se realizó, con base en el diagrama a bloques (véase figura 3.31). Es importante tener claro que un switch no proporciona por si solo conectividad con otras redes, y obviamente, tampoco proporciona conectividad con internet. Para ello es necesario otros dispositivos, por ejemplo, un router, el cual no es indispensable en este proyecto, ya que este proyecto se desarrolla en una red local.

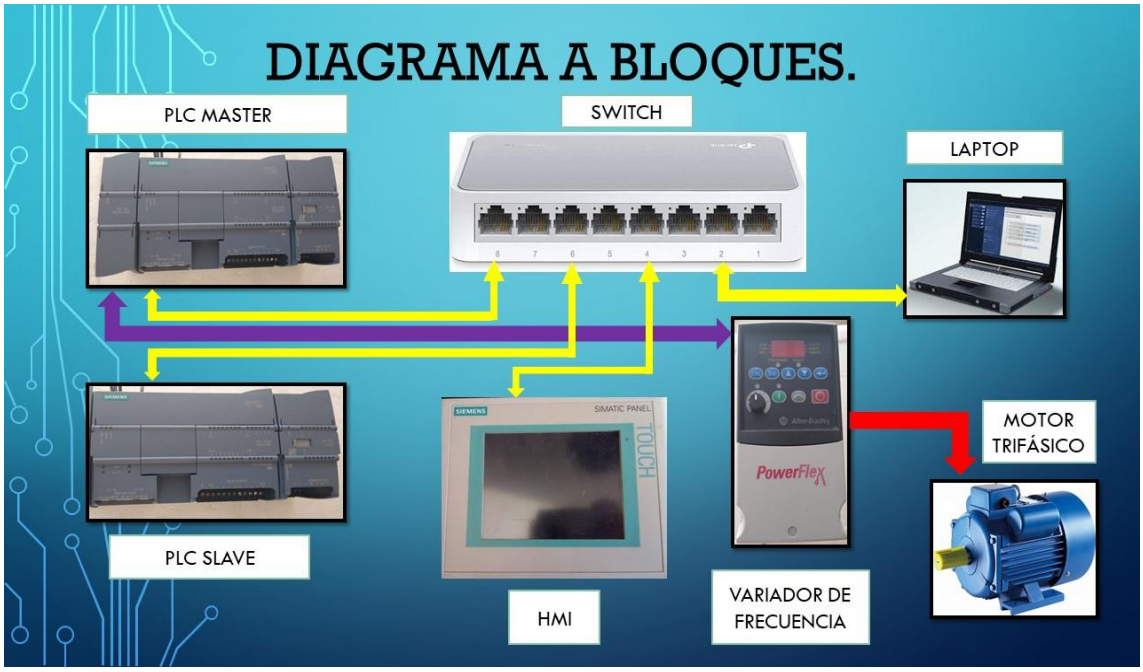


Figura 3.31. Diagrama a bloques.

Es de esta manera entonces como se realizó la conexión eléctrica (véase figura 3.32) y se comprobó la comunicación con cada dispositivo. Después se realizó la descarga de programas para cada equipo. Posteriormente se pudo ingresar datos desde Touch Panel para ser enviados al variador.

Figura 3.32. Conexión de los dispositivos en la misma red local.

3.4. Programación del HMI.

Para esta red se añadieron algunas plantillas más para la integración de las redes, en la figura 3.33, se muestra la pantalla de INICIO que contiene un menú, con el cual, se pudo navegar entre las diferentes pantallas que contiene el HMI y que se muestran a continuación.
Las pantallas diseñadas y programadas son las siguientes:

- Presentación
- Inicio
- Conectividad de la red
- Selección de modo de operación Cliente-Servidor
- Writting Logic Command Data
- Reading Logic Status Data
- Lectura de frecuencia real

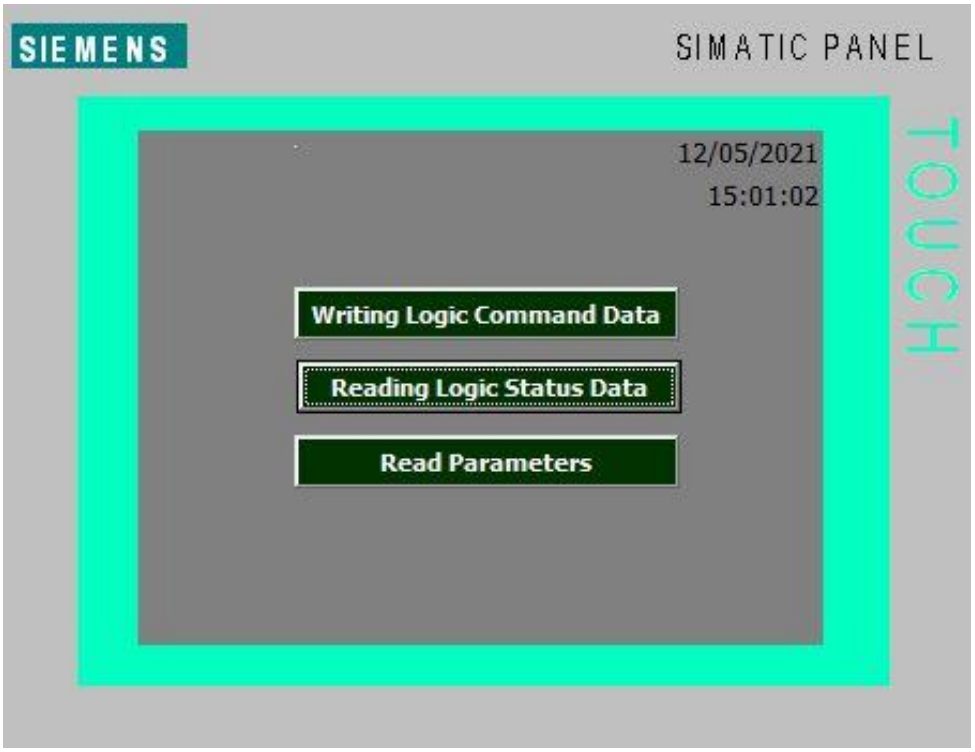


Figura 3.33. Pantalla del HMI: inicio.

