



**ESPE**  
UNIVERSIDAD DE LAS FUERZAS ARMADAS  
INNOVACIÓN PARA LA EXCELENCIA

**UNIDAD DE GESTIÓN DE  TECNOLOGÍAS**

**DEPARTAMENTO DE ELECTRÓNICA Y  
COMPUTACIÓN.**

**CARRERA DE ELECTRÓNICA MENCIÓN INSTRUMENTACIÓN  
& AVIÓNICA**

**TRABAJO DE TITULACIÓN, PREVIO A LA OBTENCIÓN DEL  
TÍTULO DE TECNÓLOGO EN ELECTRÓNICA MENCIÓN  
INSTRUMENTACIÓN & AVIÓNICA**

**TEMA: “IMPLEMENTACIÓN DE UNA RED MODBUS  
RTU CON PLC S7-1200 PARA EL CONTROL Y  
MONITOREO DE DOS ESTACIONES DE PROCESOS”.**

**AUTOR: MEJÍA HERRERA JORGE DAVID**

**DIRECTOR: Ing. Campaña Emma.**

**LATACUNGA**

**2016**



**DEPARTAMENTO DE ELECTRÓNICA Y COMPUTACIÓN  
CARRERA ELECTRÓNICA MENCIÓN INSTRUMENTACIÓN Y AVIÓNICA**

**CERTIFICACIÓN**

Certifico que el trabajo de Titulación, **“IMPLEMENTACIÓN DE UNA RED MODBUS RTU CON PLC S7-1200 PARA EL CONTROL Y MONITOREO DE DOS ESTACIONES DE PROCESOS”** realizado el Sr. **MEJÍA HERRERA JORGE DAVID**, ha sido revisado en su totalidad y analizado por el software anti-plagio, el mismo cumple con los requisitos teóricos, científicos, técnicos, metodológicos y legales establecidos por la Universidad de las Fuerzas Armadas ESPE, por lo tanto me permito acreditarlo y autorizar al señor **MEJÍA HERRERA JORGE DAVID** para que lo sustente públicamente.

Latacunga, 27 de Junio del 2016

Atentamente,

Ing. Campaña Emma.



**DEPARTAMENTO DE ELECTRÓNICA Y COMPUTACIÓN  
CARRERA ELECTRÓNICA MENCIÓN INSTRUMENTACIÓN Y AVIÓNICA**

**AUTORÍA DE RESPONSABILIDAD**

Yo, **MEJÍA HERRERA JORGE DAVID**, con cédula de identidad N° 0503801649, declaro que este trabajo de titulación **“IMPLEMENTACIÓN DE UNA RED MODBUS RTU CON PLC S7-1200 PARA EL CONTROL Y MONITOREO DE DOS ESTACIONES DE PROCESOS”** ha sido desarrollado considerando los métodos de investigación existentes, así como también se ha respetado los derechos intelectuales de terceros considerándose en las citas bibliográficas.

Consecuentemente declaro que este trabajo es de mi autoría, en virtud de ello me declaro responsable del contenido, veracidad y alcance de la investigación mencionada.

Latacunga, 27 de Junio 2016

---

Mejía Herrera Jorge David

C.I: 0503801649



**DEPARTAMENTO DE ELECTRÓNICA Y COMPUTACIÓN  
CARRERA ELECTRÓNICA MENCIÓN INSTRUMENTACIÓN Y AVIÓNICA**

**AUTORIZACIÓN (PUBLICACIÓN BIBLIOTECA VIRTUAL)**

Yo, **MEJÍA HERRERA JORGE DAVID**, autorizo a la Universidad de las Fuerzas Armadas ESPE publicar en la biblioteca Virtual de la institución el presente trabajo de titulación “**IMPLEMENTACIÓN DE UNA RED MODBUS RTU CON PLC S7-1200 PARA EL CONTROL Y MONITOREO DE DOS ESTACIONES DE PROCESOS**” cuyo contenido, ideas y criterios son de mi autoría y responsabilidad.

Latacunga, 27 de Junio 2016.

---

Mejía Herrera Jorge David

C.I: 0503801649

## **DEDICATORIA**

Me gustaría ofrecer este logro a mi hijo. Posiblemente en este momento no entiendas mis palabras, pero cuando seas capaz, quiero que te des cuenta de lo que significas para mí. Eres la razón de que me levante cada día esforzándome por el hoy y el mañana, eres mi principal motivación. Hijo, eres mi orgullo, libras mi mente de todas las adversidades que se presentan, y me impulsas cada día a prevalecer en la carrera de ofrecerte siempre lo mejor. Tú ocasionas en mí, a tener aspiraciones muy grandes, y sé que las lograré.

## **AGRADECIMIENTO**

Este logro conlleva una gran gratitud hacia Dios, quien con su bendición me ha permitido llegar a alcanzar una meta más en el transcurso académico. Mis padres Yolita y Saúl, por comprenderme y ayudarme en momentos malos y menos malos, son ustedes quienes me han enseñado a encarar las adversidades sin perder nunca la dignidad ni desfallecer en el intento. Me han dado todo lo que soy como persona, mis valores, mis principios, mi perseverancia y mi empeño, y todo ello con una gran cantidad de amor.

Mi hermano Diego, que siempre me ha acompañado en el transcurso de la vida. Mi esposa Carlita, y mi hijo que cada día me brindan tranquilidad y amor. Gracias por ayudarme cada día a cruzar con firmeza el camino de la superación, porque con su apoyo y aliento hoy he logrado uno de mis más grandes anhelos.

Con amor y agradecimiento infinito.

## ÍNDICE DE CONTENIDOS

CERTIFICACIÓN .....	ii
AUTORÍA DE RESPONSABILIDAD .....	iii
AUTORIZACIÓN (PUBLICACIÓN BIBLIOTECA VIRTUAL) .....	iv
DEDICATORIA .....	v
AGRADECIMIENTO .....	vi
ÍNDICE DE CONTENIDOS .....	vii
ÍNDICE DE TABLAS .....	x
ÍNDICE DE FIGURAS .....	xi
RESUMEN .....	xiii
ABSTRACT .....	xiv
<b>CAPÍTULO I.....</b>	<b>1</b>
<b>PLANTEAMIENTO DEL PROBLEMA.....</b>	<b>1</b>
1.1 ANTECEDENTES. ....	1
1.2 PLANTEAMIENTO DEL PROBLEMA. ....	1
1.3 JUSTIFICACIÓN .....	2
1.4 OBJETIVOS. ....	2
1.4.1 Objetivo general. ....	2
1.4.2 Objetivos específicos.....	2
1.5 Alcance.....	3
<b>CAPÍTULO II.....</b>	<b>4</b>
<b>MARCO TEÓRICO. ....</b>	<b>4</b>
2.1 Redes Industriales.....	4

2.2	Bus de Campo.....	4
2.2.1	Bus de Campo Propietario .....	5
2.2.2	Bus de Campo Abierto .....	5
2.3	ModBus .....	7
2.3.1	Protocolos .....	7
2.3.2	ModBus RTU.....	8
2.3.3	Trama RTU.....	9
2.3.4	Mapa de direcciones Modbus.....	10
2.4	Módulo de Comunicación CM1241 .....	11
2.5	PLC S7 1200 .....	12
2.5.1	Partes del PLC S7 1200.....	13
2.5.2	Tarjeta de señales.....	14
2.5.3	Módulo de señales .....	14
2.5.4	Módulo de comunicación.....	15
2.5.5	Almacenamiento de datos, áreas de memoria y direccionamiento ...	15
2.5.6	Acceso a los datos en las áreas de memoria de la CPU.....	16
2.5.7	Módulo de señales SM 1232 AQ.....	18
2.6	Instrucciones ModBus PLC S7 1200.....	20
2.6.1	MB_COMM_LOAD .....	20
2.6.2	MB_MASTER .....	21
2.6.3	MB_SLAVE .....	23
2.7	Micromaster 440.....	25
2.7.1	Características Principales.....	26
2.7.2	Descripción de parámetros.....	27

<b>CAPÍTULO III.....</b>	<b>30</b>
<b>IMPLEMENTACIÓN DE LA RED MODBUS PARA EL CONTROL DE DOS ESTACIONES DE PROCESOS .....</b>	<b>30</b>
3.1 Preliminares .....	30
3.2 Creación de la red Profinet.....	30
3.3 Creación de la red Modbus .....	33
3.4 Conexión de módulos de señales .....	35
3.5 Conexiones físicas de los dispositivos .....	37
3.6 Configuración y programación del maestro .....	39
3.7 Configuración y programación del esclavo 1.....	49
3.8 Configuración y programación del esclavo 2.....	53
3.9 HMI para el monitoreo de la red .....	57
3.10 Pruebas de funcionamiento.....	58
<b>CAPÍTULO IV .....</b>	<b>59</b>
<b>CONCLUSIONES Y RECOMENDACIONES.....</b>	<b>59</b>
4.1 Conclusiones.....	59
4.2 Recomendaciones.....	60
<b>GLOSARIO DE TÉRMINOS.....</b>	<b>61</b>
<b>REFERENCIA BIBLIOGRAFÍA.....</b>	<b>62</b>
<b>ANEXOS.....</b>	<b>¡Error! Marcador no definido.</b>

## ÍNDICE DE TABLAS

<b>Tabla 1</b> Trama del mensaje RTU.....	10
<b>Tabla 2</b> Estructura de los datos en Modbus .....	11
<b>Tabla 3</b> Memoria del proceso de las entradas.....	16
<b>Tabla 4</b> Memoria del proceso de salidas .....	17
<b>Tabla 5</b> Área de marcas .....	17
<b>Tabla 6</b> Parámetros del bloque MB_MASTER .....	23
<b>Tabla 7</b> Parámetros del Bloque MB_SLAVE .....	25
<b>Tabla 8.</b> Función de lectura y escritura que realiza el maestro.....	43

## ÍNDICE DE FIGURAS

<b>Figura 1.</b> Módulo CM 1241 RS485.....	12
<b>Figura 2.</b> Partes del PLC S7 1200.....	13
<b>Figura 3.</b> Signal Board.....	14
<b>Figura 4.</b> Partes de un módulo de señal.....	14
<b>Figura 5.</b> Partes de un módulo de comunicación .....	15
<b>Figura 6.</b> Módulo SM 1232 AQ.....	19
<b>Figura 7.</b> Terminales SM 1232 AQ.....	19
<b>Figura 8.</b> Bloque MB_COMM_LOAD .....	21
<b>Figura 9.</b> Bloque MB_MASTER.....	23
<b>Figura 10.</b> Bloque MB_SLAVE .....	24
<b>Figura 11.</b> Descripción de pines del Micromaster 440 .....	27
<b>Figura 12.</b> Red Profinet del Proyecto Técnico.....	31
<b>Figura 13.</b> Asignación de dirección IP al PLC_1 .....	31
<b>Figura 14.</b> Asignación de dirección IP al PLC_2 .....	32
<b>Figura 15.</b> Asignación de dirección IP al PLC_3 .....	32
<b>Figura 16.</b> Asignación de dirección IP al panel táctil .....	33
<b>Figura 17.</b> Selección del Módulo CM 1241 para el PLC_1.....	33
<b>Figura 18.</b> PLC_1 y módulo CM 1241 .....	34
<b>Figura 19.</b> PLC_2 y módulo CM 1241 .....	34
<b>Figura 20.</b> PLC_3 y módulo CM 1241 .....	35
<b>Figura 21.</b> Selección del módulo de señales SM 1232 AQ .....	36
<b>Figura 22.</b> Modulo SM 1232 AQ conectado al PLC_2.....	36
<b>Figura 23.</b> Modulo SM 1232 AQ conectado al PLC_2.....	37
<b>Figura 24.</b> Diagrama de terminales del PLC S7 1200 .....	37
<b>Figura 25.</b> Esquema de conexiones estación PCT 2 .....	38
<b>Figura 26.</b> Esquema de conexiones de la estación con el Micromaster 44039	38
<b>Figura 27.</b> Activación de bits de marca de sistema y ciclo .....	40
<b>Figura 28.</b> Bloque de datos del maestro.....	40
<b>Figura 29.</b> Bloque de sistema PLC_1.....	41
<b>Figura 30.</b> Configuración del maestro ModBus RTU .....	42
<b>Figura 31.</b> Instrucción para recibir datos del esclavo 1 .....	43

<b>Figura 32.</b> Contador para la secuencia de comunicación ModBus. ....	44
<b>Figura 33.</b> Bit que activa lectura del esclavo 1 .....	44
<b>Figura 34.</b> Bit que activa escritura del esclavo 1 .....	45
<b>Figura 35.</b> Bit que activa lectura del esclavo 2 .....	45
<b>Figura 36.</b> Bit que activa escritura del esclavo 2 .....	45
<b>Figura 37.</b> Escritura de datos al esclavo 1 .....	46
<b>Figura 38.</b> Escalamiento de valores a ser enviados al esclavo 1 .....	47
<b>Figura 39.</b> Instrucción para recibir datos del esclavo 2 .....	47
<b>Figura 40.</b> Escritura de datos al esclavo 2 .....	48
<b>Figura 41.</b> Escalamiento de valores a ser enviados al esclavo 2 .....	48
<b>Figura 42.</b> Activación de bits de marca de sistema del esclavo 1 .....	49
<b>Figura 43.</b> Bloque de datos del esclavo 1 .....	49
<b>Figura 44.</b> Dirección de la salida analógica 0 del esclavo 1 .....	50
<b>Figura 45.</b> Configuración del esclavo 1 .....	50
<b>Figura 46.</b> Asignación de dirección del esclavo 1 .....	51
<b>Figura 47.</b> Escalamiento de la entrada analógica 0 del esclavo 1 .....	51
<b>Figura 48.</b> Escalamiento para convertir a grados centígrados .....	52
<b>Figura 49.</b> Valor de temperatura enviado al maestro .....	52
<b>Figura 50.</b> Recepción de datos desde el maestro .....	53
<b>Figura 51.</b> Activación de bits de marca de sistema del esclavo 2 .....	53
<b>Figura 52.</b> Bloque de datos del esclavo 2 .....	54
<b>Figura 53.</b> Dirección de la salida analógica 0 del esclavo 2 .....	54
<b>Figura 54.</b> Configuración del esclavo 2 .....	55
<b>Figura 55.</b> Asignación de dirección del esclavo 2 .....	55
<b>Figura 56.</b> Lectura de la frecuencia de los pulsos del encoder .....	56
<b>Figura 57.</b> Escalamiento para obtener velocidad del motor .....	56
<b>Figura 58.</b> Datos recibidos por el esclavo 2 del maestro .....	57
<b>Figura 59.</b> HMI de la red ModBus .....	57

## **RESUMEN**

El presente Proyecto Técnico consiste en implementar una red Modbus RTU conformada por un maestro y dos esclavos, empleando tres PLC S7 1200 CPU 1214C AC/DC/RLY, tres módulos de comunicación CM 1241 RS 422/485, un panel táctil KTP 600 y dos estaciones de procesos, una de temperatura de flujo de aire denominada PCT2 y la otra para controlar la velocidad de un motor trifásico mediante un variador Micromaster 440 y monitorear la velocidad a través de un encoder óptico. Se creó una interface de usuario para manipular y visualizar las variables de las dos estaciones de procesos. El modo de comunicación empleado para esta red es Semiduplex con dos hilos, esto quiere decir que el maestro se comunicará solo con un esclavo a la vez. Para esto se programó una secuencia en el maestro empleando contadores, el valor de contaje es comparado para activar o desactivar bits de marcas internas. En la estación de procesos de temperatura se aumenta o disminuye la temperatura del flujo de aire al variar el voltaje que ingresa al bloque de control de la niquelina entre 0 y 5 Vdc. En la estación de procesos con un variador Micromaster 440 y un encoder, se mide la frecuencia de los pulsos mediante el contador rápido número 2 y se convierte esa frecuencia para obtener la velocidad de giro del eje, la velocidad se aumenta o disminuye ingresando a los terminales del Micromaster 440 en voltaje entre 0 y 10 Vcd.

### **Palabras Claves**

**PLC**

**ESTACIÓN DE PROCESOS**

**RED MODBUS**

**VARIADOR DE VELOCIDAD**

**TEMPERATURA**

## **ABSTRACT**

This Technical Project is to implement a Modbus RTU network shaped for a master and two slaves, using three PLC S7 1200 CPU 1214C AC / DC / RLY, three CM 1241 RS 422/485 communication modules, a touch panel KTP 600 and two process stations, one of airflow temperature called PCT2 and the other to control the speed of a three-phase motor through a Micromaster drive 440 and monitor the speed through an optical encoder. A user interface was created to manipulate and display the variables of the two stations processes. The communication mode used for this network is half-duplex two-wire, this means that the teacher will communicate only with a slave at a time. For this a sequence was programmed in the master using counters, the count value is compared to enable or disable bits of internal marks. In the station processes temperature increases or decreases the temperature of the air flow by varying the voltage entering the control block niquelina between 0 and 5 Vdc. In the station processes with a Micromaster drive 440 and an encoder, the frequency of the pulses is measured by the speed counter number 2 and that frequency is converted to obtain the rotational speed of the shaft. The speed is increased or decreased by accessing the Micromaster terminals 440 in voltage between 0 and 10 Vdc.

### **Keywords**

**PLC**

**STATION PROCESSES**

**MODBUS NETWORK**

**VARIABLE SPEED**

**TEMPERATURE**

---

Lic. Diego Granja P.

Jefe Secc. Dpto. Lenguas UGT

## CAPÍTULO I

### PLANTEAMIENTO DEL PROBLEMA.

#### 1.1 ANTECEDENTES.

La Unidad de Gestión de Tecnologías ESPE, tiene por misión formar los mejores profesionales, íntegros e innovadores, competitivos y entusiastas, a través del aprendizaje por logros aportando así, al desarrollo del país. La carrera de Electrónica Mención Instrumentación & Aviónica, posee el laboratorio de Automatización y Control de Procesos, sin embargo el avance tecnológico crece de forma rápida y se van desarrollando dispositivos, lo que hace necesario ir a la par de la misma.

