



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

**UNIDAD DE GESTIÓN DE  TECNOLOGÍAS**

**DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA.**

**CARRERA DE ELECTRÓNICA MENCIÓN INSTRUMENTACIÓN**

**& AVIÓNICA**

**TRABAJO DE GRADUACIÓN PARA LA OBTENCIÓN DEL  
TÍTULO DE:**

**TECNÓLOGO EN ELECTRÓNICA MENCIÓN  
INSTRUMENTACIÓN & AVIÓNICA**

**TEMA: “IMPLEMENTACIÓN DE UNA RED MODBUS CON UN  
MAESTRO Y DOS ESCLAVOS UTILIZANDO UN PLC-S7 – 1200 -  
CPU - 1215C CON MÓDULOS DE COMUNICACIÓN MODBUS 485.  
PARA PRÁCTICAS DE REDES INDUSTRIALES”.**

**AUTOR: CHANO PILLAJO RICHARD ESTALIN**

**DIRECTOR: Ing. Zahira Proaño C.**

**LATACUNGA**

**2019**



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

**CERTIFICACIÓN**

Certifico que el trabajo de Titulación, **“IMPLEMENTACIÓN DE UNA RED MODBUS CON UN MAESTRO Y DOS ESCLAVOS UTILIZANDO UN PLC-S7 – 1200 - CPU - 1215C CON MÓDULOS DE COMUNICACIÓN MODBUS 485. PARA PRÁCTICAS DE REDES INDUSTRIALES”** realizado por el Sr. **CHANO PILLAJO RICHARD ESTALIN**, 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 **CHANO PILLAJO RICHARD ESTALIN** para que lo sustente públicamente.

Latacunga, 14 de Febrero del 2019

Atentamente,

Ing. Zahira Proaño.



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

**AUTORÍA DE RESPONSABILIDAD**

Yo, **CHANO PILLAJO RICHARD ESTALIN**, con cédula de identidad N° **1723616130**, declaro que este trabajo de titulación “**IMPLEMENTACIÓN DE UNA RED MODBUS CON UN MAESTRO Y DOS ESCLAVOS UTILIZANDO UN PLC-S7 – 1200 - CPU - 1215C CON MÓDULOS DE COMUNICACIÓN MODBUS 485. PARA PRÁCTICAS DE REDES INDUSTRIALES**” 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, 14 de Febrero del 2019

---

Chano Pillajo Richard Estalin

C.I: 1723616130



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

**AUTORIZACIÓN (PUBLICACIÓN BIBLIOTECA VIRTUAL)**

Yo, **CHANO PILLAJO RICHARD ESTALIN**, 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 CON UN MAESTRO Y DOS ESCLAVOS UTILIZANDO UN PLC-S7 – 1200 - CPU - 1215C CON MÓDULOS DE COMUNICACIÓN MODBUS 485. PARA PRÁCTICAS DE REDES INDUSTRIALES”** cuyo contenido, ideas y criterios son de mi autoría y responsabilidad.

Latacunga, 14 de Febrero del 2019

---

Chano Pillajo Richard Estalin

C.I: 1723616130

## **DEDICATORIA**

Me gustaría ofrecer este logro a toda mi familia. Para mis padres Susana y Luis y a mis hermanos, por su comprensión y el apoyo en momentos buenos y malos.

Me han enseñado valores muy importantes y a no desmayar en ninguna circunstancia. A ellos agradezco la buena persona que formaron en mí con valores, principios, perseverancia, honestidad y metas.

Doy gracias a Dios por guiar mi camino siempre y por darme la oportunidad de conocer a personas muy especiales como (Sandy) quien fue la que cambió el rumbo de mi vida y me ayudo en momentos difíciles. Agradezco a todos mis maestros por compartir sus conocimientos.

## **AGRADECIMIENTO**

Este logro le doy gracias a Dios quien me ha ayudado a cumplir otra meta más en mi vida académica. Para mis padres Susana y Luis y a mis hermanos, por su comprensión y el apoyo en momentos buenos y malos.

Me han enseñado valores muy importantes y a no desmayar en ninguna circunstancia. A ellos agradezco la buena persona que formaron en mí con valores, principios, perseverancia, honestidad y metas.

Doy gracias a Dios por guiar mi camino siempre y por darme la oportunidad de conocer a personas muy especiales como (Sandy) quien fue la que cambió el rumbo de mi vida y me ayudo en momentos difíciles. Agradezco a todos mis maestros por compartir sus conocimientos a mis amigos que encontré en toda la carrera universitaria gracias por esos buenos momento.