Se muestra las opciones para habilitar y deshabilitar la comunicación de la red, como se mencionó anteriormente se debió activar el botón REQ y los botones DESCONNECT-CLIENT y DESCONNECT-SERVER para finalizar la comunicación si es el caso. En la parte inferior se encuentra el menú de navegación entre las diferentes pantallas del proceso, (véase figura 3.34).

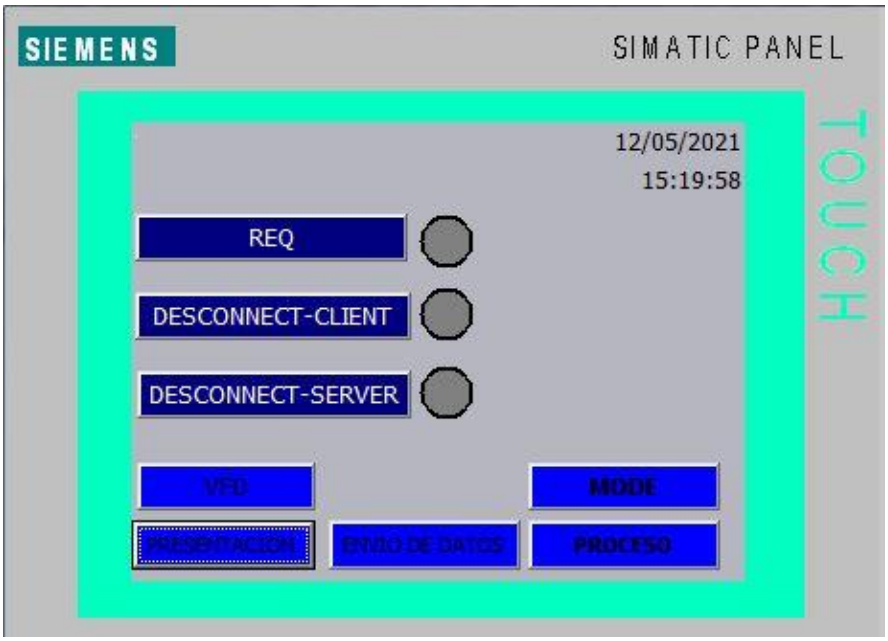


Figura 3.34. Pantalla del HMI: conectividad de la red.

Como se vio anteriormente debe de especificarse el modo de operación entre el Cliente-Servidor. En la pantalla MODE (véase figura 3.35). Para haber realizado la escritura de Cliente-Servidor se tuvo que ingresar el valor de 1 en caso contrario, para realizar la lectura de Servidor-cliente entonces se tuvo que ingresar el valor de 0.

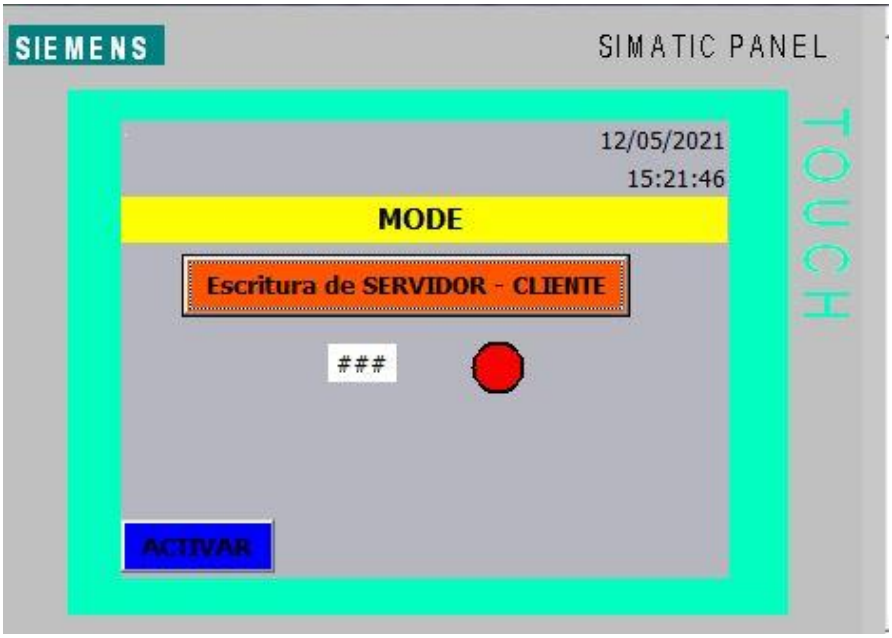


Figura 3.35. Pantalla del HMI: selección de modo de operación Cliente-Servidor.

Para ir a la pantalla de Writting Logic Command Data (véase figura 3.36) bastó con hacer clic en el primer botón como aparece en la figura anterior.