Por ello es imperioso la adquisición de nuevos módulos **Modbus RTU**. De esta manera se desarrollará prácticas, destrezas y habilidades, adquiriendo significativos conocimientos, en todos y cada uno de los estudiantes que cursan esta prestigiosa carrera.

#### 1.2 PLANTEAMIENTO DEL PROBLEMA.

El laboratorio de Automatización y Control de Procesos de la Unidad de Gestión de Tecnologías de la Universidad de Fuerzas Armadas – ESPE, está equipado con una gran variedad de módulos y dispositivos de comunicación Ethernet y Profibus. Sin embargo se ha evidenciado que se necesita repotenciar la pirámide de comunicación industrial para el desarrollo de prácticas.

De esta manera los estudiantes podrán apreciar las facilidades y tecnología actualizada, del laboratorio de Automatización y Control de Procesos, la misma que ayuda a preparar personal calificado para que el desempeño sea eficiente en las empresas. Al aportar con el proceso pertinente, los estudiantes podrán aumentar destrezas y habilidades significativas.

### **1.3 JUSTIFICACIÓN**

Actualmente, la tecnología sigue en desarrollo, renovado cada vez los equipos e instrumentos en el campo de la comunicación industrial. Es necesario que la Unidad de Gestión de Tecnologías de la Universidad de Fuerzas Armadas – ESPE disponga de los equipos necesarios en sus laboratorios, para que de esta manera los estudiantes puedan incrementar los conocimientos tanto prácticos como teóricos, complementando una educación de calidad.

Tomando en consideración la disponibilidad económica, técnica, operacional u organizacional de los recursos necesarios para llevar a cabo los objetivos y metas señaladas. Mediante los módulos CM 1241-RS422-485 se realizará una comunicación Modbus RTU por medio de un maestro y dos esclavos, realizando así el control y monitoreo de dos estaciones de procesos para prácticas de redes industriales. La implementación de la red es factible porque se utilizarán dispositivos electrónicos que se pueden comprar en el mercado local y nacional.

### **1.4 OBJETIVOS.**

#### **1.4.1 Objetivo general.**

Implementar de una Red Modbus RTU con PLC S7 1200 para el control y monitoreo de dos estaciones de procesos.

#### **1.4.2 Objetivos específicos.**

- Indagar las características de los módulos de comunicación Modbus RTU.
- Configurar los PLCs S7 1200 para que se comuniquen mediante los módulos CM 1241-RS422-485.
- Realizar la programación de los PLCs S7 1200 mediante TIA Portal.
- Realizar pruebas de funcionamiento de la red industrial.

### **1.5 Alcance.**

Este proyecto está dirigido a la carrera de Electrónica Mención Instrumentación y Aviónica de la Universidad de Fuerzas Armadas-ESPE, para la asignatura de Automatización y Control de Procesos, brindando a los estudiantes un el material didáctico donde puedan desarrollar prácticas de comunicación ModBus RTU y aplicar los conocimientos adquiridos en el aula, esto permitirá obtener mayor experiencia para posteriormente desempeñarse de mejor manera en el ámbito laboral, logrando contar con profesionales altamente capacitados y competitivos que contribuyan con el desarrollo del país.

## **CAPÍTULO II**

### **MARCO TEÓRICO.**

#### **2.1 Redes Industriales**

En principio una red industrial se puede definir como una serie de estaciones (Nodos) o equipos interconectados entre sí, para tener información más común disponible en uno, varios o cada uno de los dispositivos de red. Por tanto, las REDES INDUSTRIALES son redes conformadas por equipos de control, tales como:

- PLC's Industriales.
- Controladores
- Sistemas de Control Distribuido.
- Transductores y Actuadores.
- Módulos Inteligentes.
- Interfaces de Operador.

#### **2.2 Bus de Campo**

Muchas veces el escuchar hablar de "fieldbus" (bus de campo) es relacionarlo con "una red con características propias", que permite conectar dispositivos de campo ubicados en plantas industriales, con el fin que conversen entre ellos. (Villajulca, 2010)

Primero, es necesario definir lo que es campo. El "campo" es el área de producción o fabricación; es decir, el área donde se localizan las máquinas y transmisores, junto con sus sensores (nivel, presión, temperatura, entre otros) y actuadores (válvulas, solenoides, relés, lámparas, entre otros).

FIELDBUS o bus de campo, se refiere a tecnologías de comunicación y protocolos usados en automatización y control de procesos industriales. La tarea del bus de campo es comunicar los sensores y actuadores con sistemas inteligentes como los PLC y las computadoras, de manera que la información pueda recorrer toda la planta.

El bus de campo debe cumplir estrictamente con los requerimientos de modularidad, seguridad, flexibilidad, resistencia a interferencias, facilidad de instalación, mantenimiento y programación. Además, un bus de campo debe cubrir principalmente dos aplicaciones diferentes:

- El control secuencial de las maquinarias o fábricas; donde la protección contra el peligro de explosión no es un factor crítico. Se caracteriza por tiempos de reacción cortos, elevada velocidad de transmisión y longitud de bus de hasta 300 metros.
- El control de procesos; donde la protección contra los peligros de explosión debe ser intrínsecamente segura. Es posible tener ciclos de tiempo de 100 ms para control y se puede cubrir distancias mayores a 1500 metros.

### **2.2.1 Bus de Campo Propietario**

Son aquellos sistemas que se caracterizan por su restricción de componentes a los productos de un solo fabricante, por lo tanto, no existe compatibilidad con productos de otros fabricantes. (Villajulca, 2010)

Una de las ventajas principales de estos buses de campo cerrados o propietarios es que tienen bajo requerimiento de configuración y puesta a punto, en vista de que todos los componentes se obtienen de un solo fabricante y por lo tanto están estandarizados. La desventaja es la dependencia de los productos y precios a un fabricante.

### **2.2.2 Bus de Campo Abierto**

Los buses de campo abiertos son todo lo contrario. Son sistemas que facilitan la comunicación entre dispositivos de diferentes fabricantes sin necesidad de adaptaciones adicionales. En pocas palabras, los usuarios podrán usar o desarrollar productos basados en estos buses de campo abiertos a un costo razonable y sin mucho esfuerzo. Existe una completa disponibilidad de herramientas y componentes hardware y software. (Villajulca, 2010)

Un bus de campo debe permitir por lo menos tres cosas:

- Interconectividad: los equipos de diversos fabricantes pueden ser conectados en forma segura al bus.
- Interoperabilidad: es la habilidad para la conexión de diversos elementos de diversos fabricantes.
- Intercambio: los equipos de un fabricante pueden ser reemplazados con una funcionalidad equivalente por equipos de otros fabricantes.

La interconectividad es el común denominador. Si la interoperabilidad no puede ser lograda, la misma operación del bus de campo es limitada y tal bus de campo se convierte en una opción poco útil. La última meta es la capacidad de intercambio. Esto sólo puede ser posible si las especificaciones son completas y se cuenta con un apropiado sistema de pruebas y validación de los equipos. Si se selecciona correctamente un bus de campo para una determinada aplicación se puede ofrecer al usuario muchas ventajas tangibles e intangibles.

- Reduce la complejidad del sistema de control en términos de hardware. Permite reducir la cantidad de equipos de control, tales como: PLC, DCS, hardware, etc. Esto a su vez, reduce la necesidad de instalar grandes cabinas de control para tales equipos. Como el cableado es drásticamente reducido en el campo, la cantidad de accesorios de montaje también se reduce. Todo esto implica un ahorro directo y muy significativo de costos.
- Al reducir los requerimientos de hardware, también se reducen los tiempos de instalación y requerimientos de mano de obra. La implementación física de las instalaciones es mucho más rápida. Además, las funciones de diagnóstico de fieldbus identifican cualquier error en las conexiones.
- Debido a que se reduce la complejidad del sistema de control, la etapa de diseño (Ingeniería) se desarrolla en menor tiempo y por tanto el costo

es menor. La necesidad de diseñar complicados esquemas de control y de cableado se simplifica significativamente.

- Las modificaciones futuras, actualizaciones o ampliaciones son fácilmente implementadas y a un costo menor.
- Al seleccionar un sistema reconocido, abierto y bien establecido, el equipo de campo de su planta puede ser intercambiable con equipos de otros fabricantes sin preocuparse por problemas de incompatibilidad.
- Los tiempos de parada y pérdidas de producción son reducidos porque los procedimientos de diagnóstico y corrección de fallas del sistema fieldbus permiten tomar acciones rápidamente.

En la actualidad existen muchas opciones. Algunos fabricantes han desarrollado sistemas basados en la tecnología disponible (como la interfaz RS-232 ó el protocolo MODBUS) con las limitaciones existentes. Otras compañías, sin embargo, han desarrollado nuevos y más poderosos estándares que ofrecen el manejo de mayores cantidades de información a mayor velocidad que MODBUS. Un ejemplo de ellos, es PROFIBUS.

## **2.3 ModBus**

El protocolo industrial Modbus fue desarrollado en 1979 para permitir la comunicación entre dispositivos de automatización. Originalmente implementado como un protocolo a nivel de la aplicación para transferir datos en una capa serial, el protocolo se ha expandido para incluir implementaciones a través de protocolo serial, TCP/IP y UDP (User Datagram Protocol). Hoy en día, es un protocolo común usado por innumerables dispositivos para comunicación simple, confiable y eficiente en una variedad de redes modernas. (National Instruments, 2015)

### **2.3.1 Protocolos**

Existen dos versiones principales: MODBUS RTU para la comunicación serial convencional, y MODBUS/TCP, que emplea Ethernet como medio físico de transmisión. Funciones especialmente dignas de mención:

- **MODBUS RTU/ASCII**

- Comunicación con 16 dispositivos (esclavos) por canal (hay disponibles 1 canal)
- Velocidad de transferencia de hasta 19,2 Kbps
- **MODBUS TCP/IP**
  - Comunicación con 64 dispositivos (esclavos) (empleando la función automática de comunicación)
  - Velocidad de transferencia de 10/100 Mbaud

Los modos de transmisión definen como se envían los paquetes de datos entre maestros y esclavos, el protocolo MODBUS define dos principales modos de transmisión, con diferentes representaciones numéricas de los datos y detalles del protocolo ligeramente desiguales:

- MODBUS RTU (Remote Terminal Unit). La comunicación entre dispositivos se realiza por medio de una representación binaria compacta de los datos. Esta es la opción más usada del protocolo.
- MODBUS ASCII (American Standard Code for Information Interchange). Es una representación legible del protocolo pero menos eficiente. La comunicación entre dispositivos se hace por medio de caracteres ASCII.

Ambas implementaciones del protocolo son serie. El formato RTU finaliza la trama con una suma de control de redundancia cíclica (CRC), mientras que el formato ASCII utiliza una suma de control de redundancia longitudinal (LRC).

### **2.3.2 ModBus RTU**

Cuando los controladores son configurados para comunicarse en una red Modbus usando el modo RTU (Remote Terminal Unit), cada byte de 8 bits en un mensaje contiene dos dígitos hexadecimales de 4 bits. La principal ventaja de este modo es que su mayor densidad de carácter permite mejor rendimiento que el modo ASCII, para la misma velocidad. Cada mensaje debe ser transmitido en un flujo continuo. (Cevallos, 2007)

El formato para cada byte en modo RTU es:

**Sistema de codificación:** Binario 8-bits, hexadecimal 0-9, A-F.

Dos dígitos hexadecimales contenidos

En cada campo de 8 bits del mensaje.

**Bits por byte:**

1 bit de arranque.

8 bits de datos, el menos significativo se envía primero.

1 bit para paridad Par o Impar; ningún bit para No paridad.

1 bit de paro si se usa paridad; 2 bits si no se usa paridad.

**Campo de Comprobación de error:** Comprobación Cíclica Redundante (CRC).

### 2.3.3 Trama RTU

En modo RTU, los mensajes comienzan con un intervalo silencioso de al menos 3.5 tiempos de carácter. Esto es más fácilmente implementado como un múltiplo de tiempos de carácter a la velocidad de transmisión configurada en la red (mostrado como T1-T2-T3-T4 en la tabla 1). El primer campo transmitido es entonces la dirección del dispositivo destinatario. Los caracteres a transmitir permitidos para todos los campos son 0-A, A-F hexadecimal.

Los dispositivos conectados en red monitorizan el bus de red continuamente incluso durante los intervalos silenciosos. Cuando el primer campo (el campo de dirección) es recibido, cada dispositivo lo decodifica

para enterarse si es el dispositivo direccionado. Siguiendo al último carácter transmitido, un intervalo de al menos 3.5 tiempos de carácter señala el final del mensaje. Un nuevo mensaje puede comenzar después de este intervalo. La trama completa del mensaje debe ser transmitida como un flujo continuo. Si un intervalo silencioso de más de 1.5 tiempos de carácter tiene lugar antes de completar la trama, el dispositivo receptor desecha el mensaje incompleto y asume que el próximo byte será el campo de dirección de un mensaje.

De forma similar, si un nuevo mensaje comienza antes de que transcurran 3.5 tiempos de carácter después de un mensaje previo, el dispositivo receptor lo considerará una continuación del mensaje previo. Esto dará lugar a un error, en vista de que el valor en el campo final CRC no será válido para el mensaje combinado.

**Tabla 1**

**Trama del mensaje RTU**

INICIO	DIRECCIÓN	FUNCIÓN	DATOS	CRC	FINAL
T1-T2-T3-T4	8 BITS	8 BITS	N * 8 BITS	16 BITS	T1-T2-T3-T4

**Fuente:** (Cevallos, 2007)

### 2.3.4 Mapa de direcciones Modbus

El protocolo Modbus, en su versión original, soporta 4 tipos de datos:

- **Salidas digitales (direcciones 0001-09999):** son salidas físicas discretas. Requieren un bit que puede tomar los valores 0 o 1 y permiten acceso de escritura.
- **Entradas digitales (direcciones 10001-19999):** son entradas físicas discretas. Requieren un bit que puede tomar los valores 0 o 1 y permiten acceso de escritura/lectura.
- **Entradas analógicas (direcciones tipo 30001-39999):** son entradas físicas analógicas que funcionan con registros de 16 bits y que permiten acceso de escritura.

- **Salidas analógicas (direcciones 40001-49999):** se trata de salidas físicas analógicas o registros internos del equipo. También conocidas con el nombre de holding registers. Registros de 16 bits que permiten acceso de lectura y de escritura.

**Tabla 2**

**Estructura de los datos en Modbus**

Tipo de Datos	Tipo de Acceso	Rango de Memoria
<b>Salidas Digitales</b>	Escritura	00001-09999
<b>Entradas Digitales</b>	Escritura / Lectura	10001-19999
<b>Entradas Analógicas</b>	Escritura	30001-39999
<b>Registros de memoria</b>	Escritura / Lectura	40001-49999

**Fuente:** (Cevallos, 2007)

**2.4 Módulo de Comunicación CM1241**

En los módulos de comunicaciones CM 1241 se dispone de los siguientes protocolos estándar: (Siemens, Hoja de datos 6ES7241-1CH32-0XB0, 2015)

- **ASCII:** Para la conexión a sistemas externos con protocolos de transferencia sencillos, por ejemplo protocolos con caracteres de inicio y de fin o protocolos con caracteres de control. El programa de usuario permite consultar y controlar las señales handshake de la interfaz.
- **MODBUS:** Para la comunicación según el protocolo MODBUS con formato RTU:
- **MODBUS maestro:** Conexión maestro-esclavo con SIMATIC S7 como maestro.

- **MODBUS esclavo:** Conexión maestro-esclavo con SIMATIC S7 como esclavo; no es posible el tráfico de telegramas de esclavo a esclavo.
- **Protocolo para accionamientos USS:** En particular, se admiten las instrucciones para la conexión de accionamientos conformes al protocolo USS. Los accionamientos intercambian datos en este caso a través de RS485. Con ello es posible controlar estos accionamientos, así como leer y escribir parámetros.



**Figura 1** Módulo CM 1241 RS485

**Fuente:** (Siemens, Hoja de datos 6ES7241-1CH32-0XB0, 2015)

## 2.5 PLC S7 1200

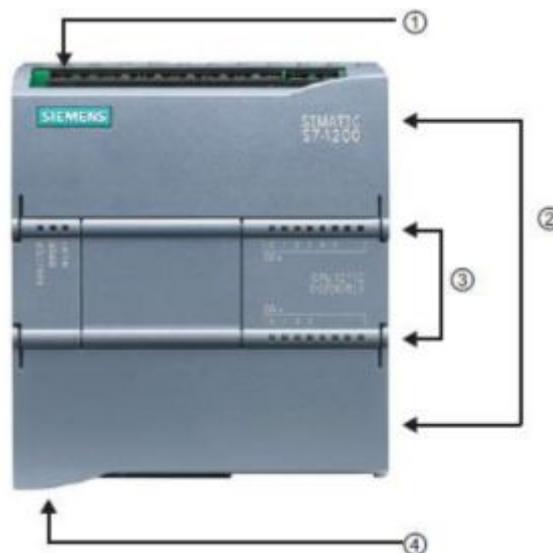
El controlador lógico programable (PLC) S7-1200 ofrece la flexibilidad y capacidad de controlar una gran variedad de dispositivos para las distintas tareas de automatización. Gracias a su diseño compacto, configuración flexible y amplio juego de instrucciones, el S7-1200 es idóneo para controlar una gran variedad de aplicaciones. (Siemens, Controlador Programable S7 1200, 2009)

La CPU incorpora un microprocesador, una fuente de alimentación integrada, así como circuitos de entrada y salida en una carcasa compacta, conformando así un potente PLC. Una vez cargado el programa en la CPU, ésta contiene la lógica necesaria para vigilar y controlar los dispositivos de la

aplicación. La CPU vigila las entradas y cambia el estado de las salidas según la lógica del programa de usuario, que puede incluir lógica booleana, instrucciones de conteo y temporización, funciones matemáticas complejas, así como comunicación con otros dispositivos inteligentes. La CPU incorpora un puerto PROFINET para la comunicación en una red PROFINET. Los módulos de comunicación están disponibles para la comunicación en redes RS485 o RS232.