## 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.....	XII
ÍNDICE DE FIGURAS.....	XIII
RESUMEN.....	XVII
ABSTRACT.....	XVIII
CAPITULO I.....	1
PLANTEAMIENTO DEL PROBLEMA.....	1
1.1 Antecedentes.....	1
1.2 Planteamiento del problema.....	2
1.3 Justificación.....	2
1.4 Objetivos.....	2
1.4.1 Objetivo general.....	2
1.4.2 Objetivos específicos.....	3
1.5 Alcance.....	3
CAPITULO II.....	4
MARCO TEÓRICO.....	4
3.1 Conocimientos básicos de automatización.....	4

3.2Comunicaciones industriales:.....	5
3.3Sistemas industriales de Control .....	6
3.3.1Sistemas de control centralizado .....	7
3.3.2Sistema de control distribuido.....	7
3.3.3Sistema de control híbrido.....	7
3.4La Pirámide CIM (Computer Integrated Manufacturing) .....	8
3.4.1Nivel de E/S (Nivel actuador/sensor).....	8
3.4.2Nivel de Campo y Proceso .....	9
3.4.3Nivel de Control (Nivel de célula) .....	9
3.4.4Nivel de gestión (Nivel de fábrica) .....	9
3.5Buses de comunicación industrial. ....	10
3.5.1Bus de Campo Propietario.....	12
3.5.2Bus de Campo Abierto .....	12
3.5.3Buses existentes en el mercado .....	13
3.6Protocolo de comunicación .....	14
3.6.1 Hart .....	14
3.6.2Foundation Fieldbus. ....	14
3.6.3Interbus.....	14
3.6.4Profibus. ....	15
3.6.5Compobus.....	15
3.6.6Modbus.....	15
3.6.7Versiones del modbus .....	15

3.6.8ModBus RTU características.....	16
3.6.9Trama RTU.....	17
3.6.10Mapa de direcciones Modbus.....	18
3.7Sistema SCADA (Supervisory Control and Data Acquisition).....	19
3.7.1Funciones y requisitos .....	19
3.7.2Componentes de hardware del SCADA.....	20
3.7.2.1Master Terminal Unit (MTU).....	20
3.7.2.2Remote Terminal Unit, RTU.....	20
3.7.3Red de comunicación del SCADA.....	21
3.7.4Instrumentos de campo.....	21
3.8HMI (Interfaz Hombre-Máquina) O MMI (Man-Machine Interface).....	21
3.9Interface OPC (Comunica el PLC y la computadora).....	22
3.10PLC (Controlador lógico programable) .....	23
3.11Instrucciones para la programación en los PLC .....	23
3.11.1BLOQUE MB_COMM_LOAD.....	23
3.11.2BLOQUE MB_MASTER .....	25
3.11.3BLOQUE MB_SLAVE.....	27
3.12Métodos de acceso al medio.....	29
3.12.1Token bus .....	29
3.12.2Maestro-esclavo.....	30
3.13TIPOS DE TOPOLOGÍAS .....	30
3.13.1Bus.....	30

3.13.2Árbol.....	31
3.13.3Anillo.....	32
3.13.4Estrella.....	33
CAPITULO III.....	34
IMPLEMENTACIÓN DE UNA RED MODBUS.....	34
4.1Operación del sistema SCADA.....	34
4.1.1Estación de temperatura (Esclavo 1).....	35
4.1.2Estación de caudal (Esclavo 2).....	35
4.2Selección de Hardware.....	35
4.2.1Controlador lógico programable.....	35
4.3Selección de software.....	36
4.3.1Interfaz hombre máquina (HMI).....	36
4.4Creación del proyecto.....	36
4.5Configuración de dispositivos.....	37
4.6Creación de la red Modbus.....	38
4.7Creación de la red profinet.....	40
4.8Programación del PLC maestro.....	43
4.9Programación del esclavo 1:.....	51
4.10Programación del esclavo 2.....	55
4.11 HMI.....	60
4.11.1NI OPC Server.....	60
4.11.2HMI.....	61

4.12 Diagrama de conexiones eléctricas de los PLC's.....	62
CAPITULO IV .....	63
CONCLUSIONES Y RECOMENDACIONES .....	63
5.1 Conclusiones .....	63
5.2 Recomendaciones .....	63
REFERENCIAS BIBLIOGRÁFICA .....	64