Se puso en marcha el actuador solo con presionar el botón START y se detuvo presionando el botón STOP. También se pudo indicar el sentido de giro del motor con tal solo haber presionado el botón FORWARD o el botón REVERSE dependiendo de las necesidades y requerimientos.

Se agregó un botón CLEAR, para limpiar las fallas que se han presentado durante el proceso.

Para indicar si está en marcha o stop el actuador, se añadieron indicadores, estos indicadores cambian de color una vez que el botón haya sido presionado, es decir, si se presiona el botón de START, el indicador que se encuentra a la derecha del botón se iluminará (color verde). Esto aplica para los demás indicadores que contiene esta pantalla.

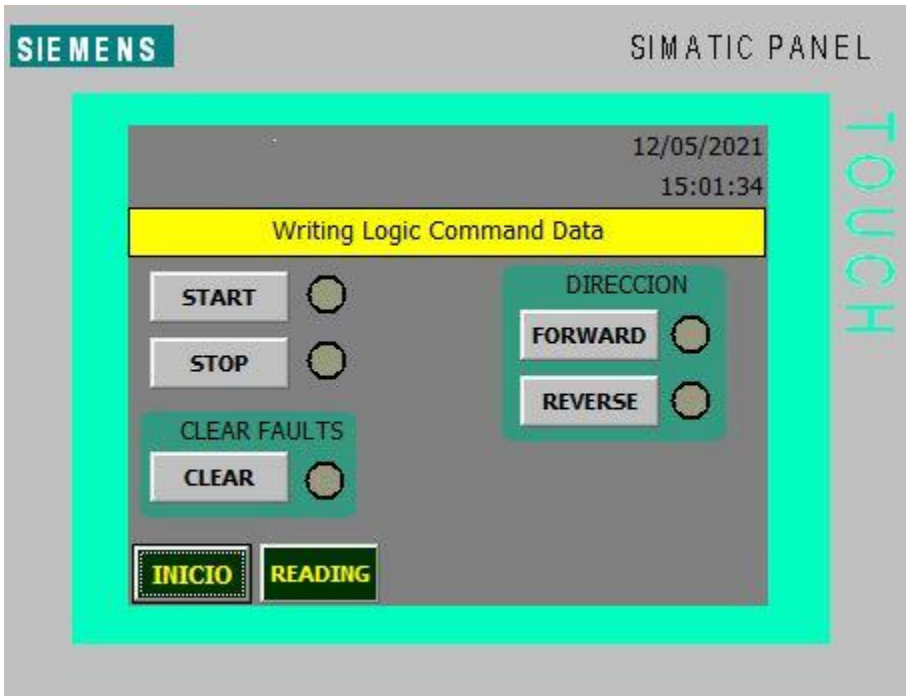


Figura 3.36. Pantalla del HMI: Writting Logic Command Data.

Para hacer un diagnóstico del actuador, es suficiente con abrir la pantalla READING LOGIC STATUS DATA (véase figura 3.37). Se puede apreciar el estado lógico del dispositivo. Esto quiere decir que no es necesario ir hasta el sitio donde se encuentra el dispositivo conectado, es suficiente con navegar en el sistema para recibir el estado en tiempo real. Con este estado se puede conocer datos, por ejemplo: si el dispositivo está listo o no, si está encendido o apagado, también se puede tener el conocimiento del

sentido de giro, de igual modo se puede conocer si está en aceleración o desaceleración. Así mismo, se conoce si presenta alguna falla o error en el sistema.

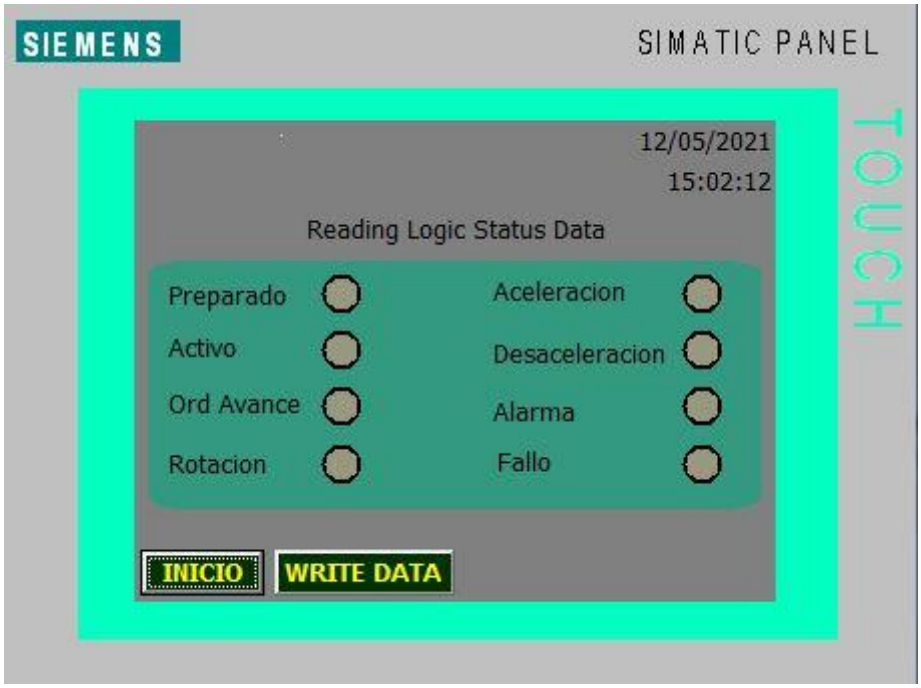


Figura 3.37. Pantalla del HMI: Reading Logic Status Data.