### 2.5.1 Partes del PLC S7 1200

Los diferentes modelos de CPUs ofrecen una gran variedad de funciones y prestaciones que permiten crear soluciones efectivas destinadas a numerosas aplicaciones.



**Figura 2** Partes del PLC S7 1200

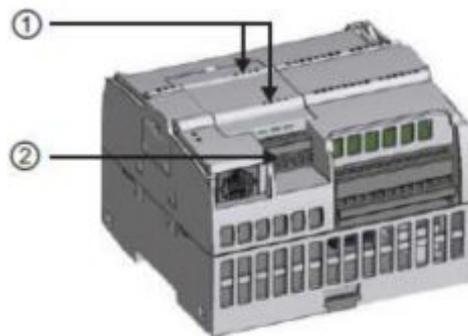
**Fuente:** (Siemens, Controlador Programable S7 1200, 2009)

- ① Conector de corriente
- ② Conectores extraíbles para el cableado de usuario (detrás de las tapas)
- ② Ranura para Memory Card (debajo de la tapa superior)
- ③ LEDs de estado para las E/S integradas
- ④ Conector PROFINET (en el lado inferior de la CPU)

### 2.5.2 Tarjeta de señales

Una Signal Board (SB) permite agregar E/S a la CPU. Es posible agregar una SB con E/S digitales o analógicas. Una SB se conecta en el frente de la CPU.

- SB con 4 E/S digitales (2 entradas DC y 2 salidas DC)
- SB con 1 entrada analógica



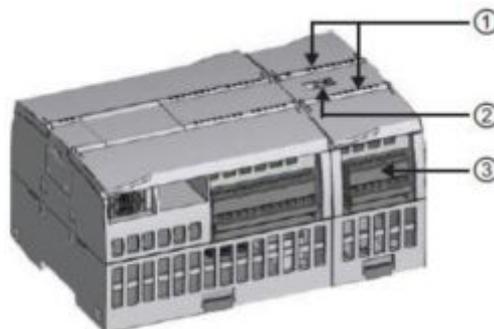
**Figura 3** Signal Board

**Fuente:** (Siemens, Controlador Programable S7 1200, 2009)

- ① LEDs de estado en la SB
- ② Conector extraíble para el cableado de usuario

### 2.5.3 Módulo de señales

Los módulos de señales se pueden utilizar para agregar funciones a la CPU, los cuales se conectan al costado derecho del mismo.



**Figura 4** Partes de un módulo de señal

**Fuente:** (Siemens, Controlador Programable S7 1200, 2009)

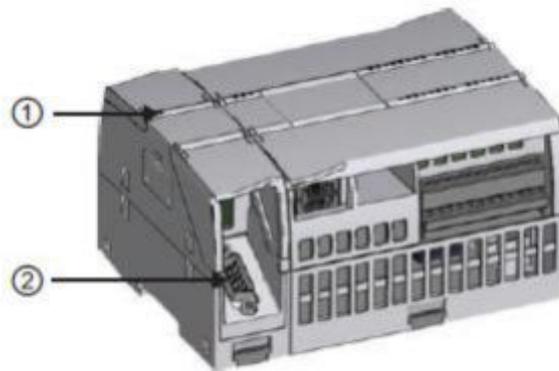
- ① LEDs de estado para las E/S del módulo de señales
- ② Conector de bus

- ③ Conector extraíble para el cableado de usuario

#### 2.5.4 Módulo de comunicación

La gama S7-1200 provee módulos de comunicación (CMs) que ofrecen funciones adicionales para el sistema. Hay dos módulos de comunicación, a saber: RS232 y RS485.

- La CPU soporta como máximo 3 módulos de comunicación
- Todo CM se conecta en lado izquierdo de la CPU (o en lado izquierdo de otro CM)



**Figura 5** Partes de un módulo de comunicación

**Fuente:** (Siemens, Controlador Programable S7 1200, 2009)

- ① LEDs de estado del módulo de comunicación
- ② Conector de comunicación

#### 2.5.5 Almacenamiento de datos, áreas de memoria y direccionamiento

La CPU ofrece varias opciones para almacenar datos durante la ejecución del programa de usuario:

- **Memoria global:** La CPU ofrece distintas áreas de memoria, incluyendo entradas (I), salidas (Q) y marcas (M). Todos los bloques lógicos pueden acceder sin restricción alguna a esta memoria.
- **Bloque de datos (DB):** Es posible incluir DBs en el programa de usuario para almacenar los datos de los bloques lógicos. Los datos almacenados se conservan cuando finaliza la ejecución del bloque

lógico asociado. Un DB "global" almacena datos que pueden ser utilizados por todos los bloques lógicos, mientras que un DB instancia almacena datos para un bloque de función (FB) específico y está estructurado según los parámetros del FB.

- **Memoria temporal:** Cada vez que se llama un bloque lógico, el sistema operativo de la CPU asigna la memoria temporal o local (L) que debe utilizarse durante la ejecución del bloque. Cuando finaliza la ejecución del bloque lógico, la CPU reasigna la memoria local para la ejecución de otros bloques lógicos.

A los datos de la mayoría de las áreas de memoria (I, Q, M, DB y L) se puede acceder como bytes, palabras o palabras dobles utilizando el formato "dirección de byte". Para acceder a un byte, una palabra o una palabra doble de datos en la memoria, la dirección debe especificarse de forma similar a la dirección de un bit. Esto incluye un identificador de área, el tamaño de los datos y la dirección de byte inicial del valor de byte, palabra o palabra doble.

Los designadores de tamaño son B (byte), W (palabra) y D (palabra doble), p. ej. IB0, MW20 ó QD8. Las direcciones tales como I0.3 y Q1.7 acceden a la memoria imagen de proceso. Para acceder a la entrada o salida física es preciso añadir ":P" a la dirección (p. ej. I0.3:P, Q1.7:P o "Stop:P"). (Siemens, Controlador Programable S7 1200, 2009)

### 2.5.6 Acceso a los datos en las áreas de memoria de la CPU

- **I (memoria imagen de proceso de las entradas):** La CPU consulta las entradas de periferia (físicas) inmediatamente antes de ejecutar el OB de ciclo en cada ciclo y escribe estos valores en la memoria imagen de proceso de las entradas. A la memoria imagen de proceso de las entradas se puede acceder en formato de bit, byte, palabra o palabra doble. Aunque se permiten accesos de lectura y escritura, generalmente sólo se leen las entradas de la memoria imagen de proceso. (Siemens, Controlador Programable S7 1200, 2009)

### Tabla 3

### Memoria del proceso de las entradas

Bit	M[dirección de byte][dirección de bit]	IO.1
Byte, palabra, palabra doble	M[tamaño][dirección de byte inicial]	IB4, IW5, ID12

**Fuente:** (Siemens, Controlador Programable S7 1200, 2009)

- **Q (memoria imagen de proceso de las salidas):** La CPU copia los valores almacenados en la imagen de proceso de las salidas en las salidas físicas. Se puede acceder en formato de bit, byte, palabra o palabra doble. De igual manera se permiten accesos de lectura y escritura a la memoria imagen de proceso de las salidas.

**Tabla 4**

### Memoria del proceso de salidas

Bit	Q[dirección de byte][dirección de bit]	Q1.1
Byte, palabra, palabra doble	M[tamaño][dirección de byte inicial]	QB5, QW10, QD40

**Fuente:** (Siemens, Controlador Programable S7 1200, 2009)

- **M (área de marcas):** El área de marcas (memoria M) puede utilizarse para relés de control y datos para almacenar el estado intermedio de una operación u otra información de control. Al área de marcas se puede acceder en formato de bit, byte, palabra o palabra doble. Se permiten accesos de lectura y escritura al área de marcas. (Siemens, Controlador Programable S7 1200, 2009)

**Tabla 5**

### Área de marcas

Bit	M[dirección de byte][dirección de bit]	M26.7
Byte, palabra, palabra	M[tamaño][dirección de	MB20, MW30, MD50

doble	byte inicial]	
-------	---------------	--

**Fuente:** (Siemens, Controlador Programable S7 1200, 2009)

- **Temp (memoria temporal):** La CPU asigna la memoria temporal según sea necesario. Asigna la memoria temporal al bloque lógico cuando éste se inicia (en caso de un OB) o se llama (en caso de una FC o un FB). La asignación de la memoria temporal a un bloque lógico puede reutilizar las mismas posiciones de memoria temporal usadas anteriormente por un OB, FC o FB diferente.

La CPU no inicializa la memoria temporal durante la asignación. Por lo que esta memoria puede contener un valor cualquiera. La memoria temporal es similar al área de marcas, con una excepción importante: el área de marcas tiene un alcance "global", en tanto que la memoria temporal tiene un alcance "local". (Siemens, Controlador Programable S7 1200, 2009)

- **DB (bloque de datos):** Los bloques de datos se utilizan para almacenar diferentes tipos de datos, incluyendo el estado intermedio de una operación u otros parámetros de control de FBs, así como estructuras de datos requeridas para numerosas instrucciones, por ejemplo. temporizadores y contadores. Es posible determinar que un bloque de datos sea de lectura/escritura o de sólo lectura. A los bloques de datos se puede acceder en formato de bit, byte, palabra o palabra doble. A los bloques de datos que se pueden leer y escribir se permiten accesos de lectura y escritura. A los bloques de datos de sólo lectura se permiten sólo los accesos de lectura. (Siemens, Controlador Programable S7 1200, 2009)

### 2.5.7 Módulo de señales SM 1232 AQ

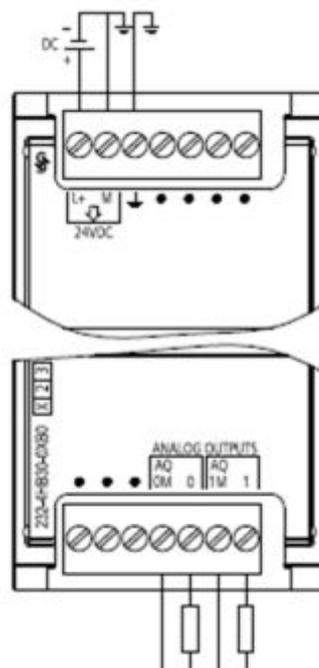
Para generar estas señales se usa un módulo SM 1232 AQ 2x14 bits. Este módulo dispone de dos salidas analógicas con una resolución de 14 bits y que entrega señales de 0-20 mA o +/- 10 V.



**Figura 6** Módulo SM 1232 AQ

**Fuente:** (Siemens, Controlador Programable S7 1200, 2009)

**SM 1232 AQ 2 x 14 bit**



**Figura 7** Terminales SM 1232 AQ

**Fuente:** (Siemens, Controlador Programable S7 1200, 2009)

## 2.6 Instrucciones ModBus PLC S7 1200

### 2.6.1 MB\_COMM\_LOAD

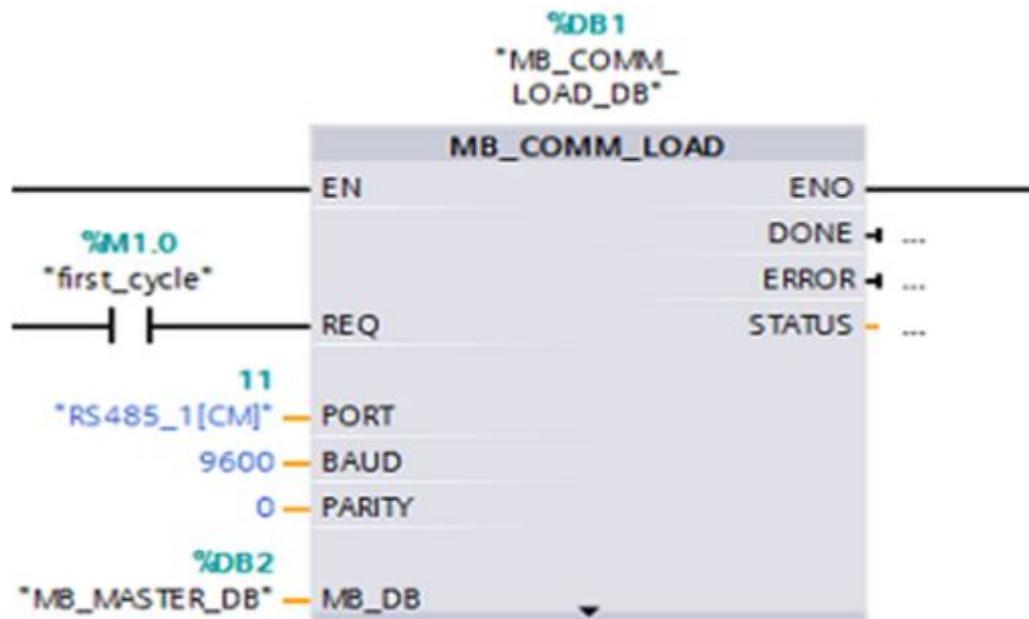
La instrucción MB\_COMM\_LOAD configura un puerto del módulo de comunicación punto a punto (PtP) CM 1241 RS485 o CM 1241 RS232 para la comunicación vía el protocolo Modbus RTU. (Siemens, Controlador Programable S7 1200, 2009)

El bloque MB\_COMM\_LOAD se usa para seleccionar el módulo de comunicación, y parametrizar el enlace con las características del maestro o el esclavo. El bloque MB\_COMM\_LOAD se debe llamar en el primer ciclo de programa (mediante la activación de la marca de sistema M1.0 en los ajustes de hardware o mediante una llamada en el OB 100 de arranque). Después de insertar el módulo de comunicación en la configuración de hardware se puede seleccionar el nombre simbólico del módulo de comunicación en el parámetro PORT. Los parámetros de comunicación BAUD (velocidad de transmisión) y PARITY (paridad) deben ser idénticos para todos los nodos.

La configuración del puerto de la interfaz RS485 en la vista de equipo del STEP 7 (TIA Portal) es irrelevante en este caso. El bloque de datos de instancia del bloque maestro o del bloque esclavo se transfiere en el parámetro MB\_DB y, por consiguiente, define el módulo de comunicación (parámetro PORT) como maestro o esclavo MODBUS. MB\_COMM\_LOAD se ejecuta para configurar un puerto para el protocolo Modbus RTU. Una vez configurado el puerto, la comunicación se realiza en el Modbus ejecutando las instrucciones MB\_SLAVE o MB\_MASTER.

MB\_COMM\_LOAD se debería llamar una vez para inicializar el puerto. Sólo es necesario volver a llamar MB\_COMM\_LOAD si debe cambiar uno de los parámetros de comunicación. Es posible llamar MB\_COMM\_LOAD desde un OB de arranque y ejecutar esta instrucción una vez, o bien utilizar la marca de sistema del primer ciclo con el fin de iniciar la llamada para ejecutarla una vez.

Es preciso utilizar una instancia de MB\_COMM\_LOAD para configurar cada uno de los puertos de todo módulo de comunicación utilizado para la comunicación Modbus. Un bloque de datos instancia MB\_COMM\_LOAD unívoco se debe asignar a cada puerto utilizado. La CPU S7-1200 está limitada a 3 módulos de comunicación.



**Figura 8** Bloque MB\_COMM\_LOAD

**Fuente:** (TIA Portal, 2013)

### 2.6.2 MB\_MASTER

La instrucción MB\_MASTER permite al programa comunicarse como maestro Modbus utilizando un puerto del módulo de comunicación punto a punto (PtP) CM 1241 RS485 o CM 1241 RS232. Es posible acceder a los datos de uno o más esclavos Modbus. (Siemens, Controlador Programable S7 1200, 2009)

Un bloque de datos instancia se asigna cuando la instrucción MB\_MASTER se inserta en el programa. El nombre de este bloque de datos instancia MB\_MASTER se utiliza cuando se especifica el parámetro MB\_DB de la instrucción MB\_COMM\_LOAD.

## Reglas de comunicación del maestro Modbus

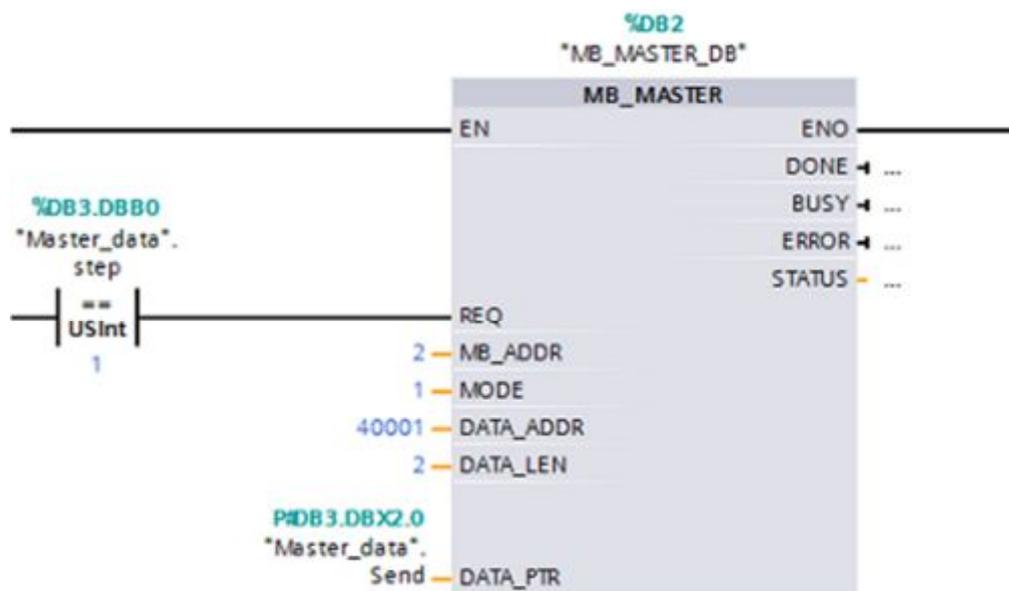
- MB\_COMM\_LOAD debe ejecutarse para configurar un puerto antes de que la instrucción MB\_MASTER pueda comunicarse con ese puerto.
- Si un puerto debe utilizarse para iniciar peticiones de maestro Modbus, MB\_SLAVE no podrá utilizar este puerto. Una o más instancias de ejecución de MB\_MASTER pueden utilizarse en este puerto.
- Las instrucciones Modbus no utilizan eventos de alarma de comunicación para controlar el proceso de comunicación. El programa debe consultar la instrucción MB\_MASTER para transmitir y recibir condiciones completas.
- Si el programa opera un maestro Modbus y utiliza MB\_MASTER para enviar una petición a un esclavo, MB\_MASTER se deberá seguir ejecutando hasta que se devuelva la respuesta del esclavo.
- Todas las ejecuciones de MB\_MASTER para un determinado puerto deben llamarse desde un mismo OB (o clase de prioridad de OB).