## ÍNDICE DE TABLAS

<b>Tabla 1.</b> Trama del mensaje RTU .....	17
<b>Tabla 2.</b> Estructura de los datos en Modbus.....	18
<b>Tabla 3</b> Parámetros del bloque MB_MASTER.....	27
<b>Tabla 4</b> Parámetros del Bloque MB_SLAVE .....	28
<b>Tabla 5</b> Función de lectura y escritura que realiza el maestro .....	44

## ÍNDICE DE FIGURAS

Figura 1. Control de lazo abierto.....	4
Figura 2 Control de lazo cerrado.....	5
Figura 3 Evolución de los Procesos Industriales.....	6
Figura 4. Pirámide CIM de comunicación .....	8
Figura 5 Bus de campo abierto PROFIBUS .....	13
Figura 6 Buses de Comunicación.....	13
Figura 7. Ejemplo de una HMI de un Sistema SCADA .....	22
Figura 8 Bloque MB_COMM_LOAD.....	25
Figura 9 Bloque MB_MASTER .....	27
Figura 10 Bloque MB_SLAVE.....	28
Figura 11 Metodología token bus .....	29
Figura 12 Métodos de transmisión de datos en una red industrial.....	30
Figuran 13 Topologías .....	31
Figura 14 Profinet .....	31
Figura 15 Tipología Árbol .....	32
Figura 16 Tipología Anillo.....	33
Figura 17 Repetidor.....	34
Figura 18 Tipología Estrella.....	34
Figura 19 Crear proyecto .....	36
Figura 20 Seleccionar Dispositivo .....	37
Figura 21 Activación de bits de marca de sistema y ciclo del maestro.....	37

Figura 22 Activación de bits de marca de sistema del esclavo 1 .....	38
Figura 23 Activación de los bits de marca de sistema del esclavo 2 .....	38
Figura 24 Selección del módulo CM 1241. ....	39
Figura 25 Maestro y CM 1241 .....	39
Figura 26 Esclavo 1 y CM 1241 .....	40
Figura 27 Esclavo 2 y CM 1241 .....	40
Figura 28 Red profinet del proyecto técnico.....	41
Figura 29 Asignación IP al PLC 1 .....	41
Figura 30 Asignación IP al PLC 2 .....	42
Figura 31 Asignación IP al PLC 3 .....	42
Figura 32 Asignación IP de la PC .....	43
Figura 33 Creación del bloque .....	43
Figura 34 Bloque de Datos del Maestro.....	44
Figura 35 Bloque del sistema del Maestro.....	44
Figura 36 Configuración del Maestro Modbus RTU .....	45
Figura 37 Instrucción para recibir datos del esclavo 1 .....	46
Figura 38 Instrucción para escribir datos al esclavo 1 .....	46
Figura 39 Instrucción para recibir datos del esclavo 2.....	47
Figura 40 Instrucción para escritura de datos del esclavo 2.....	47
Figura 41 Contador para la secuencia de comunicación Modbus.....	48
Figura 42 Bit que activa la lectura del esclavo 1 .....	48
Figura 43 Bit que activa la escritura del esclavo 1.....	48

Figura 44 Bit que activa la lectura del esclavo 2 .....	49
Figura 45 Bit que activa la escritura del esclavo 2.....	49
Figura 46 Escalamiento de valores a ser enviados al esclavo 1 .....	49
Figura 47 Escalamiento de valores a ser enviados al esclavo 2.....	50
Figura 48 No permite valores mayores a 100 .....	50
Figura 49 No permite valores menores a 0 .....	50
Figura 50 Encendido y apagado.....	51
Figura 51 PEM y Reinicio.....	51
Figura 52 Bloques de sistema del esclavo 1.....	51
Figura 53 Bloque de datos del esclavo 1.....	52
Figura 54 Dirección de salida analógica del canal 0.....	52
Figura 55 Configuración del esclavo 1 .....	53
Figura 56 Asignación de dirección del esclavo 1.....	53
Figura 57 Escalamiento de la entrada analógica 0 del esclavo 1 .....	54
Figura 58 Escalamiento para convertir a grados centígrados.....	54
Figura 59 Valor de temperatura enviado al maestro .....	55
Figura 60 Recepción de datos desde el maestro.....	55
Figura 61 Bloque de datos del esclavo 2.....	56
Figura 62 Bloque de sistema del esclavo 2 .....	56
Figura 63 Salida analógica del esclavo 2 .....	57
Figura 64 Configuración del esclavo 2 .....	57
Figura 65 Asignación de dirección del esclavo 2.....	58