Para la lectura de la frecuencia real es necesario ir a la pantalla READING (véase figura 3.38). La frecuencia se dará en Hz.

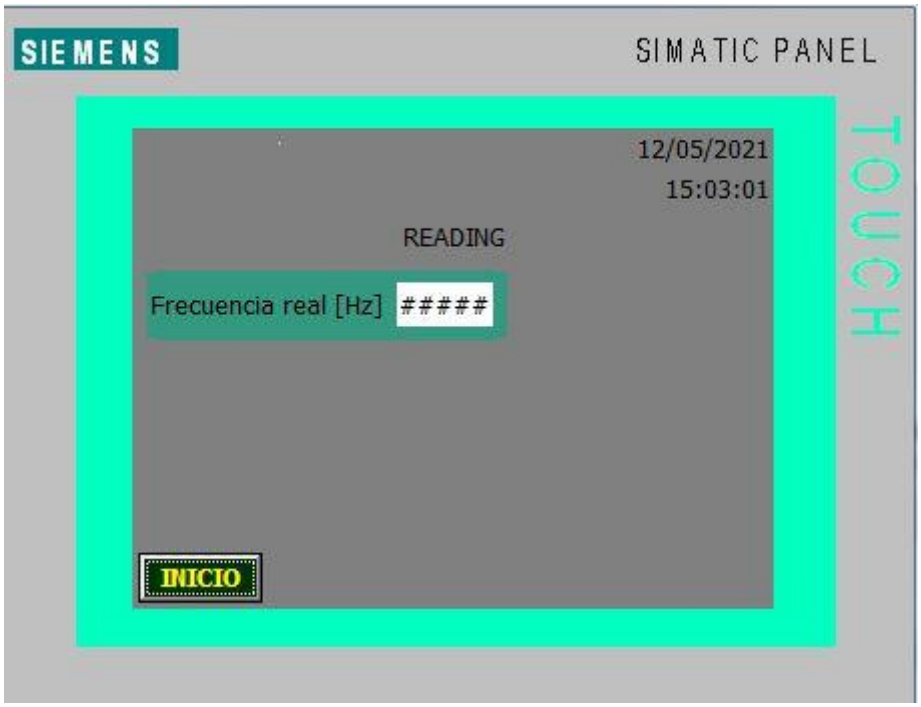


Figura 3.38. Pantalla del HMI: lectura de frecuencia real.

En la programación del HMI también se habilitó la protección de los datos y toda información que se considere importante, para ello se activó la función: Administración de usuarios. Con esta función se proporciona la seguridad solicitando autorización con un usuario y una contraseña para acceder al control de información y poder modificar (véase figura 3.39).

Cabe mencionar que, para el personal de mantenimiento, personal de confianza y autorizado, se les proporciona un usuario y una contraseña siempre y cuando se considere conveniente y necesario, en caso contrario no hay manera de como acceder a la información del HMI sin un usuario, así mismo la sesión se cierra automáticamente, o se puede de igual manera programar el cierre de sesión.

Figura 3.39. Inicio de sesión con usuario y contraseña única.

Una vez que se programó las pantallas y se dio de alta los usuarios con contraseña que tendrán acceso a la información y modificación de parámetros del proceso, el usuario tendrá que ingresar el nombre y la contraseña proporcionada para alguna modificación en el proceso (véase figura 3.40).

Figura 3.40. Ingresar usuario y contraseña.

Si el usuario ingresó un nombre o contraseña incorrecta saldrá un aviso, el cual le dirá que los datos proporcionados no son válidos (véase figura 3.41). En dado caso que algún usuario que no esté dado de alta en el sistema y quisiera entrar a modificar algún parámetro le será difícil acceder ya que todo está programado desde la plataforma, lo cual significa que no hay manera de como violar la seguridad a través del HMI.

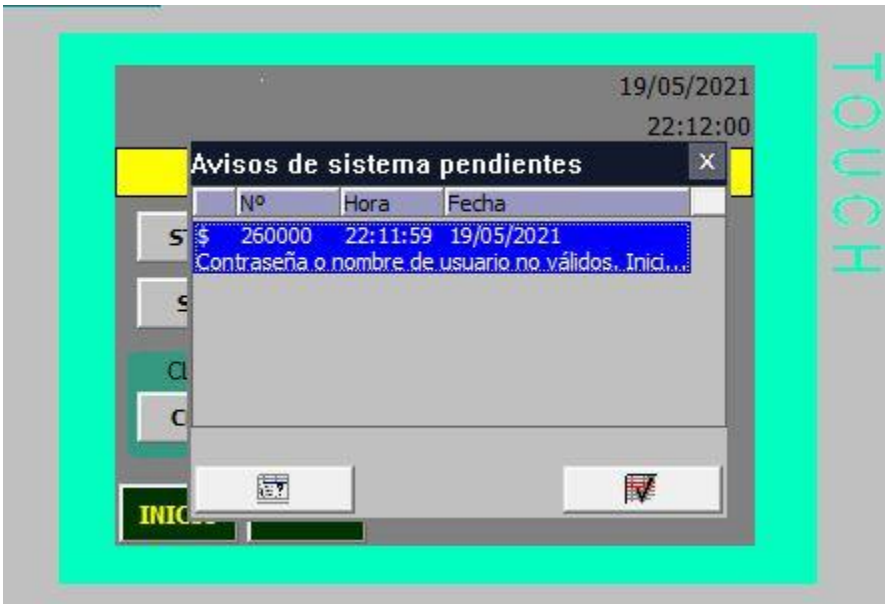


Figura 3.41. Avisos de sistema pendientes

Capítulo 4

Resultados y conclusiones

En este capítulo se dan a conocer las conclusiones de las redes de comunicación implementadas para el proyecto, comenzando por las conclusiones de metodología, la cual involucra las particulares, así mismo, se dan algunas recomendaciones para cada una de las etapas realizadas con el fin de orientar al ingeniero y lector sobre algunos puntos importantes al momento de estár diseñando e implementando este tipo de redes para procesos industriales.