### Parámetro REQ

Si el valor de REQ es FALSE (falso), no hay ninguna petición. Cuando el valor de REQ es TRUE (verdadero), hay una petición de transmitir datos a uno o varios esclavos Modbus. Esta entrada debe recibir suministro de un contacto disparado por un flanco ascendente en la primera llamada de la ejecución de MB\_MASTER. El impulso disparado por flancos llama una vez la petición de transmisión. Todas las entradas se capturan y no se modifican durante una petición y respuesta disparadas por esta entrada.

MB\_MASTER inicia internamente una máquina de estados para asegurarse de que ninguna otra MB\_MASTER pueda lanzar una petición hasta que no haya finalizado esta petición. Además, si la misma instancia de la llamada del FB MB\_MASTER se ejecuta nuevamente con la entrada REQ = TRUE (verdadero) antes de que se finalice la petición, no se realizarán

más transmisiones. No obstante, en cuanto finalice la petición, se lanzará otra petición si MB\_MASTER se ejecuta con la entrada REQ puesta a TRUE.



**Figura 9** Bloque MB\_MASTER

**Fuente:** (TIA Portal, 2013)

**Tabla 6**

Parámetros del bloque MB\_MASTER

Parámetro	Descripción
REQ	Habilita la comunicación.
MB_ADDR	Dirección estación MODBUS-RTU.
MODE	Seleccionar la dirección de la transmisión ("0" = lectura, "1" = escritura o funciones de diagnóstico adicionales).
DATA_ADDR	Especifica la dirección de inicio MODBUS.
DATA_LEN	Especifica la longitud de datos MODBUS.
DATA_PTR	Define las zonas de datos de envío y recepción locales del maestro. El parámetro DATA_PTR debe hacer referencia a un bloque de datos global creado con la opción "Sólo acceso simbólico" deshabilitada.

**Fuente:** (TIA Portal, 2013)

### 2.6.3 MB\_SLAVE

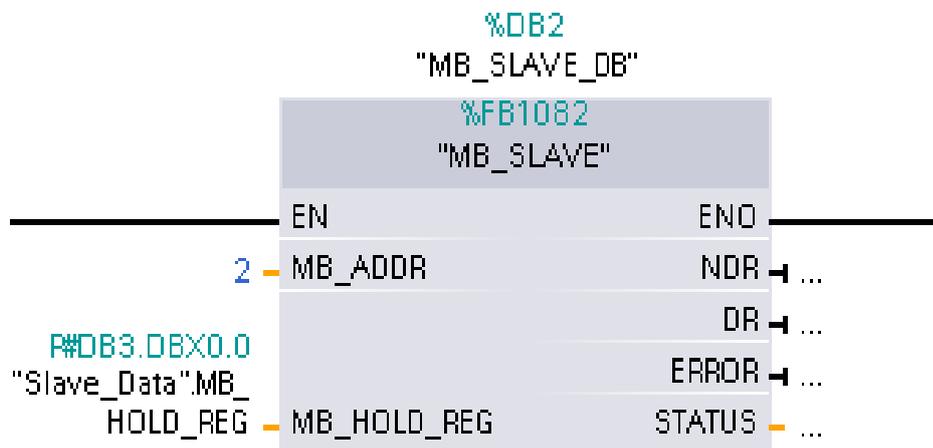
La instrucción MB\_SLAVE permite al programa comunicarse como esclavo Modbus utilizando un puerto del módulo de comunicación punto a

punto (PtP) CM 1241 RS485 o CM 1241 RS232. Un maestro Modbus RTU puede lanzar una petición y el programa responde ejecutando la instrucción MB\_SLAVE. (Siemens, Controlador Programable S7 1200, 2009)

Es preciso asignar un bloque de datos instancia unívoco al insertar la instrucción MB\_SLAVE en el programa. El nombre de este bloque de datos instancia MB\_SLAVE se utiliza cuando se especifica el parámetro MB\_DB de la instrucción MB\_COMM\_LOAD.

Los códigos de las funciones de comunicación Modbus (1, 2, 4, 5 y 15) pueden leer y escribir bits y palabras directamente en la memoria imagen de proceso de las entradas y salidas del PLC. La tabla siguiente muestra el mapeo de las direcciones Modbus en la memoria imagen de proceso de la CPU.

El bloque MB\_SLAVE se usa para definir la dirección de estación MODBUS-RTU y especificar la zona de almacenamiento de datos local para la transferencia de datos con el registro de participación.



**Figura 10** Bloque MB\_SLAVE

**Fuente:** (TIA Portal, 2013)

**Tabla 7**

Parámetros del Bloque MB\_SLAVE

Parámetro	Descripción
<b>MB_ADDR</b>	Transfiere la dirección estación MODBUS-RTU.
<b>MB_HOLD_REG</b>	Registro de participación del esclavo. El parámetro MB_HOLD_REG debe hacer referencia a un bloque de datos global creado con la opción "Sólo acceso simbólico" deshabilitada.

**Fuente:** (TIA Portal, 2013)

Cuando se tienen que transferir múltiples registros (palabras), con el parámetro "MB\_HOLD\_REG" se tiene que transferir el nombre simbólico de un array o estructura que cubra el volumen de datos especificado del maestro (DATA\_ADDR y DATA\_PTR). La primera palabra de este array o estructura se corresponde con la dirección de inicio 40001 del registro de participación de MODBUS. Los parámetros se pueden conmutar con las entradas y salidas de los bloques o mediante el acceso al correspondiente bloque de datos de instancia.

## 2.7 Micromaster 440

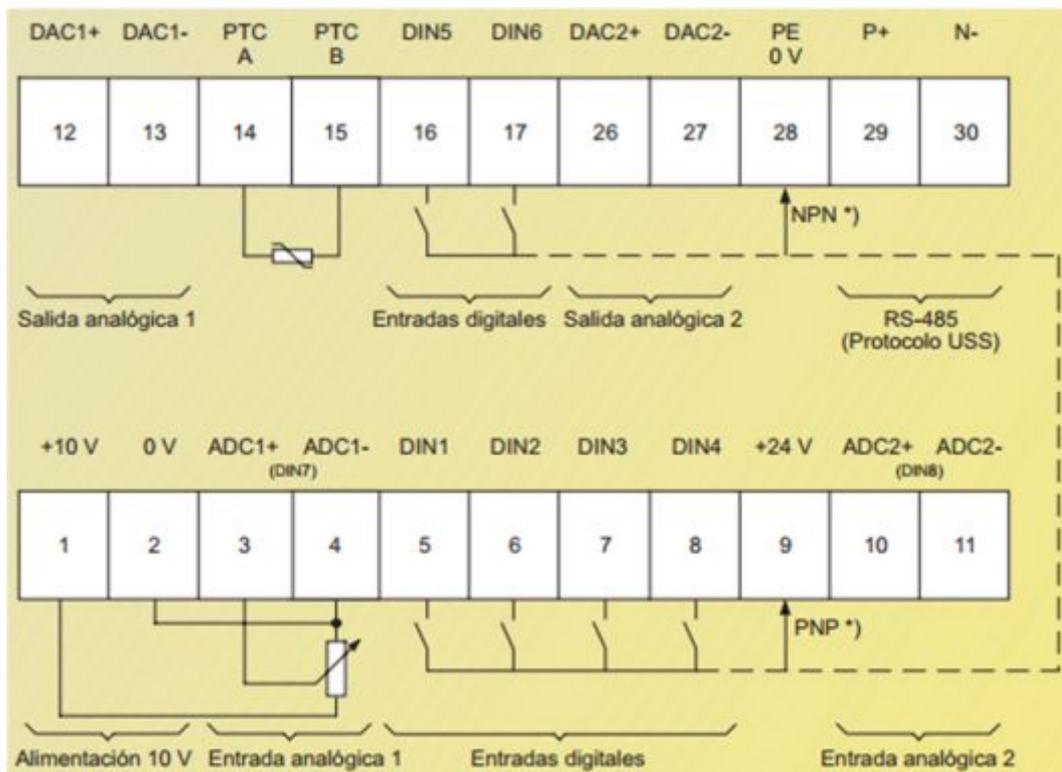
La serie MICROMASTER 440 es una gama de convertidores de frecuencia (también denominados variadores) para modificar la velocidad de motores trifásicos. Los distintos modelos disponibles abarcan un rango de potencias desde 120 W para entrada monofásica hasta 75 kW con entrada trifásica. Los convertidores están controlados por microprocesador y utilizan tecnología IGBT (Insulated Gate Bipolar Transistor) de última generación. Esto los hace fiables y versátiles. Un método especial de modulación por ancho de impulsos con frecuencia de pulsación seleccionable permite un funcionamiento silencioso del motor. Extensas funciones de protección, ofrecen una protección excelente tanto del convertidor como del motor. El MICROMASTER 440, con sus ajustes por defecto realizados en fábrica, es ideal para una gran gama de aplicaciones sencillas de control de motores, también puede utilizarse para aplicaciones más avanzadas de control, haciendo uso de su funcionalidad al completo. El MICROMASTER 440

puede utilizarse tanto en aplicaciones donde se encuentre aislado como integrado en sistemas de automatización. (García & Pubill, 2008)

### 2.7.1 Características Principales

- Fácil de instalar.
- Puesta en marcha sencilla.
- Diseño robusto en cuanto a CEM.
- Puede funcionar en alimentación de línea IT.
- Tiempo de respuesta a señales de mando rápido y repetible.
- Amplio número de parámetros que permite la configuración de una gama extensa de aplicaciones.
- Conexión sencilla de cables.
- Relés de salida.
- Salidas analógicas (0...20 mA)
- 6 entradas digitales NPN/PNP aisladas y conmutables.
- entradas analógicas.
  - AIN1: 0. 10 V, 0 . 20 mA y -10 a +10 V.
  - AIN2: 0 . 10 V, 0 . 20 mA.
- Las 2 entradas analógicas se pueden utilizar como la 7 y 8 entrada digital.
- Tecnología BiCo.
- Diseño modular para configuración extremadamente flexible.
- Altas frecuencias de pulsación para funcionamiento silencioso del motor.
- Información de estado detallada y funciones de mensaje integradas.
- Opciones externas para comunicación por PC, panel BOP (Basic Operator panel), panel AOP (Advanced Operator Panel) y módulo de comunicación PROFIBUS.

La distribución de pines para realizar el control mediante los bornes terminales y mando a distancia se observa en la siguiente figura.



**Figura 11** Descripción de pines del Micromaster 440

**Fuente:** (Garcia & Pubill, 2008)

### 2.7.2 Descripción de parámetros

- **r0000**

Muestra la visualización seleccionada por el usuario en P0005.

Pulsando el botón "Fn" durante 2 segundos el usuario puede ver los valores de

1 La tensión en el circuito intermedio.

2 La corriente de salida.

3 La frecuencia de salida.

4 La tensión de salida y el ajuste de r0000 elegido (definido en P0005).

- **P0005**

Selecciona la visualización para el parámetro r0000 (Visualización accionamiento).

Ajustes:

21 Frecuencia real.

25 Tensión de salida.

26 Tensión circuito intermedio.

27 Corriente de salida.

Indicaciones:

Estos ajustes sólo se refieren a números de parámetro de sólo lectura (rxxxx).

- **P0003**

Define el nivel de acceso a los juegos de parámetros. Para las aplicaciones más simples es suficiente con el ajuste por defecto.

Ajustes:

0 Lista de parámetros de usuario

1 Estándar

2 Extendido

3 Experto

4 Servicio: Protegido contraseña

- **P0700**

Selecciona la fuente para la orden digital.

Posibles ajustes:

0 Ajuste por defecto de fábrica

1 BOP (teclado)

2 Terminal

4 USS en conexión BOP

5 USS en conexión COM

6 CB en conexión COM

Nota: Cambiando de P0700 = 1 a P0700 = 2 se resetean todas las entradas digitales de los ajustes por defecto.

- **P1000**

Selecciona la fuente de consigna de frecuencia.

Posibles ajustes:

0 Sin consigna principal.

1 Consigna MOP.

- 2 Consigna analógica.
- 3 Frecuencia fija.
- 4 USS en conexión BOP.
- 5 USS en conexión COM.
- 6 CB en conexión COM.
- 7 Consigna analógica 2

- **P1058**

Frecuencia en Hz en Modo Jog para motor girando a la derecha.

- **P1059**

Frecuencia en Hz en Modo Jog para motor girando a la izquierda.

- **P1060**

Tiempo de aceleración JOG tiempo de aceleración en s de 0 a frecuencia máxima (P1082). La aceleración JOG la limitan P1058 y P1059.

- **P1082**

Ajusta la frecuencia de motor máxima [Hz] a la cual el motor. Funcionará independientemente de la consigna de frecuencia.

El valor que se ajuste aquí sirve para ambos sentidos de giro.

- **P1120**

Tiempo que necesita el motor por ejemplo para acelerar desde el punto muerto hasta la frecuencia máx. P1082.

Parametrizar un tiempo demasiado corto puede producir la alarma A0501 (limitación de la corriente) o que se desconecte el convertidor con el fallo F0001 (sobrecorriente).

- **P1121**

Tiempo que necesita el motor por ejemplo para desacelerar desde la frecuencia máx. P1082, hasta el punto muerto.

Parametrizar un tiempo demasiado corto puede producir la alarma A0501 (limitación de la corriente) o A0502 (límite por sobretensión) o que se desconecte el convertidor con el fallo F0001 (sobrecorriente) o F0002 (sobretensión)

## CAPÍTULO III

### IMPLEMENTACIÓN DE LA RED MODBUS PARA EL CONTROL DE DOS ESTACIONES DE PROCESOS

#### 3.1 Preliminares

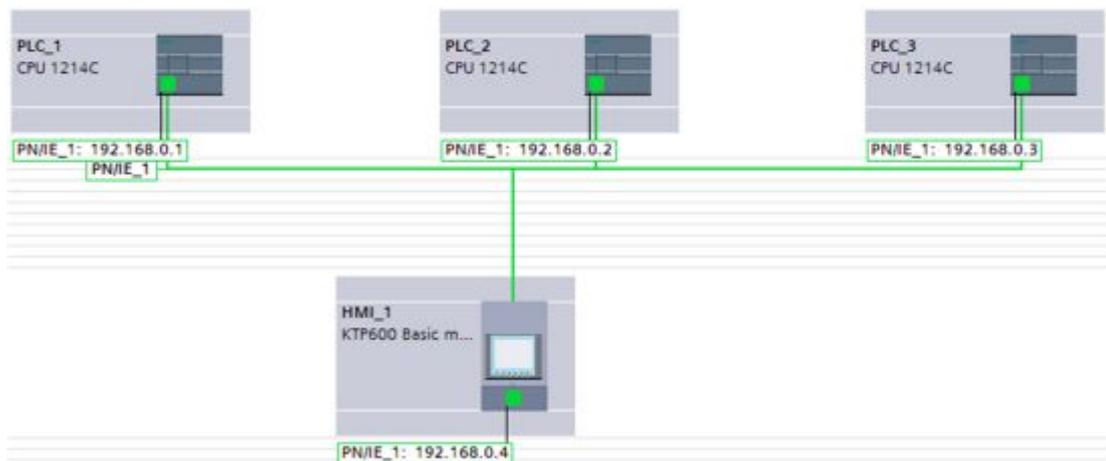
La red Modbus estará formada por 1 maestro y 2 esclavos. El maestro es el PLC\_1, el esclavo 1 es el PLC\_2 y el esclavo 2 es el PLC\_3. Los datos de los esclavos se visualizarán en el panel táctil KTP 600.

Los dispositivos empleados para el desarrollo del presente proyecto técnico son:

- 3 PLCs S7 1200 CPU 1214C AC/DC/RLY
- 2 Módulos de salidas analógicas SM 1232 AQ
- 1 Switch
- 1 estación de proceso de temperatura de Flujo de Aire PCT2
- 1 estación para el control de motor trifásico con variador de velocidad Micromaster 440
- Cable de comunicación Profibus
- Cable de comunicación Ethernet
- 1 Panel Táctil KTP 600

#### 3.2 Creación de la red Profinet

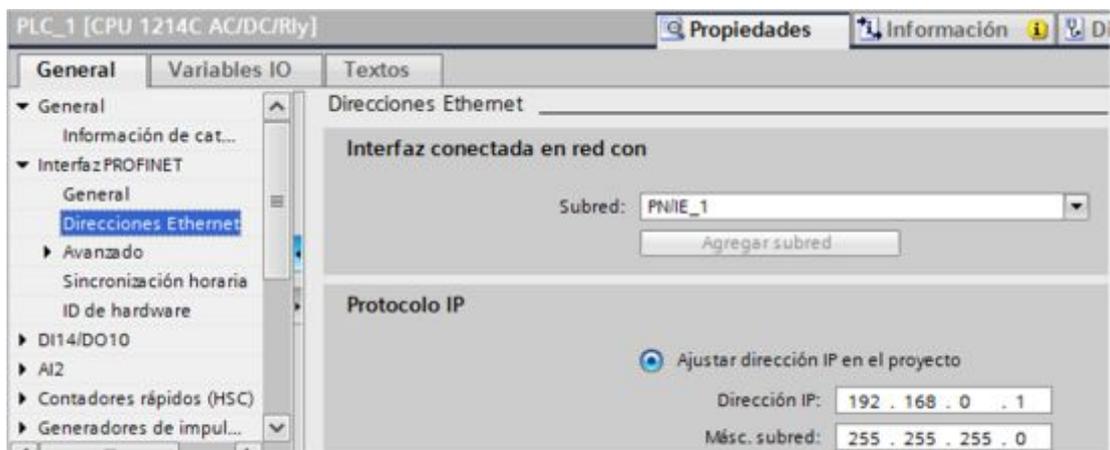
Primero conecte los tres PLCs y el panel táctil KTP 600 en red mediante la interfaz ProfiNet con la ayuda de un switch. Asigne la dirección 192.168.0.1 al PLC\_1, 192.168.0.2 al PLC\_2, 192.168.0.3 al PLC\_3 y la 192.168.0.4 al panel táctil KTP 600.