Figura 66 Resta la entrada del sensor de caudal .....	58
Figura 67 Resultado de la ecuación .....	59
Figura 68 Valor de caudal enviado al maestro.....	59
Figura 69 Recepción de datos desde el maestro.....	60
Figura 70 Tags para el HMI.....	60
Figura 71 HMI en Labview.....	61
Figura 72 Programacion del HMI .....	61
Figura 73 Esquema de conexiones de la PCT-2.....	62
Figura 74 Esquema de conexión de la estación de caudal. ....	62

## **RESUMEN**

El siguiente proyecto técnico fue realizado con la finalidad de obtener el título de Tecnólogo en Electrónica mención Instrumentación y Aviónica con la implementación de una red modbus con acceso al medio maestro - esclavo utilizando un PLC S7 1200 con CPU 1215C y módulos de comunicación modbus 485.

La red modbus cuenta con tres PLCs uno designado como el maestro y los dos restantes como esclavos que controlan de forma independiente una estación de temperatura y una estación de flujo de caudal.

En el primer proceso, el maestro inicia la comunicación enviando un set point de temperatura que recibe desde el HMI al esclavo 1 (estación de temperatura), el cual envía las ordenes a la niquelina y a su vez el sensor de temperatura envía al maestro la variación de temperatura real para visualizarla en el HMI.

En el segundo proceso, el maestro inicia la comunicación enviando un set point de porcentaje de bombeo recibido desde el HMI al esclavo 2 (estación de flujo de caudal), enviando las órdenes indirectamente a la bomba a través del variador de velocidad al cual está conectado, con el objetivo de controlar el flujo de caudal que pasará por la tubería. A la vez el transmisor indicador de flujo envía al maestro el valor del flujo instantáneo que circula por la tubería.

Al iniciar el sistema SCADA se verificó que los datos en el HMI sean los mismos que se visualizan en el proceso.

### **PALABRAS CLAVES:**

- **REDES DE COMUNICACIÓN INDUSTRIAL**
- **MODBUS**
- **SISTEMA SCADA**
- **HMI**
- **MAESTRO ESCLAVO**

## ABSTRACT

The present study was carried out in order to obtain the title of Instrumentation and Avionics Electronics Technologies with the implementation of a modbus network with access to the master-slave medium using a PLC S7 1200 with CPU 1215C and Modbus 485 communication modules.

The Modbus network has three PLCs, one designated as the master and the other two as slaves that independently control a temperature station and a flow station.

In the first process, the master initiates communication by sending a temperature set point of temperature that receives from the HMI to the slave 1 (temperature station), which sends the commands to the nickel and at the same time the temperature sensor sends the variation to the master of real temperature to display it in the HMI.

In the second process, the master initiates the communication by sending a set point of percentage point of pumping received from the HMI to slave 2 (flow station), sending the commands indirectly to the pump through the variable speed drive to which it is connected, with the objective of controlling the flow of flow that will pass through the pipeline. At the same time, the flow indicator transmitter sends to the master the value of the instantaneous flow that circulates through the pipe.

When starting the SCADA system, it was verified that the data in the HMI is the same as that displayed in the process.

### KEYWORDS:

- **INDUSTRIAL COMBINATION NETWORKS**
- **MODBUS**
- **SCADA SYSTEM**
- **HMI**
- **SLAVE MASTER**

---

**CHECKED BY:**

**MARÍA ELISA COQUE**

**ENGLISH TEACHER UGT**

# CAPITULO I

## PLANTEAMIENTO DEL PROBLEMA

IMPLEMENTACIÓN DE UNA RED MODBUS CON UN MAESTRO Y DOS ESCLAVOS UTILIZANDO UN PLC-S7-1200-CPU-1215C CON MÓDULOS DE COMUNICACIÓN MODBUS 485. PARA PRÁCTICAS DE REDES INDUSTRIALES

### 1.1 Antecedentes

Según (De Francesch Saavedra, 2015) quien desarrollo el proyecto de grado con el tema “Implementación de una red de supervisión para riego mecanizado con protocolo Modbus, PLCs y Matlab”, explica a detalle el protocolo Modbus, la forma en que se construye una trama de comunicación y la sintaxis del mismo, además se usan equipos de automatización, ejemplo de ellos son los PLCs. Todo esto basado en el proyecto de irrigación Agrolmos. Finalmente, el estudio ha permitido la integración de varios sistemas para implementar el actual sistema de control de riego del proyecto Agrolmos asegurando una buena producción y rentabilidad de estos terrenos.