Finalmente se mencionan las opciones de mejora y la conclusión general.

4.1. Conclusiones metodología.

La metodología de programación resulta importante para la organización de las funciones de la aplicación que se vaya a desarrollar, así mismo, es de ayuda para seguir una secuencia de configuración del equipo hasta la puesta en marcha. Además, puede ser de gran utilidad ya que se puede ocupar en todas las integraciones de proyectos de programación de equipos industriales.

4.1.1. MODBUS TCP

Actualmente, este protocolo es muy utilizado para transferencia de datos y por supuesto para supervisar el proceso, ya que se utiliza el ETHERNET que es de acceso libre y un estándar Aquí no se requiere de ningún modulo convertidor, ya que ambos equipos son de la misma marca SIEMENS, por tanto manejan el mismo tipo de comunicación.

El diseño y la implementación de la red de comunicación utilizando el protocolo MODBUS TCP se concluye satisfactoriamente, por un lado, por otro lado, es importante identificar los parámetros a configurar y tener claro qué tipos de datos se van a configurar, esto para una comunicación eficiente entre el Cliente-Servidor. Cabe mencionar que ambos dispositivos tendrán que estár en el mismo dominio de IPs.

- Verificar que el número de parte sea el mismo que se da de alta en la plataforma de programación y el equipo físico.
- Identificar que las IPs estén dentro del mismo dominio, es decir, en la misma red.

- Configurar adecuadamente todos los equipos que se vayan a conectar en la misma red, así como módulos de expansión I/Os, tarjetas de comunicaciones, entre otros módulos.
- Habilitar los recursos al inicio de configuración de software que crea que se utilizaran durante la implementación.
- Analizar las instrucciones de programación tanto el equipo cliente como el servidor online para confirmar que haya una correcta comunicación.

Finalmente, concluimos que el protocolo MODBUS TCP permite la integración y la supervisión de los dispositivos distribuidos dentro de una misma red, lo que hace posible que un solo ingeniero pueda supervisar y modificar los parámetros individuales de los controladores. Además, podrá atender remotamente múltiples procesos simultáneamente desde el mismo punto de monitorización.

4.1.2. MODBUS RTU.

Se implementó en la red de comunicación industrial MODBUS RTU un variador de frecuencia, un PLC, y un HMI para supervisar y controlar las variables de un proceso, lo cual facilita un soporte inmediato a los controladores en dado caso que lo requiera.

- Conocer los registros a los cuales acceder para las funciones de escritura y lectura, facilitan la implementación.
- Conocer las bases de este protocolo, ayuda mucho para ofrecer soluciones en la transferencia de datos de control a distancia.
- Reconocer la información importante y necesaria ayudará a solucionar eficazmente problemas futuros en los dispositivos.
- Tener áreas trasversales ayuda bastante, ya que se requiere conocimientos de control eléctrico, programación y diseño de HMIs para un mejor desempeño.

Finalmente, concluimos que algunos parámetros permanecen por default, esto podría ser por estándar o para que trabaje en sus óptimas condiciones el dispositivo.

4.1.3. MODBUS TCP- RTU.

Las dos redes de comunicación industrial antes implementadas se integran en una sola red considerando todos los parámetros compatibles. Se debe considerar y definir el tipo de dato que se va a enviar desde el controlador Cliente a partir de la red TCP, el cual llegará al esclavo en la red RTU y finalmente lo transmitirá al variador de frecuencia.

- Considerar las pruebas de comunicación por etapas, será la mejor opción de hacer la implementación.
- Colocar marcas de estado dentro del programa para encontrar rápidamente fallas y errores de comunicación o coherencia.
- Controlar la habilitación y deshabilitación de cada red reducirá fallas en las conexiones entre cliente y servidor.

Finalmente, se concluye que estos dos protocolos son de gran utilidad, ya que MODBUS se ha renovado con una nueva configuración, la cual utiliza el ETHERNET como medio, por un lado y por otro lado, MODBUS RTU sigue siendo utilizado ya que tiene muchas prestaciones con otro tipo de controladores cuando se trata de la transferencia de datos de control.

4.2. Programación PLC

En la programación del PLC se utiliza programación estructurada, con el fin de que el programa tenga un orden y se comprenda. Es importante que la programación se divida en funciones para casos de mantenimiento o mejoras.
Así mismo, se recomienda que haya comentarios en todo el programa para saber qué hace cada parte del programa.

- Tener presente las versiones de las instrucciones con las que se va a trabajar en la programación.
- Realizar la aplicación en subfunciones facilitará la programación.
- Conocer la plataforma de programación es una gran ventaja para todo ingeniero que se dedique a la automatización.
- Colocar nombres simples y relacionados a las variables físicas a controlar, ayudará bastante al momento de buscar alguna falla.
- Se recomienda hacer un listado de las entradas y salidas que se vayan a utilizar con el fin de evitar confusiones.