**Figura 12** Red Profinet del Proyecto Técnico

**Fuente:** (TIA Portal, 2013)

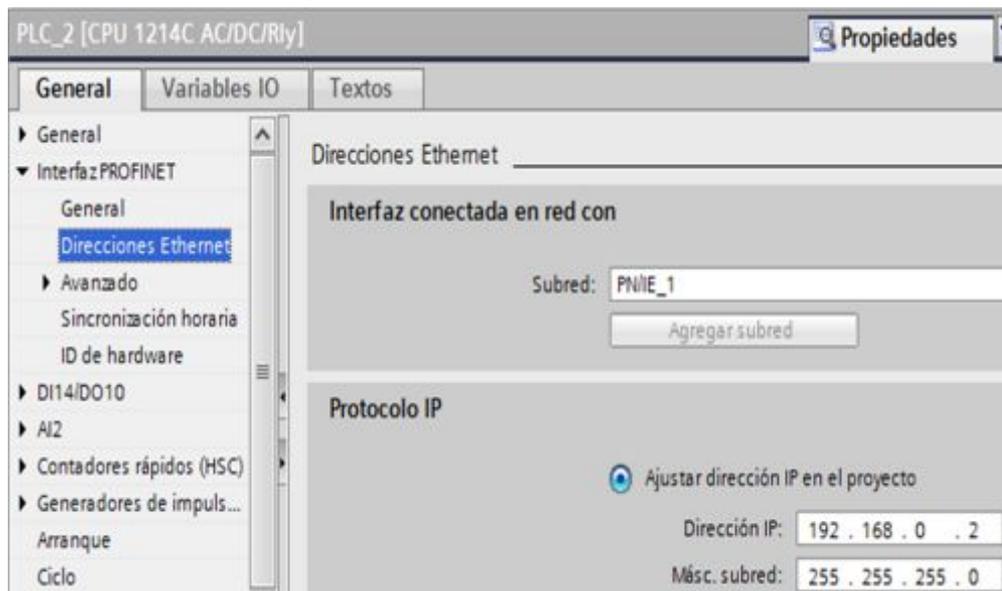
Para asignar la dirección IP en el PLC 1 ingrese a Propiedades/General/Direcciones Ethernet.



**Figura 13** Asignación de dirección IP al PLC\_1

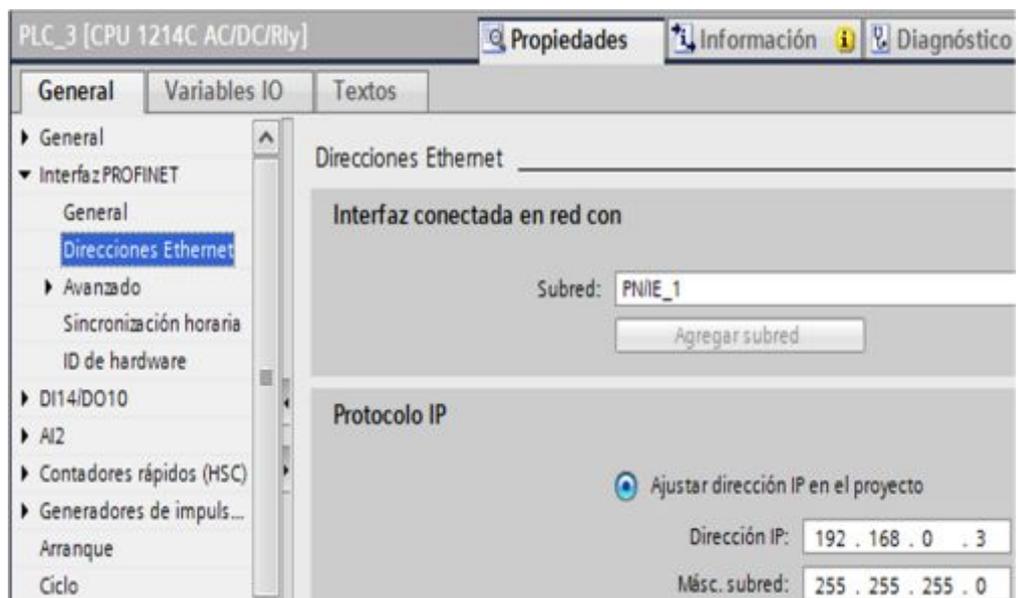
**Fuente:** (TIA Portal, 2013)

El mismo procedimiento siga para asignar las direcciones IP al PLC\_2, PLC\_3 y panel táctil KTP 600.



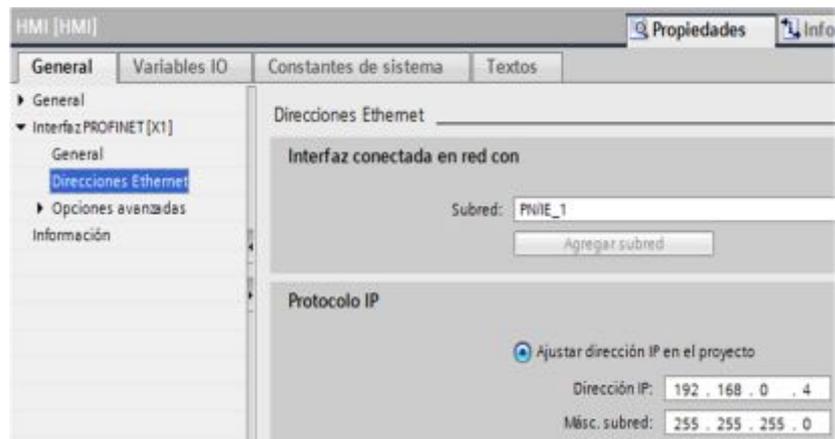
**Figura 14** Asignación de dirección IP al PLC\_2

**Fuente:** (TIA Portal, 2013)



**Figura 15** Asignación de dirección IP al PLC\_3

**Fuente:** (TIA Portal, 2013)



**Figura 16** Asignación de dirección IP al panel táctil

**Fuente:** (TIA Portal, 2013)

La dirección IP del computador debe ser diferente a la asignada a los dispositivos de la red ProfiNet.

### 3.3 Creación de la red Modbus

Para crear la red ModBus se utilizó el módulo de comunicación para el PLC S7 1200, CM 1241 (RS 422/485).

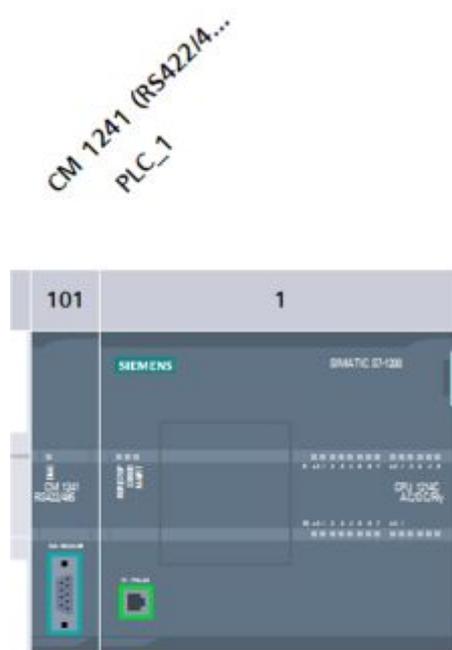
Desde el catálogo de dispositivos seleccione Módulos e comunicación/Punto a Punto a Punto/CM 1241 (RS 422/485)/ 6ES7241-1CH32-0XB0.



**Figura 17** Selección del Módulo CM 1241 para el PLC\_1

**Fuente:** (TIA Portal, 2013)

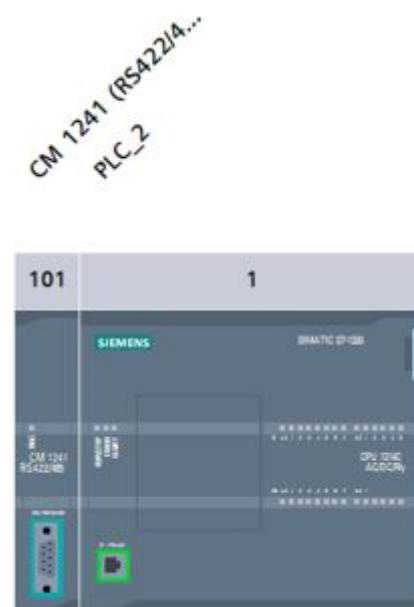
El módulo CM1241 y el PLC\_1 conectados se visualizan en la siguiente figura.



**Figura 18** PLC\_1 y módulo CM 1241

**Fuente:** (TIA Portal, 2013)

Repita los pasos para el PLC\_2 y para el PLC\_3.



**Figura 19** PLC\_2 y módulo CM 1241

Fuente: (TIA Portal, 2013)

CM 1241 (RS422/IA...  
PLC\_3



Figura 20 PLC\_3 y módulo CM 1241

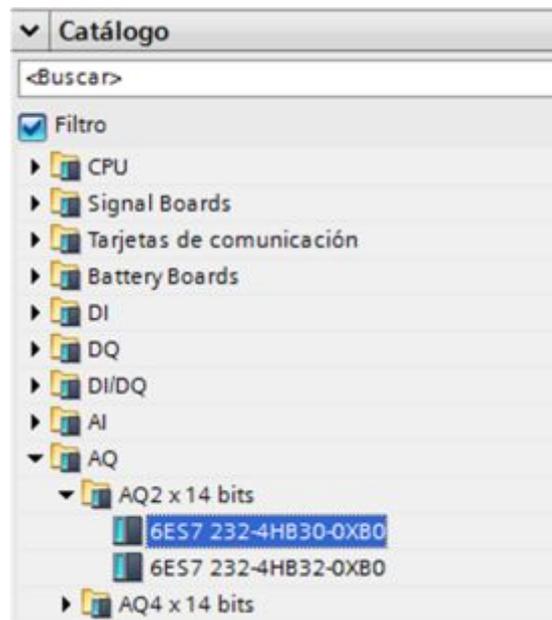
Fuente: (TIA Portal, 2013)

Con un cable Profibus DP conecte los tres módulos CM 1241 para formar la red ModBus.

### 3.4 Conexión de módulos de señales

Como el objetivo de este proyecto es monitorear dos estaciones de procesos mediante la red ModBus RTU, es necesario agregar al PLC\_2 y PLC\_3 módulos de salidas analógicas necesarios para modificar el estado de las variables de cada uno de los procesos empleados.

Del catálogo de dispositivos seleccione AQ/AQ x 14 bits/6ES7 232-4HB30-0XB0



**Figura 21** Selección del módulo de señales SM 1232 AQ

**Fuente:** (TIA Portal, 2013)

La siguiente figura muestra el PLC\_2 con sus módulos conectados



**Figura 22** Modulo SM 1232 AQ conectado al PLC\_2

**Fuente:** (TIA Portal, 2013)

Repita el mismo procedimiento para el PLC\_3.

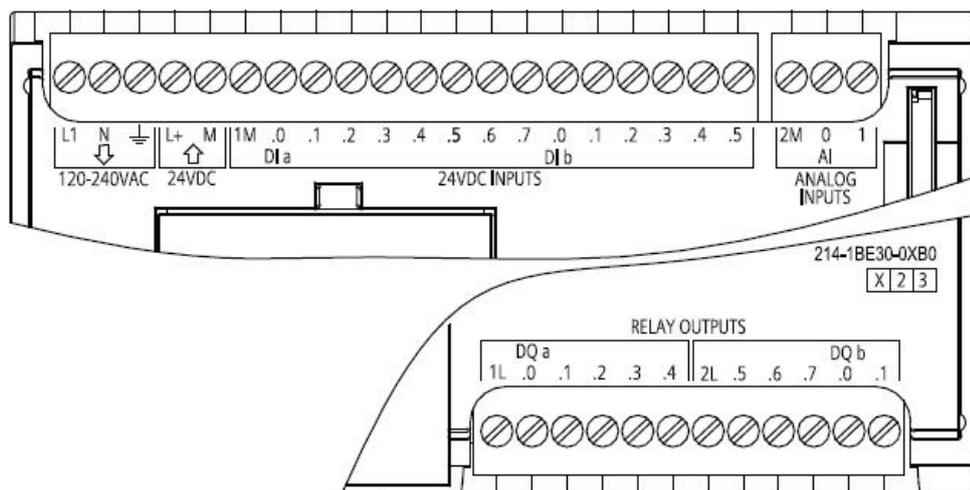


**Figura 23** Modulo SM 1232 AQ conectado al PLC\_2

**Fuente:** (TIA Portal, 2013)

### 3.5 Conexiones físicas de los dispositivos

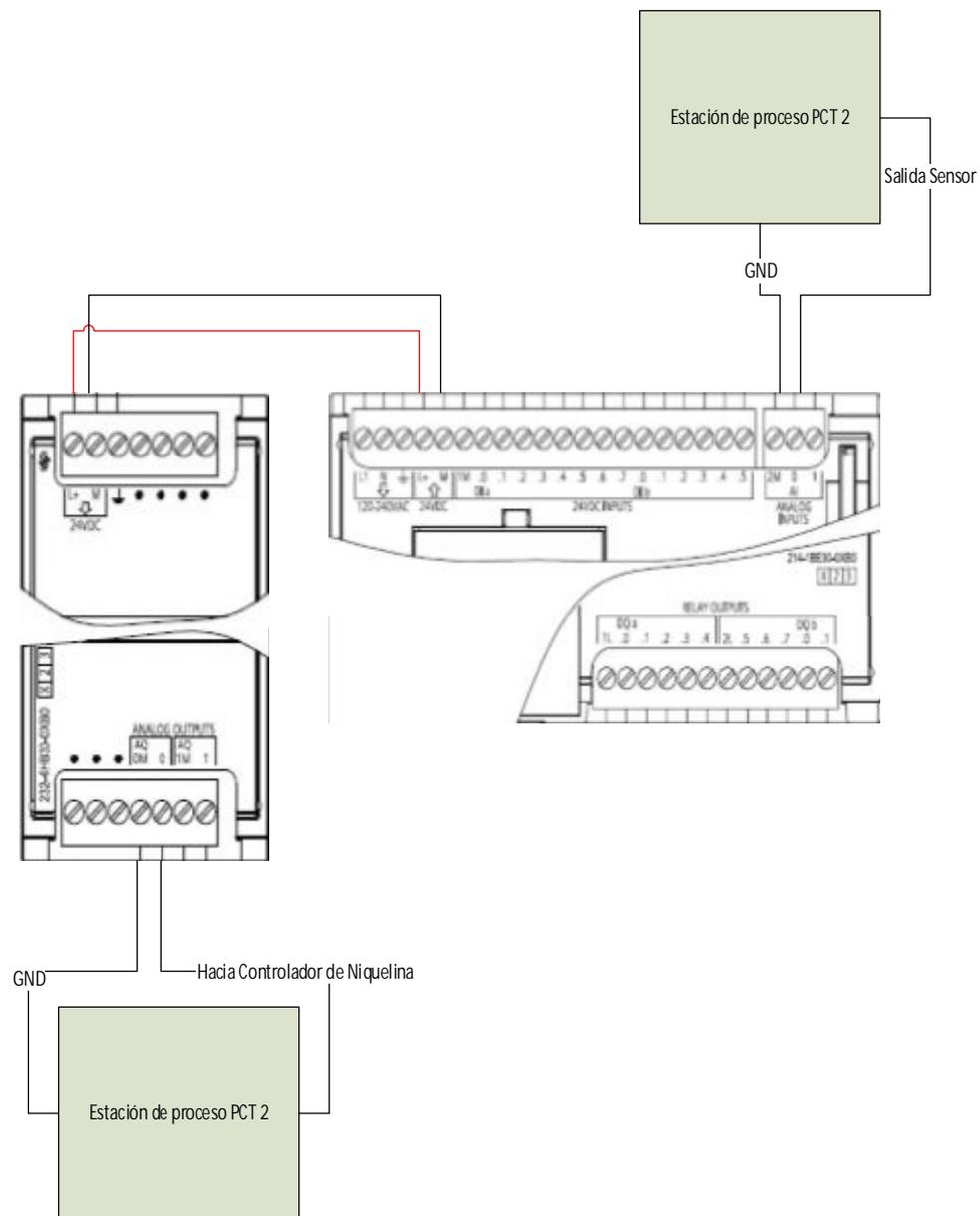
La figura 24 muestra el diagrama de los terminales del PLC S7 1200 empleado, en base a esta figura se explicará las conexiones eléctricas realizadas en cada estación de procesos.



