

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

IMPLEMENTACIÓN DE UN SISTEMA DE CONTROL INDUSTRIAL (ICS) DIDÁCTICO MODULAR PARA EL LABORATORIO DE REDES INDUSTRIALES

DISEÑO E IMPLEMENTACIÓN DE UN MÓDULO DIDÁCTICO QUE EMULE UNA CELDA DE CONTROL INDUSTRIAL BASADA EN UNA RED MODBUS

**TRABAJO DE INTEGRACIÓN CURRICULAR PRESENTADO COMO
REQUISITO PARA LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN
ELECTRÓNICA Y AUTOMATIZACIÓN**

LUIS FERNANDO HEREDIA CONCHAMBAY

luis.heredia@epn.edu.ec

DIRECTOR: Dra. Ing. SILVANA DEL PILAR GAMBOA BENÍTEZ

silvana.gamboa@epn.edu.ec

DMQ, septiembre 2023

CERTIFICACIONES

Yo, LUIS FERNANDO HEREDIA CONCHAMBAY declaro que el trabajo de integración curricular aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

LUIS FERNANDO HEREDIA CONCHAMBAY

Certifico que el presente trabajo de integración curricular fue desarrollado por LUIS FERNANDO HEREDIA CONCHAMBAY, bajo mi supervisión.

DRA.-ING. SILVANA DEL PILAR GAMBOA BENÍTEZ
DIRECTOR

DECLARACIÓN DE AUTORÍA

A través de la presente declaración, afirmamos que el trabajo de integración curricular aquí descrito, así como el (los) producto(s) resultante(s) del mismo, son públicos y estarán a disposición de la comunidad a través del repositorio institucional de la Escuela Politécnica Nacional; sin embargo, la titularidad de los derechos patrimoniales nos corresponde a los autores que hemos contribuido en el desarrollo del presente trabajo; observando para el efecto las disposiciones establecidas por el órgano competente en propiedad intelectual, la normativa interna y demás normas.

LUIS FERNANDO HEREDIA CONCHAMBAY

DRA.-ING. SILVANA DEL PILAR GAMBOA BENÍTEZ

DEDICATORIA

Dedico este trabajo con todo mi corazón a mis padres, su bendición a diario a lo largo de mi vida me protege y me lleva por el camino del bien. Por eso les doy mi trabajo en ofrenda por su paciencia y amor.

AGRADECIMIENTO

En un principio, quiero expresar mi gratitud hacia mis padres y hermana, quienes constantemente me han ofrecido su apoyo absoluto para lograr mis metas personales y educativas. Su afecto ha sido mi motivación constante para perseguir mis objetivos sin rendirme ante las dificultades.

Agradezco a mi familia que me ha acompañado y nunca me ha dejado desfallecer, a mis tíos, Patricio, Fernando, Marco, Augusta, Lorena y Esteban, quienes me han enseñado el valor de la dedicación y constancia.

A mi abuelita Ana María, quien ha sido un pilar fundamental en mi vida, pues me llena de bendiciones y me da fortaleza.

Deseo expresar mi más sincero agradecimiento a mi tutora Dra. SILVANA DEL PILAR GAMBOA BENÍTEZ por su compromiso y calma; sin sus precisas observaciones y correcciones, no habría sido capaz de alcanzar este anhelado logro. Valoro enormemente su orientación y los consejos brindados, los cuales llevaré conmigo de manera perdurable en mi futura carrera.

Varios maestros han formado parte de mi trayecto académico, y a todos ellos les extiendo mi gratitud, de manera muy especial a Mateo Vásquez, quien me ha mostrado un apoyo incondicional para el desarrollo de este trabajo.

Agradecer a mis más cercanos amigos, su amistad ha sido un regalo invaluable que ha iluminado cada uno de mis días, Darwin, Erick, Santy, Arbey, Dave, Gabo, Stefany, Karlita, Steven, Freddy, Leimer, Angelo, cada uno de ustedes han dejado una huella imborrable en mi vida, y estoy profundamente agradecido por ello.

Y al final, pero no menos importante, quiero de todo corazón agradecer a Gaby, quién me brindó un apoyo incondicional. Tu paciencia y comprensión son tesoros invaluables que atesoro con todo mi corazón.

ÍNDICE DE CONTENIDO

CERTIFICACIONES	I
DECLARACIÓN DE AUTORÍA	II
DEDICATORIA	III
AGRADECIMIENTO	IV
RESUMEN.....	VII
ABSTRACT	VIII
1 INTRODUCCIÓN.....	1
1.1 OBJETIVO GENERAL.....	1
1.2 OBJETIVOS ESPECÍFICOS.....	1
1.3 ALCANCE	1
1.4 MARCO TEÓRICO	3
1.4.1 PROTOCOLO MODBUS	3
1.4.1.1 Interfaz de Comunicación	4
1.4.1.2 Topología de Red	4
1.4.2 VARIANTES MODBUS.....	5
1.4.3 MENSAJES MODBUS.....	8
1.4.4 MAPA DE DIRECCIONES MODBUS	8
1.4.5 REDES MODBUS	9
1.5 REQUERIMIENTOS PARA EL DISEÑO DE LA MAQUETA DE PROCESO.....	11
1.6 REQUERIMIENTOS DE SOFTWARE PARA LA PROGRAMACIÓN DE COMPONENTES.....	12
2 METODOLOGÍA.....	13
2.1 REQUERIMIENTOS DE LA RED A DISEÑAR	13
2.2 SELECCIÓN DE EQUIPOS.....	14
2.3 DISEÑO DEL SISTEMA DE CONTROL BASADO EN MODBUS	16
2.3.1 SERVIDORES	18
2.3.2 CLIENTE	18
2.3.3 INTERFAZ GRÁFICA	18
2.3.4 SISTEMA INTEGRADO DE SEGURIDAD.....	18
2.4 DISEÑO DE INTERFAZ GRÁFICA.....	18
2.4.1 ARQUITECTURA DE NAVEGACIÓN DE LA INTERFAZ.....	19
2.4.1.1 Primer Nivel.....	19
2.4.1.2 Segundo Nivel	20
2.4.1.3 Tercer Nivel	20
2.4.2 ALARMAS Y TENDECIAS.....	20
2.4.3 DISEÑO DE VENTANAS.....	20

2.5	SELECCIÓN DE SENSORES Y ACTUADORES DE LA MAQUETA DE PROCESO	22
2.6	IMPLEMENTACIÓN DE LA RED	26
2.6.1	CONFIGURACIÓN DE DISPOSITIVOS	26
2.6.1.1	Configuración del dispositivo Siemens S7-1200 como Cliente.....	26
2.6.1.2	Configuración de Transmisión de datos del Servidor Micrologix MI-1400 hacia el Cliente.....	32
2.6.1.3	Configuración de transmisión de datos del Servidor Modicon M580 hacia el Cliente	33
2.6.1.4	Configuración para la transmisión de datos entre la Interfaz Gráfica y el Cliente	36
2.6.1.5	Configuración del Sistema Integrado De Seguridad.....	39
2.6.2	MAQUETA DE PROCESO	41
2.6.3	INTERFAZ GRÁFICA	42
2.7	VALIDACIÓN DE LA RED	44
3	RESULTADOS, CONCLUSIONES Y RECOMENDACIONES	46
3.1	RESULTADOS	46
3.2	CONCLUSIONES.....	54
3.3	RECOMENDACIONES.....	54
4	REFERENCIAS BIBLIOGRÁFICAS.....	56
5	ANEXOS.....	57
	ANEXO I	58
	Parámetros de estado de comunicación posibles para el PLC Siemens S7-1200	58
	ANEXO II	59
	Configuración de PLC Schneider M580 disponible en el Laboratorio de Redes Industriales.....	59
	ANEXO III	66
	Codificación y decodificación de datos tipo discreto para el envío y recepción desde y hacia la interfaz gráfica	66
	ANEXO IV	73
	Configuración del servidor de datos	73
	ANEXO V	76
	Diseño PCB para interconexión de elementos que conforman la maqueta de proceso y esquemas de conexión.	76
	ANEXO VI.....	79
	Manual de usuario.....	79

RESUMEN

Este trabajo se enfoca en el diseño e implementación de un sistema de control industrial (ICS) modular que simula una celda de control industrial a través de una red basada en el protocolo Modbus. El enfoque metodológico abarca desde una revisión exhaustiva del protocolo Modbus, el establecimiento de los requisitos de diseño, el diseño de la celda de control y la interfaz de monitoreo, hasta la implementación del diseño, su verificación funcional y su integración en un sistema de control de procesos industriales.

Los aspectos sobresalientes incluyen tanto el desarrollo de hardware como de software, la configuración de un servidor Modbus y la creación de una Interfaz Hombre-Máquina (HMI) para la supervisión del proceso. A lo largo del estudio, se proporciona un sólido fundamento teórico sobre los protocolos de comunicación, las topologías de red y el esquema de direccionamiento Modbus.

El presente trabajo concluye con una discusión en torno a la realización y validación de un sistema de control implementado que aprovecha el protocolo Modbus. Se analizan los resultados que confirman una exitosa comunicación de datos entre servidores, clientes e interfaz mediante la red Modbus.

PALABRAS CLAVE: Automatización, Aprendizaje interactivo, Emulación, Módulo didáctico, Red Modbus, Tecnología.

ABSTRACT

This work focuses on the design and implementation of a modular Industrial Control System (ICS) that simulates an industrial control cell through a network based on the Modbus protocol. The methodological approach encompasses a comprehensive review of the Modbus protocol, the establishment of design requirements, the design of the control cell and the monitoring interface, up to the implementation of the design, its functional verification, and its integration into an industrial process control system.

Prominent aspects include both hardware and software development, the configuration of a Modbus server, and the creation of a Human-Machine Interface (HMI) for process monitoring. Throughout the study, a solid theoretical foundation is provided on communication protocols, network topologies, and the Modbus addressing scheme.

This paper concludes with a discussion regarding the realization and validation of an implemented control system that leverages the Modbus protocol. The results are analyzed, confirming successful data communication between servers, clients, and the interface through the Modbus network.

KEYWORDS: Automation, Interactive learning, Emulation, Didactic module, Modbus Network, Technology.

1 INTRODUCCIÓN

El trabajo propuesto consiste en el desarrollo de un módulo didáctico que emula una sección de un sistema de control industrial (ICS), concretamente una celda de control basada en una red Modbus que pueda ser utilizado en el Laboratorio de Redes Industriales para emular la sección de un sistema de control distribuido y que pueda integrarse al ICS didáctico modular del Laboratorio de Redes Industriales. Además, se propone el diseño de una interfaz para el monitoreo y supervisión del proceso a través de una aplicación diseñada en Intouch.

1.1 OBJETIVO GENERAL

Diseñar e Implementar un módulo didáctico que emule una celda de control industrial basada en una red Modbus.

1.2 OBJETIVOS ESPECÍFICOS

Realizar una investigación bibliográfica sobre el funcionamiento y las características de las redes Modbus, sus componentes y su uso en el monitoreo y control de procesos industriales.

Establecer los requerimientos de diseño de la celda de control industrial basada en una red Modbus y los requerimientos para el desarrollo de la interfaz de monitoreo y control.

Diseñar la celda de control industrial basada en una red Modbus y la interfaz de monitoreo y control.

Implementar la celda de control diseñada, así como la interfaz de monitoreo y control.

Validar el funcionamiento de la celda de control industrial basada en una red Modbus y la interfaz de monitoreo y control.

1.3 ALCANCE

A. Fase Teórica.

Se realizará una síntesis bibliográfica acerca del protocolo Modbus, enfatizando en su uso en ambientes industriales para control, supervisión y monitoreo de procesos de producción.

Se elaborará un resumen bibliográfico sobre las características más importantes de los componentes de la celda de control industrial basada en una red Modbus.

Se establecerá el conjunto de requerimientos de hardware para la implementación del módulo Modbus y la interfaz de monitoreo y control para la emulación de una celda de control industrial, con elementos adecuados para su funcionamiento.

Se seleccionará los requerimientos de software para la programación de los distintos componentes del módulo didáctico emulador de una celda de control industrial, con el objetivo de adquirir información de elementos de campo para realizar el control de actuadores en el nivel de campo, y poder interactuar a través de una aplicación con el proceso mediante una interfaz para el monitoreo y supervisión de variables de importancia desarrollada en Intouch.

B. Fase de diseño.

Se diseñará la distribución del hardware en la celda a implementar con el fin de tener una disposición adecuada de los componentes.

Se realizará una selección de los sensores y actuadores para realizar las funciones de adquisición de datos del proceso, así como su control, y se diseñará el acondicionamiento de las señales tanto de entradas como salidas.

Se establecerá las configuraciones necesarias para habilitar el protocolo de comunicación Modbus en los controladores de la red.

Se realizará el diseño del software del prototipo, que incluye la programación del controlador lógico programable para procesar la información adquirida, y realizar acciones de control con dispositivos dentro de la red Modbus.

Se realizará el diseño del HMI (Interfaz Hombre Máquina) para la supervisión y monitoreo que del proceso emulado que sea asociado al módulo didáctico para que permita al usuario interactuar con las variables.

C. Fase de implementación.

Se implementará el hardware del prototipo de acuerdo al diseño implementado en la etapa anterior.

Se desarrollará el software diseñado previamente el cual incluirá la programación del controlador lógico programable y de la interfaz de monitoreo y control previamente diseñada en Intouch.

Se realizará la configuración del servidor Modbus para realizar las funciones de adquisición de datos para el HMI con el objetivo de que el usuario interactúe con el proceso mediante la interfaz de monitoreo y control.

D. Fase de validación/análisis de resultados/pruebas de funcionamiento.

Se realizarán pruebas de funcionamiento del módulo didáctico mediante la emulación de un proceso industrial, con el objeto de verificar el funcionamiento de la celda de control basada en Modbus, al adquirir información de datos de campo, realizar acciones de control, y tener una adecuada visualización en la interfaz de operador.

1.4 MARCO TEÓRICO

A continuación, en esta sección se presenta información teórica que permite comprender la problemática y la solución propuesta en este trabajo de integración curricular. Para lo cual se describe el protocolo Modbus enfatizando su uso en ambientes industriales, se describe la introducción al protocolo Modbus TCP, sus ventajas y redes Modbus con las características más importantes.

1.4.1 PROTOCOLO MODBUS

El protocolo de comunicación Modbus se introduce en la industria en la década de 1970 patentado por la empresa “Modicon” el cual es un protocolo de la capa de aplicación, es decir, este protocolo ocupa el nivel 7 del modelo OSI [1]. Inicialmente se consideró como una interfaz punto a punto entre PLCs de la marca Modicon, el protocolo ha ido desarrollando diferentes aplicaciones y se ha dado apertura en redes multipunto mediante el protocolo TCP, suprimiendo también, la restricción de utilizarse solamente en equipos Modicon [2].

Los equipos que se comuniquen bajo el protocolo Modbus lo realizan bajo una arquitectura cliente/servidor como se visualiza en la Figura 1.1 donde el encargado de iniciar la comunicación entre los dispositivos es el cliente, esta transmisión de datos puede ser uno a uno (Unicast) o uno a todos (Broadcast) [3].

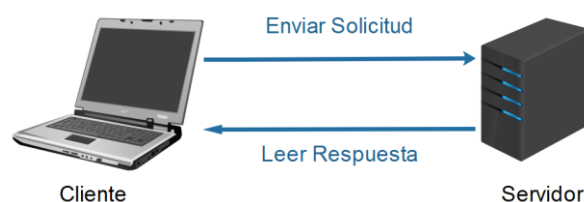


Figura 1.1. Arquitectura Cliente/Servidor.

Utilizando el protocolo Modbus, es posible emplear diversos dispositivos con la finalidad de llevar a cabo operaciones a distancia. Estos dispositivos pueden comprender PLC, dispositivos de entrada y salida, HMI, paneles de control, entre otros [3].

1.4.1.1 Interfaz de Comunicación

El protocolo Modbus está diseñado de manera flexible para su adaptación en diversos interfaces de comunicación, como RS-485 o RS-232, en función de las necesidades específicas y la infraestructura disponible. Además, en concordancia con el avance tecnológico, se ha evolucionado para poder ser utilizado en entornos de redes más extensos al implementarse a través de TCP en redes Ethernet, lo que proporciona una mayor versatilidad en la conectividad y el intercambio de datos entre dispositivos en un contexto contemporáneo y amplio [3]. En el caso de hacer uso de la tecnología Ethernet se requiere cable de par trenzado categoría 5e o superior, así como, conectores RJ-45 para los terminales como se presenta en la Figura 1.2.

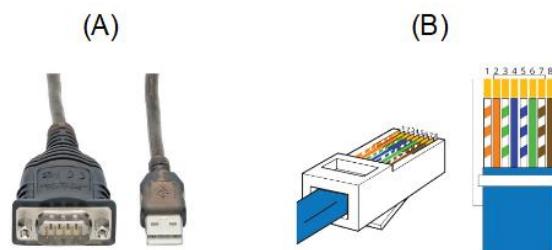


Figura 1.2. Cableado para protocolo Modbus con (A) RS-485 y (B) Ethernet.

1.4.1.2 Topología de Red

Los componentes usados para la automatización de un proceso como lo son los sensores, actuadores, PLCs, entre otros; necesitan intercambiar información por lo cual deben conectarse físicamente con cierta estructura. Las topologías de red se refieren a las diferentes estructuras en las que se pueden intercomunicar los dispositivos que conforman una red para generar el intercambio de datos [4].

Las topologías de red conllevan una topología física que se encuentra a nivel de capa física y consiste en la forma en que los cables interconectan los elementos de la red; y la topología lógica que se encuentra a nivel de capa de enlace de datos, consiste en la forma en que una red gestiona el acceso al medio físico de transmisión [5].

De las diferentes estructuras de las topologías de red se busca la máxima fiabilidad, óptimo rendimiento, mínimo tiempo de respuesta y encaminar el tráfico con el costo mínimo [4].

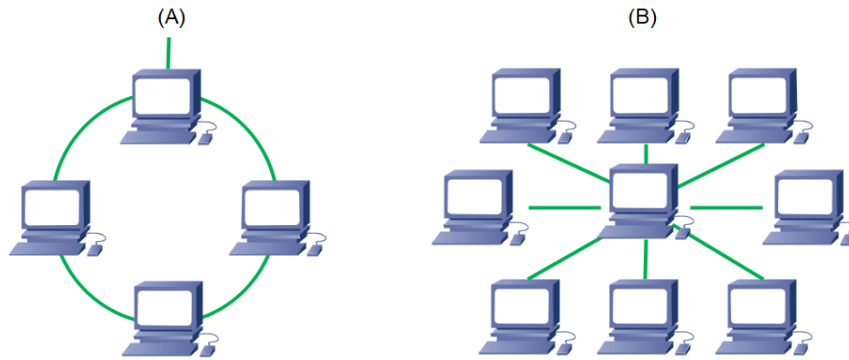


Figura 1.3. Topologías de red (A) en Anillo y (B) en Estrella.

Entre las diferentes topologías de red se encuentran las expuestas en la Figura 1.3 que corresponden a:

- Topología de anillo: En esta topología se conectan los nodos en serie y la transmisión de mensajes se realiza pasando por todos los nodos que sean necesarios hasta llegar al destino tal y como se visualiza en la Figura 1.3A. Al usar este tipo de topología es necesario utilizar un protocolo que permita eliminar bucles, sin embargo, con esta topología se logra reducir el cuello de botella dado que existen dos accesos, además que brinda un buen nivel de resiliencia [5].
- Topología en estrella: Para esta topología se tiene conectados todos los nodos a un nodo central y este es el encargado de gestionar la comunicación como se presenta en la Figura 1.3B. En este tipo de topologías se encuentra que ante cualquier perturbación en el nodo central se tendría el fallo de toda la red, sin embargo, a diferencia de la topología en anillo, los mensajes no pasan por tantos nodos [5].

1.4.2 VARIANTES MODBUS

El protocolo Modbus, a lo largo del tiempo, ha desarrollado varias variantes que le permiten adaptarse a diferentes necesidades y tecnologías. Estas variantes hacen referencia a cómo se codifican y envían los datos entre dispositivos. Además, determinan el medio de comunicación por donde los datos son transmitidos, ya sea un cable, una red inalámbrica u otro canal similar [6].

Entre las variantes se tiene:

- **Modbus ASCII:** su principal característica es que realiza la transmisión de datos en formato ASCII. Además, se basa en conexiones de serie [6]. La trama en Modbus ASCII se muestra en la Figura 1.4.

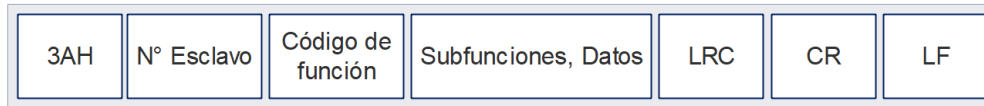


Figura 1.4. Codificación de la trama en Modbus ASCII.

- **Modbus RTU:** realiza la transmisión de datos en formato binario. Se basa en conexiones de serie como RS-485 [6]. La trama en Modbus RTU se muestra en la Figura 1.5.



Figura 1.5. Codificación de la trama en Modbus RTU.

En las tramas expuesta se pueden evidenciar los siguientes campos:

- Número de esclavo: campo donde se especifica el número de esclavo destinatario de la trama. Puede ir del 1 al 247, donde el maestro puede tomar cualquier dirección en ese rango [1].
- Código de función: campo donde se detalla la operación que se desea que realice el esclavo. Puede ser Lectura/Escritura u ordenes de control [1].
- Subfunciones y Datos: dentro de este campo se envía la información para ejecutar la operación seleccionada. Aquellos dispositivos esclavos no incluyen este campo al momento de responder [1].
- LRC o CRC (Control de errores): en el caso del modo ASCII se tiene la suma de comprobación del mensaje y en RTU requiere el cálculo por medio de una formula polinómica [1].

LRC o CRC (Control de errores): en el caso del modo ASCII se tiene la suma de comprobación del mensaje y en RTU requiere el cálculo por medio de una formula polinómica [1].

- **Modbus TCP:** realiza una comunicación manejando el proceso de mensajería Modbus mediante una red de internet utilizando los protocolos TCP [1]. Al ser una variación del protocolo Modbus, la estructura de la trama no se complica dado que en la capa de aplicación TCP se encapsula la trama base del protocolo Modbus como se muestra a continuación en la Figura 1.6 [1].

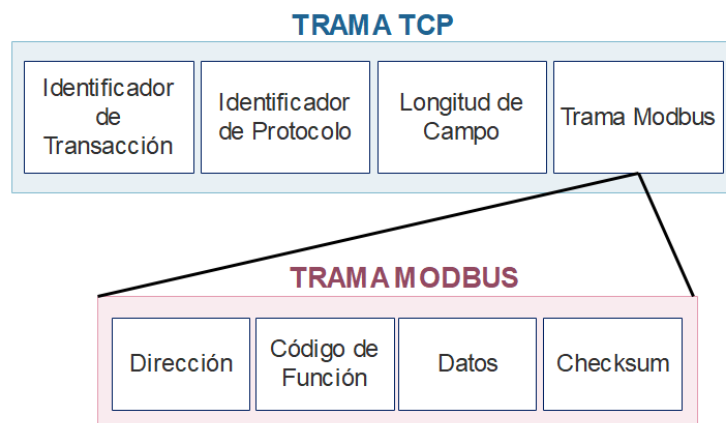


Figura 1.6. Encapsulado de la trama Modbus en TCP.

En la trama expuesta se pueden evidenciar los siguientes nuevos campos:

- **Identificador de transacción:** corresponde a un número de 2 bytes que correlaciona las respuestas del servidor con las solicitudes, cuando el cliente envía una solicitud incluye un valor único en este campo y al momento que el servidor responde, también incluye este mismo valor; lo que permite evitar confusiones en la comunicación [1].
- **Identificador de protocolo:** corresponde a un número de 2 bytes que es importante en caso de que en una red coexistan múltiples protocolos. En caso del ser protocolo Modbus, el valor es de 0 [1].
- **Longitud de campo:** corresponde a un número que especifica la cantidad de bytes de datos presentes en la unidad de datos Modbus. En caso de no coincidir, se puede producir una comunicación errónea [1].

En entornos de automatización, las siguientes ventajas correspondientes al protocolo Modbus TCP son sobresalientes:

- Este tipo de protocolo aprovecha la red Ethernet, la cual es bastante utilizada en entornos industriales, lo cual permite una comunicación rápida, confiable, su integración simple y permite conexiones a larga distancia lo cual es adecuado para aplicaciones distribuidas [1].
- La implementación y mantenimiento de redes con este protocolo suele ser menos costoso en comparación a otras soluciones, esto es debido a que se usa la tecnología Ethernet [1].

Esta variación del protocolo Modbus lleva una larga historia en el mercado, por lo que además de contar con una amplia gama de documentación, también cuenta con diversidad de dispositivos y equipos compatibles incluso de diferentes proveedores [3] [1].

1.4.3 MENSAJES MODBUS

Para establecer la comunicación entre el Cliente y el Servidor, en Modbus se establece cuatro tipos de mensajes para esta interacción, estos se muestran en la Figura 1.7:

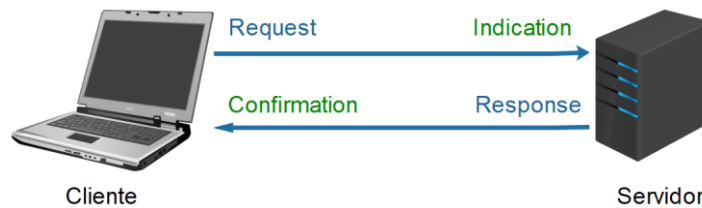


Figura 1.7. Mensajes Modbus.

Estos mensajes de son:

- Modbus Request: corresponde al mensaje enviado por el cliente a la red para iniciar la comunicación [3].
- Modbus Indication: es el mensaje que recibe el servidor de requerimiento enviado por el cliente [3].
- Modbus Response: corresponde al mensaje de respuesta que envía el servidor frente al requerimiento realizado por el cliente [3].
- Modbus Confirmation: es el mensaje que recibe el cliente de respuesta por parte del servidor al requerimiento realizado [3].

A estos mensajes, se le suma el Modbus Exception Response, que corresponde al mensaje que se envía desde el servidor en caso de existir un error [3].

1.4.4 MAPA DE DIRECCIONES MODBUS

Se refiere a la asignación de direcciones que se emplea en el protocolo Modbus para acceder a diversos registros y componentes en un dispositivo dentro una red Modbus. Este mapa de direcciones se convierte en un elemento esencial que habilita que un dispositivo solicite o modifique un dato específico en otro dispositivo [6].