El según trabajo revisado de (Marín Iturrarte, 2015) quien desarrollo el proyecto de grado con el tema “Control de PLCs Siemens S7-1200 mediante el protocolo MODBUS a través del programa LABVIEW para realización de prácticas de comunicación industrial” Se desarrolla una aplicación en la cual se extrae información de un encoder que controla la velocidad de un motor. La adquisición de datos es en tiempo real, ya que el dato que proporciona el encoder es la velocidad del motor en el momento que se observa dicha variable. Modbus TCP cumple con los requisitos que se desean para la aplicación sobre la que se acaba de describir.

El último trabajo revisado de (Ojeda Sosa, 2017) quien desarrollo el proyecto de grado con el tema “Diseño, Construcción y Automatización de Sistema de Etiquetado de Envases Pet para Proceso de Producción de Bebidas en la Empresa Yoguis S.R.L”. Para el funcionamiento del sistema de transporte se considera necesario el uso de un motor reductor y un variador de frecuencia el cual pueda tener enlace con un PLC (Controlador Lógico Programable) a través de protocolos de comunicación, a fin de poder controlar la velocidad de la producción a través de un HMI, PC o cualquier

dispositivo Android (ya sea Tablets y/o smartphones) mediante diversas aplicaciones móviles disponibles en la actualidad. Para el proceso de etiquetado, se considera el uso de un motor paso a paso, el cual será comandado por el PLC mediante funciones avanzadas (tren de pulsos) y a través de un driver de control.

## **1.2 Planteamiento del problema**

La tecnología evoluciona constantemente, actualmente el laboratorio de Instrumentación de la Unidad de Gestión de Tecnologías “ESPE”, ya que no cuenta con PLCs actualizados para la realización de prácticas de redes industriales.

Por tanto, es necesario que el laboratorio de instrumentación cuente con estos equipos con la finalidad de preparar profesionales capaces de solucionar problemas en el campo de la industria y mejorar su eficiencia en la misma.

## **1.3 Justificación**

Los beneficiarios del presente proyecto serán los alumnos de la Unidad de Gestión de Tecnologías-ESPE, directamente los estudiantes de la carrera de Electrónica Mención Instrumentación y Aviónica, fortaleciendo los conocimientos teóricos y prácticos adquiridos en las aulas, permitiendo alcanzar un nivel de competitividad en el ámbito profesional, formando académicamente mano de obra calificada y cualificada para así aportar al desarrollo del país.

El presente proyecto tiene la finalidad permitir que los nuevos estudiantes conozcan y fortalezcan sus conocimientos en el área de protocolos de comunicación y así puedan dar soluciones rápidas y eficaces en la industria convirtiéndose en profesionales capacitados y puedan competir con fortalezas en el campo laboral.

## **1.4 Objetivos**

### **1.4.1 Objetivo general**

Implementar una red modbus con un maestro y dos esclavos utilizando un plc-s7-1200-cpu-1215c con módulos de comunicación modbus 485. Para prácticas de redes industriales

### 1.4.2 Objetivos específicos

- Comprobar que las estaciones que ya están realizadas las cuales serán los esclavos tengan Una buena comunicación entre el plc y los transmisores.
- Habilitar y configurar la red modbus.
- Adquirir la respuesta de los esclavos cuando el maestro lo requiera.
- Realizar un HMI en labview para el monitoreo de los transmisores que tienen los esclavos.

### 1.5 Alcance

Este proyecto tiene como finalidad de habilitar una red MODBUS con un maestro que dará órdenes a los dos esclavos quienes responderán solo cuando el maestro lo requiera se y se creara un HMI en labview para monitorear a las estaciones ya programadas y sacar un registro de las variables medidas en los esclavos, este proyecto servirá para que los estudiantes de la unidad de gestión de tecnologías ESPE puedan tener un conocimiento más amplio en el campo de lo que comprende las redes industriales.

Los módulos de comunicación modbus serán prestados por la universidad y los PLC S7-1200-CPU-1215C serán donados al Laboratorio de Instrumentación de la UNIDAD DE GESTIÓN DE TECNOLOGÍAS de la ESPE para fines académicos.