**Figura 24** Diagrama de terminales del PLC S7 1200

**Fuente:** (Siemens, Controlador Programable S7 1200, 2009)

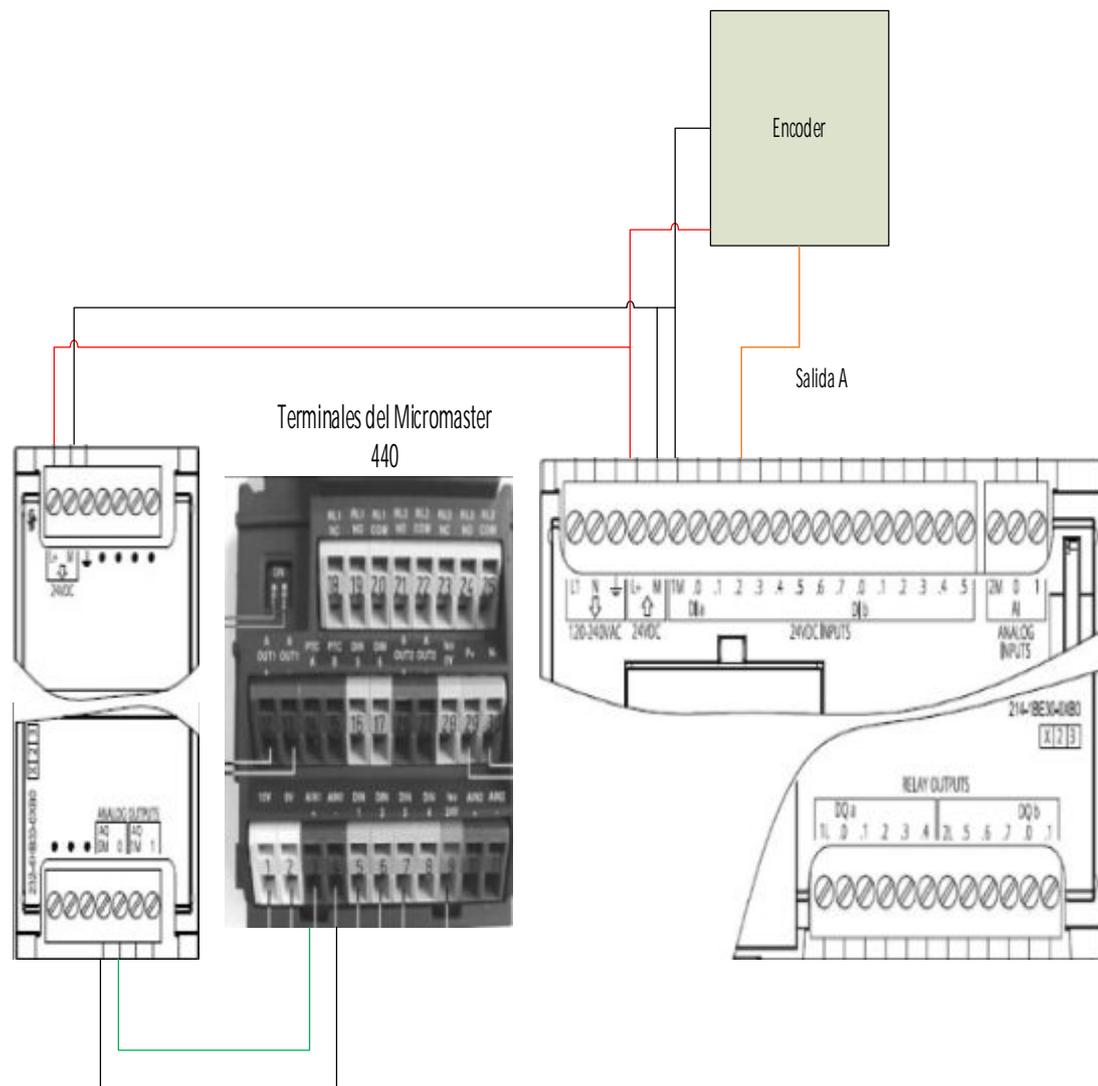
Para el monitoreo de la estación de flujo de aire PCT 2 fue necesario realizar las siguientes conexiones:



**Figura 25** Esquema de conexiones estación PCT 2

**Fuente:** (Siemens, Controlador Programable S7 1200, 2009)

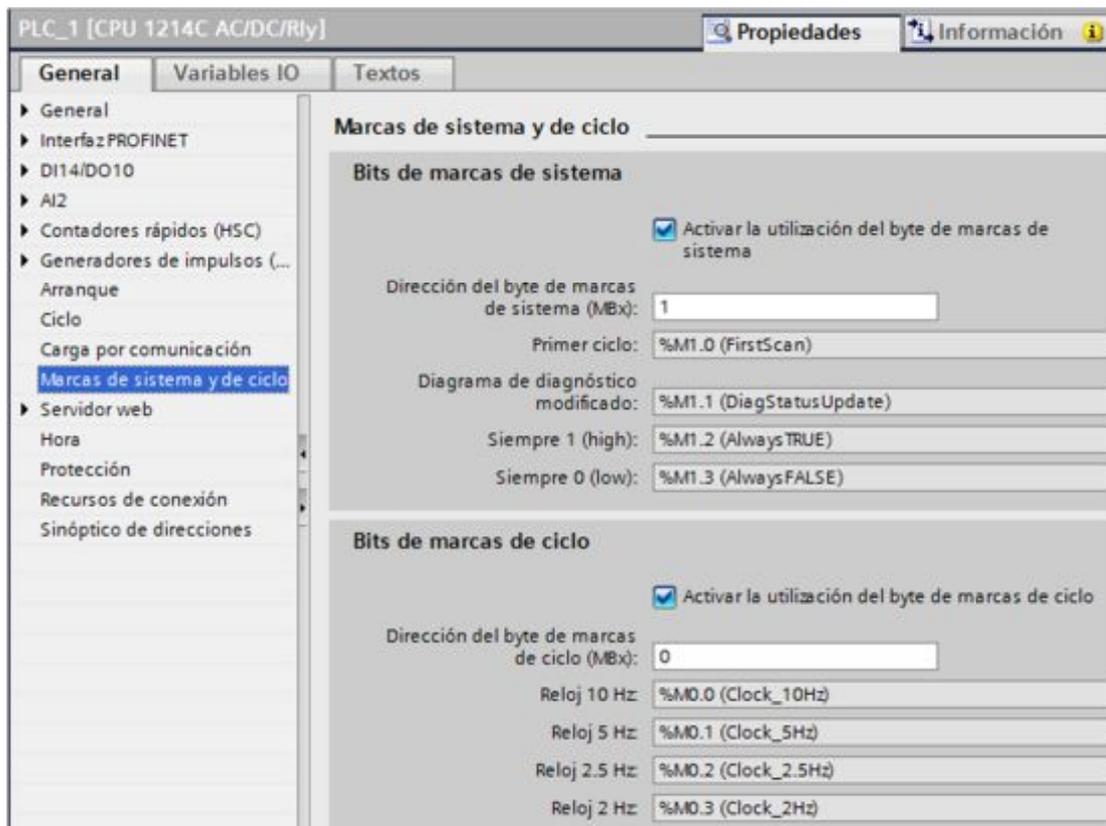
Para el monitoreo de la estación de control de velocidad de un motor trifásico mediante el variador Micromaster 440, fue necesario realizar las siguientes conexiones:



**Figura 26** Esquema de conexiones de la estación con el Micromaster 440

### 3.6 Configuración y programación del maestro

El dispositivo maestro de la red es el PLC\_1, este es el encargado de solicitar las peticiones o enviar datos a los dispositivos esclavos cada 200 ms. En primer lugar se activan los bits de sistema y de marca necesarios en el PLC maestro.



**Figura 27** Activación de bits de marca de sistema y ciclo

**Fuente:** (TIA Portal, 2013)

También es necesario crear un bloque de datos para intercambiar los datos entre el maestro y los esclavos.

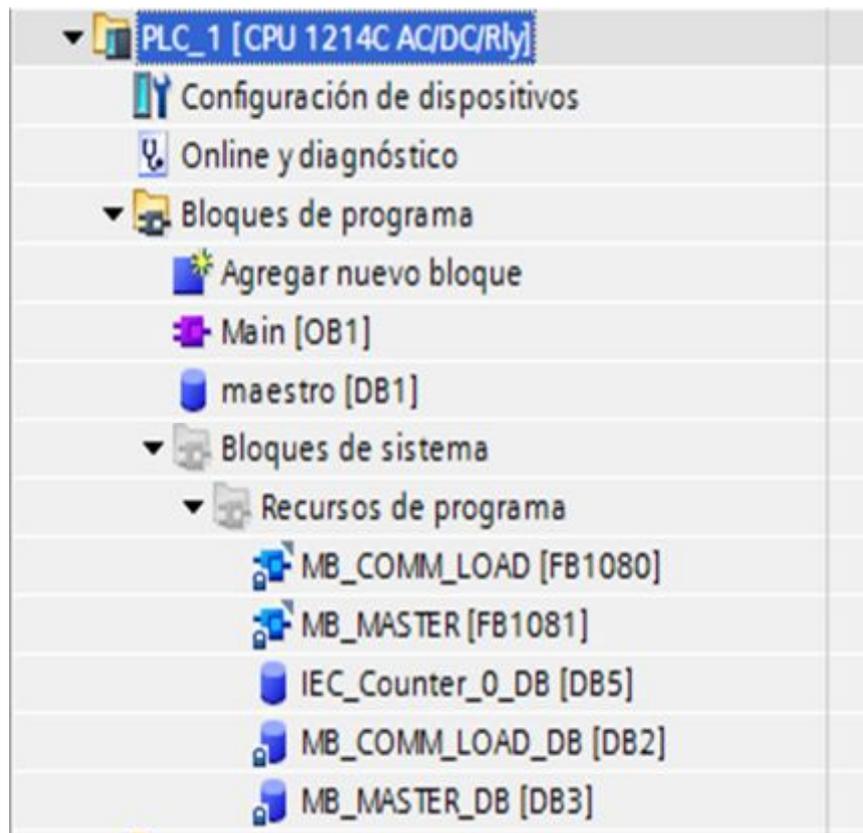
maestro			
	Nombre	Tipo de datos	Offset
1	Static		
2	recibir	Array [0..1] of Real	0.0
3	recibir[0]	Real	0.0
4	recibir[1]	Real	4.0
5	recibir1	Array [0..1] of Int	8.0
6	recibir1[0]	Int	0.0
7	recibir1[1]	Int	2.0
8	enviar	Array [0..1] of Int	12.0
9	enviar[0]	Int	0.0
10	enviar[1]	Int	2.0
11	enviar1	Array [0..1] of Int	16.0
12	enviar1[0]	Int	0.0
13	enviar1[1]	Int	2.0

**Figura 28** Bloque de datos del maestro

**Fuente:** (TIA Portal, 2013)

El bloque de datos de la figura 28 consta de cuatro datos tipo Array, llamados recibir, recibir1, enviar y enviar1. Los datos recibir y enviar sirven para intercambiar datos entre el maestro y esclavo1, mientras que los datos recibir1 y enviar1 sirven para intercambiar datos entre el maestro y el esclavo2.

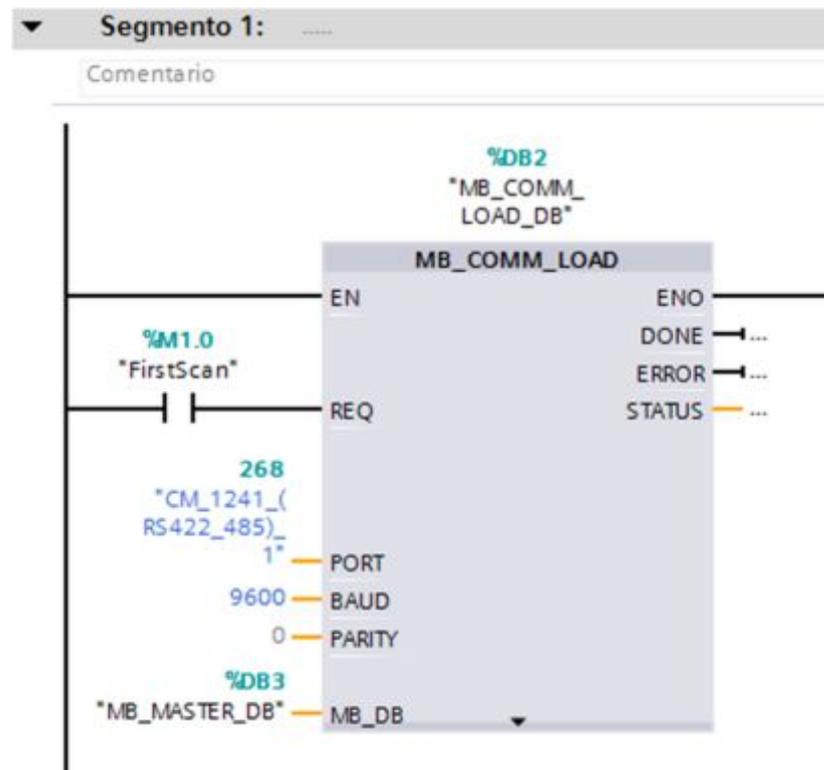
Los bloques de sistema que contiene el PLC\_1 se muestran en la figura 29.



**Figura 29** Bloque de sistema PLC\_1

**Fuente:** (TIA Portal, 2013)

El bloque MB\_COMM\_LOAD\_DB, sirve para configurar la comunicación ModBus RTU del maestro, se ejecuta una sola vez gracias al bit de marca de sistema FirstScan que corresponde al bit de marca interna %M1.0

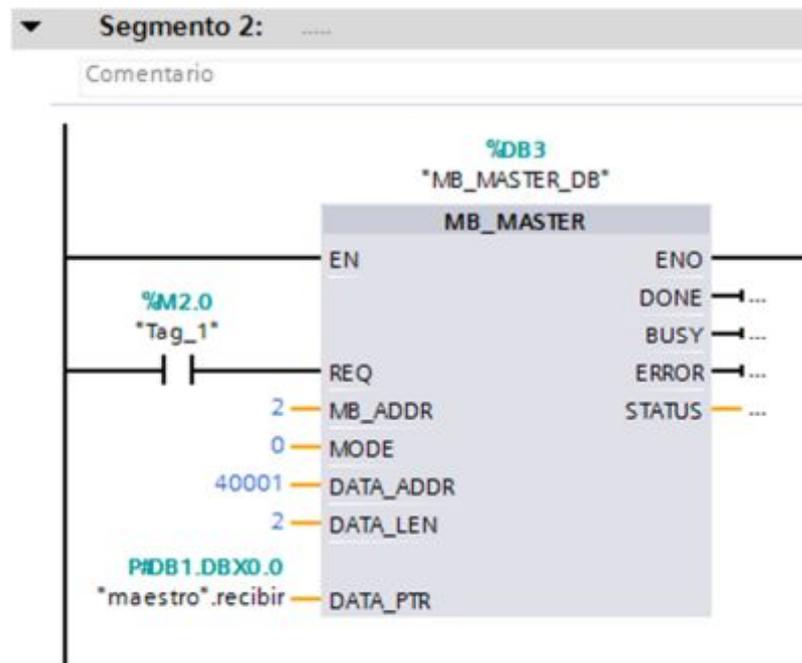


**Figura 30** Configuración del maestro ModBus RTU

**Fuente:** (TIA Portal, 2013)

Parte del protocolo de comunicación es los 9600 que representa la velocidad de transmisión y la paridad cuyo valor es 0 que representa ninguna. En la entrada del bloque de sistema denominada MB\_DB, se especifica la función mediante la cual el maestro accede o envía datos a los esclavos.

El segmento 2 del programa del maestro se crea el canal de comunicación entre el maestro y el esclavo 1 para recibir datos. La dirección (MB\_ADDR) es 2 porque en una red Modbus el maestro tiene la dirección 1. El MODE es 0 porque se realiza la operación de lectura, se inicia en la dirección 40001 porque corresponde a los registros finalmente en DATA\_PTR, se especifica que datos del bloque creado en el maestro se va a emplear, en este caso maestro.recibir.



**Figura 31** Instrucción para recibir datos del esclavo 1

**Fuente:** (TIA Portal, 2013)

La entrada REQ necesita un valor verdadero; es decir, que se cierre el contacto %M2.0, para que se ejecute la instrucción de lectura o escritura en el maestro.

El maestro solicita o envía información a los esclavos mediante una secuencia; es decir, el maestro realiza una petición de lectura o escrita a un esclavo a la vez. Por tal motivo se creó un contador de 0 a 3 para cumplir con este requerimiento.

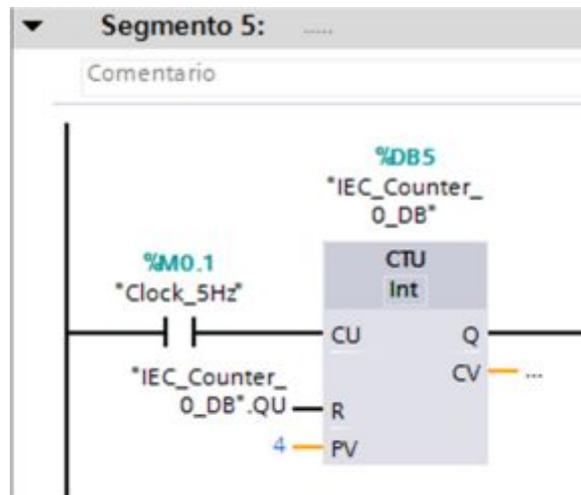
**Tabla 8.**

Función de lectura y escritura que realiza el maestro

Valor del Contador	Bits de marca interna				Función
	%M2.0	%M2.1	%M2.2	%M2.3	
0	1	0	0	0	Lee datos del esclavo 1
1	0	1	0	0	Escribe datos en el esclavo 1
2	0	0	1	0	Lee datos del esclavo 2
3	0	0	0	1	Escribe datos en el esclavo 2

El contador es ascendente, alcanza un valor de conteo máximo de 3 porque el cuarto sirve como reseteo del contador. El intervalo de contaje es 200ms

debido a que tiene conectada en la entrada CPU la marca de ciclo %M0.1 que genera una señal de reloj con una frecuencia de 5 Hz.