Este mapa de direcciones varía dependiendo de la variante del protocolo usado, en Modbus RTU se maneja el mapa presentado en la Tabla 1.1:

Tabla 1.1. Mapa de registros Modbus.

Direcciones Modbus	Función	Descripción
1-9999	Lectura/Escritura	Salida digital (COIL Register)
10001-199999	Solo lectura	Entrada digital (CONTACT Register)
30001-39999	Solo lectura	Entrada analógica (INPUT Register)
40001-49999	Lectura/Escritura	Salida analógica (HOLDING Register)

1.4.5 REDES MODBUS

Una red Modbus hace referencia a una infraestructura física que conecta a diferentes dispositivos mediante el protocolo Modbus. Se encuentra conformada por un cliente y varios servidores los cuales tienen la tarea de recolectar información de campo mediante un sistema de adquisición de datos. Una red Modbus proporciona el medio para llevar a cabo la comunicación, esta red puede ser RS-485 o Ethernet, sin embargo, dependiente de la variante del protocolo, la red puede tener diferentes topologías o requerimientos de cableado [7]. Para visualizar un proceso de manera más dinámica, se requiere una interfaz donde se permita interpretar los datos obtenidos y mostrarlos de una manera detallada. Para una red Modbus TCP, se considera un puente de enlace entre los servidores y el cliente, al manejar una distribución ethernet se debe utilizar un switch tal y como se visualiza en la Figura 1.8.

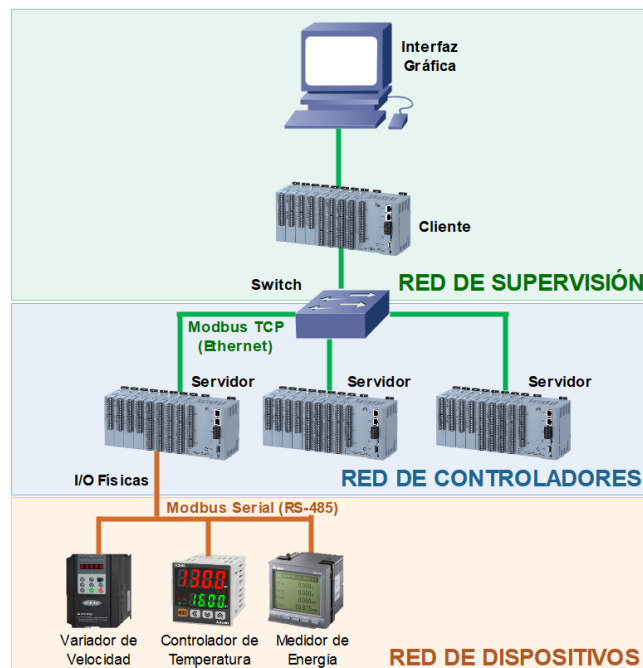


Figura 1.8. Red Modbus.

La implementación de redes Modbus se puede realizar mediante distintos medios físicos como lo es RS-485 y Ethernet. A continuación, se tallan algunas propiedades de estas tecnologías:

En la Tabla 1.2 se especifica para red Modbus con RS-485:

Tabla 1.2. Características físicas con RS-485 [8].

Característica	Descripción
Cable	Par trenzado con conectores DB-9.
Velocidad	Puede ir desde los 9600 bps hasta los 115200 bps.
Distancia	Puede alcanzar una distancia de hasta 1200m.
Topología	La topología más utilizada corresponde a la topología de bus, es decir, todos los nodos se conectan a un único medio de transmisión.
Resistencia al Ruido	Utiliza señal diferencial, por lo que cualquier ruido tiende a ser cancelado, es decir, tiene buena resistencia al ruido y puede operar en entornos industriales ruidosos.

En la Tabla 1.3 se especifica para red Modbus con Ethernet:

Tabla 1.3. Características físicas con Ethernet [9].

Característica	Descripción
Cable	Cables Cat 5e o Cat 6 con conectores RJ45.
Velocidad	La velocidad puede ser de 10 Mbps, 100 Mbps y hasta de 1Gbps.
Distancia	Puede alcanzar hasta 100 metros sin necesidad de repetidores
Topología	Utiliza la topología estrella, es decir, los dispositivos están conectados a un switch.
Resistencia al Ruido	Este tipo de tecnología utiliza técnicas de codificación y balanceo de señales para reducir la susceptibilidad al ruido.

De manera resumida, se tiene que las redes Modbus implementada por medio de Ethernet presentan mayor velocidad, flexibilidad en la topología, pueden alcanzar largas distancias, y son menos susceptibles al ruido [9].

1.5 REQUERIMIENTOS PARA EL DISEÑO DE LA MAQUETA DE PROCESO

Para emular un proceso automatizado se requiere de una maqueta de proceso que represente una planta física que cumpla con las siguientes características:

- Se pueda obtener datos de variables físicas.
- Tamaño compacto.
- Cuento con un panel de control.
- Se pueda manipular las variables que componen el proceso mediante actuadores.

Se propone el control de nivel en dos tanques de una maqueta de proceso disponible en el Laboratorio de Redes Industriales la cual se muestra en la Figura 1.9, para lo cual, se requiere conocer el nivel mediante sensores analógicos o digitales, también es necesario controlar el flujo del líquido que ingresa o sale de cada tanque, esto se lo puede realizar mediante bombas de agua o válvulas. La maqueta debe contar con un tablero que le permita al usuario controlar el arranque del proceso, accionar los actuadores de manera manual y visualizar mediante luces indicadoras el estado de actuadores o del proceso. La maqueta tiene las siguientes dimensiones:

- Largo: 80 [cm]
- Ancho: 20 [cm]
- Altura: 90[cm]



Figura 1.9. Maqueta de proceso.

1.6 REQUERIMIENTOS DE SOFTWARE PARA LA PROGRAMACIÓN DE COMPONENTES.

Para los requerimientos de software para programar los diferentes componentes que forman parte de la celda de control son de fundamentales ya que se consideran aspectos como:

- Funciones y acciones específicas que realiza el software.
- Capacidad de crear, leer, actualizar y eliminar datos.
- Limitaciones y restricciones del software.
- Restricciones de presupuesto, normativas y regulación.

2 METODOLOGÍA

Este capítulo comprende la fase de diseño e implementación de este proyecto, considerando como parte fundamental los requerimientos para la elaboración de la celda de control basada en el protocolo Modbus para la interconexión de los componentes que la conforman.

2.1 REQUERIMIENTOS DE LA RED A DISEÑAR

La celda diseñada debe cumplir con el objetivo principal de llevar datos obtenidos de un grupo de mediciones de variables físicas a través de una red de comunicación Modbus conformada por controladores lógicos programables de diferente marca y/o fabricante, hacia una interfaz gráfica, considerando un tiempo de respuesta mínimo y evitando la pérdida de información durante el proceso.

En base al funcionamiento mencionado se pueden establecer los siguientes requerimientos:

- **Dispositivos compatibles con el protocolo Modbus:** Los dispositivos que formen parte de la red deben ser compatibles con el protocolo Modbus, tanto el cliente como los servidores, se puede encontrar una gran variedad de dispositivos que cumplen los requerimientos necesarios, entre estos se encuentran controladores lógicos programables, unidades terminales remotas, variadores de frecuencia, dispositivos de adquisición de datos, algunos tipos de sensores y controladores de temperatura.
- **Red de Comunicación:** Se requiere establecer una conexión entre cada controlador lógico utilizando una red Ethernet con un correcto direccionamiento de cada dispositivo.
- **Interfaz gráfica:** Que brinde al usuario una interacción con las variables que se encuentran en el proceso, se deben considerar también las tendencias en tiempo real de las variables, así como, alarmas e históricos para un mejor conocimiento del proceso.
- **Maqueta de proceso:** Para la validación de la celda a desarrollar, se propone una maqueta que represente una planta física que permite obtener las variables físicas necesarias para validar su funcionamiento.

2.2 SELECCIÓN DE EQUIPOS

Uno de los puntos importantes de la celda a diseñar es la interconexión entre dispositivos compatibles con el protocolo Modbus de diferente marca y/o fabricante, esto con el objetivo de plantear el diseño de la celda de manera universal y aplicable a cualquier dispositivo que se tenga a disposición y cumpla con las cualidades necesarias.

Los equipos que serán utilizados para la adquisición de datos deben contar con las siguientes características:

- Ser compatible con el protocolo Modbus TCP
- Puerto Ethernet

Para el monitoreo del proceso se requiere un computador con una buena capacidad de procesamiento y cuente con el software necesario para el desarrollo de la interfaz, también debe contar con un puerto Ethernet o contar con una conexión Wifi.

Finalmente, para interconectar los equipos de la celda, se requiere un switch con un número de puertos de acuerdo con el número de dispositivos que conforman la red.

De acuerdo con las características mencionadas anteriormente, los equipos que cumplen con los requerimientos y se tienen a disposición son los siguientes:

- **Siemens S7-1200 1214 AC/DC/Relay:** Se presenta en la Figura 2.1, es un controlador lógico programables de la familia Siemens S7-1200, utilizado en procesos de automatización industrial que cuenta con un puerto Ethernet y es compatible con el protocolo Modbus [10].



Figura 2.1. PLC Siemens S7-1200 1214 AC/DC Relay [10].

- **Allen-Bradley Micrologix ML-1400 serie C:** Se presenta en la Figura 2.2, es un controlador lógico programable fabricado por la empresa AllenBradley, utilizado en

aplicaciones industriales que cuenta con un puerto Ethernet y es compatible con el protocolo Modbus [11].



Figura 2.2. PLC Micrologix ML-1400 serie C [11].

- **Schneider MODICON M580:** Se presenta en la Figura 2.3, es un controlador lógico programable fabricado por la empresa Schneider Electric, este es utilizado para aplicaciones industriales complejas, entre sus características principales cuenta con un puerto Ethernet y es compatible con el protocolo Modbus [12].



Figura 2.3. PLC Schneider MODICON M580 [12].

- **Allen-Bradley Micro 850:** Se presenta en la Figura 2.4, es un controlador lógico programable fabricado por la empresa Allen-Bradley, de la familia Micro800, cuenta con un puerto Ethernet y es compatible con el protocolo Modbus [13].



Figura 2.4. PLC Allen-Bradley Micro 850 [13].

- **Computador Quasad:** Se presenta en la Figura 2.5, este computador de gran capacidad de procesamiento cuenta con, 16GB de memoria RAM, un procesador Intel Core i7, cuenta con un puerto Ethernet y ha sido designado para llevar a cabo las actividades relacionadas con el proyecto de titulación.



Figura 2.5. Computadora Quasad.

2.3 DISEÑO DEL SISTEMA DE CONTROL BASADO EN MODBUS

Inicialmente se propone una arquitectura de red para conocer las interacciones que se van a presentar en el sistema de control basado en Modbus, en la Figura 2.6 se muestra la arquitectura de red donde se cuenta con un DCS conformado por la interfaz gráfica, un cliente y dos servidores, también con un SIS conformado por dos servidores y la interfaz.

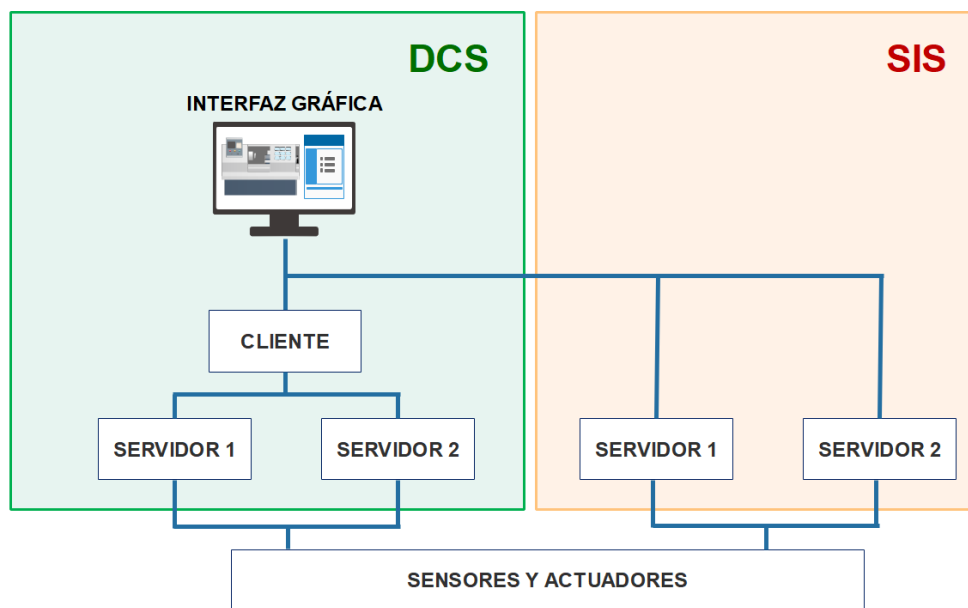


Figura 2.6. Arquitectura de red.

Ahora se diseña la red basada en protocolo Modbus TCP, teniendo en cuenta los dispositivos que la conformarán en relación CLIENTE/SERVIDOR tomando en cuenta la funcionalidad, como sistema de adquisición de datos, interfaz gráfica, o sistema integrado de seguridad.

Para conocer varios de los casos posibles que se pueden encontrar en una red Modbus conformado por dispositivos de diferente marca y/o fabricante se establece una distribución de la siguiente manera:

- Siemens S7-1200 1214 AC/DC Relay (CLIENTE)
- Micrologix ML-1400 serie C (SERVIDOR)
- Schneider M580 (SERVIDOR)
- Schneider Micro 850 (SIS)
- Computador (INTERFAZ GRÁFICA)

A continuación, se muestra un esquema del sistema de control basado en Modbus en la Figura 2.7, donde se puede notar los servidores que se encuentran tomando los datos del proceso, un switch que interconecta todo el sistema de control, el cliente y la interfaz gráfica, todo interconectado sobre una red Ethernet.

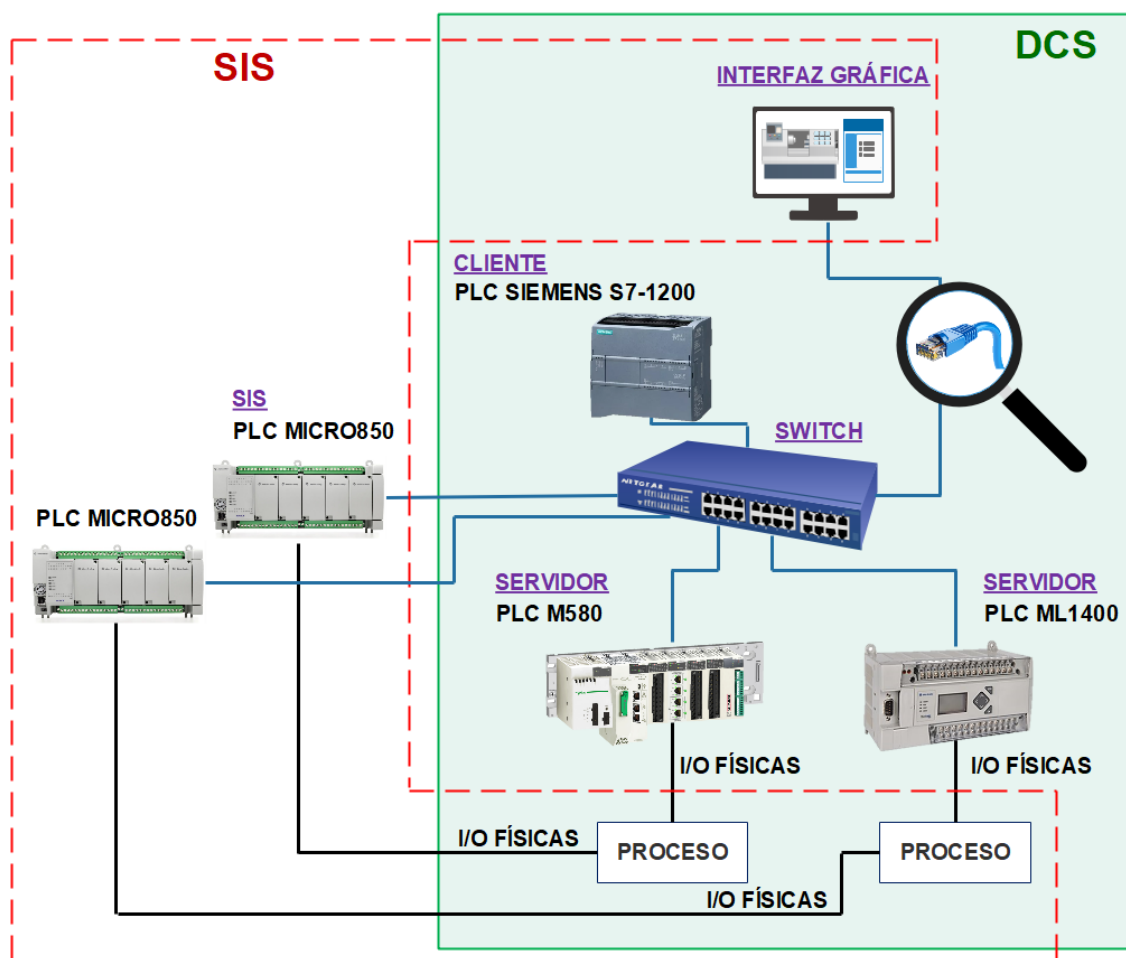


Figura 2.7. Sistema de control basado en Red Modbus.

2.3.1 SERVIDORES

Los servidores son los encargados de recolectar los datos de las variables físicas obtenidas del proceso automatizado, agrupar, y enviar de manera ordenada hacia el cliente cuando este realice una petición.

2.3.2 CLIENTE

El cliente es el encargado de realizar la petición de datos a cualquiera de los servidores, para la lectura/escritura de uno o varios registros. La información obtenida se organiza de forma sistemática para posteriormente ser enviada nuevamente hacia la interfaz para el monitoreo de las variables.

2.3.3 INTERFAZ GRÁFICA

La interfaz gráfica se encarga del monitoreo y configuración de parámetros necesarios para la operación del proceso automatizado, el dispositivo donde se realice la interfaz toma el papel de cliente para recopilar los datos de aquellos dispositivos que figuran como clientes en un nivel más inferior, es decir, los dispositivos que figuraban como clientes ahora pasan a ser servidores del cliente de mayor nivel.

Se requiere que la interfaz tenga la capacidad de mostrar los datos del proceso automatizado obtenidos a través del sistema de control descentralizado conformado por una red cliente/servidores, y también por un sistema integrado de seguridad completamente independiente.

2.3.4 SISTEMA INTEGRADO DE SEGURIDAD

El SIS, es un sistema independiente del DCS el cual recopila información del proceso automatizado y lo envía directamente hacia el interfaz. También tiene la independencia de actuar a través de sus actuadores en casos críticos de operación, esto para evitar el daño temporal o permanente de los dispositivos que conforman el proceso.

2.4 DISEÑO DE INTERFAZ GRÁFICA

La interfaz gráfica por desarrollarse juega un papel muy importante ya que, permitirá visualizar en tiempo real el comportamiento de las variables físicas del proceso, comprobando si efectivamente se realiza de manera correcta la comunicación entre los dispositivos de la red basada en protocolo Modbus.

2.4.1 ARQUITECTURA DE NAVEGACIÓN DE LA INTERFAZ

Para el diseño de una interfaz intuitiva y amigable con el usuario se consideran tres niveles de navegación como se presenta en la Figura 2.8, en cada nivel se establecen diferentes prioridades y secciones que se encuentran relacionados directamente con el nivel de acceso del usuario que haya iniciado sesión en ese momento.

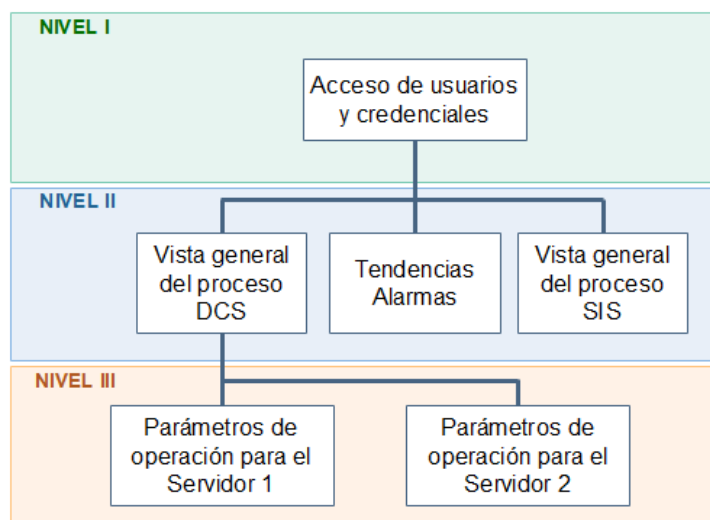


Figura 2.8. Arquitectura de navegación de la interfaz.

2.4.1.1 Primer Nivel

En este nivel se realiza la validación de usuario y contraseña para diferentes niveles de acceso, dependiendo del usuario se permite o no el ingresar nuevos usuarios con diferentes niveles de acceso.

Se consideran los siguientes niveles de acceso:

- **Gerente:** Cuenta con el permiso de más alto nivel, puede asignar nuevos usuarios, visualizar y modificar los parámetros de operación del proceso.
- **Ingeniero:** Se asigna un permiso para visualización de todas las ventanas, y puede también, modificar los parámetros de operación del proceso.
- **Operador:** Su nivel de acceso únicamente le permite ingresar a la ventana de vista general, no le es permitido ingresar a la ventana de configuración de los parámetros de operación del proceso.

2.4.1.2 Segundo Nivel

En este nivel se muestra las variables del proceso, los valores establecidos para la operación del proceso, gráficas de las tendencias de las variables, y alarmas con un registro con el objetivo de conocer anomalías en el funcionamiento del sistema.

Se considera recursos visuales para que el usuario pueda identificar el comportamiento del proceso en tiempo real. También se puede considerar una segunda ventana en el mismo nivel que cuente con las mismas características mencionadas anteriormente, con la diferencia que los datos que se muestran son obtenidos por el sistema integrado de seguridad.

2.4.1.3 Tercer Nivel

Para acceder a este nivel se requiere un nivel de acceso de gerente o ingeniero, ya que se pueden realizar cambios en los parámetros de operación del proceso, también se puede enfocar en diferentes subprocesos para contar con una visualización más detallada, se puede realizar una separación de acuerdo con la información que maneja cada servidor.

2.4.2 ALARMAS Y TENDECIAS

Las alarmas que se van a mostrar en la interfaz deben mostrarse en caso de una operación fuera de un rango seguro de trabajo, también se debe considerar la hora y fecha en la que se presentó la alarma para contar con un registro del comportamiento del sistema. Para casos extremos se debe mostrar una ventana emergente que no permita reanudar la operación del sistema a menos que se realice una aceptación por parte del usuario.

Se considera una gráfica que muestre en tiempo real la variación de la o las variables de mayor importancia, como nivel, temperatura, presión, etc.

2.4.3 DISEÑO DE VENTANAS

Cada ventana que conforme la interfaz gráfica debe mantener una estructura definida, esto con el objetivo de presentar un entorno esquematizado para el usuario. Se considera un esquema como el que se muestra en la Figuras 2.9, 2.10, 2.11 donde se puede notar una distribución de datos de tal manera que el usuario pueda encontrar en cada ventana la información en ubicaciones específicas como, la fecha y hora, el estado de conexión de los servidores, los parámetros definidos de funcionamiento del proceso, y las opciones para la navegación entre ventanas.

USUARIO	DATOS DEL ESTADO DE CONEXIÓN DE LOS SERVIDORES	PARÁMETROS DE FUNCIONAMIENTO	FECHA HORA
VISTA GENERAL DEL PROCESO		TENDENCIAS ALARMAS	
			NAVEGACION ENTRE VENTANAS

Figura 2.9. Ventana general del proceso.

USUARIO	DATOS DEL ESTADO DE CONEXIÓN DE LOS SERVIDORES	PARÁMETROS DE FUNCIONAMIENTO	FECHA HORA
PARÁMETROS DE CONFIGURACIÓN DEL PROCESO		VISTA GENERAL DEL SUBPROCESO	
			NAVEGACION ENTRE VENTANAS

Figura 2.10. Ventana de subproceso.

USUARIO			FECHA HORA
USUARIO Y CONTRASEÑA			
			NAVEGACION ENTRE VENTANAS

Figura 2.11. Ventana de acceso.

Se toma en consideración también, ventanas emergentes para dar un aviso de emergencia en caso de un mal funcionamiento de la red, estas ventanas podrán cerrarse únicamente si el usuario autoriza mediante la selección de un comando, esto para evitar el daño de los actuadores o una mala operación del proceso.

2.5 SELECCIÓN DE SENSORES Y ACTUADORES DE LA MAQUETA DE PROCESO

Para conocer el nivel de los tanques es necesario disponer de sensores, por lo cual se toman en cuenta de tipo digitales (flotadores) para el SIS y analógicos.

Sensor digital: El sensor digital utilizado es un tipo flotador que, dependiendo de la posición en la que se coloque se puede trabajar como NA (normalmente abierto), o NC (normalmente cerrado).

En la Figura 2.12 se muestra el flotador utilizado para determinar el nivel bajo:



Figura 2.12. Imagen referencial del sensor flotador vertical.

Este sensor, cuenta con las siguientes especificaciones mostradas en la Tabla 2.1:

Tabla 2.1. Características sensor flotador.

Material	Pólimero y PP
Voltaje de Switch	50 Vdc
Corriente del Switch	0.5 A

El siguiente tipo de sensor cumple con el mismo objetivo de indicar un estado NA o NC, con la diferencia que su disposición de fabricación y colocación es diferente, tal como se muestra en la Figura 2.13, es utilizado para conocer el nivel alto.



Figura 2.13 Imagen referencial del sensor flotador horizontal.

Sensor analógico: El sensor utilizado es el HC-SR04 mostrado en la Figura 2.14, el cual cuenta con las siguientes características:



Figura 2.14 Imagen referencial del sensor ultrasónico HC-SR04.

Este sensor, cuenta con las siguientes especificaciones mostradas en la Tabla 2.2:

Tabla 2.2. Características sensor ultrasónico.