En ultimas, se recomienda habilitar la función *KNOW HOW*, la cual se utiliza para proteger aquellos bloques de programas a los que no queremos que accedan personas no autorizadas.

4.3. Programación HMI

El diseño y programación del sistema SCADA es un proceso muy esencial e indispensable en aplicaciones y proyectos donde se requiera supervisar y visualizar procesos, puesto que realizar la programación de diferentes pantallas, niveles de acceso con contraseña para usuarios para proteger el acceso y modificación de información. Así mismo, mostrar donde ha ocurrido una falla, mostrar avisos de seguridad, y toma de decisiones eficazmente para que el proceso continúe lo más pronto posible.

- Las variables deben de coincidir con las del PLC para poder llamarlas.
- Se recomienda colocar contraseñas a todo usuario que intervenga en el proceso.
- Todos los dispositivos deberán tener su IP en el mismo dominio, así mismo, conectados a la misma red.
- Hacer la programación lo más sencilla y entendible para el personal que vaya a vigilar el proceso a través del HMI.
- Conocer las funciones dentro de la plataforma de programación para desarrollar las pantallas del proceso, será indispensable, ya que mejorará la visualización y cumplirá con los objetivos del proceso.

Finalmente, el sistema SCADA se compone de la información necesaria del proceso, ya que no sirve de nada si las pantallas se ven saturadas de información irrelevante para el personal operativo, por lo que es importante conocer el proceso y saber qué información mostrar.

4.4. Programación VFD

Los variadores de velocidad de este tipo traen consigo el puerto serial RS-485 para utilizar el protocolo de comunicación MODBUS RTU, lo que hace muy interesante este tipo de dispositivos, ya que se considera la seguridad tanto en el proceso como en la información de este, puesto que se puede deshabilitar el panel de control que trae el dispositivo y controlar el proceso a través de un PLC como en este caso.

- Conocer el manual completo del dispositivo es de gran utilidad ante cualquier emergencia o falla técnica.
- En cualquier sector industrial, se van a encontrar con variadores de frecuencia por lo que es bueno conocer estos dispositivos y capacitarse.

Para finalizar, los variadores ofrecen grandes beneficios y ventajas en una aplicación donde involucren actuadores como lo son motores trifásicos debido a que, se pueden proteger las salidas a altos voltajes.

4.5. Opciones de mejora

Los PLCs son equipos inteligentes por lo cual, podría crearse una rutina con la cual después de un determinado tiempo sin ingresar datos de control, pueda desconectarse automáticamente la red, es decir cerrar la comunicación. Esto es una mejora con la cual aumenta la eficiencia de la red de comunicación y por ende aumenta la disponibilidad para otros clientes que requieran hacer solicitudes a los mismos u otros servidores conectados.

En el HMI se encuentran muchas herramientas de gran utilidad por lo cual, llevar un registro de fallas, alarmas, warnings, etc. Es importante para el departamento de mantenimiento, por lo tanto, es una mejora necesaria que ayudaría a visualizar los mensajes de todo el proceso

4.6. Conclusión general.

MODBUS es un protocolo sobresaliente debido a sus prestaciones, compatibilidad y acceso gratuito. Por esto mismo MODBUS utiliza el ETHERNET como un medio de comunicación, puesto que es muy utilizado en la industria para la comunicación de dispositivos industriales y de fácil implementación.

La integración de las dos redes de comunicación industrial fue satisfactoria sin necesidad de utilizar algún convertidor de datos, puesto que todo se hace en la programación de los PLCs. La programación estructura es indispensable para que cualquier ingeniero o personal operativo pueda comprender que es lo que hace dicho

programa. Este tipo de programación siempre se va a encontrar en todo controlador programable ya que es obligatorio programar en algún estándar.

Es importante conocer los protocolos de comunicación, en dado caso de escalar algún proyecto o aplicación en la mejora de la comunicación de equipos industriales para poder ofrecer propuestas optimas y desarrollarse en el menor tiempo posible reduciendo costos.

Cabe mencionar que existen varias maneras de activar la seguridad para la protección de información del proceso y no accedan usuarios no autorizados. Dentro de la plataforma se puede activar la función *KNOW HOW* para proteger con contraseña los bloques de programación, por si fuera poco, en la programación del HMI se ofrece la opción de acceso con contraseña para usuarios, que también podrá ser aplicada en la visualización del proceso.

El presente proyecto ha logrado cumplir con el objetivo general y con los particulares, especialmente demostrar que las redes industriales pueden implementarse y adaptarse en toda aplicación que requiera comunicar controladores programables y dispositivos de control industrial dentro de una red local.

Tabla de costos.

Dispositivo	Costo	Referencia
Un PLC S7-1200, CPU: 1215C DC/DC/RLY	$ 10,792.35 MXN	[25]
Un PLC S7-1200, CPU: 1212C AC/DC/RLY	$ 7,624.09 MXN	[26]
Un switch TP LINK	$ 209 MXN	[27]
Un SIMATIC PANEL TP177	$ 34,417.53 MXN	[28]
Un módulo de comunicación SIEMENS. CM 1241 RS422/ RS485.	$ 8,839 MXN	[29]
Un variador de frecuencia POWERFLEX 4 ALLEN BRADLEY	$ 6,809.75 MXN	[30]
Un conector DB9 macho	$ 48 MXN	[31]
Plugs RJ45 (14 piezas)	$104.65 MXN caja con 100 piezas	[32]
Cable UTP CAT. 6 (20 m)	$ 13 MXN por metro lineal $ 260 por 20 m	[33]
Ingeniería	$21000	
TOTAL	$ 90,104.37 MXN	

Trabajos futuros.