**Figura 32** Contador para la secuencia de comunicación ModBus.

**Fuente:** (TIA Portal, 2013)

La lectura que realiza el maestro al esclavo 1 se ejecuta cuando se activa la marca %M2.0



**Figura 33** Bit que activa lectura del esclavo 1

**Fuente:** (TIA Portal, 2013)

La escritura que realiza el maestro al esclavo 1 se ejecuta cuando se activa la marca %M2.1



**Figura 34** Bit que activa escritura del esclavo 1

**Fuente:** (TIA Portal, 2013)

La lectura que realiza el maestro al esclavo 2 se ejecuta cuando se activa la marca %M2.2



**Figura 35** Bit que activa lectura del esclavo 2

**Fuente:** (TIA Portal, 2013)

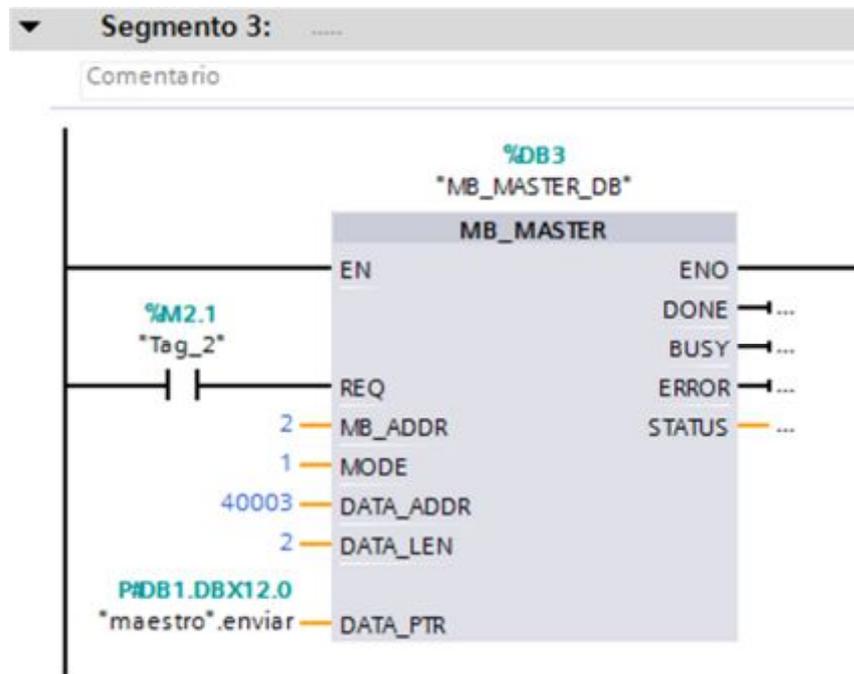
La escritura que realiza el maestro al esclavo 2 se ejecuta cuando se activa la marca %M2.3



**Figura 36** Bit que activa escritura del esclavo 2

**Fuente:** (TIA Portal, 2013)

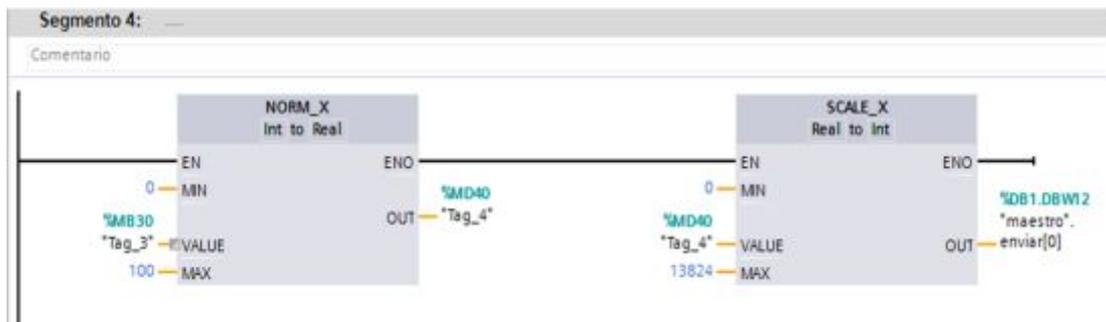
En el segmento 3 se escribe datos al esclavo 1 mediante la dirección 40003 y el dato enviar del bloque de datos del maestro. Para esto la entrada MODE es igual a 1.



**Figura 37** Escritura de datos al esclavo 1

**Fuente:** (TIA Portal, 2013)

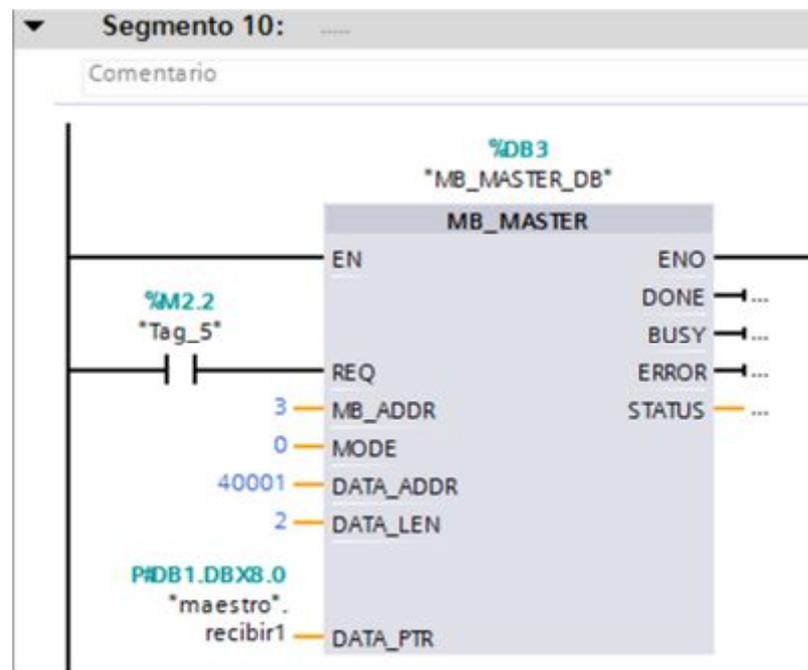
Como al esclavo 1 está conectada la estación de proceso de temperatura de flujo de aire PCT 2, es necesario enviar desde el panel táctil un valor entre 0 y 100 para que este sea convertido a un valor entre 0 y 13824 empleando las funciones NORM\_X y SCALE\_X, este valor llegara por ModBus al PLC\_2, este enviara al módulo de salidas analógicas SM 1232 AQ para que sea convertido a un valor de voltaje de 0 a 5 Vdc.



**Figura 38** Escalamiento de valores a ser enviados al esclavo 1

**Fuente:** (TIA Portal, 2013)

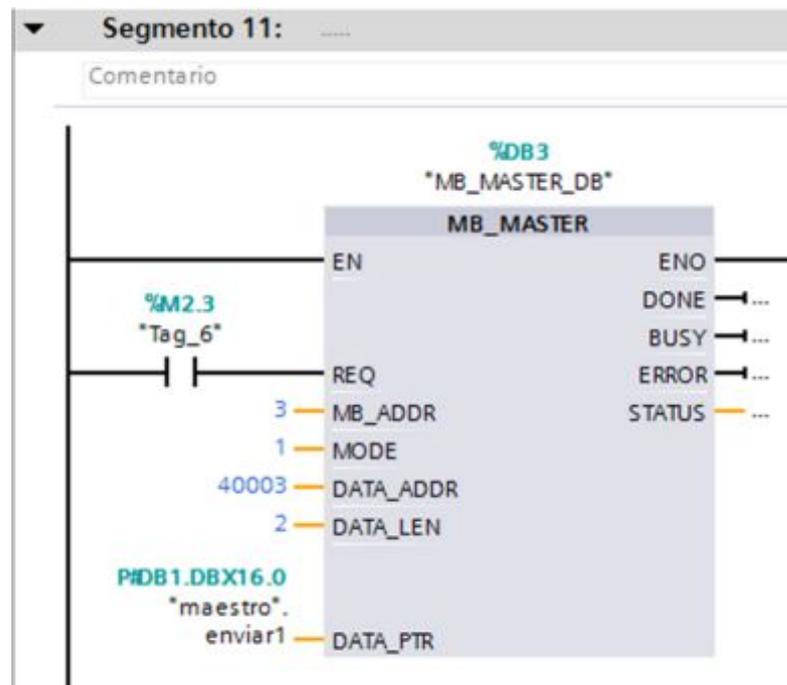
El segmento 10 del programa del maestro se crea el canal de comunicación entre el maestro y el esclavo 2 para recibir datos. La dirección (MB\_ADDR) es 3. El MODE es 0 porque se realiza la operación de lectura, se inicia en la dirección 40001 porque corresponde a los registros, finalmente en DATA\_PTR, se especifica que datos del bloque creado en el maestro se va a emplear, en este caso maestro.recibir1.



**Figura 39** Instrucción para recibir datos del esclavo 2

**Fuente:** (TIA Portal, 2013)

En el segmento 11 se escribe datos al esclavo 2 mediante la dirección 40003 y el dato enviar1 del bloque de datos del maestro. Para esto la entrada MODE es igual a 1.



**Figura 40** Escritura de datos al esclavo 2

**Fuente:** (TIA Portal, 2013)

Como al esclavo 2 está conectada la estación de proceso control de un motor trifásico mediante el Micromaster 440, es necesario enviar desde el panel táctil un valor entre 0 y 100 para que este sea convertido a un valor entre 0 y 27648 empleando las funciones NORM\_X y SCALE\_X, este valor llegará por ModBus al PLC\_3, este enviará al módulo de salidas analógicas SM 1232 AQ para que sea convertido a un valor de voltaje de 0 a 10 Vdc.

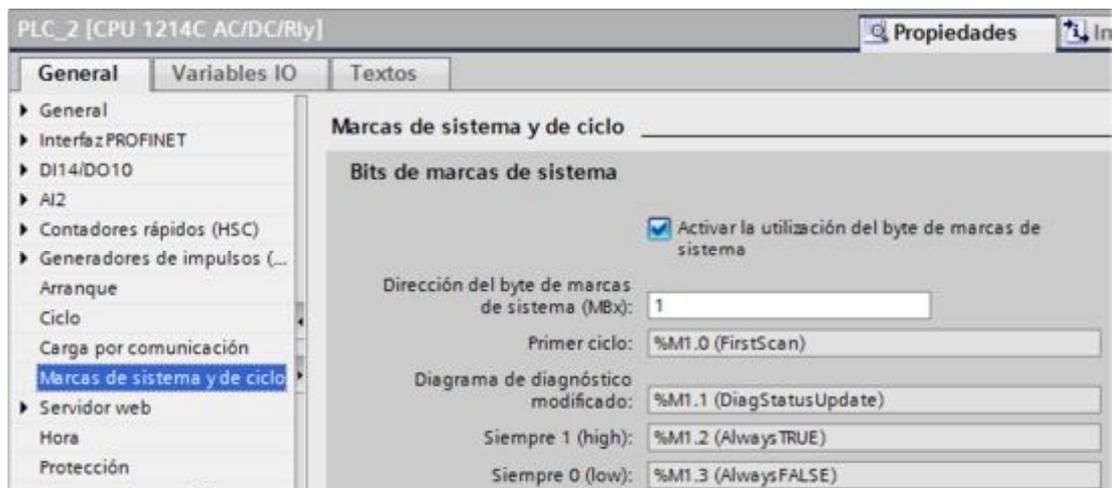


**Figura 41** Escalamiento de valores a ser enviados al esclavo 2

**Fuente:** (TIA Portal, 2013)

### 3.7 Configuración y programación del esclavo 1

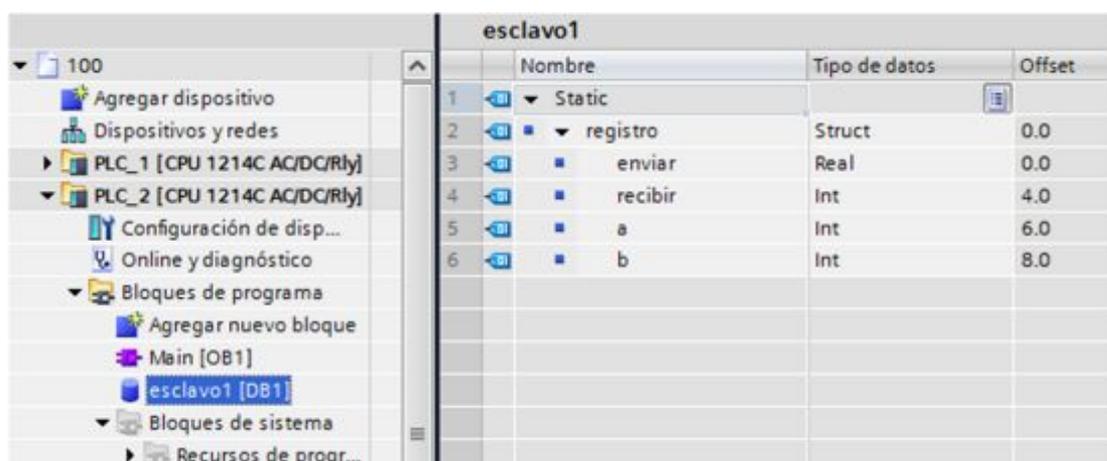
El dispositivo esclavo 1 de la red es el PLC\_2, este es el encargado de recibir o enviar datos al maestro. En primer lugar se activan los bits de sistema en el PLC esclavo.



**Figura 42** Activación de bits de marca de sistema del esclavo 1

**Fuente:** (TIA Portal, 2013)

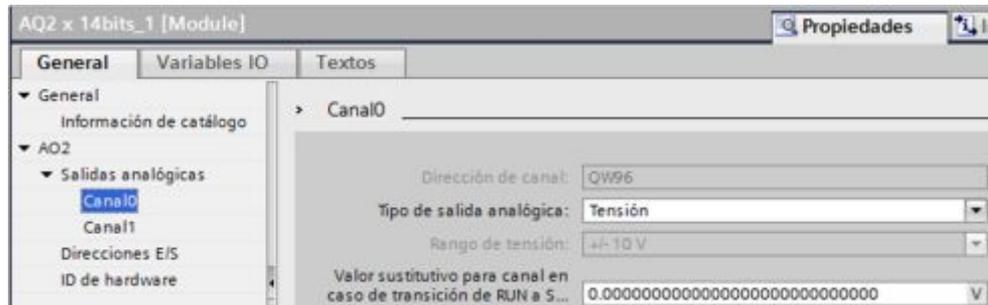
También es necesario crear un bloque de datos para intercambiar los mismos entre el esclavo 1 y el maestro.



**Figura 43** Bloque de datos del esclavo 1

**Fuente:** (TIA Portal, 2013)

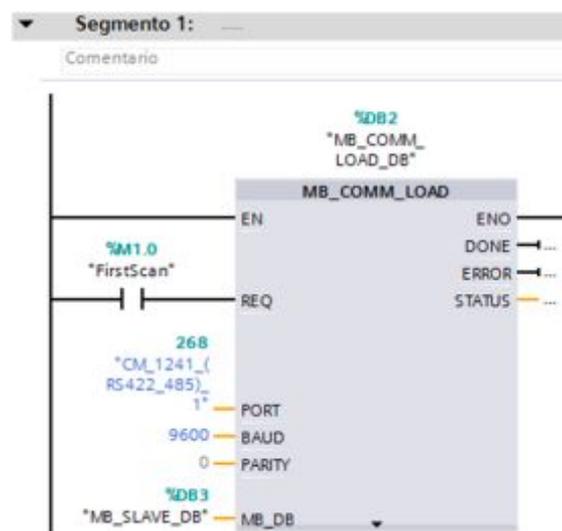
El bloque de datos de la figura 43 consta de un dato tipo estructura, el mismo que contiene 4 elementos necesarios para el intercambio de datos entre el esclavo 1 y el maestro. El módulo de señales analógicas SM 1232 AQ conectado al PLC\_2, tiene como dirección de la salida cero la QW96.



**Figura 44** Dirección de la salida analógica 0 del esclavo 1

**Fuente:** (TIA Portal, 2013)

El bloque MB\_COMM\_LOAD\_DB, sirve para configurar la comunicación ModBus RTU del esclavo 1, se ejecuta una sola vez gracias al bit de marca de sistema FirstScan que corresponde al bit de marca interna %M1.0. Parte del protocolo de comunicación es los 9600 que representa la velocidad de transmisión y la paridad cuyo valor es 0 que representa ninguna. En la entrada del bloque de sistema denominada MB\_DB, se especifica la función mediante la cual el esclavo recibe o envía datos a maestro.



**Figura 45** Configuración del esclavo 1

**Fuente:** (TIA Portal, 2013)

En los esclavos se emplea la instrucción MB\_SLAVE\_DB, en esta se asigna la dirección del esclavo y el bloque de datos necesario para el intercambio de datos entre el esclavo 1 y el maestro.

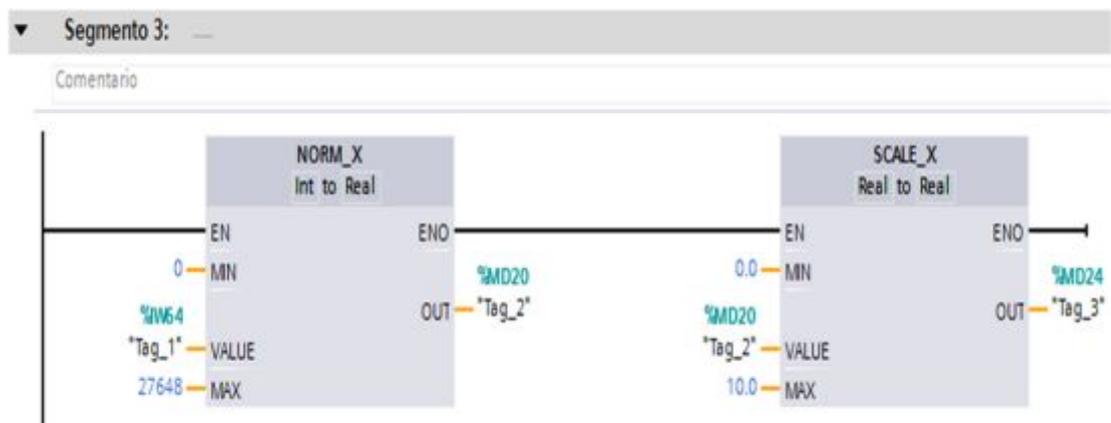


**Figura 46** Asignación de dirección del esclavo 1

**Fuente:** (TIA Portal, 2013)

Este esclavo lee por la entrada analógica cero, que viene integrada en el mismo PLC, la señal acondicionada del sensor de temperatura (IC). El segmento 3 escala el valor de 0 a 27648 a un valor de 0 a 10 que representa el voltaje máximo que puede medir. Este valor se guarda en la marca %MD24.