Voltaje de alimentación	+5 Vdc
Corriente de trabajo	15 mA
Resolución	0.3cm

Con la ayuda de una tarjeta de desarrollo (ARDUINO) se conoce e interpreta la medida que registra el sensor ultrasónico en tiempo real, mediante código de programación se limita la lectura de la señal para el valor del alto en el recipiente y se genera una señal PWM a través de un pin asignado en la tarjeta de desarrollo.

Se implementa un filtro capacitivo para la señal PWM con el objetivo de obtener el valor medio generando, una señal DC entre 0 y 5 voltios.

Bombas de agua: Para controlar el flujo de agua entre los recipientes del sistema se utilizan mini bombas de agua que se muestran en la Figura 2.15, con las siguientes características:



Figura 2.15 Imagen referencial de la bomba de agua.

Este tipo de bomba cuenta con las siguientes especificaciones de la Tabla 2.3:

Tabla 2.3. Características bomba.

Voltaje de alimentación	12 Vdc
Corriente de trabajo	0.5 a 0.7 A
Elevación máxima de succión	2m
Empuje vertical máximo	3m

Pulsadores y selectores: Para contar con un mando manual es necesario utilizar pulsadores y selectores conectados a las entradas digitales de los programadores lógicos, se muestra una imagen referencial en la Figura 2.16.



Figura 2.16 Imagen referencial de los pulsadores y selectores.

Estos pulsadores y selectores cuentan con las siguientes especificaciones mostradas en la Tabla 2.4:

Tabla 2.4. Características pulsadores y selectores.

Voltaje de aislamiento	600[V]
Corriente térmica	10[A]
Fuerza de operación	13 a 15 [N]

Luces indicadoras: Para brindar un indicativo visual del estado del proceso se requiere utilizar luces indicadoras, en este caso se seleccionan el modelo como se muestra en la Figura 2.17.



Figura 2.17 Imagen referencial de las luces indicadoras.

Estas luces indicadoras cuentan con las siguientes especificaciones mostradas en la Tabla 2.5:

Tabla 2.5. Características luces indicadoras.

Voltaje de alimentación	AC/DC de 6 a 127 [$V_{DC} - V_{AC}$]
Rango de corriente	16 a 127 [$mA_{DC} - mA_{AC}$]

Tablero

En la Figura 2.18 se puede observar el tablero diseñado con el señalamiento donde fueron conectados cada uno de los pulsadores, selectores y luces indicadoras.

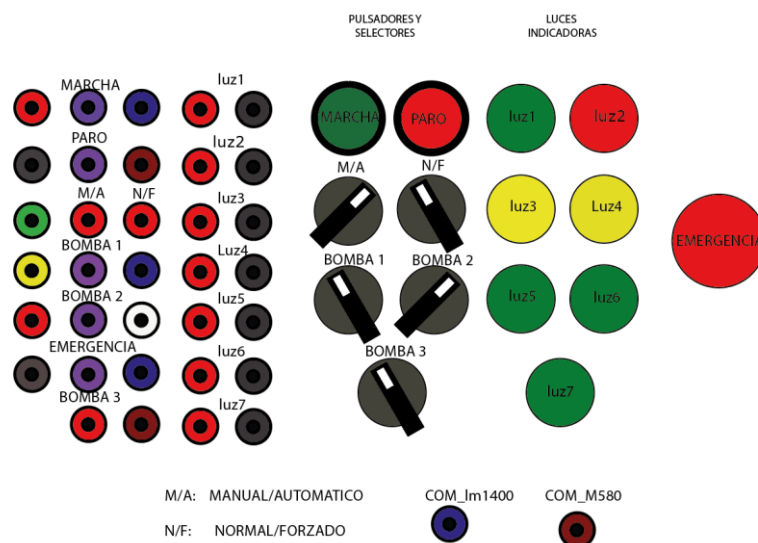


Figura 2.18 Imagen referencial del tablero diseñado.

Se propone un colocar los sensores y actuadores como se muestra en la Figura 2.19, se puede notar el panel de control que cuenta con borneras en el lado frontal de la maqueta, esto debido a que se considera un panel didáctico.

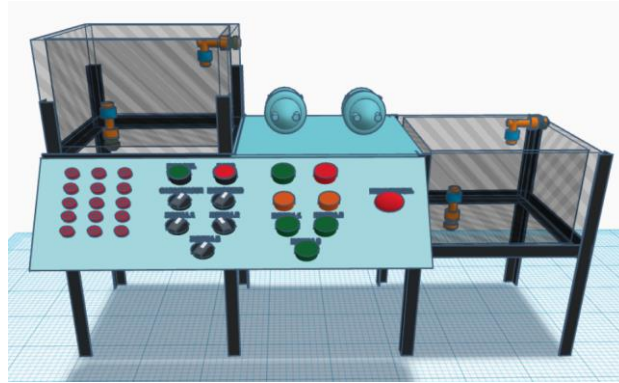


Figura 2.19 Diseño 3D de la maqueta de proceso

2.6 IMPLEMENTACIÓN DE LA RED

En este punto se toma en consideración los requerimientos de diseño mencionados anteriormente, para la puesta en marcha de la red, se distribuye este apartado en las siguientes secciones:

- **Configuración de dispositivos:** Configuración y programación necesaria para la correcta transmisión de datos.
- **Maqueta de proceso:** Elaboración de la maqueta que representará un proceso automatizado
- **Interfaz gráfica:** Se muestra la interfaz final que se ha desarrollado.

2.6.1 CONFIGURACIÓN DE DISPOSITIVOS

Los dispositivos se configuran de acuerdo con la funcionalidad que desempeñan en la red de comunicación basada en el protocolo Modbus, es por ello que, se considera presentar los parámetros necesarios para cada dispositivo y función que desempeñan.

2.6.1.1 Configuración del dispositivo Siemens S7-1200 como Cliente

Como primer punto, se debe crear un “Data block” que contenga un array con el tipo de datos que se van a utilizar para un manejo adecuado de los datos de la comunicación, entre los grupos se deben considerar del tipo “Bool” para los Coil y Contact, y tipo “Word” para los Input Register y Holding Register como se muestra en la Figura 2.20.

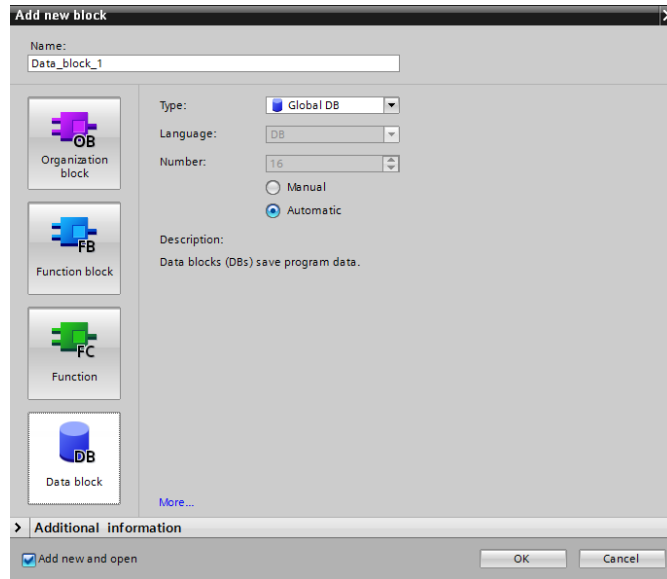


Figura 2.20 Creación Data Block.

Para manejar un orden y organización a detalle con los datos que se van a recibir y transmitir desde y hacia los servidores se utiliza la siguiente nomenclatura, “Identificador del servidor_tipo de dato”, como identificadores de servidor se utiliza “M11400” que hace referencia al servidor con el PLC Micrologix ML1400 y “M580” que hace referencia al PLC Modicon M580, mientras que el tipo de dato hace referencia a los datos tipo Coil, Contact, Input Register y Holding Register, esto se puede observar en la Figura 2.21.

Datos				
	Name	Data type	Offset	Start value
1	Static			
2	ml1400_coil	Array[0..15] of Bool	0.0	
3	ml1400_contact	Array[0..15] of Bool	2.0	
4	ml1400_input	Array[0..31] of Word	4.0	
5	ml1400_holding	Array[0..31] of Word	68.0	
6	m580_coil	Array[0..15] of Bool	132.0	
7	m580_contact	Array[0..15] of Bool	134.0	
8	m580_input	Array[0..31] of Word	136.0	
9	m580_holding	Array[0..31] of Word	200.0	

Figura 2.21 Datos designados en el “Data Block”.

Para cada registro se asigna un rango de datos definidos, en el caso de los datos tipo “Bool” se crea un arreglo de 16 datos, mientras que, para los datos tipo “Word” se crea un arreglo con 32 datos, en la Figura 2.22 se visualiza los 16 datos creados en el arreglo correspondiente para el dato Coil del servidor Micrologix ML1400.

Datos					
	Name	Data type	Offset	Start value	R
1	Static				
2	ml1400_coil	Array[0..15] of Bool	0.0		
3	ml1400_coil[0]	Bool	0.0	false	
4	ml1400_coil[1]	Bool	0.1	false	
5	ml1400_coil[2]	Bool	0.2	false	
6	ml1400_coil[3]	Bool	0.3	false	
7	ml1400_coil[4]	Bool	0.4	false	
8	ml1400_coil[5]	Bool	0.5	false	
9	ml1400_coil[6]	Bool	0.6	false	
10	ml1400_coil[7]	Bool	0.7	false	
11	ml1400_coil[8]	Bool	1.0	false	
12	ml1400_coil[9]	Bool	1.1	false	
13	ml1400_coil[10]	Bool	1.2	false	
14	ml1400_coil[11]	Bool	1.3	false	
15	ml1400_coil[12]	Bool	1.4	false	
16	ml1400_coil[13]	Bool	1.5	false	
17	ml1400_coil[14]	Bool	1.6	false	
18	ml1400_coil[15]	Bool	1.7	false	

Figura 2.22 Espacios a detalle en el Data Block.

Como siguiente paso, es necesario configurar un bloque de cliente apuntando a cada servidor de la red, este bloque se lo puede encontrar ingresando en el set de INSTRUCTIONS seleccionando la opción de COMMUNICATION, OTHERS y finalmente MODBUS TCP como se muestra en la Figura 2.23.

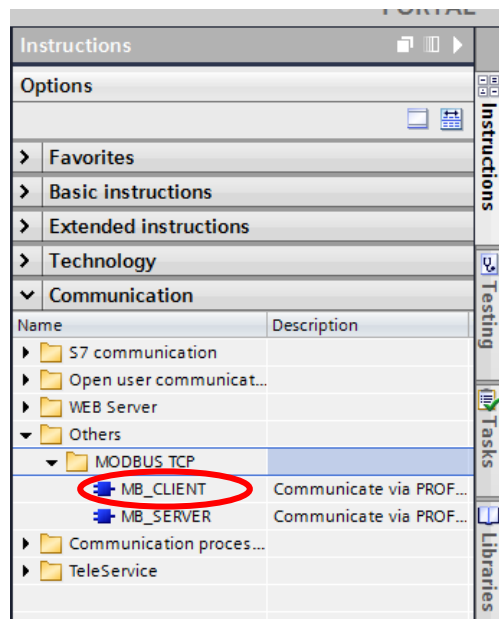


Figura 2.23 Creación Cliente Modbus.

Como se puede notar en la Figura 2.24, el bloque crea por sí solo un data block con los diferentes parámetros que se pueden configurar, para acceder a los parámetros de configuración, en el bloque de programas en la pestaña “Program resources” se encuentra el data block con el nombre asignado.

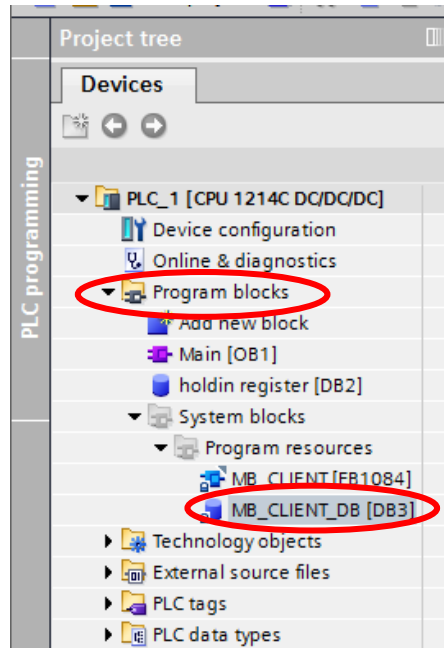


Figura 2.24 Configuración del cliente Modbus.

Son necesarios configurar los siguientes parámetros:

- **Señal o pulso de requerimiento:** Señal que realiza el requerimiento para solicitar datos al servidor al cual se esté apuntando.
- **Dirección IP del o los servidores:** Dirección IP asignada a cada servidor.
- **Modo (lectura o escritura):** Indica en el requerimiento si se desea leer o escribir en un registro del servidor.
- **Número de los datos:** Limita el número de datos que se va a enviar o recibir en cada requerimiento.
- **Dirección donde se van a almacenar los datos adquiridos:** Es el espacio asignado en el “Data Block” creado anteriormente para almacenar los datos de la comunicación.
- **Puerto (502):** El protocolo Modbus TCP utiliza el puerto de comunicación 502
- **Dirección de los registros donde se van a tomar los datos:** En este espacio se debe colocar la dirección Modbus de registro que se requiere leer o escribir en el servidor.

Como se muestra en la Figura 2.25, el bloque “MB_CLIENT” se configura para cada tipo de dato y cada servidor que se requiera, en este caso se considera dos datos tipos “Bool”

y dos registros tipo "Word" para ambos servidores. Al ser varios requerimientos se debe utilizar una secuencia para enviar cada mensaje de manera consecutiva, para ello, se utilizan los siguientes parámetros:

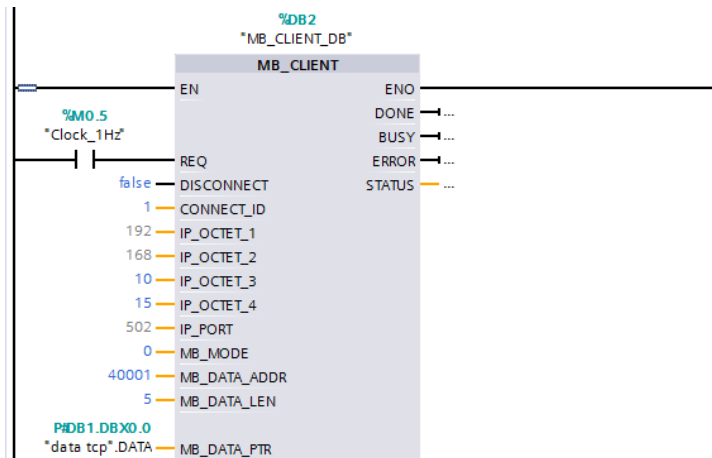


Figura 2.25 Bloque de configuración para el cliente.

- **Done:** Indica si el requerimiento se realizó de manera correcta
- **Error:** Indicador de que el requerimiento tuvo un error.
- **Status:** Indica el estado del requerimiento, pueden presentarse varios estados como, requerimiento en proceso, sin conexión, requerimientos cruzados, los mensajes de estado se encuentran detallados de manera completa en el ANEXO I.

Con la ayuda de los parámetros mencionados anteriormente, se configura el envío de datos con la secuencia que se presenta en la Figura 2.26. La idea principal de la secuencia de mensajes consiste en utilizar los indicadores "Done y Error" del mensaje anterior, para habilitar el siguiente mensaje, también se debe realizar una conmutación entre los servidores, es decir, al terminar los mensajes de un servidor estos se deshabilitan y se habilitan los mensajes del siguiente servidor.

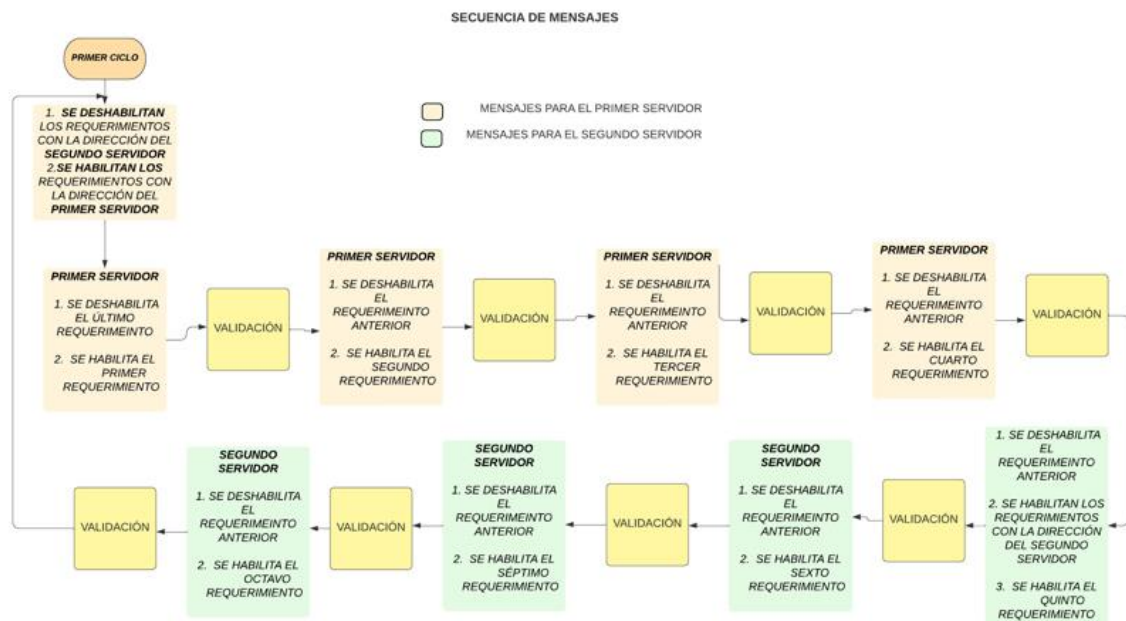


Figura 2.26 Secuencia de Mensajes.

Los indicadores “Done, Error y Status” son utilizados en el proceso de validación el cual consiste en verificar si el mensaje se realizó de manera correcta, si presenta un error, o por el contrario no realiza ninguna acción pues el requerimiento presenta un valor en “Status” diferente a cero, con lo cual se considera un tiempo definido por el usuario y pasado ese tiempo se habilita el mensaje siguiente.

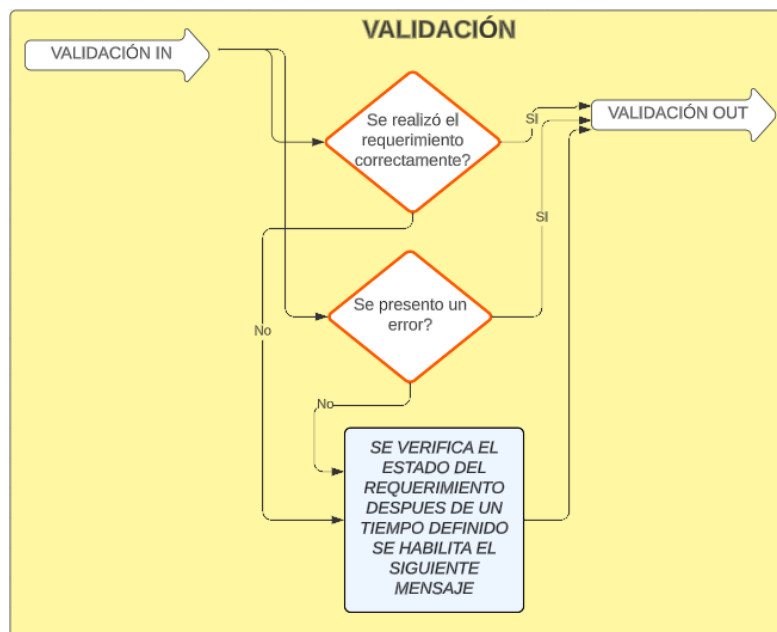


Figura 2.27 Validación de mensajes.

La Figura 2.27 representa la lógica del proceso de validación, si el mensaje se ha realizado de manera correcta o presenta un error, se activan los indicadores “Done o Error” respectivamente, dando paso al mensaje siguiente. Por otro lado, si la comunicación no se ha realizado debido a un problema en los parámetros del bloque de configuración, se activa un contador que permite definir un tiempo de espera, este a su vez, al terminar el tiempo establecido activa un indicador dando paso al siguiente mensaje.

2.6.1.2 Configuración de Transmisión de datos del Servidor Micrologix MI-1400 hacia el Cliente

Es necesario configurar el canal uno asignando una dirección IP, y habilitar el protocolo Modbus TCP como se muestra en la Figura 2.28.

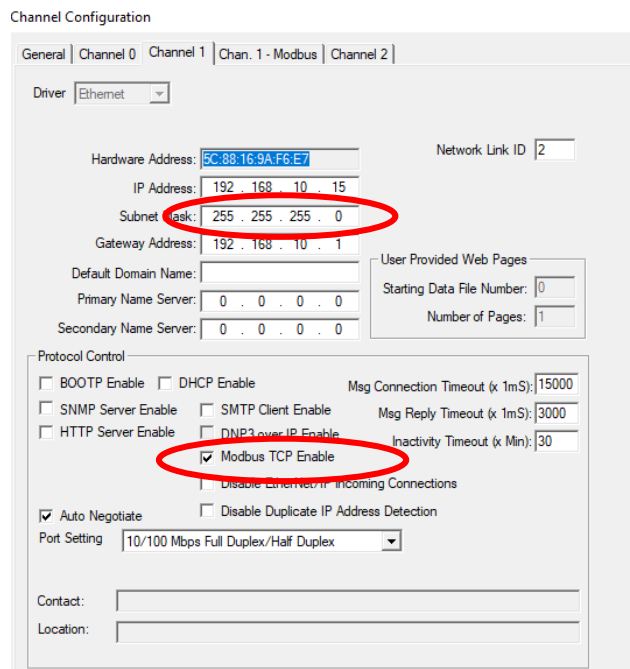


Figura 2.28 Configuración del servidor.

En la pestaña del canal UNO Modbus se configuran los Coil Register, Contact Register, Input Register, Holding Register, donde se debe asignar el número del “Data File” con el cual será identificado en los espacios de memoria, se puede establecer una prioridad para el cliente colocando la dirección IP de este.

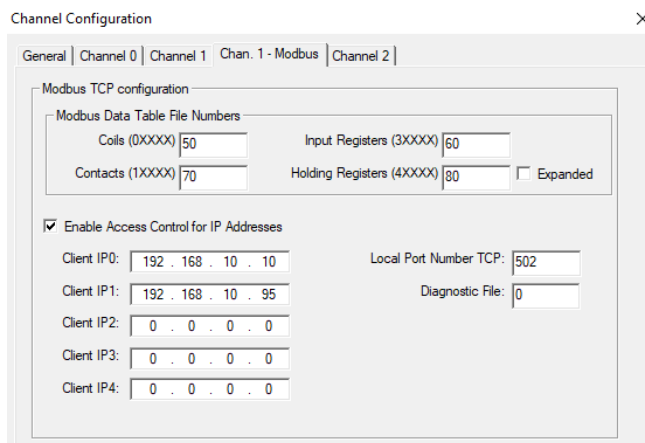


Figura 2.29 Configuración de los registros Modbus en el servidor.

Para el direccionamiento de los registros se debe aclarar que, los valores asignados en la configuración de la Figura 2.29 son utilizados internamente en la programación del programador lógico, es decir, en la dirección 60 se van a almacenar datos enteros, sin embargo, para acceder a esos datos desde un cliente mediante el protocolo Modbus es necesario colocar la dirección Modbus 3XXXX.

En la Tabla 2.6 se muestran ejemplos más detallados para comprender de mejor manera la asignación del espacio de memoria y el direccionamiento de los datos para el protocolo Modbus.

Tabla 2.6. Ejemplo de direccionamiento de datos en protocolo Modbus.

Descripción	Registro	Dirección Interna	Dirección Modbus
MARCHA	COIL	B50:0/0	00001
MARCHA_HMI	CONTACT	B70:0/1	10002
Nivel1	INPUT Register	N60:2	30003
SetPoint1	HOLDING Register	N80:3	40004

2.6.1.3 Configuración de transmisión de datos del Servidor Modicon M580 hacia el Cliente

Inicialmente se debe asignar una dirección IP al PLC, en este caso, seleccionando el módulo de comunicaciones DIO, en la pestaña "IPConfig" se puede asignar la dirección IP y una máscara de subred como se muestra en la Figura 2.30.

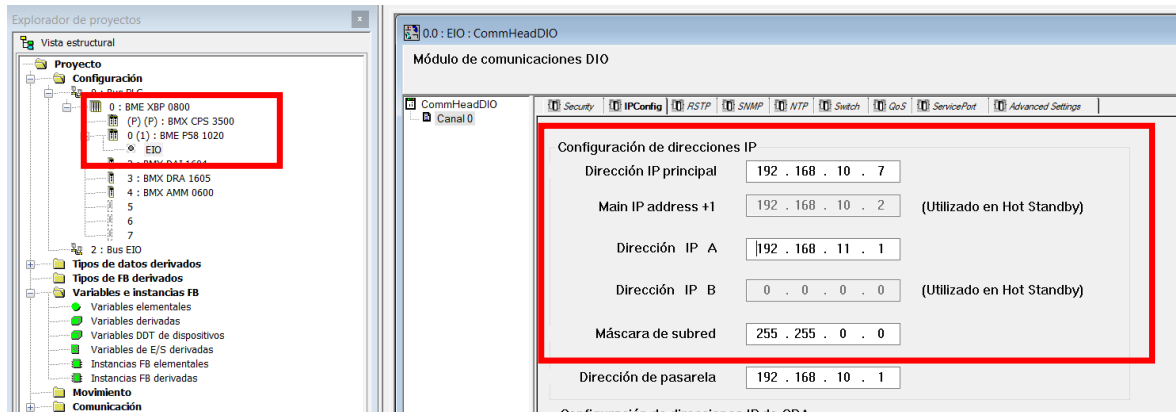


Figura 2.30 Asignación de dirección IP a PLC Modicon M580.

Al abrir el módulo de comunicación BME P58 1020, se puede observar que el canal TCP/IP toma la dirección IP y la máscara de subred para su configuración, esto se puede visualizar en la Figura 2.31.