El proyecto presente se podrá aplicar en el control de un sistema de llenado de fertilizante de la empresa IT Automation RH S.A. De C.V. por supuesto que deberán hacerse las adecuaciones correspondientes a los parámetros de acuerdo con la aplicación, por lo que se podrá adecuar la interfaz HMI para el proceso que se pretenda automatizar y controlar.
Este mismo proyecto podría migrarse a otra versión de PLC siempre y cuando se consideren las respectivas condiciones de trabajo de los equipos.

Bibliografía.

[1] Anónimo. *Automatización Industrial. Modbus* Fortec [citado el 2 de marzo de 2021]. Disponible en: https://aquiro.weebly.com/modbus.html#

[2] Anónimo *Autómatas Programables* 2001. [citado el 2 de mayo de 2021] Disponible en: http://www.sc.ehu.es/sbweb/webcentro/automatica/WebCQMH1/PAGINA%20PRINCIPAL/PLC/plc.htm#Funciones%20b%C3%A1sicas%20de%20un%20PLC

[3] Anónimo. *Elementos y equipos eléctricos*. [citado el 4 de mayo de 2021] Disponible en: https://web.archive.org/web/20160222071927/http://www.efn.uncor.edu/departamentos/electro/cat/eye_archivos/apuntes/a_practico/CAP%209%20Pco.pdf

[4] Carmona J. Cortes D. Domínguez M. Gazo A. González J. L. & Rodríguez F. 2012 *Administración Avanzada de Redes TCP/IP* 1ª Ed. [citado el 7 de marzo de 2021]. Disponible en. http://www.manolodominguez.com/content/common/pdf/cv/publications/AdministracionAvanzadaDeRedesTCPIP.pdf ,

[5] Departamento de Ingeniería Eléctrica, Electrónica y de Control. *Controladores Industriales Inteligentes.* [citado el 25 de abril de 2021] Disponible en: http://www.ieec.uned.es/investigacion/Dipseil/PAC/archivos/Informacion_de_referencia_ISE6_1_1.pdf

[6] Estrada J. *Protocolos de comunicación: Modbus TCP/IP*. [citado 10 de marzo de 2021]. Disponible en: Protocolos de comunicación: MODBUS TCP/IP - Blog Logicbus

[7] Estrada J. *MODBUS*. [citado 19 de marzo de 2021]. Disponible en: ModBus Logicbus S.A de C.V

[8] Fuentes M, *Historia y Hardware del PLC*. [citado el 27 de abril de 2021]. Disponible en: http://files.electricidadceah.webnode.cl/20000133269efa6be2e/Historia%20y%20Hardware%20del%20PLC.pdf

[9] Guerrero V., R. Yuste, & Martínez L., *Autómatas Programables SIEMENS Grafcet y Guía Gemma con TIA Portal* 1ª ed. España, MARCOMBO, 2017, pp. 243 – 248.

[10] Guerrero F. Ruiz A. & Barandica A. *Implementación de una red MODBUS/ TCP*. 2004 [citado 20 de marzo de 2021]. Disponible en: https://biblat.unam.mx/hevila/Ingenieriaycompetitividad/2004/vol6/no2/4.pdf

[11] Guerrero V. Yuste R. & Martínez L. *Comunicaciones industriales,* 1ª ed., MARCOMBO, 2009.

[12] Hurtado J. M. *Comunicaciones industriales*, Departamento: Electricidad-Electrónica I.E.S., Himilce – Linares. Pp. 6 – 13.

[13] Peciña L. *Comunicaciones industriales y WinCC,* 1ª ed., MARCOMBO, 2018.

[14] Rodríguez A. *Sistemas SCADA*, 3ª ed., México: MARCOMBO, 2013.

[15] Rodríguez A. *Comunicaciones industriales,* 1ª ed., Barcelona, MARCOMBO, 2008.

[16] Rockwell Automation. *Manual de usuario. Adjustable Frequency AC Drive*, Rockwell Automation, USA, 2017. [citado el 7 de marzo de 2021]. Disponible en: https://literature.rockwellautomation.com/idc/groups/literature/documents/um/22a-um001_-en-e.pdf

[17] Rockwell Automation. *PowerFlex 4 Adjustable Frequency Drive FRN 1.xx – 2.xx User Manual, Publicación 22A- UM001K-EN-E*, Rockwell Automation, july 2017, [citado 13 de marzo de 2021]. Disponible en: 22A-UM001K-EN-E PowerFlex 4 Adjustable Frequency AC Drive FRN 7.xx User Manual (rockwellautomation.com)

[18] Rodríguez J. Cerda L. & Sánchez R. *Automatismos Industriales.* Paraninfo, 1ª Ed. 2014. España. Pp 309-315

[19] SIEMENS AG. *Datasheet PLC 1200, SIEMENS*, Alemania 2014. [citado el 9 de marzo de 2021]. Disponible en: https://docs.rs-online.com/e75a/0900766b8112975b.pdf

[20] S&P *Variadores de frecuencia*. El blog de la ventilación eficiente 2020 [citado el 10 de marzo de 2021]. Disponible en: https://www.solerpalau.com/es-es/blog/variador-de-frecuencia/