## CAPITULO II

### MARCO TEÓRICO

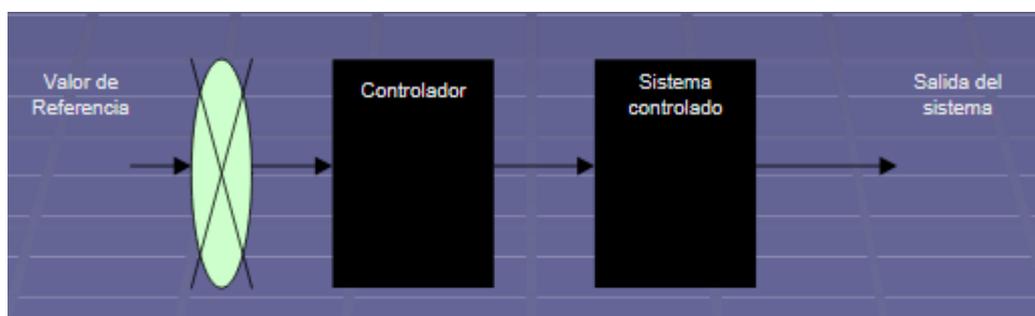
#### 2.1 Conocimientos básicos de automatización

**Automatizar:** Consiste en aplicar un procedimiento automático a un aparato, proceso o sistema, convirtiendo ciertos movimientos corporales en movimiento automático. (Salinas, 2011)

**Automático:** Disciplina que trata de los métodos y procedimientos cuya finalidad es la sustitución del operador humano por un operador artificial en la ejecución de una tarea física o mental previamente programada. (Moreno, 2007)

**Sistema de control:** Un sistema de control manipula indirectamente los valores de un sistema controlado. Su objetivo es gobernar un sistema sin que el operador intervenga directamente sobre sus elementos. El operador manipula valores de referencia y el sistema de control se encarga de transmitirlos al sistema controlado a través de los accionamientos de sus salidas. (Moreno, 2007)

**Sistemas de Control de Lazo Abierto:** Los sistemas de control de lazo abierto son aquellos en los que la salida no tiene efecto sobre la acción del controlador, es decir, la salida ni se mide ni se realimenta para compararla con la entrada. Para cada valor de referencia corresponde una condición de operación fijada. (Moreno, 2007)



**Figura 1. Control de lazo abierto**

Fuente: (Moreno, 2007)

**Sistemas de Control de Lazo Cerrado:** Un sistema de control de lazo cerrado es aquel en el que la señal de salida tiene efecto directo sobre la acción del controlador. La señal de error actuante (que es la diferencia entre la señal de entrada y la realimentación) entra al control para reducir el error y llevar la salida del sistema al valor deseado. (Moreno, 2007)



**Figura 2 Control de lazo cerrado**

Fuente: (Moreno, 2007)

**La variable controlada:** ésta es la variable que se debe mantener o controlar dentro de un valor deseado (Salinas, 2011)

**Variable manipulada:** ésta es la variable que utiliza el sistema para mantener en el punto de control a la variable controlada. (Salinas, 2011)

## 2.2 Comunicaciones industriales:

La aparición de los autómatas programables marcó un cambio en la industria de ensamblaje y líneas de producción. El concepto de línea de producción automatizada apareció entonces como un hecho real. Este pequeño dispositivo permitió que los procesos industriales fuesen desde entonces más eficientes, precisos, y lo que es más importante, reprogramables, eliminando el gran costo que se producía al reemplazar el complejo sistema de control basado en relés y contactores, tanto por tamaño como por vida útil. (Torres Hurtado, 2015)



**Figura 3 Evolución de los Procesos Industriales**

Fuente: (Torres Hurtado, 2015)

Pero la mejora en los procesos de automatización pasa hoy en día por el desarrollo de las redes de comunicación. La intercomunicación de sistemas y procesos industriales no es un concepto nuevo, pues es ampliamente conocido el uso de sistemas como IEEE-488 y RS485/422 que durante más de 20 años han sido capaces de ofrecer los requerimientos necesarios en las instalaciones de baja y media complejidad en cuanto a las capacidades de intercomunicación se refiere. Este tipo de enlaces entre sistemas que han empleado esencialmente para equipos de instrumentación y sistemas de automatización, donde es necesaria una baja tasa de transferencia de datos entre equipos, pero que en gran número de casos hoy en día ya no puede responder a las necesidades de intercomunicación entre dispositivos que se demandan. Por ello, redes de comunicación como PROFIBUS, PROFINET y HART han conseguido ser indispensables en un entorno de trabajo donde cada día es más necesaria la integración global. (Torres Hurtado, 2015)