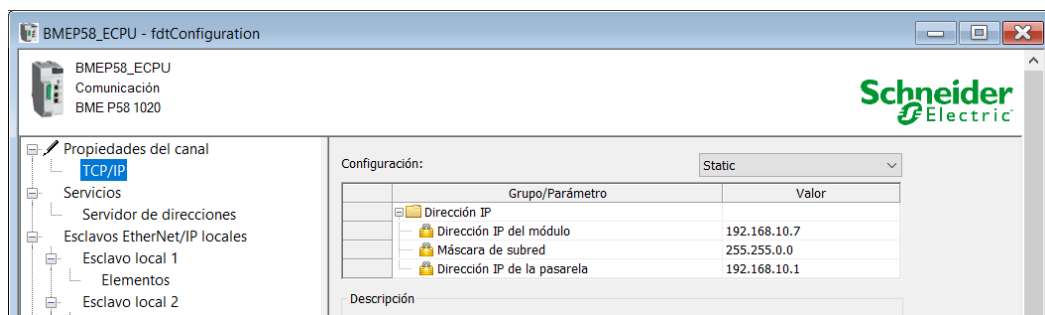


Figura 2.31 Configuración de Canal TCP/IP.

Para acceder a la configuración del canal TCP/IP se lo hace dando clic derecho en el CPU general de la configuración para el PLC Modicon M580 como se muestra en la Figura 2.32. Al utilizar un módulo didáctico establecido del laboratorio de Redes Industriales se recomienda tomar en consideración la configuración que se detalla en el Anexo II.

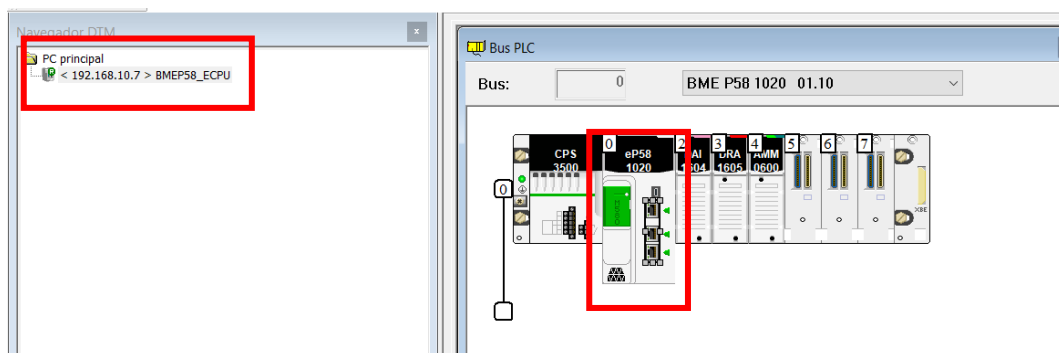


Figura 2.32. Ingreso a configuración de módulo de comunicación.

Una vez configurada la dirección IP del PLC Modicon M580, el cliente puede acceder a las variables creadas a través de su dirección Modbus. Por ejemplo, para un rango de valores dentro de las variables del programa como se muestra en la Figura 2.33, donde se designan variables tipo “EBOOL” para cada dato Coil.

Nombre	Tipo	Dirección	Valor
coil_1	EBOOL	%M400	
coil_2	EBOOL	%M401	
coil_3	EBOOL	%M402	
coil_4	EBOOL	%M403	
coil_5	EBOOL	%M404	
contact_1	EBOOL	%M1	
contact_2	FROOI	%M2	

Figura 2.33 Definición de variables de comunicación.

Las direcciones de las variables establecidas anteriormente deben colocarse en el cliente con la función de escritura o lectura, para este caso se define el requerimiento de escritura sobre las variables que contienen las direcciones desde 00401 hasta 00404, la configuración de parámetros en el bloque del cliente se muestra en la Figura 2.34.

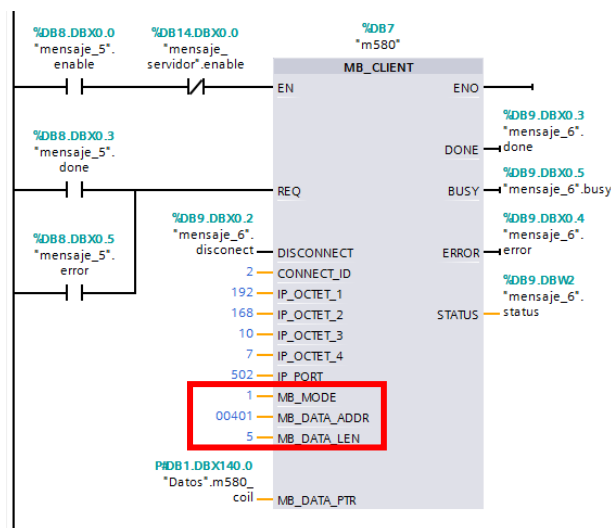


Figura 2.34 Bloque de requerimiento de escritura de datos tipo “Bool”.

Para mantener un orden y organización adecuados es necesario definir un rango de valores que sean designados para cada tipo de variables, con ello, se puede definir un rango de direcciones que el cliente puede direccionar para leer o escribir cada tipo de variable requerida, esto con la finalidad de no reescribir algún dato que podría generar errores.

Se puede mencionar un ejemplo para el requerimiento de lectura de una variable tipo entera, se muestra en la Figura 2.35 la definición de variables enteras en el software Unity Pro-XL las cuales han sido designadas para los Input Registers.

input_1	INT	%MW200
input_2	INT	%MW202
input_3	INT	%MW204
input_4	INT	%MW206
input_5	INT	%MW208

Figura 2.35 Definición de variables de comunicación.

Para este rango de datos, el cliente se configura con los parámetros que se muestran en la Figura 2.36.

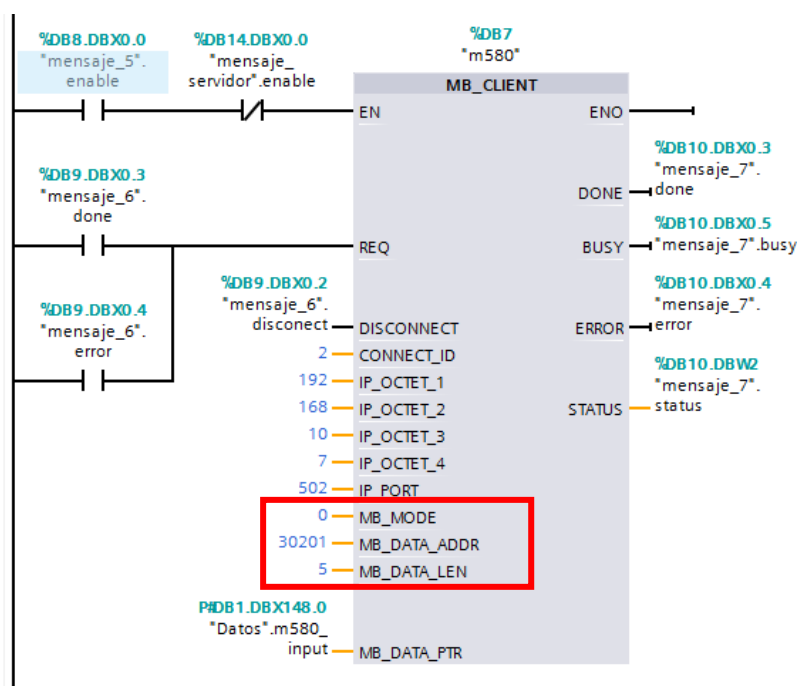


Figura 2.36 Bloque de requerimiento de lectura de datos tipo enteros.

2.6.1.4 Configuración para la transmisión de datos entre la Interfaz Gráfica y el Cliente

Para que se efectúe la transmisión de datos entre la interfaz y el cliente es necesario aclarar que, el dispositivo asignado como cliente pasa a cumplir las funciones de un servidor y la interfaz toma el papel de cliente.

Es por ello por lo que se requiere configurar el bloque “MB_Server” en el dispositivo Siemens S7-1200, tal y como se presenta en la Figura 2.37. Este dispositivo para la

transmisión de datos únicamente utiliza el registro holding, es decir, solamente se envían y reciben datos tipo “Word”.

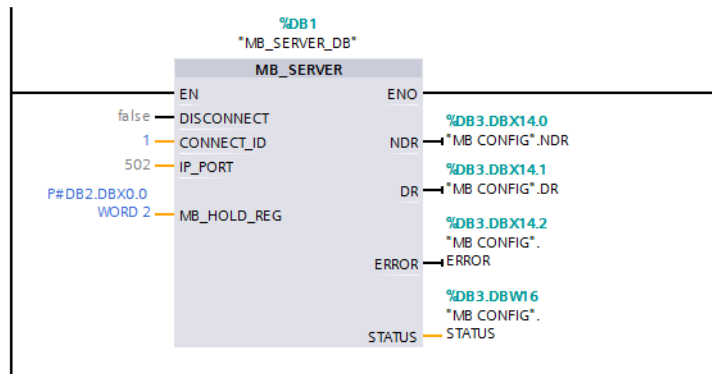


Figura 2.37 Bloque de configuración para el servidor.

Los parámetros para configurar son:

- Puerto (502 Modbus)
- Dirección del “Data block” donde se encuentran los datos para la comunicación

La interrogante se presenta con la necesidad de transmitir datos tipo “Bool”, ya que, no sería óptimo utilizar un espacio de 32 bits (Word) para enviar un dato de un bit. Por esa razón se utiliza una codificación y decodificación para transformar los bits en datos tipo “Word” tanto para enviar como para recibir, en el Anexo III se muestra a detalle el proceso para realizar la conversión de datos.

Los datos que recibe la interfaz deben ser configurados de tal manera que sean direccionados desde un espacio de memoria del servidor a través de un servidor de datos que se configura en el software System Manage Console en el computador

A. CONFIGURACIÓN DEL SERVIDOR DE DATOS

El servidor de datos es utilizado para interconectar el cliente con la interfaz gráfica, para lo cual con la ayuda del software System Manage Console se realiza el levantamiento del servidor DASMBTCP de la siguiente manera:

- 1 Crear un nuevo NODO.
- 2 Agregar un puerto TCP/IP.
- 3 Agregar un “Modbus PLC”.
- 4 Configurar la dirección IP.

- 5 Establecer el número de bit que utiliza el PLC para la comunicación.
- 6 Establecer el tamaño de los registros de comunicación.
- 7 Crear un tópico para identificación en Intouch.

El proceso detallado de los pasos a seguir para la configuración del servidor de datos se encuentra en el ANEXO IV.

B. CONFIGURACIÓN DE VARIABLES EN INTOUCH

El siguiente paso se desarrolla en el software Intouch, donde se debe crear un “AccessName”, tal y como se muestra en la Figura 2.38, este se utiliza para direccionar los datos que se van a enviar y recibir.

Figura 2.38 Creación del Access Name.

- **Access Name:** Representa el identificador del enlace entre el cliente y el servidor, el cual debe ser configurado en cada variable requerida.
- **Node Name:** Corresponde a la identificación del nodo o dispositivo en una red de comunicación.
- **Topic Name:** Actúa como una etiqueta o nombre para un canal de comunicación.

La variable requerida se debe configurar como una variable tipo entrada/salida entera, como un caso especial para el modo de funcionamiento del PLC siemens S/-1200 como servidor, solamente se pueden enviar y recibir datos tipo “holding”, es decir, desde el computador se debe enviar un dato tipo entero que contenga los datos discretos y posterior a ello, el servidor debe decodificar para interpretar los datos bit por bit. Los tipos de variable se pueden visualizar en la Figura 2.39.

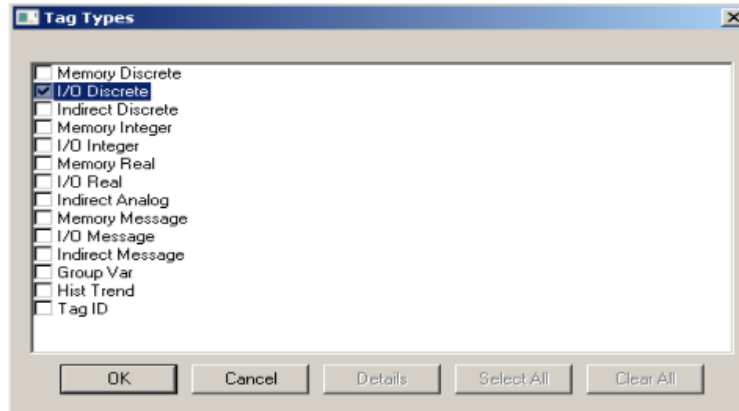


Figura 2.39 Tipos de datos en Intouch.

Se debe especificar también si el dato en la interfaz es únicamente de lectura o escritura, detallando el rango de valores permitidos y configurando el “Acces Name”, así como, la dirección Modbus que se requiere obtener desde el servidor tal y como se presenta en la Figura 2.40.

En este punto se debe mencionar que al ser un dato tipo “holding”, se puede realizar la lectura y escritura de este.

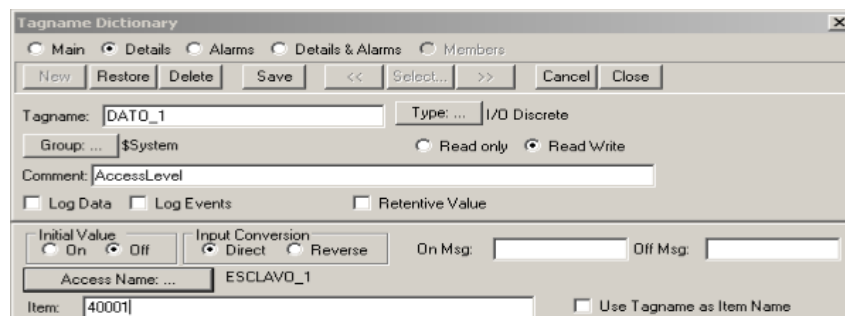


Figura 2.40. Creación de una variable en Intouch.

2.6.1.5 Configuración del Sistema Integrado De Seguridad

El sistema integrado de seguridad conformado por dos controladores lógicos Micro 850, también forman parte de la red Modbus, por ello, deben ser configurados para que se puedan acceder a los diferentes registros.

Como primer punto, se debe asignar la dirección IP al controlador lógico, como en la Figura 2.41, ya que el protocolo se encuentra montado bajo el protocolo Modbus TCP.

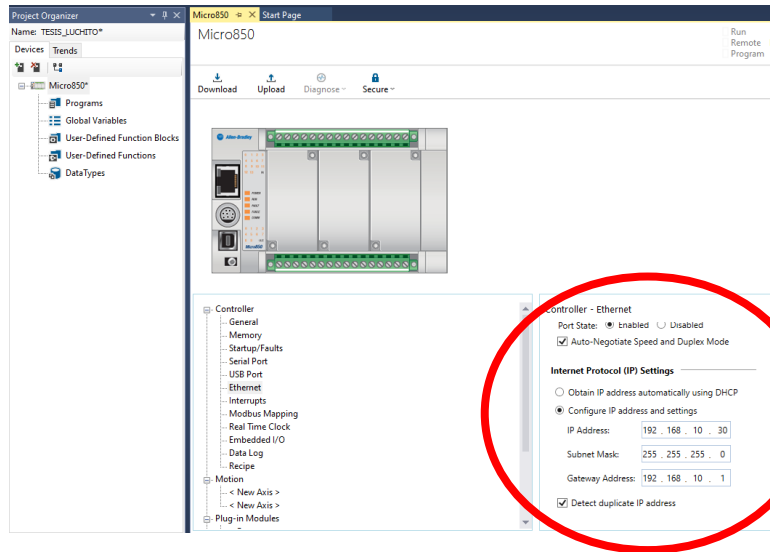


Figura 2.41 Configuración de IP del PLC Micro 850.

En la opción “Modbus Mapping” se debe asignar una variable con la dirección que va ligada con es variable como se presenta en la Figura 2.42.

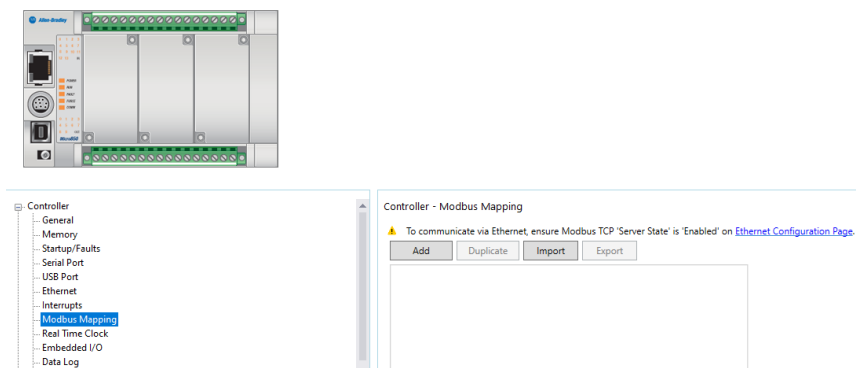


Figura 2.42 Creación de variables en el PLC Micro 850.

Para asignar una variable con la dirección, como se puede observar en la Figura 2.43 es necesario tener creadas variables del tipo entero o discreto para asignar a los registros respectivos de lectura y escritura.

Name	Alias	Data Type	Dimension	Project Value	Initial Value
NIVEL1		INT			
MAXIMO		INT		85	
ALERTA1		BOOL			
ALERTA2		BOOL			
MINIMO		INT			10
*					

Figura 2.43 Variables correspondientes al SIS.

Una vez asignada las variables, se puede acceder a ella mediante la dirección establecida como se muestra en la Figura 2.44.

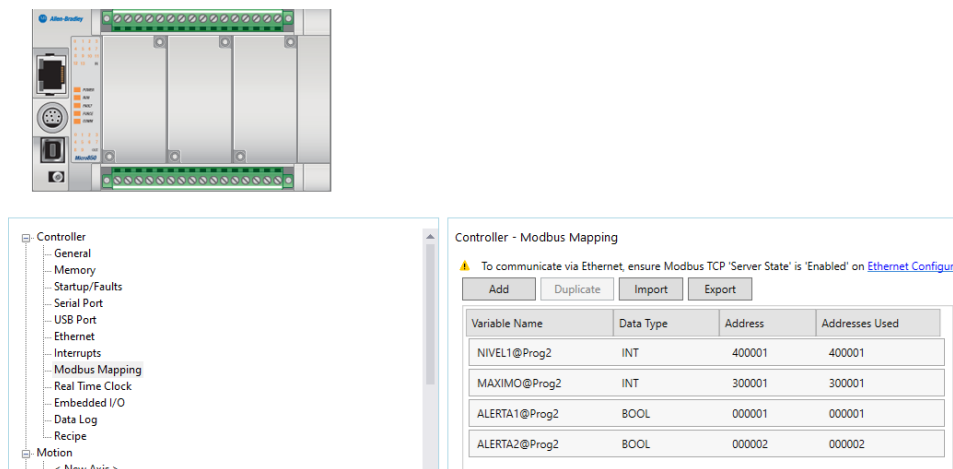


Figura 2.44 Mapa de variables Modbus.

Para acceder a los registros del SIS en la interfaz gráfica, es necesario repetir el proceso de la Sección 2.6.1.4, asignando un “Modbus PLC” con la dirección de los PLC que forman parte del SIS y creando también un “Topic” para el identificador del “AccesName” que se debe configurar en Intouch.

2.6.2 MAQUETA DE PROCESO

A continuación, en la Figura 2.45 se muestra la maqueta de proceso donde se puede apreciar los sensores digitales y el panel de control.



Figura 2.45 Maqueta de proceso implementada.

Para la interconexión de los diferentes elementos de la maqueta de proceso, se realiza una PCB, el diseño y los diagramas de conexión de la maqueta se muestra en el Anexo V.

2.6.3 INTERFAZ GRÁFICA

En la Figura 2.46 se muestra la vista general del proceso implementado, en el cual se puede observar los valores operación establecidos para su funcionamiento, también varios indicadores de los estados de las bombas, gráficas de las lecturas de nivel, así como registros de alarmas.

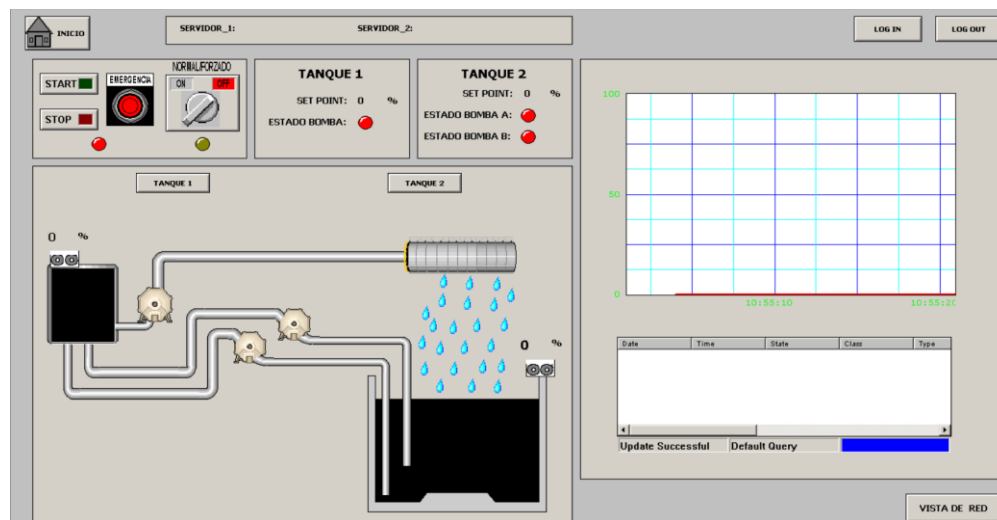


Figura 2.46 Vista general del proceso implementado.

En la Figura 2.47 se puede observar la ventana correspondiente al recipiente “elevado”, es posible realizar modificaciones en los parámetros de funcionamiento y se puede activar o desactivar la bomba dependiendo si el sistema se encuentra en modo manual o automático, también se puede manipular el modo de trabajo, esto si es normal o forzado, ya que en cada caso el flujo con el cual se envía el líquido hacia el tanque dos es mayor o menor

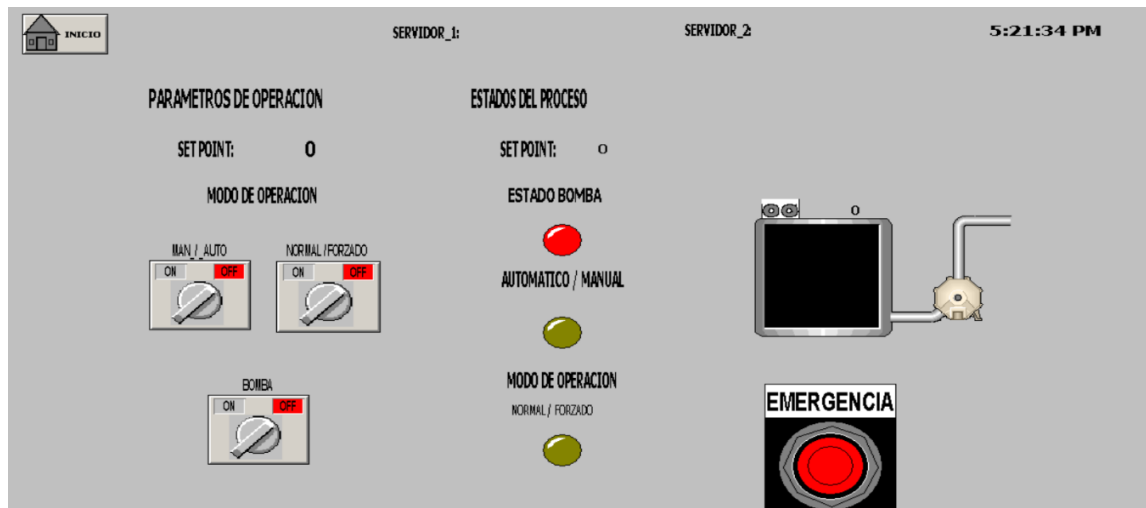


Figura 2.47 Ventana correspondiente al tanque 1(elevado).

En la Figura 2.48 se muestra la ventana correspondiente al recipiente “cisterna”, en el cual se puede modificar los parámetros de configuración, colocar en marcha o paro el proceso y accionar o desactivar las bombas.

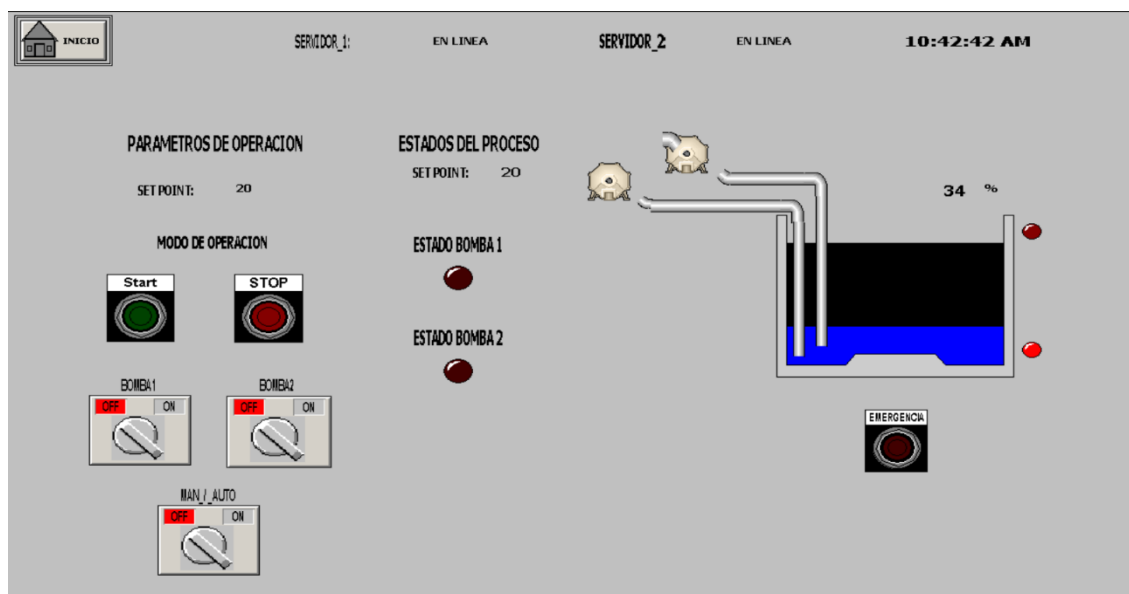


Figura 2.48 Ventana correspondiente al tanque 2 (cisterna).

Finalmente, en la misma interfaz se cuenta con una ventana que realiza un intercambio de datos con el SIS, la cual se presenta en la Figura 2.49, donde se pueden observar las variables en tiempo real y el estado de los sensores flotadores utilizados.