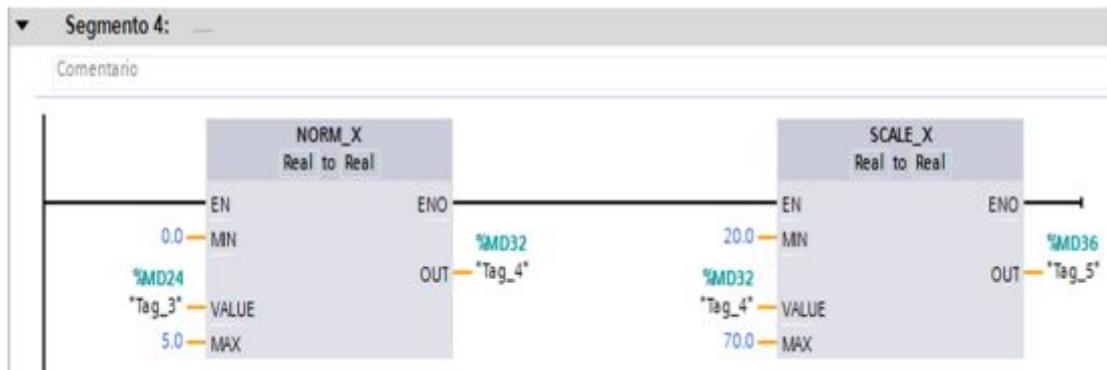
#### Escalamiento de la entrada analógica 0 del esclavo 1



**Figura 47** Escalamiento de la entrada analógica 0 del esclavo 1

**Fuente:** (TIA Portal, 2013)

El valor presente en la marca %MD24 nuevamente es escalado para convertir un valor de 0 a 5Vdc en un valor de 20 a 70 que representan los grados centígrados medidos por el sensor de temperatura.



**Figura 48** Escalamiento para convertir a grados centígrados

**Fuente:** (TIA Portal, 2013)

El valor presente en la marca interna %MD36 es enviado mediante el bloque de datos creado en el esclavo 1 al maestro.



**Figura 49** Valor de temperatura enviado al maestro

**Fuente:** (TIA Portal, 2013)

El esclavo 1 recibe los datos del maestro mediante el bloque de datos esclavo1.registro.jrecibir, este dato es movido a la dirección %QW96 que corresponde al canal de salida analógico cero.

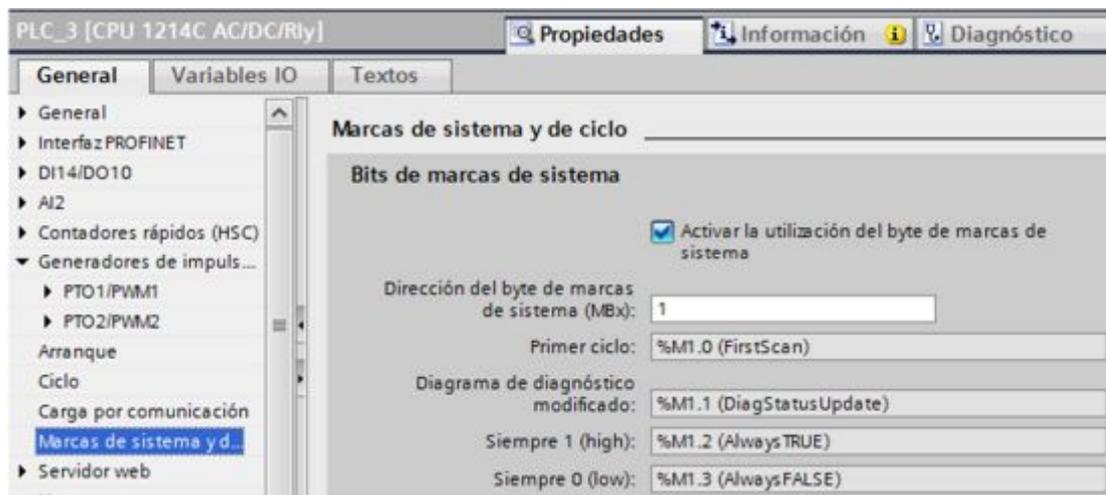


**Figura 50** Recepción de datos desde el maestro

**Fuente:** (TIA Portal, 2013)

### 3.8 Configuración y programación del esclavo 2

El dispositivo esclavo 2 de la red es el PLC\_3, este es el encargado de recibir o enviar datos al maestro. En primer lugar se activan los bits de sistema en el PLC esclavo.



**Figura 51** Activación de bits de marca de sistema del esclavo 2

**Fuente:** (TIA Portal, 2013)

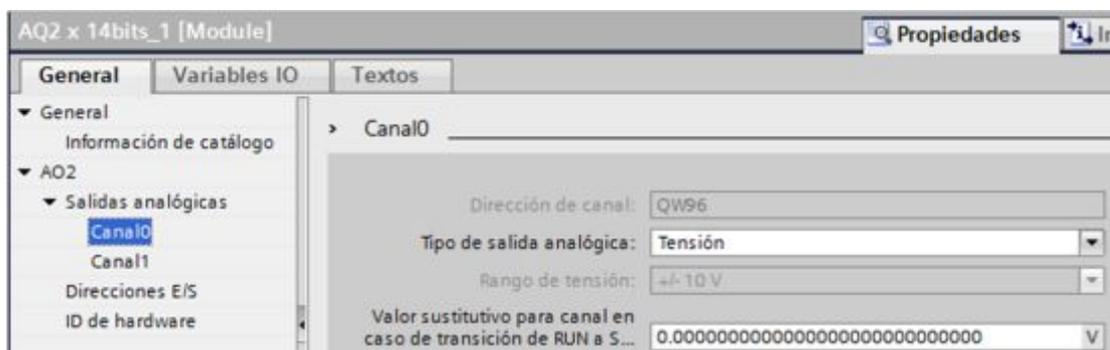
También, es necesario crear un bloque de datos para intercambiar los mismos entre el esclavo 2 y el maestro.

esclavo2		
Nombre	Tipo de datos	Offset
1 Static		
2 registro	Struct	0.0
3 enviar	Int	0.0
4 recibir	Int	2.0
5 a	Int	4.0
6 b	Int	6.0

**Figura 52** Bloque de datos del esclavo 2

**Fuente:** (TIA Portal, 2013)

El bloque de datos de la figura 52 consta de un dato tipo estructura, el mismo que contiene 4 elementos necesarios para el intercambio de datos entre el esclavo 2 y el maestro. El módulo de señales analógicas SM 1232 AQ conectado al PLC\_3, tiene como dirección de la salida cero la QW96.



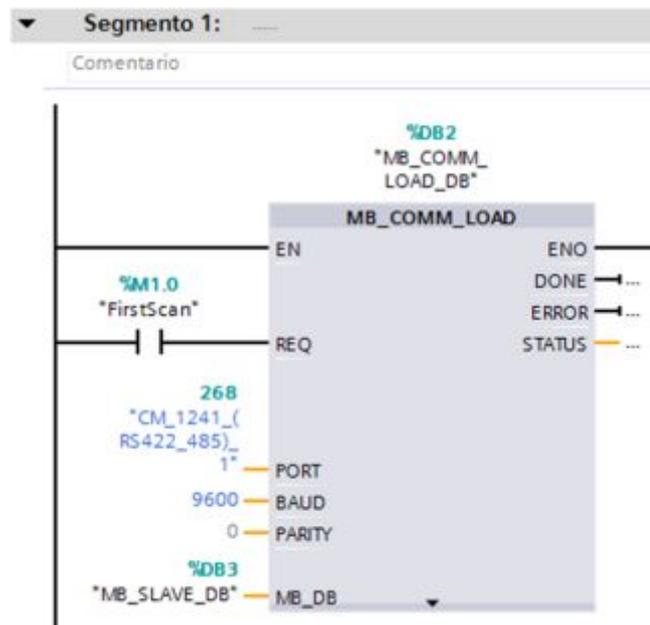
**Figura 53** Dirección de la salida analógica 0 del esclavo 2

**Fuente:** (TIA Portal, 2013)

El bloque MB\_COMM\_LOAD\_DB, sirve para configurar la comunicación ModBus RTU del esclavo 2, se ejecuta una sola vez gracias al bit de marca de sistema FirstScan que corresponde al bit de marca interna %M1.0.

Parte del protocolo de comunicación es los 9600 que representa la velocidad de transmisión y la paridad cuyo valor es 0 que representa ninguna.

En la entrada del bloque de sistema denominada MB\_DB, se especifica la función mediante la cual el esclavo recibe o envía datos a maestro.



**Figura 54** Configuración del esclavo 2

**Fuente:** (TIA Portal, 2013)

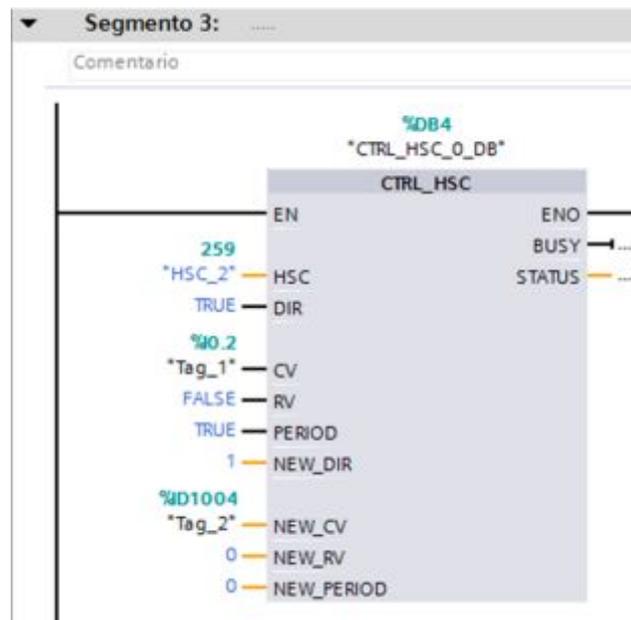
En los esclavos se emplea la instrucción MB\_SLAVE\_DB, en esta se asigna la dirección del esclavo y el bloque de datos necesario para el intercambio de datos entre el esclavo 2 y el maestro.



**Figura 55** Asignación de dirección del esclavo 2

**Fuente:** (TIA Portal, 2013)

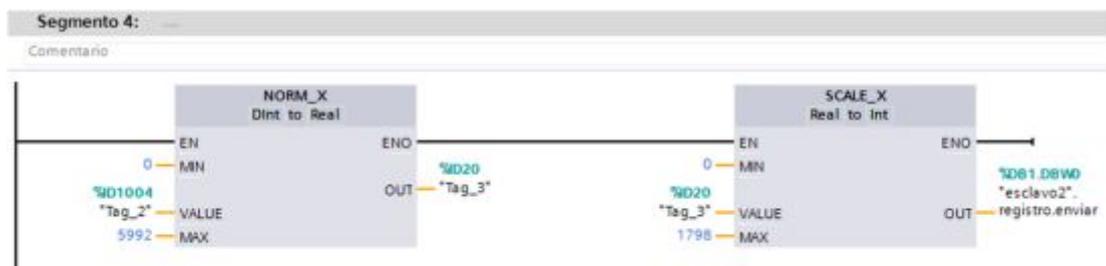
Este esclavo lee los pulsos de un encoder conectado a la entrada digital 2 del PLC, la %I0.2 es la entrada del contador rápido número 2, con la instrucción CTRL\_HSC\_0\_DB se lee la frecuencia de los pulsos entregados por el encoder y los guarda en la marca %ID1004.



**Figura 56** Lectura de la frecuencia de los pulsos del encoder

**Fuente:** (TIA Portal, 2013)

La frecuencia máxima que se lee mediante la marca %ID1004 cuando el motor alcanza su máxima velocidad es 5992 Hz, esta frecuencia equivale a un valor de 1798 rpm, por lo que es necesario escalar el valor de la frecuencia para convertir e rpm.



**Figura 57** Escalamiento para obtener velocidad del motor

**Fuente:** (TIA Portal, 2013)

El valor de la velocidad del motor se envía al maestro mediante el bloque de datos esclavo2.registro.enviar

El esclavo 2 recibe los datos del maestro mediante el bloque de datos esclavo2.registro. A este dato es movido a la dirección %QW96 que corresponde al canal de salida analógico cero.



**Figura 58** Datos recibidos por el esclavo 2 del maestro

**Fuente:** (TIA Portal, 2013)

### 3.9 HMI para el monitoreo de la red

Para crear la interface Humano Maquina de la red ModBus implementada, se empleó el panel táctil KTP 600, como son dos estaciones que están presentes en la red, se dividió la pantalla del panel táctil en dos partes, en la primera parte se monitorea el valor de la temperatura de la estación PCT 2 y también se ingresa en el campo de E/S etiquetado como control un valor de 0 a 100 para que se modifique la potencia de la niquelina y de esta manera variar la temperatura.

La segunda parte de la pantalla es para monitorear la velocidad del motor conectado al esclavo 2, también se ingresa en el campo de E/S etiquetado como control un valor de 0 a 100 para modificar la velocidad por intermedio del variador de velocidad Micromaster 440.



**Figura 59** HMI de la red ModBus

**Fuente:** (TIA Portal, 2013)

### 3.10 Pruebas de funcionamiento

Es necesario conectar los dispositivos que conforman la red como se indican en las figuras 25 y 26. La red ModBus RTU implementada está sin errores cuando los leds indicadores tanto de los PLCs como de los módulos de comunicación y señales están de color verde. En ese momento se visualizará el valor de la temperatura en grados centígrados.

En el campo de Entrada/Salida etiquetado como Control de la estación de temperatura ingrese un valor de 50, observe que la temperatura empieza a subir. Para comprobar el funcionamiento del esclavo 2 donde está conectado el variador de velocidad Micromaster 440, lo primero que se debe hacer es verificar si el parámetro 700 tiene el valor de 1 y el parámetro 1000 tiene un valor de 2. Esto asegura que la puesta en marcha del motor se realice a través del panel BOP y la variación de velocidad mediante los terminales donde está conectado el módulo de salidas analógicas del PLC\_3.

En el campo de Entrada/Salida etiquetado como Control en la estación de motor trifásico ingrese un valor de 50, encienda el motor desde el panel BOP, entonces observará un valor de velocidad de 900 rpm. Varié los controles de la estación de temperatura como del motor trifásico y observe los cambios en cada variable del proceso.

## CAPÍTULO IV

### CONCLUSIONES Y RECOMENDACIONES

#### 4.1 Conclusiones

- Se implementó una red Modbus RTU con un maestro y dos esclavos, para el monitoreo y variación de las señales existentes en la estación de proceso de temperatura PCT 2 y la estación de motor trifásico mediante un panel táctil KTP 600.
- El modo de operación de los módulos de comunicación CM 1241 en este proyecto como característica principal es Semiduplex 485 a 2 hilos; es decir, el maestro solo podrá realizar peticiones o envíos a un esclavo a la vez.
- Se configuró y programó en el PLC maestro una secuencia mediante un contador para que permita enviar y recibir datos de los esclavos uno por uno.
- Se creó una red Profinet entre los tres PLCs que conforman la red asignando direcciones IP distintas para cargar programas o monitorear en línea lo que permitió depurar errores en la comunicación.
- Se realizaron pruebas de funcionamiento logrando obtener resultados satisfactorios al controlar y monitorear las dos estaciones de procesos.

## 4.2 Recomendaciones

- Observe que los leds indicadores tanto de los PLCs como de los módulos estén de color verde, esto indica que la red está funcionando correctamente.
- Verifique que los parámetros 700 y 1000 del variador de velocidad Micromaster 440 estén como los expuestos en las pruebas de funcionamiento.
- En el modo de operación de la red Semiduplex no realice peticiones ni envío de datos a más de un esclavo.
- A futuro se investigue para ver la posibilidad de realizar secuencias de petición de un maestro Modbus sin emplear contadores.

**GLOSARIO DE TÉRMINOS.**

ASCII	Código Estándar Americano para el Intercambio de Información
BAUD	Velocidad de Transmision
BOP	Panel Básico del Operador
CM	Módulo de Comunicación
CRC	Control de Redundancia Cíclica
IP	Protocolo de Internet
LRC	Control de Redundancia Longitudinal
MASTER	Maestro
PARITY	Paridad
PLC	Controlador Lógico Programable
RTU	Unidad Terminal Remota
SLAVE	Esclavo
SM	Módulo de Señal
TCP	Protocolo de Control de Transmisión
TIA	Automatización Totalmente Integrada

## REFERENCIA BIBLIOGRÁFICA.

Cevallos, W. (Noviembre de 2007). Protocolo Modbus. Quito.

Garcia, J., & Pubill, J. (18 de Octubre de 2008). Introducción al Mircomaster 440 Parámetros. Barcelona, España.

microsonic. (18 de Noviembre de 2014). *Principio ultrasónico*. Obtenido de <http://www.microsonic.de/es/Interesting-facts.htm>

National Instruments. (15 de Octubre de 2015). *Introducción a Modbus*. Obtenido de <http://www.ni.com/white-paper/7675/es/>

Siemens. (11 de 2009). Controlador Programable S7 1200.

Siemens. (Enero de 2015). Hoja de datos 6ES7241-1CH32-0XB0.

TIA Portal, V. (2013). Software.

Villajulca, J. C. (19 de Julio de 2010). *Los Buses de Campo*. Obtenido de <http://www.instrumentacionycontrol.net/cursos-libres/automatizacion/curso-supervision-procesos-por-computadora/item/271-los-buses-de-campo-directo-al-grano.html>

# ANEXOS