### 2.3 Sistemas industriales de Control

Se distinguen tres tipos de sistemas de control industrial: control centralizado, control híbrido y control distribuido. La importancia de las áreas a realizar, o la posibilidad de subdividir la tarea de control del proceso o conjunto de máquinas en esas funciones autónomas, determinará en muchos casos la elección de un tipo u otro de control. (Torres Hurtado, 2015)

### **2.3.1 Sistemas de control centralizado**

Esta aproximación es la que se sigue en el caso de sistemas poco complejos ,donde un proceso puede ser gestionado directamente mediante un único elemento de control encargado de realizar todas las tareas del proceso de producción y que puede incluir un sistema de monitorización y supervisión. Conforme las necesidades de producción han requerido mayor complejidad, una tendencia ha sido la de emplear elementos de control más complejos y potentes, manteniendo en un único elemento todo el control del proceso, con la complejidad que ello supone, ya que se hace necesario hacer llegar todas las señales de sensores y cablear todos los actuadores allá donde se encuentren. (Moreno, 2007)

### **2.3.2 Sistema de control distribuido**

La opción de control distribuido requiere que puedan considerarse procesos, grupos de procesos o áreas funcionales susceptibles de ser definidas por un algoritmo de control que pueda realizarse de forma autónoma. A cada unidad se destinará un autómatas (o elemento de control) dimensionado de acuerdo con los requerimientos del proceso considerado. Debido a interdependencia que existe entre las operaciones que tienen lugar en cada proceso, hay que tener en cuenta que es necesario interconectar los autómatas entre sí mediante entradas y salidas digitales, o a través de una red de comunicaciones para intercambio de datos y estados. Por tanto, el autómatas o elemento de control evaluado debe permitir las comunicaciones. (Torres Hurtado, 2015)

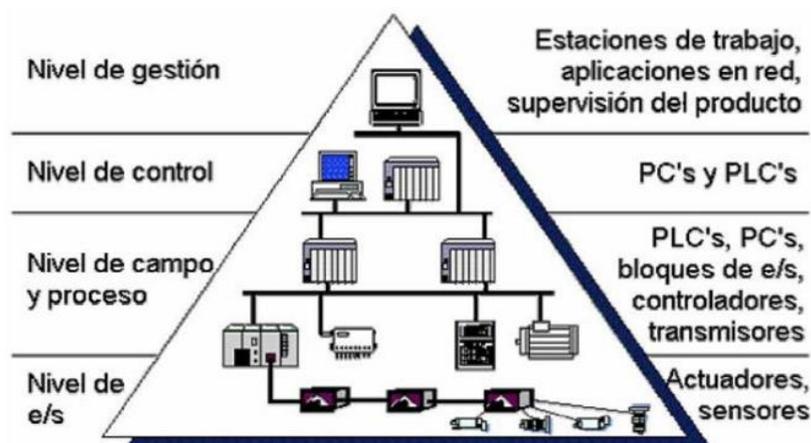
### **2.3.3 Sistema de control híbrido**

El control híbrido no está muy bien definido, ya que este tipo de gestión de planta puede considerarse a cualquier estrategia de distribución de elementos de control a medio camino entre el control distribuido y el control centralizado. En numerosas ocasiones no resulta sencillo separar los procesos de manera completamente autónoma, por lo que se debe recurrir a la gestión de varios procesos desde una misma unidad de control, debido a que la complejidad de la separación es mayor que la complejidad que supone su gestión conjunta. Por otro lado, una estrategia de este tipo también conduce a una gestión estructurada, de modo que existen elementos de control

de nivel superior que supervisan e intercomunican los procesos autónomos más sencillos, siendo los encargados de gestionar la información común. Para este tipo de gestión también es necesario el uso de redes de comunicación. (Salinas, 2011)

## 2.4 La Pirámide CIM (Computer Integrated Manufacturing)

El ideal de factoría completamente automatizada (Computer Integrated Manufacturing) se representa como una pirámide en la que en los niveles bajos se encuentran los sensores y actuadores; en los niveles intermedios se interconectan estos elementos para funcionar cooperativamente realizando funciones más o menos sincronizadas y finalmente, en el nivel superior aparece la red informática técnico-administrativa donde se recogen informaciones de estado, registros históricos, datos de partida, consignas, etc. (Torres Hurtado, 2015)