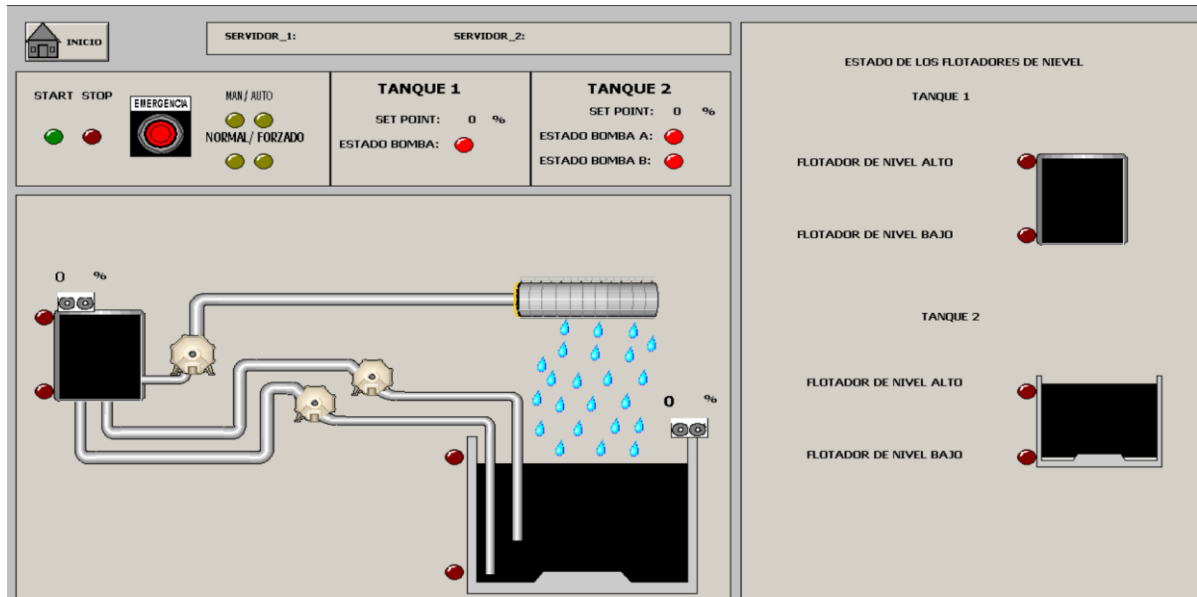


Figura 2.49 Ventana para el SIS.

2.7 VALIDACIÓN DE LA RED

La celda de control basado en el protocolo Modbus tiene como objetivo tomar datos de un proceso a través de los servidores y mostrarlos en una interfaz gráfica, es por ello que, para la validación del funcionamiento de la red, se propone los siguientes parámetros de operación para el abastecimiento de agua de una cisterna:

- El control de nivel emulará el sistema de abastecimiento de agua (cisterna) con dos modos de consumo (base y pico).
- El usuario debe definir el nivel medio de ambos tanques, esto, desde la interfaz gráfica.
- Al pulsar marcha, en la cisterna el nivel será controlado mediante el servidor Micrologix que accionará dos bombas de manera alternada una vez que el valor supere un 15% más del valor definido. Para el caso de consumo “Pico”, el agua sobrepasará el valor del 15% más del valor seteado debido a que se aumenta el flujo de líquido hacia la cisterna, por lo cual, para evitar un desbordamiento de líquido se deberá encender la bomba complementaria.
- Para el tanque elevado, el control de nivel será realizado por el servidor M580, el cual, dependiendo de la posición del selector de modo de trabajo, accionará la bomba con un voltaje de 10 o 12 voltios dependiendo el consumo base o pico. De igual manera el valor medio será configurado por el usuario mediante la interfaz

- El panel deberá indicar en todo momento el accionamiento de las bombas o la señal de arranque del proceso.

La Figura 2.50 representa un esquema de la idea conceptual utilizada para la validación de la red, donde se cuenta con dos recipientes que intercambian líquido, en este caso agua.

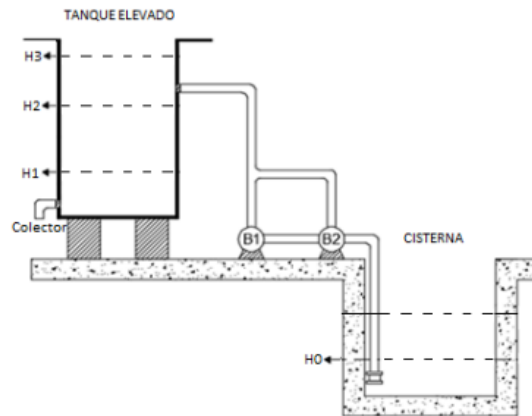


Figura 2.50. Imagen referencial del proceso a utilizar para validar el funcionamiento.

3 RESULTADOS, CONCLUSIONES Y RECOMENDACIONES

3.1 RESULTADOS

En la corroboración del funcionamiento de la celda de control basada en el protocolo de comunicación Modbus TCP, es necesario evidenciar el envío y recepción de los datos de los diferentes registros utilizados, por lo cual, para cada servidor se realizará una verificación del envío de Contact e Input Register hacia el cliente y la interfaz, de igual manera, se realiza la recepción de Coil y Holding Register desde la interfaz y el cliente hacia el servidor.

Inicialmente se focaliza la validación de comunicación en el servidor Micrologix ML-1400, se realiza el envío de un dato tipo "bool" (Contact) correspondiente a la variable "APAGADO_HMI" que indica el estado en paro del proceso, en la Figura 3.1 se observa la luz indicadora del panel principal de la maqueta de proceso que indica el estado de paro del sistema.



Figura 3.1 Panel principal de la maqueta de proceso.

Se muestra en la Figura 3.2A el dato registrado en el PLC Micrologix ML-1400 en la dirección B70:0/1, en la Figura 3.2B se observa el dato reflejado en el cliente Siemens S7-1200, y finalmente se puede observar en la Figura 3.2C la luz indicadora correspondiente a la interfaz gráfica. Con ello, queda verificado el envío de un dato tipo "Bool" del servidor Micrologix MI-1400 hacia el cliente Siemens S7-1200 y la Interfaz gráfica.

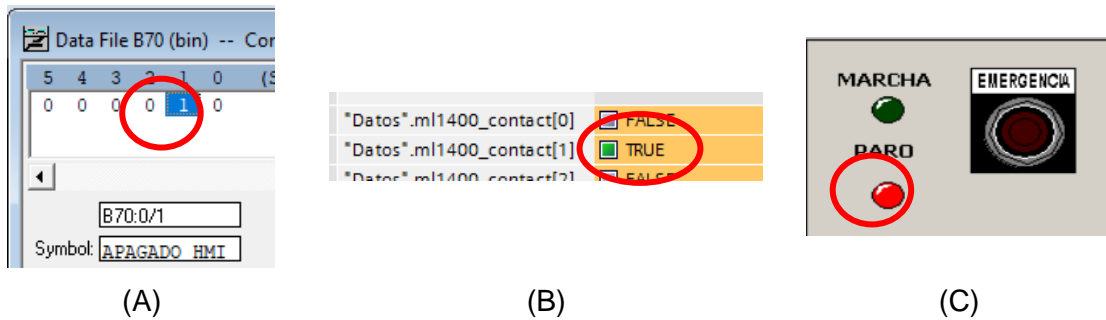


Figura 3.2 Dato Contact registrado en el Servidor Micrologix MI-1400 (A) Cliente S7-1200 (B) e Interfaz gráfica (C).

A continuación, se realiza el envío de un registro tipo entero (Input Register) correspondiente al valor del nivel de agua en la cisterna, se puede observar en la Figura 3.3 que el recipiente “cisterna” de la maqueta de proceso tiene liquido en su interior.

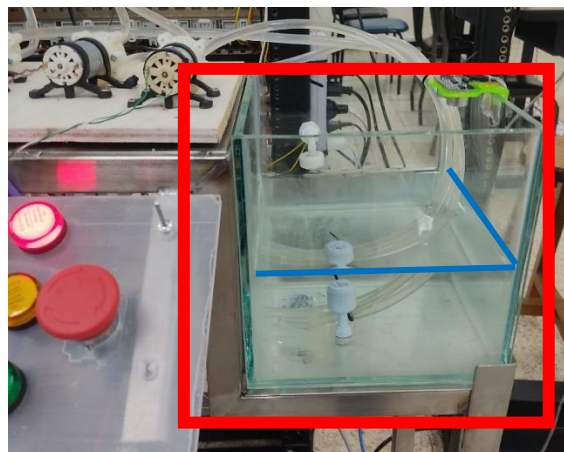


Figura 3.3 Imagen del recipiente “cisterna” de la maqueta de proceso.

El valor del nivel registrado en la dirección N60:0 del servidor Micrologix MI1400 como se muestra en la figura 3.4A, también se puede observar en la Figura 3.4B el dato reflejado en el cliente Siemens S7-1200 y finalmente en la interfaz gráfica figura 3.4C. Con ello, queda verificado el envío de un dato tipo entero del servidor Micrologix MI-1400 hacia el cliente Siemens S7-1200 y la Interfaz gráfica.

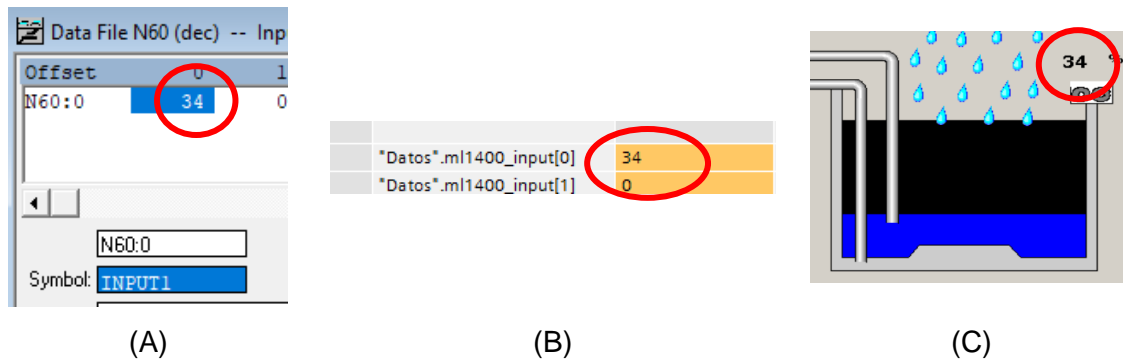


Figura 3.4 Dato Input registrado en el Servidor Micrologix MI-1400 (A) Cliente S7-1200 (B) e Interfaz gráfica (C).

Como siguiente punto, se realiza la recepción de un dato tipo “Bool” (Coil) enviado desde la interfaz gráfica hacia el servidor Micrologix MI1400. En la Figura 3.5 se muestra el pulsante de emergencia ubicado en el interfaz utilizado para la validación.



Figura 3.5 Pulsador de emergencia ubicado en la interfaz gráfica

Como se puede observar en la Figura 3.6A el dato es registrado en el Cliente Siemens S7-1200 y en el servidor Micrologix MI-1400 Figura 3.6B. Con ello, queda verificada la recepción de un dato tipo “Bool” desde la interfaz gráfica hacia el servidor Micrologix MI-1400.

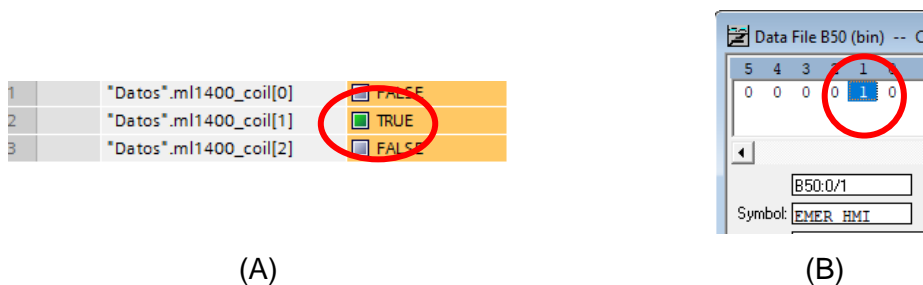


Figura 3.6 . Dato Coil registrado en el Cliente S7-1200 (A) y Servidor Micrologix MI-1400 (A)

Para finalizar la comprobación de comunicación para el servidor Micrologix MI-1400, se realiza la recepción de un dato tipo entero (Holding Register). En la Figura 3.7 se puede observar el dato “SET POINT” el cual es enviado desde la interfaz gráfica.

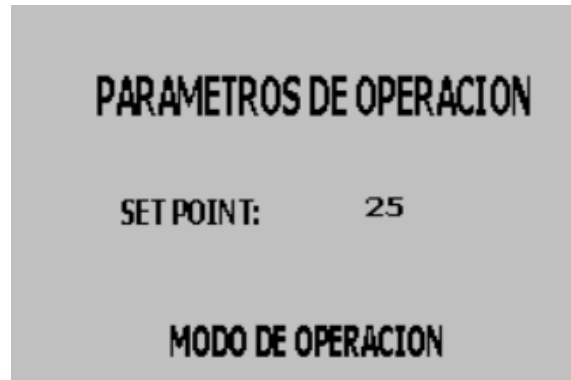


Figura 3.7 Valor SET POINT definido en la interfaz gráfica

En la Figura 3.8A se puede observar el valor registrado en el Cliente Siemens S7-1200, también como se muestra en la Figura 3.8B el dato se registra en el Servidor Micrologix MI-1400 en la variable “SP_Level” ubicado en la dirección N80:0. Con ello, queda verificada la recepción de un dato tipo “Holding Register” desde la interfaz gráfica hacia el servidor.

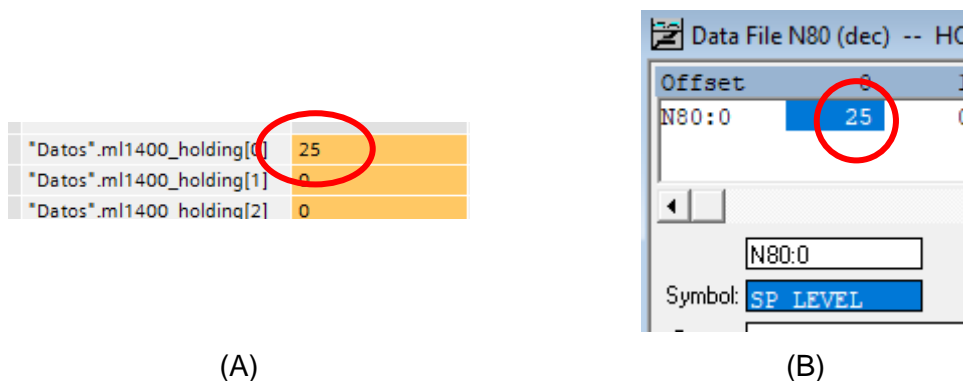


Figura 3.8 Dato Holding registrado en el Cliente S7-1200 (A) y Servidor Micrologix MI-1400 (B)

A continuación, se focaliza la validación de comunicación en el servidor correspondiente al PLC Modicon M580, se realiza el envío de un dato tipo “bool” (Contact) correspondiente a la variable “CONTACT_2” que indica el modo de operación manual/automático, en la Figura 3.9 se observa la luz indicadora y el selector correspondiente del panel principal de la maqueta de proceso que indica el modo de operación.



Figura 3.9 Panel principal de la maqueta de proceso.

Se muestra en la Figura 3.10A el dato registrado en el servidor en la variable “CONTACT_2”, en la Figura 3.10B se observa el dato reflejado en el cliente Siemens S7-1200, y finalmente se puede observar en la Figura 3.10C la luz indicadora correspondiente a la interfaz gráfica. Con ello, queda verificado el envío de un dato tipo “Bool” del servidor M-580 hacia el cliente Siemens S7-1200 y la Interfaz gráfica.

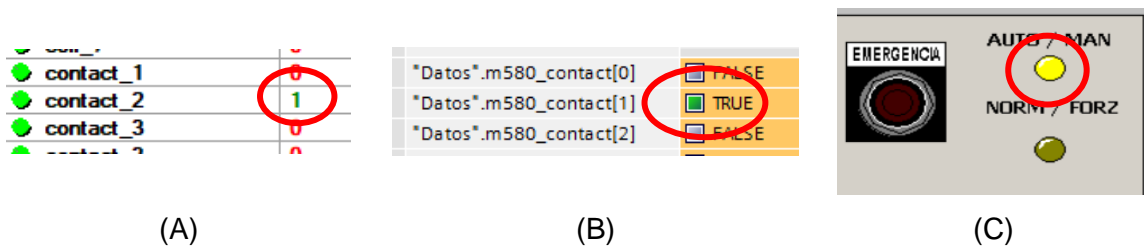


Figura 3.10 Dato Contact registrado en el Servidor Modicon M580 (A) Cliente S7-12 (B) e Interfaz gráfica (C)

A continuación, se realiza el envío de un registro tipo entero (Input Register) correspondiente al valor del nivel de agua en el recipiente elevado, se puede observar en la Figura 3.11 que el recipiente “cisterna” de la maqueta de proceso tiene liquido en su interior.

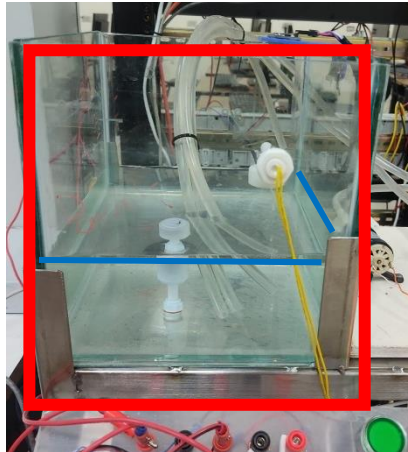


Figura 3.11. nagen del recipiente “tanque elevado” de la maqueta de proceso.

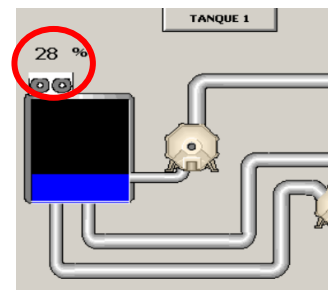
El valor del nivel registrado en la variable “Input_1” del servidor como se muestra en la figura 3.12A, también se puede observar en la Figura 3.12B el dato reflejado en el cliente Siemens S7-1200 y finalmente en la interfaz gráfica Figura 3.4C. Con ello, queda verificado el envío de un dato tipo entero del servidor Modicon M580 hacia el cliente Siemens S7-1200 y la Interfaz gráfica.

... ● holding_2	0
... ● input_1	28
... ● input_2	0

(A)

"Datos".m580_input[0]	28
"Datos".m580_input[2]	0
"Datos".m580_input[4]	0

(B)



(C)

Figura 3.12 Dato Input registrado en el Servidor Modicon M580 (A) Cliente S7-1200 (B) e Interfaz gráfica (C)

Como siguiente punto, se realiza la recepción de un dato tipo “Bool” (Coil) enviado desde la interfaz gráfica hacia el servidor Modicon M580. En la Figura 3.13 se muestra el pulsante de emergencia ubicado en el interfaz utilizado para la validación.



Figura 3.13 Pulsador de emergencia ubicado en la interfaz gráfica.

Como se puede observar en la Figura 3.14A el dato es registrado en el Cliente Siemens S7-1200 y en el servidor Modicon M580 Figura 3.14B. Con ello, queda verificada la recepción de un dato tipo “Bool” desde la interfaz gráfica hacia el servidor.

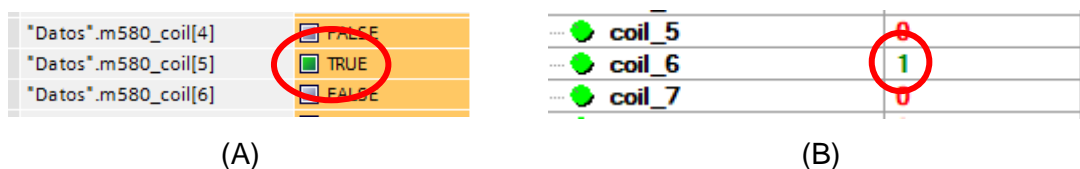


Figura 3.14 Dato Input registrado en el Servidor Modicon M580 (A) y Cliente S7-1200 (B)

Para finalizar la comprobación de comunicación para el servidor Modicon M580, se realiza la recepción de un dato tipo entero (Holding Register). En la Figura 3.15 se puede observar el dato “SET POINT” el cual es enviado desde la interfaz gráfica.

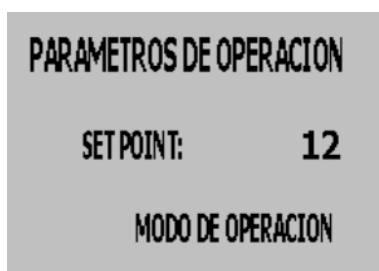


Figura 3.15 Valor SET POINT definido en la interfaz gráfica.

En la Figura 3.16A se puede observar el valor registrado en el Cliente Siemens S7-1200, también como se muestra en la Figura 3.16B el dato se registra en el servidor, en la variable “Holding_1”. Con ello, queda verificada la recepción de un dato tipo “Holding Register” desde la interfaz gráfica hacia el servidor.

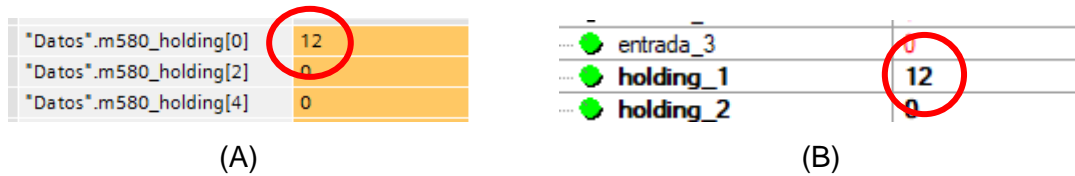


Figura 3.16 Dato Holding registrado en el Cliente S7-1200 (A) y Servidor Modicon M580 (B)

Como punto final se focaliza la validación de los servidores del SIS, en la Figura 3.17 se puede observar el valor de nivel y estado del flotador de nivel bajo, estos valores son enviados hacia el servidor Micro 850.



Figura 3.17 Valor SET POINT definido en la interfaz gráfica.

Los valores enviados desde la interfaz gráfica se pueden observar en la Figura 3.18, el valor del nivel se ve registrado en la variable "nivel" y el estado del flotador de nivel bajo en la variable "nivel_bajo".

Name	Logical Value
nivel	20
nivel_bajo	<input checked="" type="checkbox"/>
nivel_alto	<input type="checkbox"/>

Figura 3.18 Valor SET POINT definido en la interfaz gráfica.

La comunicación se encuentra verificada ya que, los datos desde los servidores hacia el interfaz, o viceversa se realiza adecuadamente.

3.2 CONCLUSIONES

En el presente trabajo de integración curricular se desarrolló el diseño e implementación de un sistema de control industrial didáctico (ICS) modular basado en una red Modbus TCP. Validando el uso del sistema para tareas de automatización industrial, mostrando también una aplicación práctica exitosa a través de su implementación.

Un aspecto destacado fue la implementación exitosa de una red Modbus TCP que incorporó dispositivos de diferente marca y fabricante compatibles con el protocolo Modbus, los cuales interactuaron sin problemas dentro de la red Modbus. Esta demostración práctica verifica la naturaleza interoperable y estándar del protocolo Modbus, al permitir que dispositivos de diferentes fabricantes se comuniquen y colaboren de manera efectiva y en tiempo real.

Es importante mencionar que el proceso de validación integrando una interfaz gráfica permitió corroborar la efectividad de la comunicación de la celda de control basada en el protocolo Modbus, proyectando los datos de la maqueta de proceso en un entorno virtual, reflejando las variaciones de manera instantánea.

Además de proporcionar conocimientos importantes sobre el diseño e implementación del protocolo Modbus en sistemas de automatización industrial, el presente trabajo también realizó valiosas contribuciones a la enseñanza y el aprendizaje práctico. Se implementó exitosamente una maqueta de proceso, ahora ubicada en el Laboratorio de Redes Industriales donde se prevé que proporcione a los estudiantes una herramienta que permitirá complementar el aprendizaje sobre la flexibilidad y la versatilidad del protocolo Modbus en la integración de sistemas heterogéneos, lo cual es crucial para su formación en la industria de la automatización industrial.

3.3 RECOMENDACIONES

Trabajos de integración curricular futuros podrían adentrarse en la ampliación del rango de dispositivos de entrada compatibles con el protocolo Modbus, es decir, podrían explorar diferentes tipos de sensores, actuadores, entre otros dispositivos que operen directamente con Modbus sin la necesidad de tener una conexión con un controlador lógico.

Sería valioso examinar la respuesta del sistema Modbus ante el incremento en la cantidad de dispositivos conectados y el impacto en el rendimiento general. Es decir, evaluar hasta qué punto, el sistema puede mantener una comunicación fluida y efectiva. Además, que,

en este mismo contexto, sería provechoso explorar estrategias para mejorar la gestión de datos.

Además, investigar formas de mejorar la seguridad en la comunicación entre el servidor y el cliente también podría representar un potencial de investigación valioso, abordando la necesidad de ciberseguridad en los sistemas de control industrial.

Con el fin de abrir nuevas oportunidades para casos donde la ubicación remota es crítica, se debe plantear un desafío de investigar la adaptación de la comunicación Modbus en entornos donde la conexión por cable no es viable, es decir, implementar el protocolo Modbus en redes inalámbricas y móviles.