[21] SIEMENS AG. *Datasheet PLC 1200, SIEMENS*, Alemania 2015. [citado el 15 de marzo de 2021]. Disponible en: https://docs.rs-online.com/7e61/0900766b813973fb.pdf

[22] Santiago. *PLC*. [citado el 30 de abril de 2021]. Disponible en: http://plcsantiago.blogspot.com/2017/07/tipos-de-plc.html

[23] Wright J. *The Debate Over Whi-ch PLC Programming Language is the State of the Art*. 1999 Journal of Industrial Technology, vol. 15, núm. 4

[24] Weis O. *MODBUS VS RS485*. 2020 Eltima Publishing [citado el 23 de abril de 2021] Disponible en: https://www.virtual-serial-port.org/es/articles/modbus-vs-rs485/#messages

[25] PLC S7-1200, CPU: 1215C DC/DC/RLY. Sitio web: 6ES7215-1HG40-0XB0 | Siemens | WIAutomation

[26] PLC S7-1200, CPU: 1212C AC/DC/RLY. Sitio web: 6ES7212-1BE31-0XB0 | Siemens Simatic S7-1200 - CPU 1212C | PLC-City

[27] Switch marca TP LINK con 8 puertos RJ-45. Sitio web: Switch Escritorio 8ptos Tl-sf1008d Tp Link | Mercado Libre

[28] Touch Panel Siemens Simatic HMI TP 177B. Sitio web: 6AV6642-0BA01-1AX1 SIEMENS SIMATIC HMI TP 177B | Siemens | WIAutomation

[29] Módulo de comunicación CM1241 RS-422/ 485 Siemens. Sitio web: Modulo Comunicacion Cm1241 Rs-422 Siemens 6es7241-1ch31-0xb0 | Mercado Libre

[30] Variador de frecuencia, PowerFlex4 AC Drive, 120VAC. Sitio web: 22A-V4P5N104 Allen-Bradley (wiautomation.com)

[31] Jack Conector DB9 con capucha. Sitio web: Jack Conector Db9 Para Soldar C/capucha | Mercado Libre

[32] Plug Conector RJ-45 Cable Red Internet UTP, Categoría 6. Sitio web: Plug Conector Rj45 Cable Red Internet Utp Cat 6 Caja 100pzs | Mercado Libre

[33] Cable UTP, Categoría 6. Sitio web: Cable UTP CAT6, azul Steren Tienda en Línea

Apéndice.

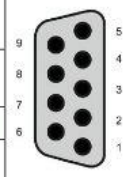
A. Datos técnicos de la tarjeta CM 1241 RS422/485

Datos técnicos	CM 1241 RS422/485
Pérdida de potencia (disipación)	1,1 W
A partir de +5 V DC	220 mA

Datos técnicos	CM 1241 RS422/485
Referencia	6ES7241-1CH32-0XB0
Dimensiones A x A x P (mm)	30 x 100 x 75
Peso	155 gramos

Datos técnicos	CM 1241 RS422/485
Tipo	RS422 o RS485, conector hembra sub-D de 9 polos
Rango de tensión en modo común	-7 V a +12 V, 1 segundo, 3 VRMS continuo
Tensión de salida diferencial del transmisor	2 V mín. a R_L = 100 Ω 1,5 V mín. a R_L = 54 Ω
Terminación y polarización	10K Ω a +5 V en B, pin PROFIBUS 3 10K Ω a GND en A, pin PROFIBUS 8 Opciones de polarización interna disponibles o bien sin polarización interna. En todos los casos se requiere terminación externa; consulte Polarizar y terminar un conector de red RS485 (Página 1110) y Configurar RS422 y RS485 en el Manual de sistema del controlador programable S7-1200. (Página 1164)
Impedancia de entrada del receptor	5,4K Ω mín. incluyendo terminación
Umbral/sensibilidad del receptor	+/- 0,2 V mín., 60 mV de histéresis típica
Aislamiento Señal RS485 a conexión a masa Señal RS485 a lógica de la CPU	707 V DC (ensayo de tipo)
Longitud de cable, apantallado	máx. 1000 m (en función de la velocidad de transferencia)
Velocidad de transferencia	300 baudios, 600 baudios, 1,2 kbits, 2,4 kbits, 4,8 kbits, 9,6 kbits (valor predeterminado), 19,2 kbits, 38,4 kbits, 57,6 kbits, 76,8 kbits, 115,2 kbits
Paridad	Sin paridad (valor predeterminado), par, impar, marca (bit de paridad siempre puesto a 1), espacio (bit de paridad siempre puesto a 0)
Número de bits de parada	1 (valor predeterminado), 2
Control de flujo	Se soporta XON/XOFF para el modo RS422
Tiempo de espera	0 a 65535 ms

Pines del conector DB9

Pin	Descripción	Conector (hembra)	Pin	Descripción
1	Masa lógica o de comunicación		6 PWR	+5 V con resistor en serie de 100 ohmios: Salida
2 TxD+ [1]	Conectada para RS422 Sin uso para RS485: salida		7	Sin conexión
3 TxD+ [2]	Señal B (RxD/TxD+): entrada/salida		8 TXD- [2]	Señal A (RxD/TxD-): Entrada/salida
4 RTS [3]	Petición de transmitir (nivel TTL) salida		9 TXD- [1]	Conectada para RS422 Sin uso para RS485: salida
5 GND	Masa lógica o de comunicación		SHELL	Conexión a masa