4 REFERENCIAS BIBLIOGRÁFICAS

- [1] D. Reynders, S. Mackay, y E. Wright, “Industrial Data Communications”, Escuela Politécnica Nacional, Quito, 2005.
- [2] M. Liberatori, *Redes de Datos y sus Protocolos*, 1a ed., vol. 1. Universidad Nacional de Mar del Plata, 2018.
- [3] Automation Networks, “MODBUS”, 2019. [En línea]. Disponible en: <http://automation-networks.es/glossary/modbus>. (consultado el 1 de agosto de 2023).
- [4] UAL, “Topología de red”, en *REDES*, Universidad de Almería, 2003. [En línea]. Disponible en: <http://ual.dyndns.org/biblioteca/redes/pdf/unidad%2003.pdf> (consultado el 10 de julio de 2023).
- [5] “Redes de comunicación: Topología y enlaces”, en *Sistemas Industriales Distribuidos*, Universitat de València, 2010. [En línea]. Disponible en: https://www.uv.es/rosado/courses/sid/Capitulo2_rev0.pdf (consultado el 6 de julio de 2023).
- [6] A. Swales, “Open Modbus/TCP Specification”, *Schneider Electric*, marzo de 2000. http://www.dankohn.info/projects/Fieldpoint_module/Open_ModbusTCP_Standard.pdf (consultado el 6 de julio de 2023).
- [7] G. Thomas, “Introduction to the Modbus Protocol”, 2008, jul. 2008. [En línea]. Disponible en: <https://www.ccontrols.com/pdf/Extv9n4.pdf> (consultado el 9 de julio de 2023).
- [8] O. Weis, “Guía de la comunicación RS485”, *Electronic Team INC*, 2021. <https://www.eltima.com/es/article/rs485-communication-guide/> (consultado el 17 de agosto de 2023).
- [9] N. Goldenberg y A. Wool, “Accurate modeling of Modbus/TCP for intrusion detection in SCADA systems”, *International Journal of Critical Infrastructure Protection*, vol. 6, núm. 2, jun. 2013, [En línea]. Disponible en: <https://www.sciencedirect.com/science/article/abs/pii/S1874548213000243> (consultado el 14 de agosto de 2023).
- [10] SIEMENS, “SIMATIC S7-1200, CPU 1214C”, *Datasheet*. 2022. [En línea]. Disponible en: <https://docs.rs-online.com/2040/0900766b81639d65.pdf> (consultado el 16 de agosto de 2023).
- [11] Rockwell Automation, “Controladores programables MicroLogix 1400”, *Datasheet*. 2021. [En línea]. Disponible en: https://literature.rockwellautomation.com/idc/groups/literature/documents/in/1766-in001_-es-p.pdf (consultado el 16 de agosto de 2023).
- [12] Schneider Electric, “Modicon M580 Automation Platform”, *Datasheet*. 2018.
- [13] Rockwell Automation, “Micro830, Micro850, and Micro870 Programmable Controllers”, *Datasheet*. 2023.

ANEXOS

ANEXO I. Parámetros de estado de comunicación posibles para el PLC Siemens S7-1200.

ANEXO II. Configuración de PLC Schneider M580 disponible en el Laboratorio de Redes Industriales.

ANEXO III. Codificación y decodificación de datos tipo discreto para el envío y recepción desde y hacia la interfaz gráfica.

ANEXO IV. Configuración del servidor de datos

ANEXO V. Diseño PCB para interconexión de elementos que conforman la maqueta de proceso y esquemas de conexión.

ANEXO VI. Manual de usuario

ANEXO I

Parámetros de estado de comunicación posibles para el PLC Siemens S7-1200

Se pueden presentar varios casos durante la comunicación entre los dispositivos de la red Modbus, esto es un identificativo del estado de comunicación observable en el cliente, en este caso el controlador lógico Siemens S7-1200 que comanda el flujo de los datos, de manera detallada se presenta la siguiente Tabla:

Tabla I.1. Parámetros de estado de comunicación posibles.

Estados de información general	
ESTADO(W#16#)	DESCRIPCIÓN
0000	Instrucción ejecutada sin error
7000	No existe requerimiento
7001	Existe requerimiento, pero el proceso se encuentra activo
7002	El requerimiento es irrelevante, el proceso se encuentra ejecutando
7003	Conexión finalizada.
Estados de error en el protocolo	
8380	Trama Modbus recibida es incorrecta, pocos datos recibidos.
8381	Código de función no compatible
8382	Error en el tamaño del dato
8383	Dato fuera del rango de direcciones Modbus
8384	Error en el valor del dato
8385	Código de diagnóstico no compatible
Estados de error en los parámetros	
80BB	Valor ACTIVE_EST inválido
8187	Puntero de datos no compatible, área muy pequeña
818C	El puntero referencia a un bloque optimizado, utilice uno con parámetros estándar

ANEXO II

Configuración de PLC Schneider M580 disponible en el Laboratorio de Redes Industriales

Al abrir el programa Unity Pro-XL, se despliega la ventana presentada en la Figura II.1.

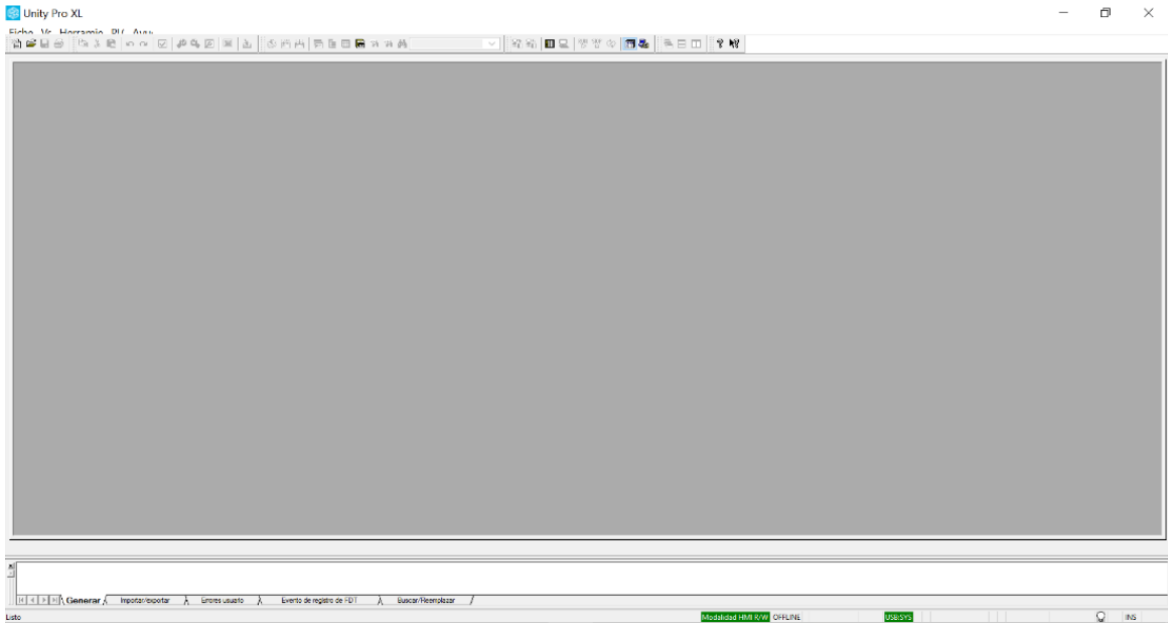


Figura II.1 Ventana inicial Unity Pro XL.

En la parte superior izquierda, en la pestaña “**Fichero**”, dar clic y seleccionar “**Nuevo**” como se muestra a continuación en la Figura II.2.

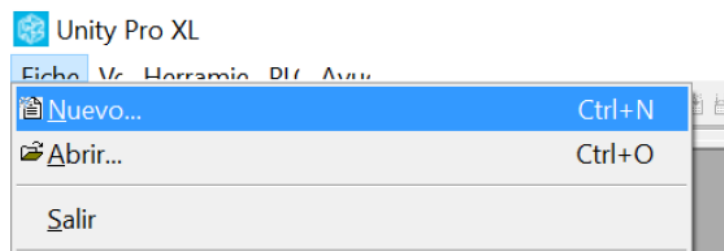


Figura II.2 Crear nuevo fichero.

Posterior a este paso, se despliega la ventana presentada en la Figura II.3.

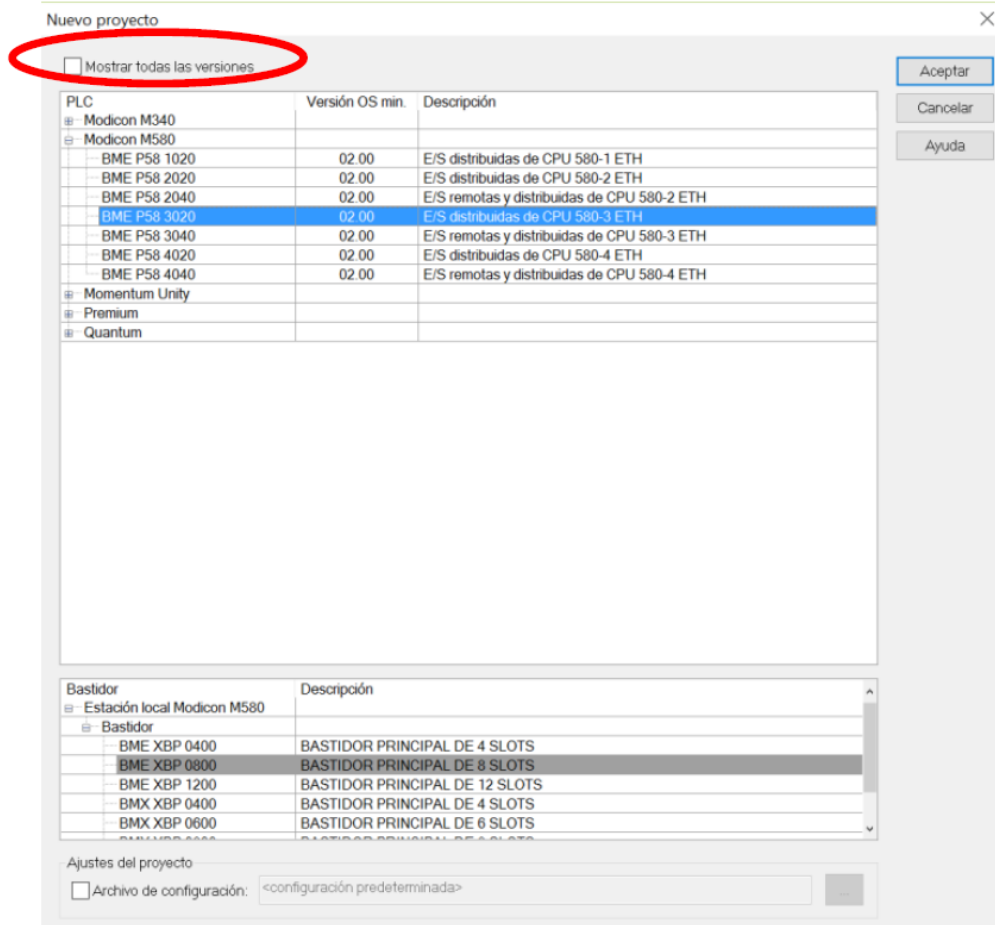


Figura II.3. Ventana de nuevo proyecto.

Se debe hacer check en el cuadro que dice **“Mostrar todas las versiones”**, para que aparezca la **Versión OS min. “01.10”** que es la versión de firmware del PLC disponible para esta implementación.

Dentro de la lista desplegada del PLC Modicon M580, se debe seleccionar el CPU el cual es **“BME P58 1020”**, Versión OS min. **“01.10”**, tal y como se muestra en la Tabla II.1.

Tabla II.1. Configuración PLC.

PLC	Versión OS min.	Descripción
666BME P58 1020	01.10	E/S distribuidas de CPU 580-1 ETH

En la parte inferior se desplegarán algunas opciones para seleccionar el Bastidor, se debe elegir el Bastidor **“BME XBP 0800”** cuya descripción es **“BASTIDOR PRINCIPAL DE 8 SLOTS”** y posteriormente se presiona **“Aceptar”** tal y como se muestra en la Figura II.4.

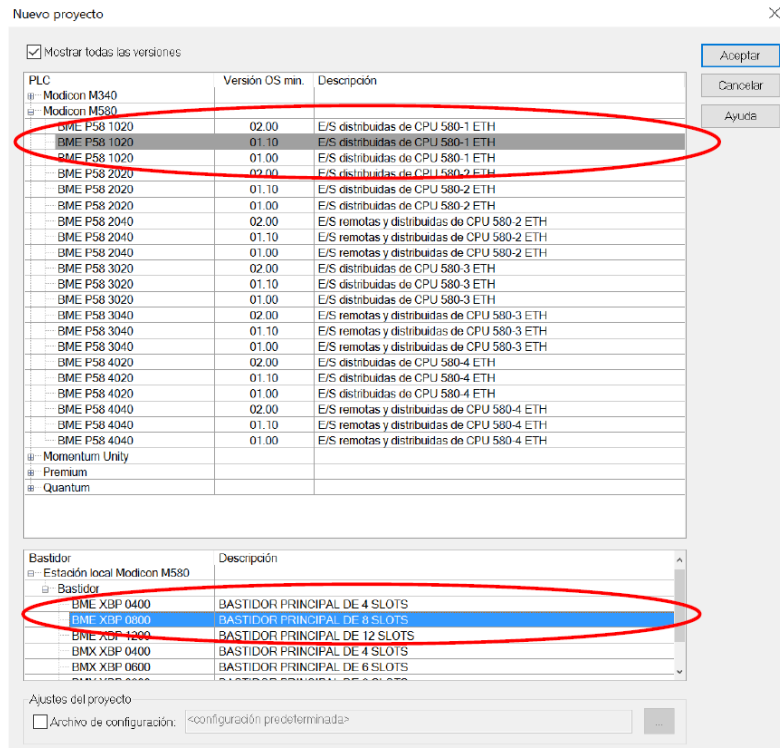


Figura II.4 Configuración CPU y bastidor del PLC.

Posterior a esto, como se observa en la Figura II.5, aparece en la parte izquierda de la pantalla una ventana llamada “**Explorador de Proyectos**”, en la cual se encuentra todo lo necesario para configurar al PLC, creación de secciones de programa, tablas de animación, pantallas de operador (HMI), entre otros.

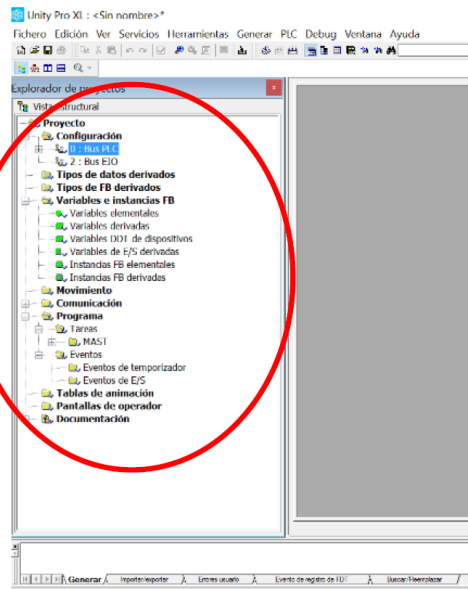


Figura II.5 Ventana de Explorador de Proyectos.

Dentro de esta ventana se cuenta con la pestaña “Proyecto” donde se tiene una carpeta que dice “Configuración” y dentro de esta se encuentran dos buses uno que dice “0: Bus PLC” y otro que dice “2: BUS EIO”.

Para seleccionar los módulos de entradas, salidas tanto digitales como analógicas y la fuente de alimentación correspondiente a lo que se tiene físicamente dentro de los tableros del PLC disponibles, se debe dar doble clic sobre “0: Bus PLC” y aparecen las siguientes ventanas “Bus PLC” y “Catálogo de Hardware” como se indica en la Figura II.6 a continuación.

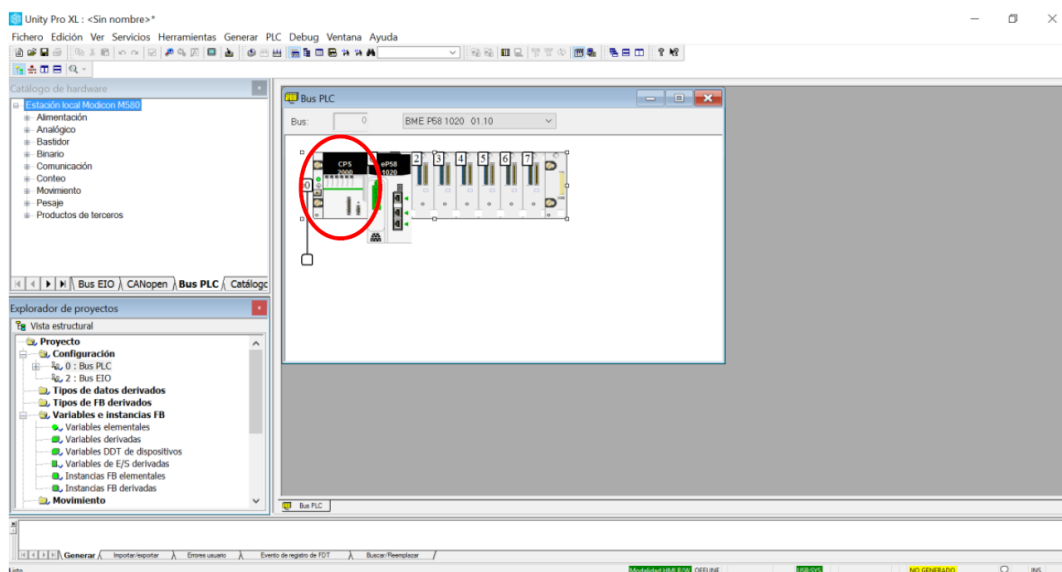


Figura II.6 Configuración del Proyecto.

La fuente del CPU que aparece por defecto no es la correcta, por lo que es necesario seleccionarla con un clic y presionar el botón de suprimir. Una vez que queda vacío el slot donde se encontraba la fuente, es necesario dar doble clic sobre este espacio y se abrirá una ventana llamada “Nuevo Dispositivo”, donde se debe seleccionar la fuente “BMX CPS 3500” como se indica en la Figura II.7.

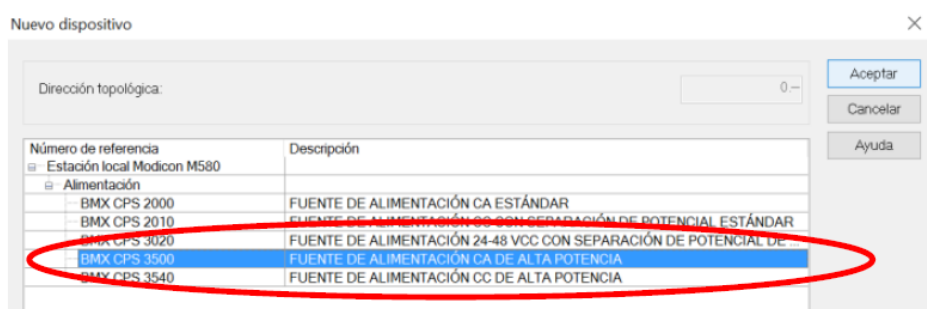


Figura II.7 Configuración de la Fuente de alimentación.

Una vez seleccionada la fuente de alimentación correcta, se deben ir seleccionando uno a uno los módulos de entradas, salidas digitales y analógicas en el orden que se encuentran colocados físicamente en el tablero como se indica en la Figura II.8.

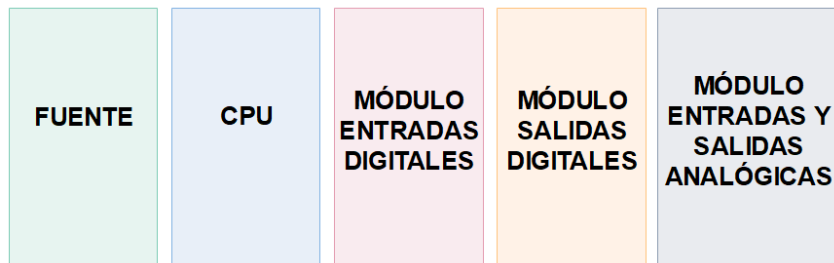


Figura II.8 Orden módulos físicos.

- **Configuración de entradas digitales**

Para la selección del módulo de entradas digitales, en la ventana “**Bus PLC**” se selecciona el slot con el número **2** con doble clic y aparece la ventana mostrada en la Figura II.9.

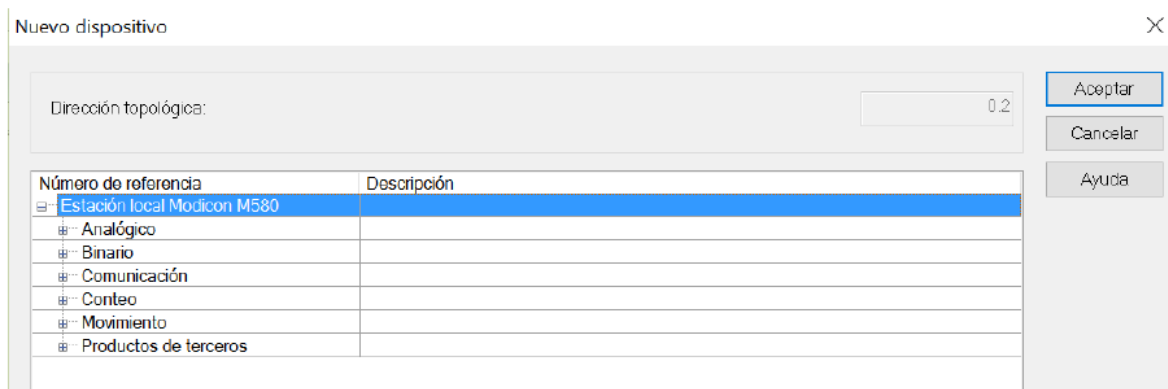


Figura II.9 Configuración nuevo dispositivo.

A continuación, se debe desplazar el menú de la opción “**Binario**” donde se encuentra diferentes módulos de entrada como salida, para este caso se selecciona el “**BMX DAI 1604**” y en el menú de la parte inferior se debe escoger “**Tipo de datos de E/S: Topológico**” y posteriormente hacer clic en “**Aceptar**” como se muestra en la Figura II.10.

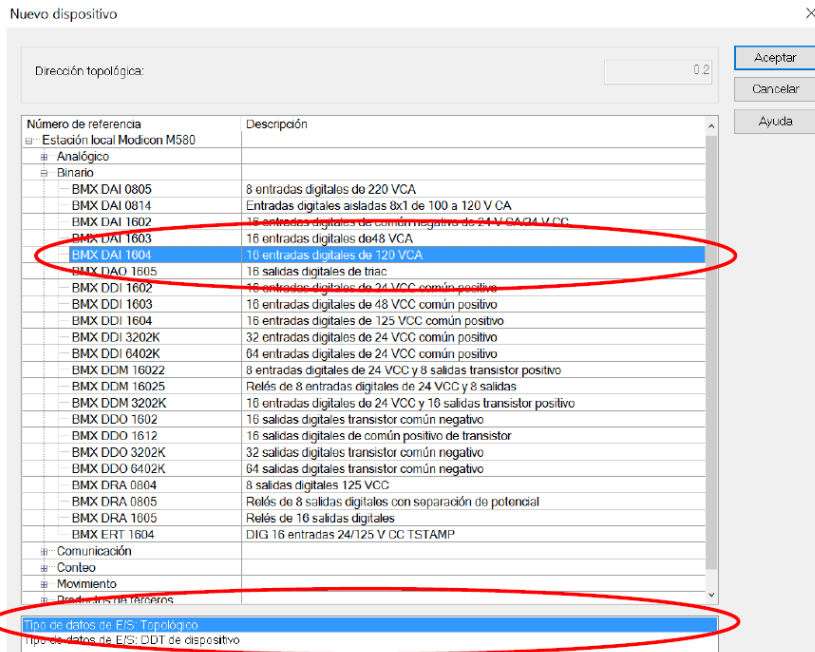


Figura II.10 Configuración entrada digital.

- Configuración de salidas digitales

De manera similar, para la configuración de las salidas digitales, en la ventana “**Bus PLC**” se selecciona el slot 3. Donde al desplegarse el menú, nuevamente se selecciona “**Binario**” y la opción “**BMX DRA 1605**” y en el menú de la parte inferior se selecciona de igual forma “**Tipo de datos de E/S: Topológico**”, tal y como se presenta en la Figura II.11.

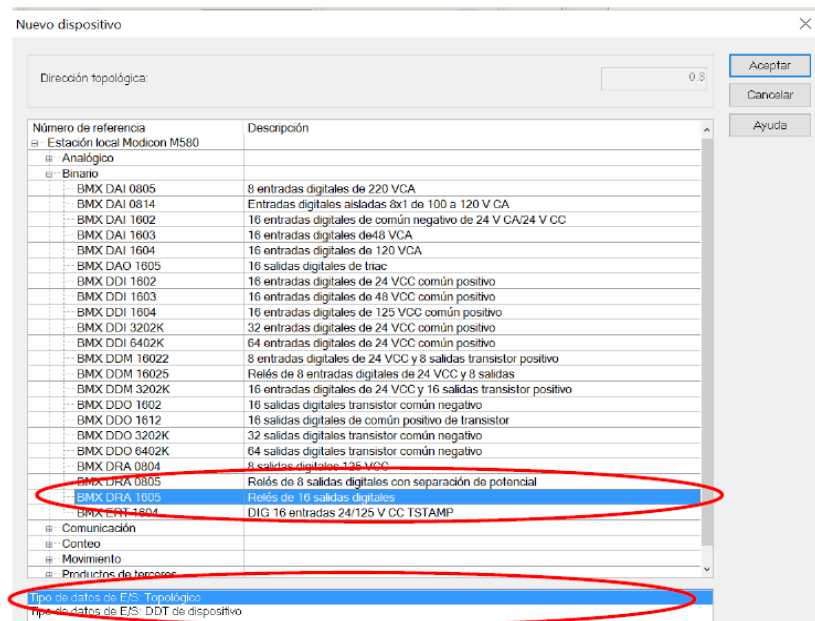


Figura II.11 Configuración salida digital.

- **Configuración de entradas y salidas analógicas**

De manera similar, para la configuración entradas y salidas analógicas, en la ventana “**Bus PLC**” se selecciona el slot 4. Donde al desplegarse el menú, se selecciona “**Analógico**” y la opción “**BMX AMM 0600**” y en el menú de la parte inferior se selecciona de igual forma “**Tipo de datos de E/S: Topológico**”, tal y como se presenta en la Figura II.12.

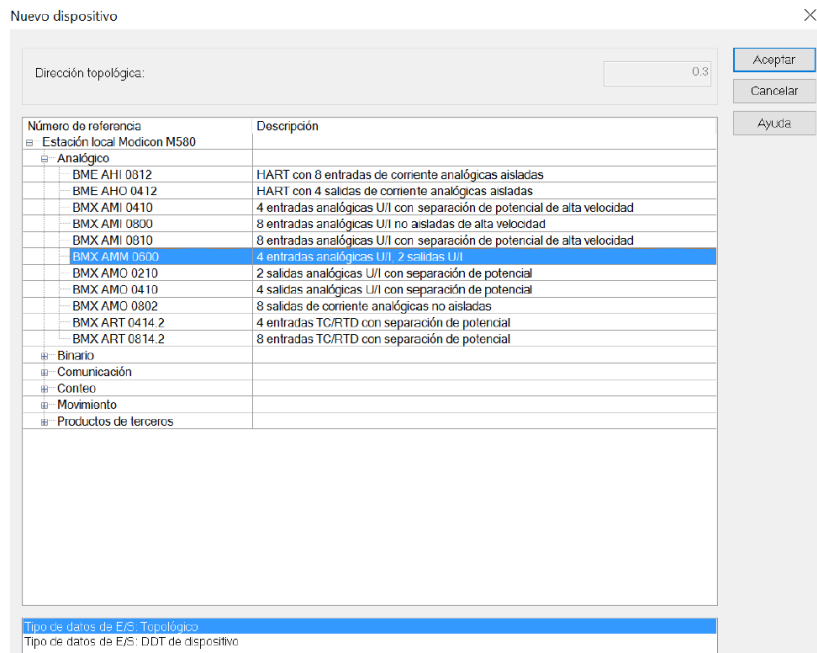


Figura II.12 Nuevo dispositivo analógico.

Una vez completada la selección de los módulos, la ventana “**Bus PLC**” debe quedar tal y como se muestra en la Figura II.13.

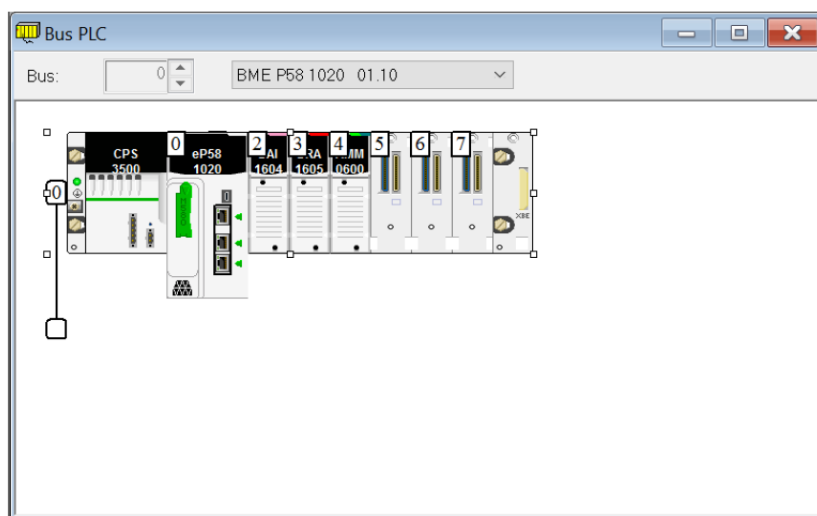


Figura II.13 Bus PLC.

ANEXO III

Codificación y decodificación de datos tipo discreto para el envío y recepción desde y hacia la interfaz gráfica

Tomando en cuenta como primer punto el envío desde el cliente Siemens S7-1200 hacia la interfaz se establece una codificación para agrupar los datos discretos que llegan desde los servidores en un dato tipo “Word” para enviarlo a la interfaz, para lo cual es necesario crear un bloque de función donde se tenga como entrada un arreglo de datos discretos y como salida un dato tipo “word”. Es necesario crear un “Function Block” como se muestra en la Figura III.1.

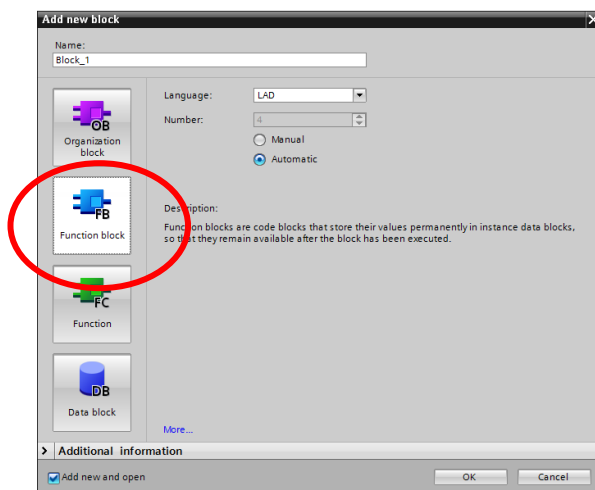


Figura III.1 Creación de Function block

Se establece el dato de entrada como un arreglo de 16 datos tipo “Bool”, la salida se define como un dato tipo “Word”, para la conversión es necesario crear datos temporales tipo “Bool” en total 16, como se muestra en la Figura III.2.

	Name	Data type	Default value	Retain	Accessible f...	Visible in ...	Setpoint
1	Input						
2	▶ datos_IN	Array[0..16] of Bool		Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
3	Output						
4	word_output	Word	16#0	Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
5	InOut						
6	<Add new>						
7	Static						
8	staticWord	Word	16#0	Set in IDB	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
9	staticWord_1	Array[0..15] of Bool		Set in IDB	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
10	Temp						
11	bit0	Bool			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
12	bit1	Bool			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
13	bit2	Bool			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
14	bit3	Bool			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Figura III.1 Establecimiento de datos de entrada, salida y temporales.

Para anexar los bits de entrada a cada bit correspondiente en los bits que conforman el dato tipo "Word" se crean datos estáticos, el primero como tipo "Word" y el segundo un arreglo de datos tipo "Bool", es necesario seleccionar la opción "Set in IDB" en ambos datos creados, en el arreglo de datos tipo "Bool" primero se debe colocar el comando AT y dar clic en el dato estático tipo Word, en la Figura III.3 se muestra el resultado final de la configuración.

	Name	Data type	Default value	Retain	Accessible f...	Visible in ...
1	Input					
2	▶ datos_IN	Array[0..16] of Bool		Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
3	Output					
4	word_output	Word	16#0	Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
5	InOut					
6	<Add new>					
7	Static					
8	staticWord	Word	16#0	Set in IDB	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
9	▶ staticWord_1	AT*staticWord* Array[0..15] of Bool		Set in IDB	<input type="checkbox"/>	<input type="checkbox"/>
10	lemp					
11	bit0	Bool				

Figura III.2 Configuración de datos anexados.

En la entrada se va a colocar un arreglo de datos tipo "Bool" correspondiente a los datos discretos de cada servidor, los cuales mediante el bloque "FieldRead" se obtiene el bit en la posición correspondiente, como se muestra en la Figura III.4 se debe colocar el primer dato del arreglo y el bit se obtiene variando la posición "INDEX".

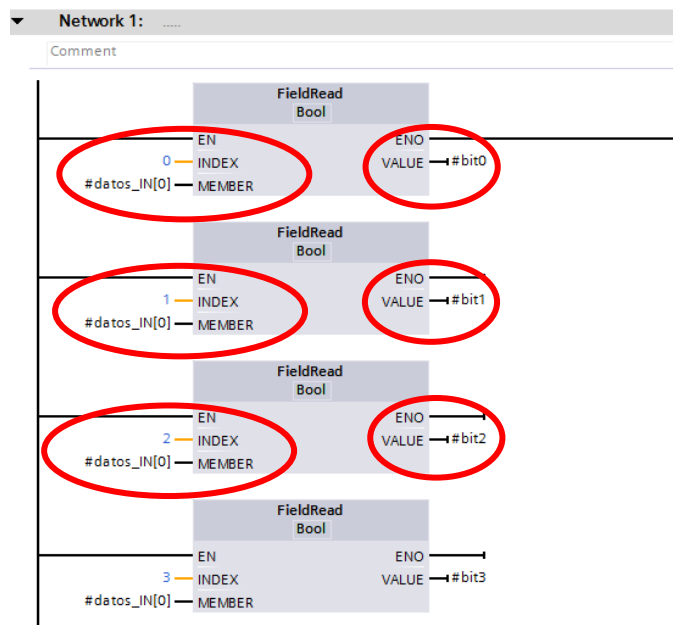


Figura III.3 Configuración de datos anexados.

Cada bit obtenido anteriormente pasa a formar parte del dato tipo “Word” que se encuentra anexado al arreglo de datos tipo “Bool”, esta configuración permite establecer un valor de uno o cero con el esquema que se muestra a continuación en la Figura III.5.



Figura III.4 Esquema para establecimiento de datos que conforman el dato “Word”

De manera más visual, se puede recurrir a la Figura III.6 donde se muestra la idea conceptual de la conversión, la cual consiste en que cada bit por separado conforma un grupo de bits formando un dato tipo “Word”.

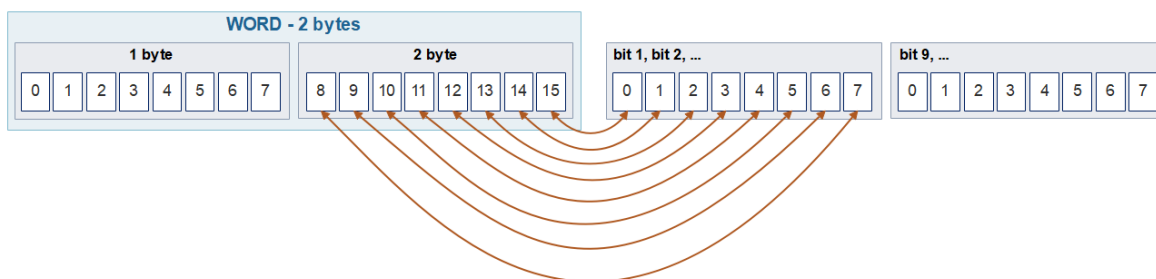


Figura III.5 Conversión de datos tipo “Bool” a tipo “Word”

Finalmente, el dato estático tipo “Word” mediante el bloque “Move” es enviado hacia el dato de salida completando la conversión como se muestra en la Figura III.7.

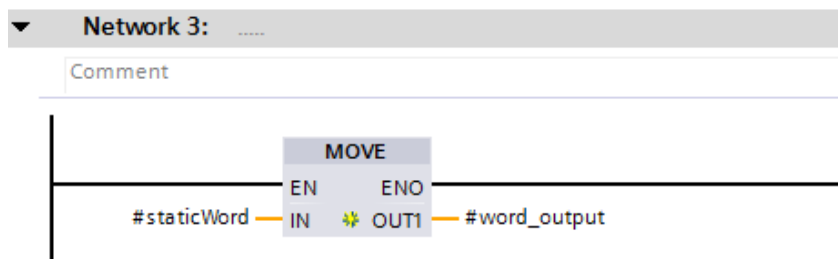


Figura III.6 Etapa final de la conversión de datos tipo “Bool” a tipo “Word”

En el desarrollo del programa se coloca el bloque de función creado y se le asigna de manera correcta los datos de entrada y salida, tomando en consideración que Siemens reagrupa los octetos de manera cruzada, como se muestra en la Figura III.8

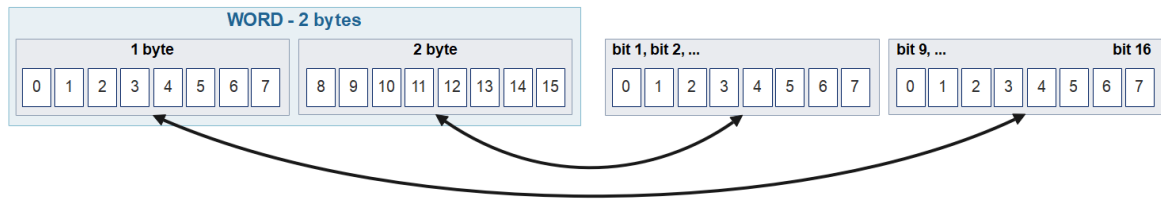


Figura III.7 Reagrupación de octetos para un dato de 32 bits

Por facilidad de interpretación se utiliza el bloque “SWAP” para reorganizar los octetos y observar que el primer bit del arreglo ocupa la primera posición de derecha a izquierda del dato tipo “Word”, en el desarrollo principal del programa se tendría un esquema como se muestra en la Figura III.9.

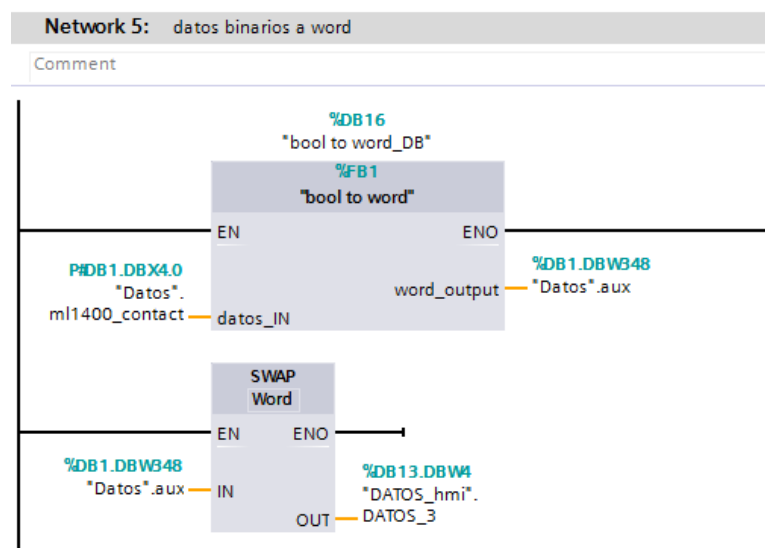


Figura III.8 Esquema para la conversión de datos tipo “Bool” a tipo “Word”

El dato ahora debe ser decodificado mediante código en la interfaz, Intouch no cuenta con una herramienta o función que permita realizar la conversión del dato tipo “Word” a datos tipo discreto, con cual, se definen variables para cada bit y se realiza una comparación “AND” como se muestra en la Figura III.10, con ello en cada “Tag” se almacena cero o uno de acuerdo a si se envió o no un bit en esa posición.

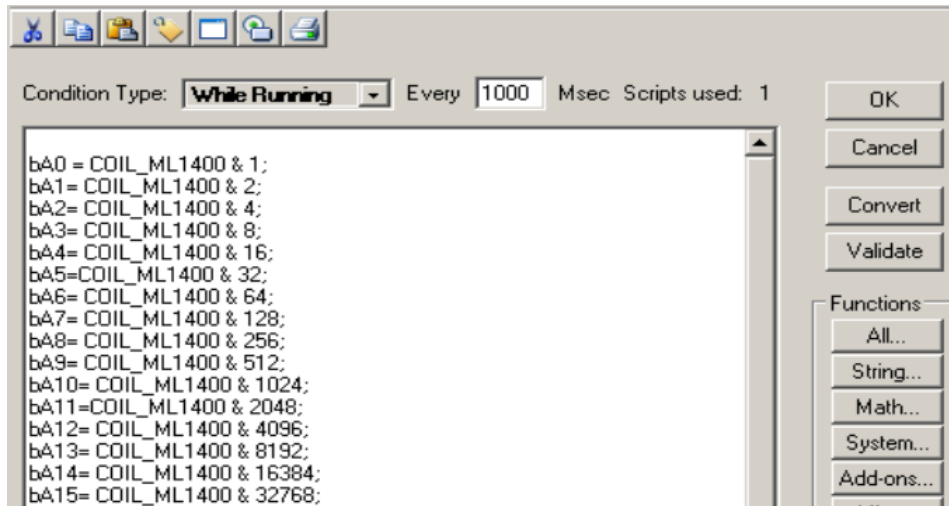


Figura III.9 Conversión de dato tipo “Word” a tipo discreto en Intouch.

En sentido contrario, tomando en cuenta un punto de partida desde la interfaz hacia el PLC Siemens S7-1200 se construye un dato tipo “Word” a partir de datos discretos proporcionados por la interfaz, los cuales son interpretados y decodificados por el PLC a través de un bloque de función.

En la Figura III.11 se muestra el algoritmo utilizado para agrupar los datos discretos de la interfaz en un dato tipo entero, el algoritmo cumple con la ecuación 1 para convertir datos binarios en datos enteros.

$$\text{Dato entero} = B_0x2^0 + B_1x2^1 + B_2x2^2 + \dots + B_nx2^n \quad (1)$$

Donde:

N: es la posición del bit

B: corresponde al bit, si es cero o es uno.

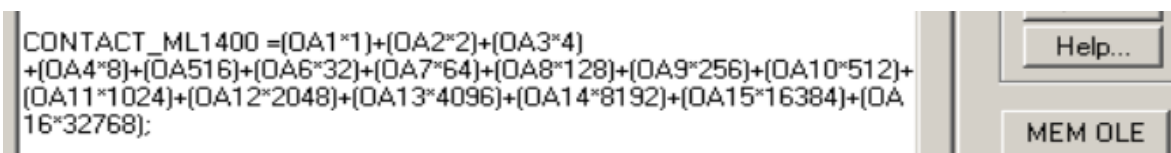


Figura III.10 Algoritmo para convertir binarios a enteros.

Este dato es enviado hacia el PLC Siemens S7-1200 como dato tipo “Word”, el cual se encarga de convertir este dato en un arreglo de datos tipo “Bool”. Como primer punto se crea un bloque de función con un dato de entrada tipo “Word”, en la salida un arreglo de

datos tipo "Bool", y finalmente variables estáticas anexadas como se mencionó anteriormente, el esquema de configuración se final se muestra en la Figura III.12

	Name	Data type	Default value	Retain	Accessible f...	Visible in ...	Setpoint
1	Input				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2	word_in	Word	16#0	Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
3	Output				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4	datos_out	Array[0..16] of Bool		Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
5	InOut				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6	<Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
7	Static				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
8	staticBool	Word	16#0	Set in IDB	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
9	VALOR AT "staticBool"	Array[0..15] of Bool		Set in IDB	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
10	Temp				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
11	<Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
12	Constant				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
13	<Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Figura III.11 Establecimiento de datos de entrada, salida, temporales y datos anexados

En la Figura III.13 se puede observar el esquema que toma un bit del grupo de bits que conforman el dato "Word" y lo establece en uno o cero a cada bit de salida de acuerdo a su posición.

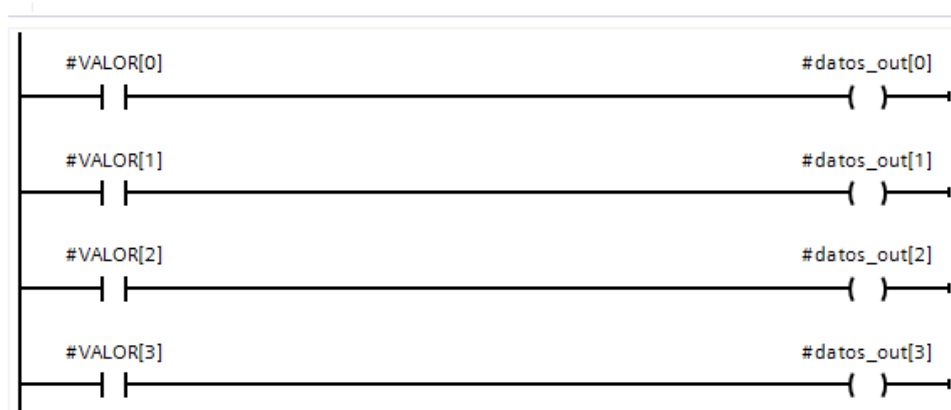


Figura III.12 Esquema para establecimiento de datos que conforman el arreglo de datos tipo "Bool"

Finalmente, los datos son enviados al arreglo de salida, como se muestra en la Figura III.14.

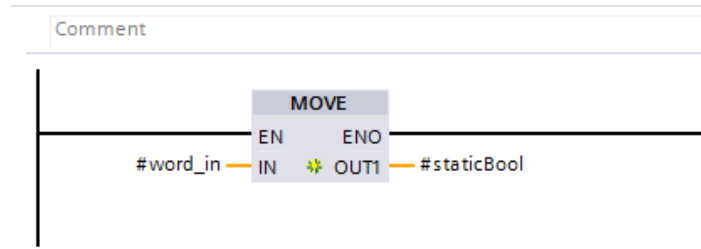


Figura III.13 Etapa final de la conversión de datos tipo “Word” a un arreglo de datos tipo “Bool”

En el programa principal también se debe considerar la ayuda del bloque “SWAP” para una mejor interpretación del usuario, se muestra en la Figura III.15.

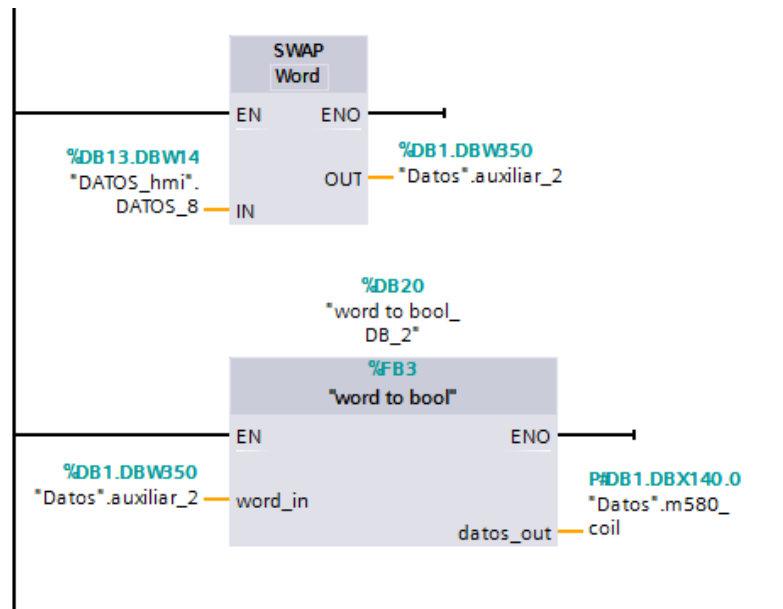


Figura III.14 Esquema para la conversión de datos tipo “Word” a un arreglo de datos tipo “Bool”

ANEXO IV

Configuración del servidor de datos

Para el servidor de datos es necesario configurar con la ayuda del software “System Management Console” el servidor DASMBTCP, como primer punto se crea un nodo tal y como se muestra en la Figura IV.1.

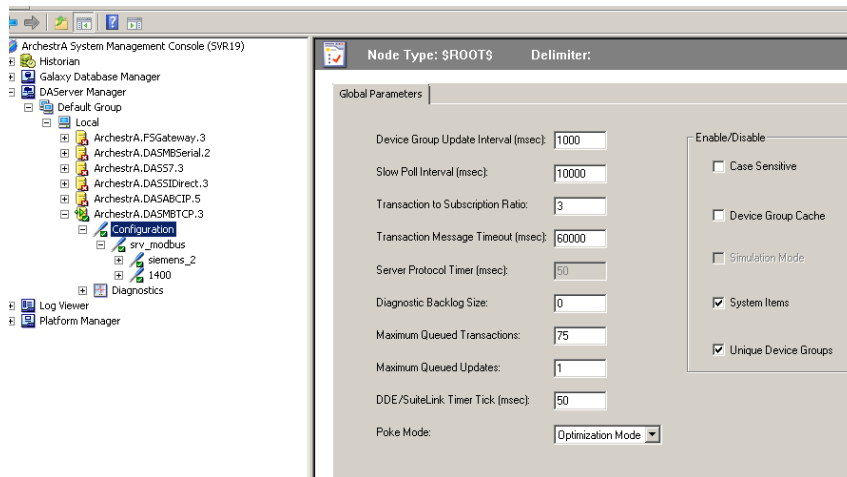


Figura IV.1 Creación de nodo en el servidor DASMBTCP.

Al dar clic derecho sobre el nodo creado, se puede abrir un puerto TCP, el cual ya se encuentra configurado con el puerto 502 por defecto como se visualiza en la Figura IV.2.

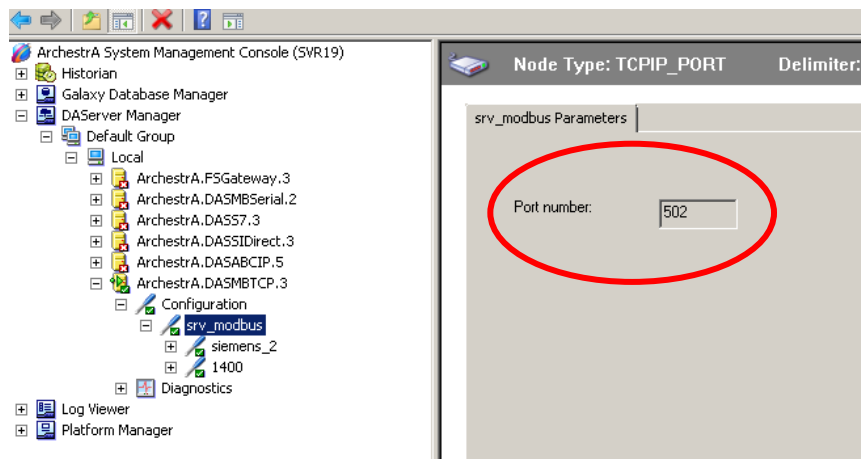


Figura IV.2 Creación del puerto TCP en el servidor DASMBTCP.

En el puerto seleccionado anteriormente, se crea un “Modbus PLC”, que permite el enlace entre el computador y el programador lógico, aquí se debe colocar la dirección IP del dispositivo con el cual se desea realizar el vínculo tal y como en la Figura IV.3.

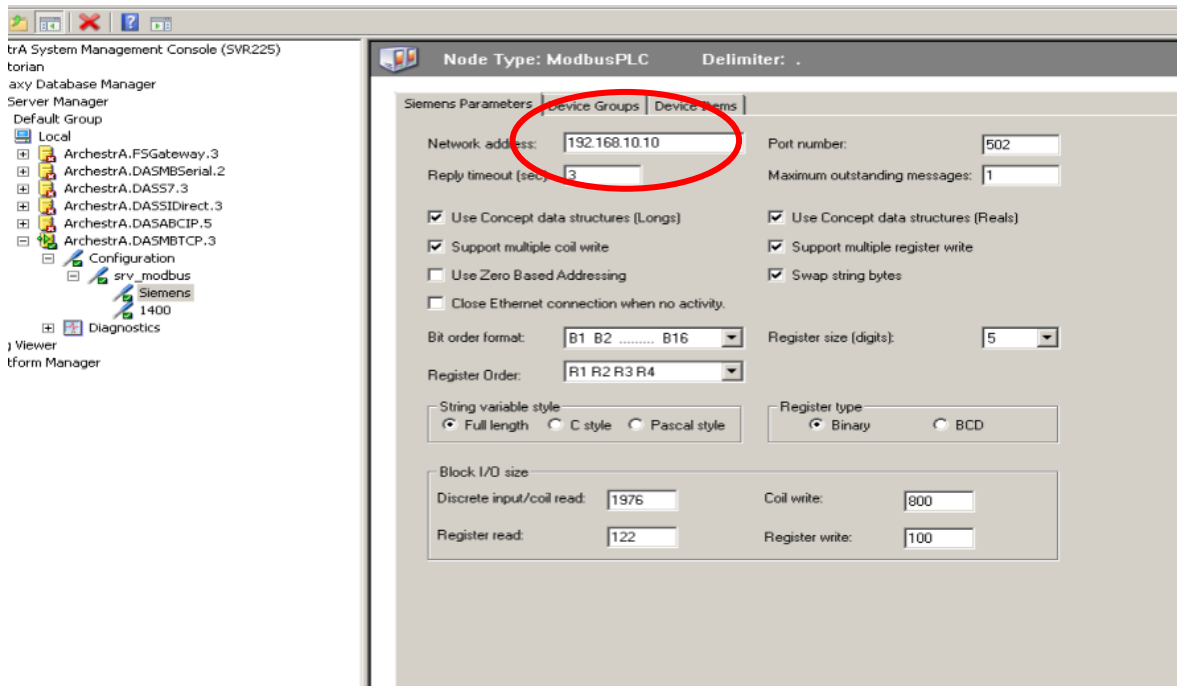


Figura IV.3 Creación del enlace “Modbus PLC” en el servidor DASMBTCP.

Adicional a ello, se debe tomar en consideración el tamaño de los registros que utiliza el PLC para realizar el direccionamiento de los registros, en este caso son 5 bits, también, para mantener un orden más detallado se puede limitar el tamaño de los registros que se pueden utilizar como en la Figura IV.4.

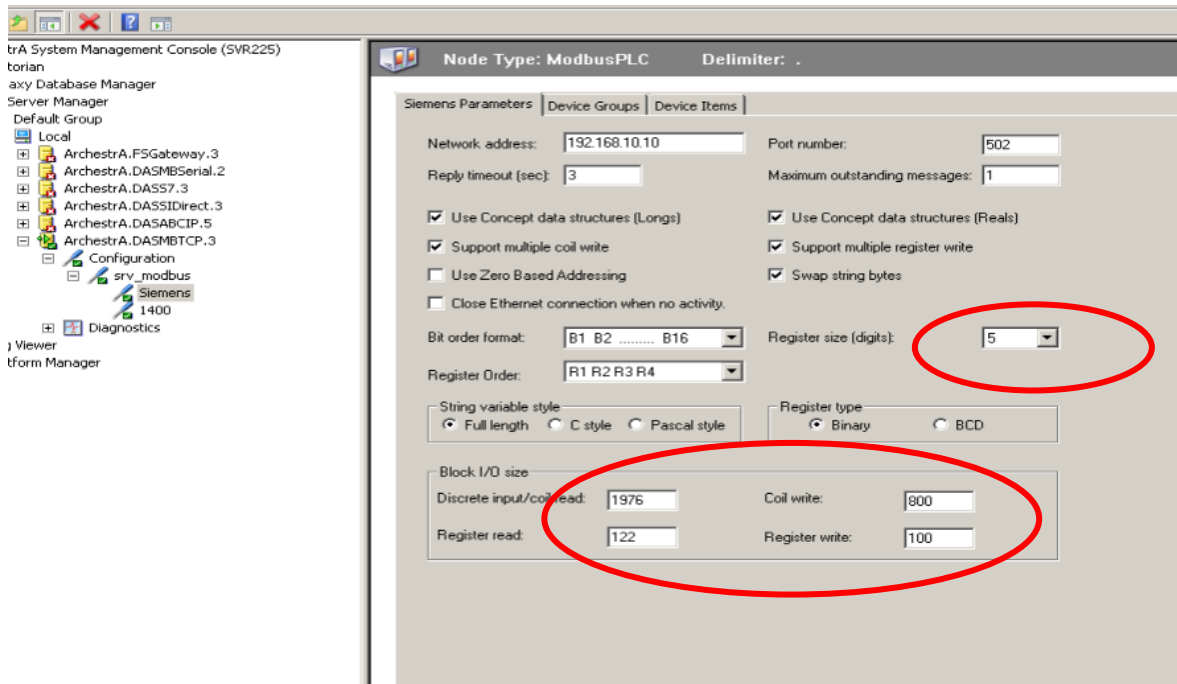


Figura IV.4 Configuración de parámetros del “Modbus PLC” en el servidor DASMBTCP.

ANEXO V

Diseño PCB para interconexión de elementos que conforman la maqueta de proceso y esquemas de conexión.

La maqueta cuenta con varios elementos que, para mayor facilidad, confort y diseño estético, es necesario que sean montadas en una placa con terminales que faciliten la conexión, también se debe encontrar señalizada de manera correcta para conocer donde se debe conectar cada elemento como se visualiza en la Figura V.1.

Cabe mencionar que las salidas de los programadores lógicos cuentan con salidas tipo relé, sin embargo, el usar dichas salidas de manera directa podría incurrir en un fallo comprometedor para el PLC, es por ello por lo que se utilizan relés externos, en este caso también se encuentran montados en la placa.

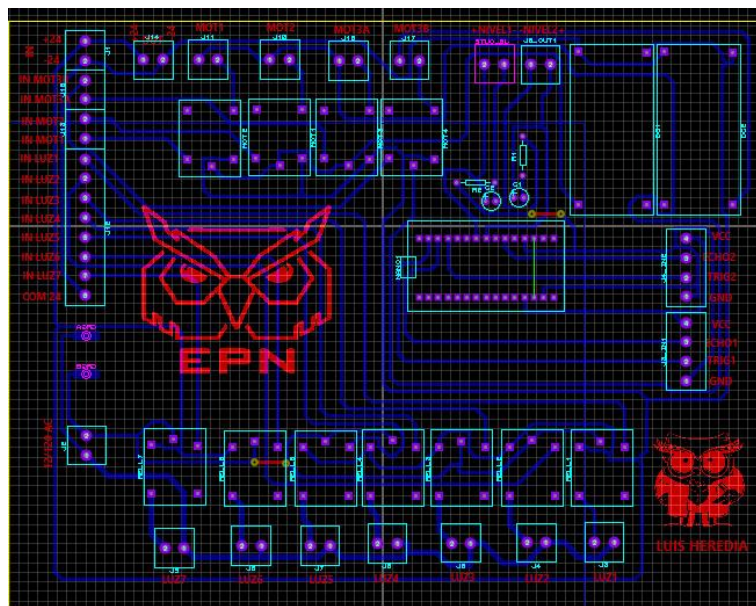


Figura V.1 Diseño final PCB de la placa

Como resultado se obtiene una placa con los relés necesarios para activar las bombas y las luces indicadoras, además cuenta con borneras debidamente etiquetadas para una fácil interpretación de la conexión como se muestra en la Figura V.2.

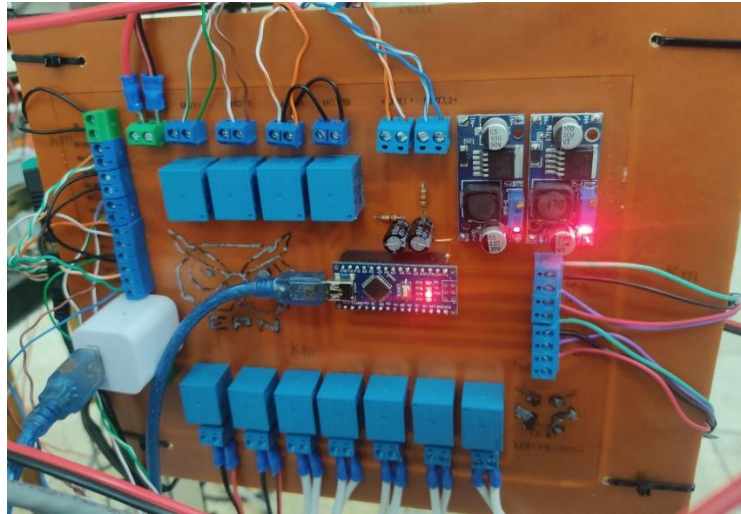


Figura V.2 Implementación de la placa final.

Esquema de conexión.

A continuación, en la Figura V.3 se muestra el esquema de conexión para el PLC Micrologix ML1400, se debe considerar la salida de la fuente de 24 voltios colocada en la placa implementada anteriormente.

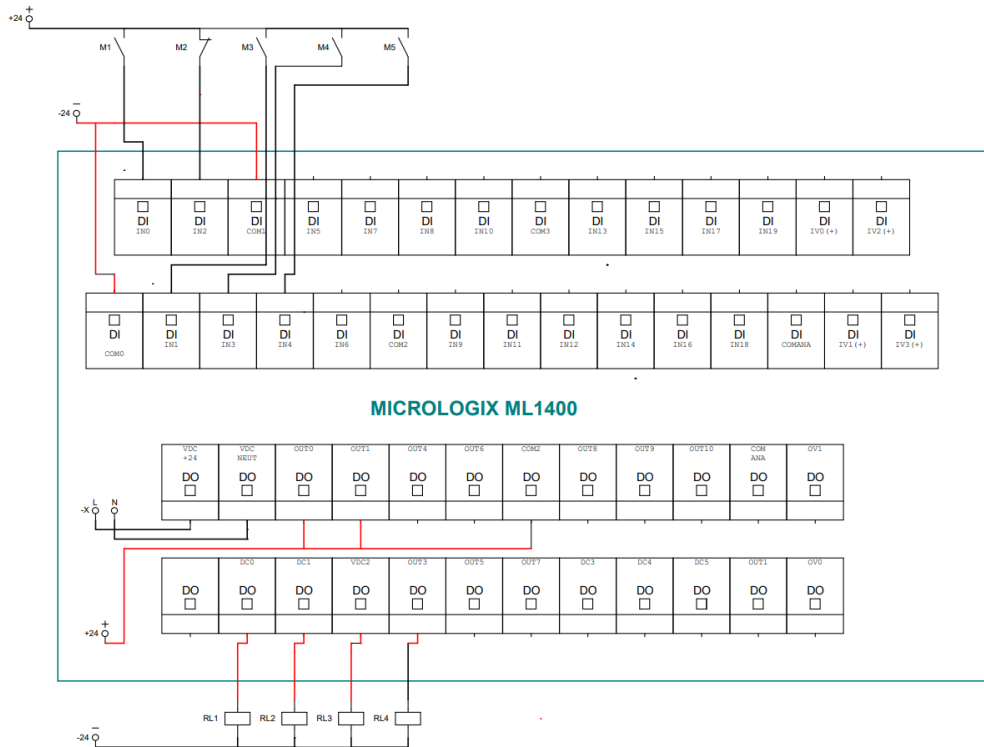


Figura V.3 Esquema de conexión para el PLC Micrologix ML1400.

En la Figura V.4, se indica la conexión para el PLC M580 realizada a través del módulo disponible en el laboratorio de Redes Industriales, con lo cual identifica las entradas y salidas, cabe mencionar que para las entradas se maneja un voltaje AC de 120 [V].

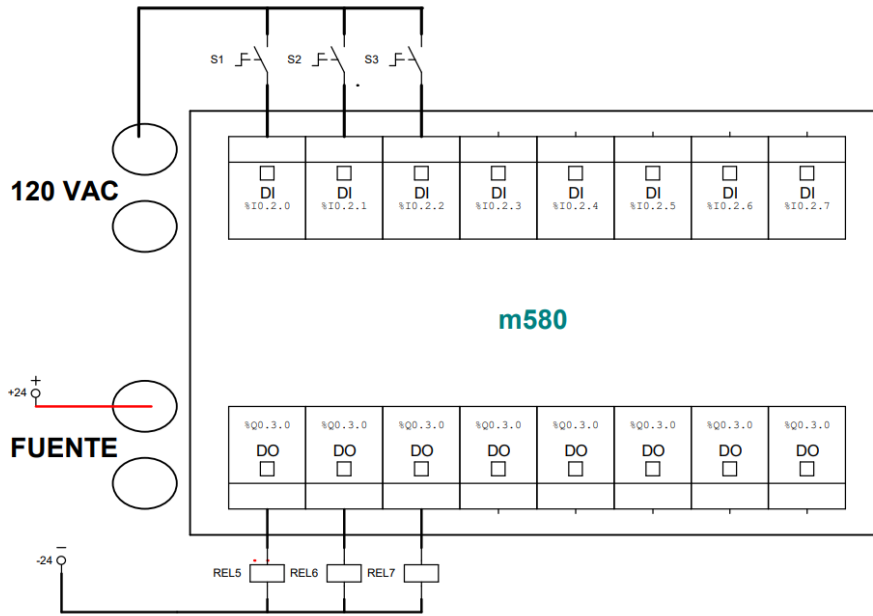


Figura V.4 Esquema de conexión para el PLC M580.

ANEXO VI

Manual de usuario

La interfaz comienza mostrando la una ventana de inicio de sesión Figura VI.1, donde se podrá acceder como a la ventana principal como “Operador”, “Ingeniero” o “Gerente”, cada uno de estos usuarios tiene un rango diferente de seguridad los cuales se detallan en la Tabla VI.1.

Tabla VI.1. Detalle de usuarios.

USUARIO	CONTRASEÑA	PERMISOS OTORGADOS
Operador	-	VISUALIZAR
Ingeniero	1234	VISUALIZAR/MODIFICAR
Gerente	1234	VISUALIZAR/MODIFICAR

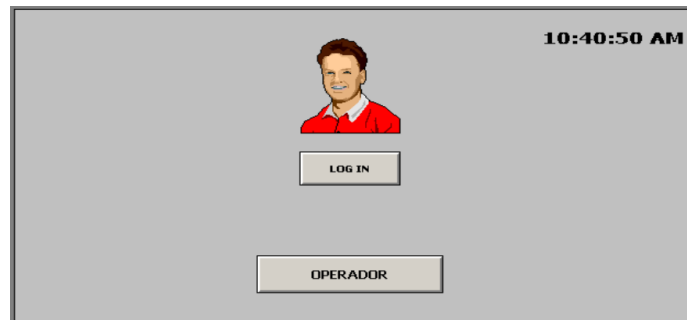


Figura VI.1. Ventana de inicio de sesión.

En la figura V.I.2 se pueden observar las opciones de navegación para cada una de las ventanas.

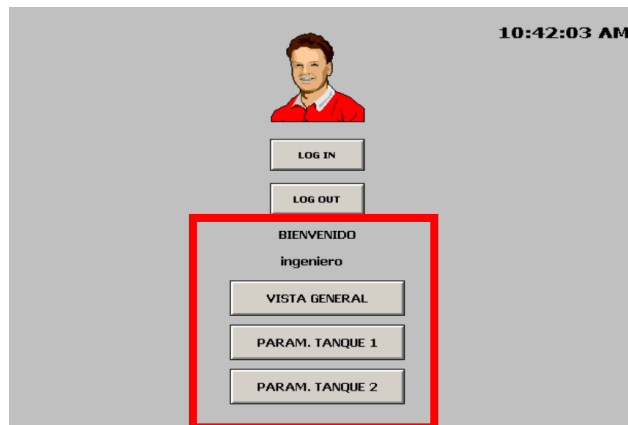


Figura VI.2. Opciones de navegación.

En la ventana “Vista General”, se pueden contar con las opciones para ingresar a las ventanas correspondientes al “Tanque 1”, “Tanque 2” y “Vista de red”, estas opciones se encuentran distribuidas como se muestra en la Figura VI.3

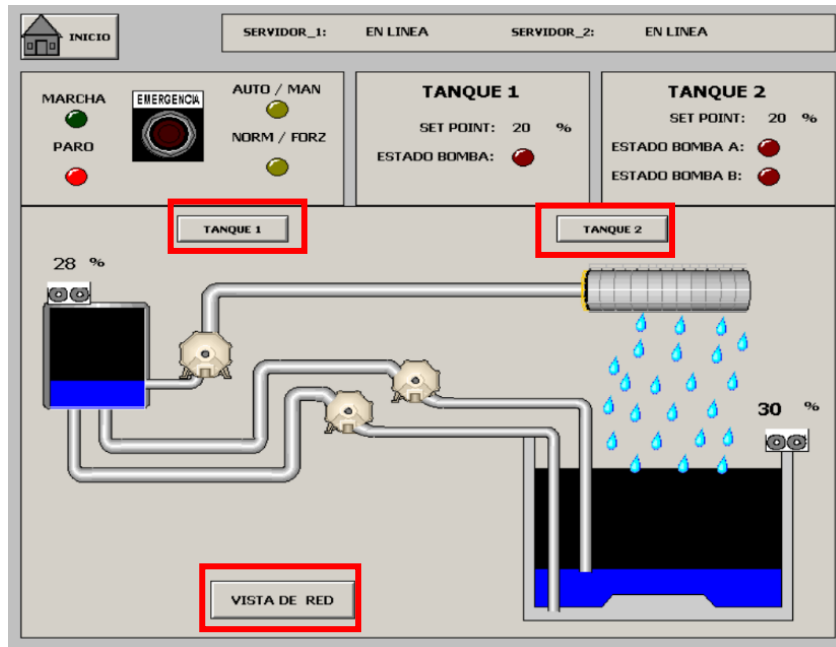


Figura VI.3. Opciones de navegación.

En la opción “Vista de Red” se puede acceder a una ventana que cuenta con la información de la distribución de la red Modbus, también tiene opciones de navegación que permiten ingresar a las ventanas de cada tanque y la ventana correspondiente al SIS como se muestra en la Figura VI.4.

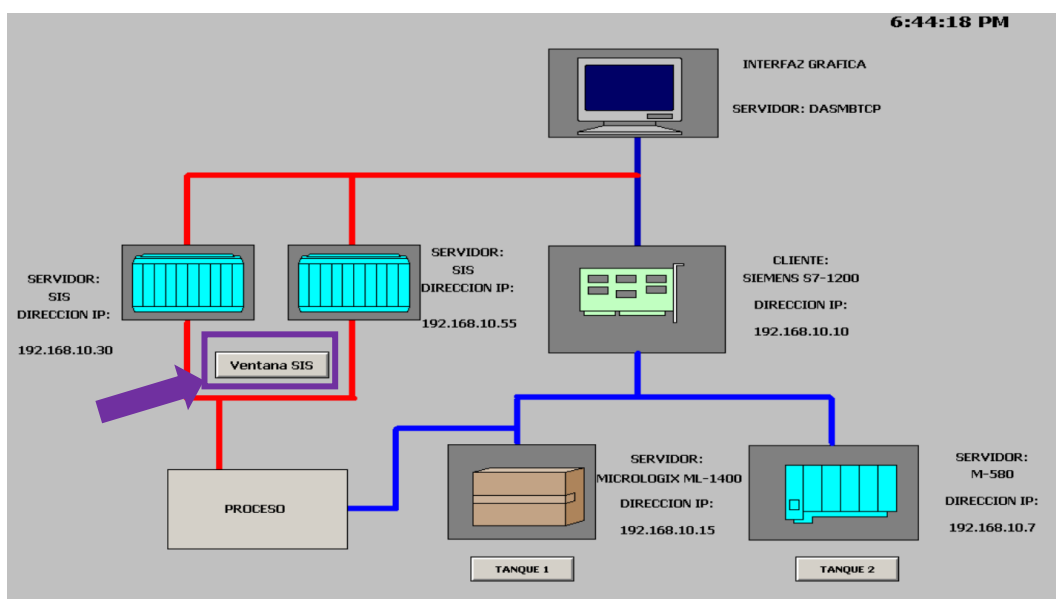


Figura VI.4. Opciones de navegación.

En la parte superior de cada ventana se puede observar el estado de conexión de cada servidor, en caso de desconectarse uno de los servidores, se muestra un mensaje parpadeante que indica que el servidor se encuentra desconectado, esto se observa en la Figura VI.5.

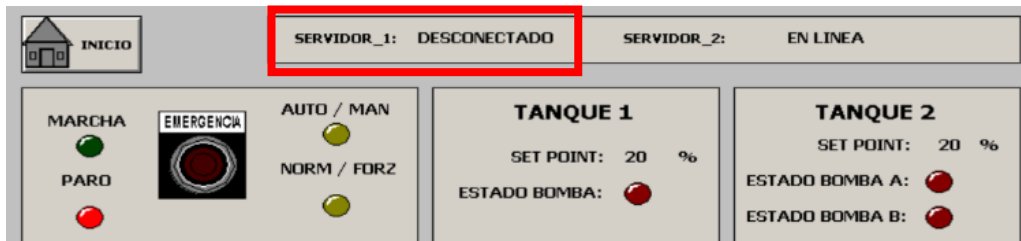


Figura VI.5. Mensaje de desconexión de servidor

Al detectar un estado de emergencia por parte del SIS, para evitar un desbordamiento de líquido, o que las bombas funcionen en vacío, se detiene el proceso, se muestra una ventana de aviso como se muestra en la Figura VI.6, con la opción “Aceptar” se puede reanudar, sin embargo, si no ha sido solucionado el problema la ventana seguirá mostrándose.



Figura VI.6. Ventana de alerta.

A continuación, se muestra el funcionamiento del proceso ponerlo en marcha, en la Figura VI.7 se muestra la ventana “Vista General” donde se puede notar que se encuentra activada la luz indicadora de marcha, así como, la luz indicadora correspondiente a la bomba 2 y los niveles en cada uno de los tanques.

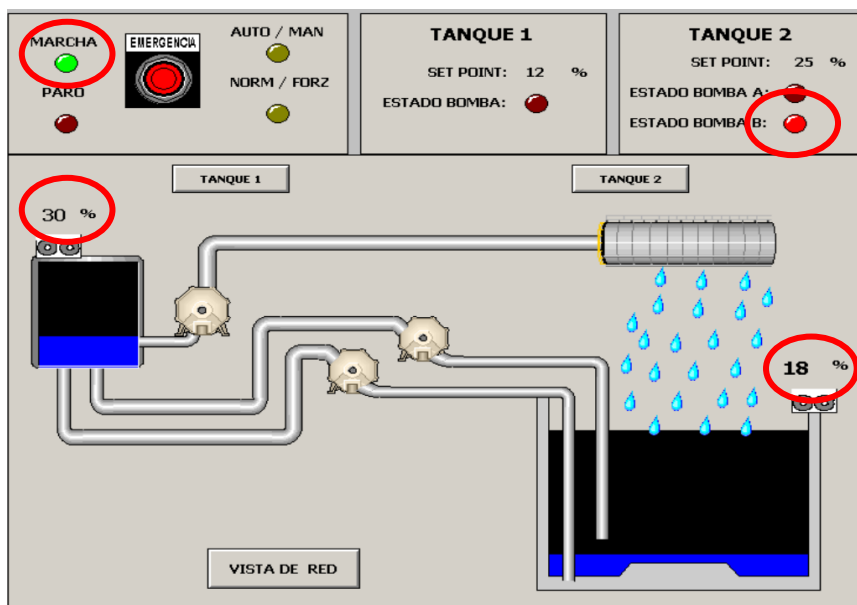


Figura VI.7. Vista General del proceso.

En la Figura VI.8 se puede observar las luces indicadoras de marcha, y la correspondiente a la bomba 2.

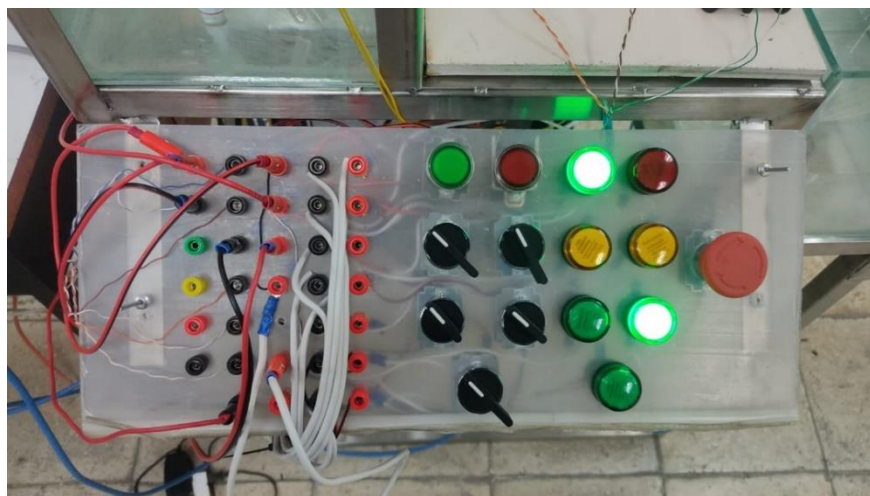


Figura VI.8. Panel principal de control.

Al transcurrir un tiempo de haber colocado en marcha al proceso, se puede observar en la Figura VI.9 el cambio de nivel en la gráfica correspondiente, notándose el control ON/OFF con histéresis aplicado.

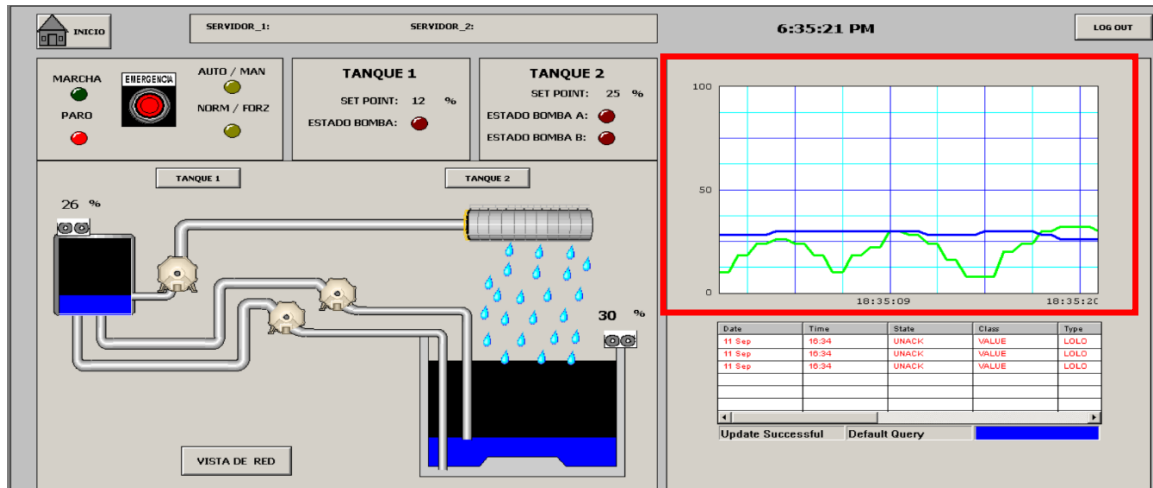


Figura VI.9. Vista General del proceso.