**Figura 4. Pirámide CIM de comunicación**

Fuente: (Torres Hurtado, 2015)

### 2.4.1 Nivel de E/S (Nivel actuador/sensor)

También llamado nivel de instrumentación. Está formado por los elementos de medida (sensores) y actuadores) distribuidos en una línea de producción. Son los elementos más directamente relacionados con el proceso productivo. Así, los actuadores son los encargados de ejecutar las órdenes de los elementos de control para modificar el proceso productivo y los sensores miden variables en el proceso de producción. Ejemplos de sensores son aquellos que permiten medir nivel de líquidos,

caudal, temperatura, presión, posición, etc. Como ejemplos actuadores se tienen los motores, válvulas, calentadores, taladros, cizallas, etc. (Serna Salazar, 2011)

#### **2.4.2 Nivel de Campo y Proceso**

En este nivel se sitúan los elementos capaces de gestionar los actuadores y sensores del nivel anterior, tales como autómatas programables o equipos de aplicación específica basados en microprocesador como robots, máquinas herramienta o controladores de motor. Estos dispositivos permiten que los actuadores y sensores funcionen de forma conjunta para ser capaces de controlar el proceso industrial deseado. (Vicente, 2016)

#### **2.4.3 Nivel de Control (Nivel de célula)**

Todos los dispositivos de control existentes en planta es posible monitorizarlos si existe un sistema de comunicación adecuado, capaz de comunicar estos elementos con otros tipo de dispositivos dedicados a la gestión y supervisión, que habitualmente están constituidos por computadores o sistemas de visualización tales como pantallas industriales. En este nivel es posible visualizar cómo se están llevando a cabo los procesos de planta y, a través de entornos SCADA (Supervisión, Control y Adquisición de Datos), poseer una “imagen virtual de la planta” de modo que ésta se puede recorrer de manera detallada, o bien mediante pantallas de resumen ser capaces de disponer de un “panel virtual” donde se muestren las posibles alarmas, fallos o alteraciones en cualquiera de los procesos que se llevan a cabo. (Vicente, 2016)

#### **2.4.4 Nivel de gestión (Nivel de fábrica)**

El nivel de gestión estará principalmente constituido por computadores, ya que se encuentra más alejado de los procesos productivos. De hecho, en este nivel no es relevante el estado y la supervisión de los procesos de planta, en cambio, sí adquiere importancia toda la información relativa a la producción y su gestión asociada, es decir, a través del nivel de control es posible obtener información global de todos los niveles inferiores de una o varias plantas. Con esta información, los gestores de la empresa pueden extraer estadísticas acerca de los costes de fabricación, rendimiento de la planta, estrategias de ventas para liberar

posibles excesos de producto almacenado y, en general, disponer de datos que permitan a los niveles directivos la toma de decisiones conducentes a una mejor optimización en el funcionamiento de la planta, todo ello de una manera rápida y flexible. Las comunicaciones con este nivel de la pirámide industrial ya no necesitan ser de tipo estrictamente industrial, es decir, muy robustas, de corto tiempo de respuesta, sino que ahora lo verdaderamente importante son los datos que se transmiten, informes que pueden tener un tamaño medio-grande. Por este motivo habitualmente se emplean redes de comunicación menos costosas, como redes Ethernet, que se adaptan mejor al tipo de datos que se desean transmitir y, además, permiten la comunicación eficaz entre los diferentes computadores del mismo nivel de gestión. (Torres Hurtado, 2015)

## **2.5 Buses de comunicación industrial.**

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)

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 controladores, de manera que la información que ellos puedan brindar pueda recorrer el sistema de información de toda la planta. (Villajulca, 2010)

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: (Villajulca, 2010)

- 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. (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. (Villajulca, 2010)
- Interoperabilidad: es la habilidad para la conexión de diversos elementos de diversos fabricantes, (Villajulca, 2010)
- Intercambio: los equipos de un fabricante pueden ser reemplazados con una funcionalidad equivalente por equipos de otros fabricantes.

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.
- 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. (Villajulca, 2010)
- 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.
- Las modificaciones futuras, actualizaciones o ampliaciones son fácilmente implementadas y a un costo menor. (Villajulca, 2010)
- 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. (Villajulca, 2010)
- 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. (Villajulca, 2010)

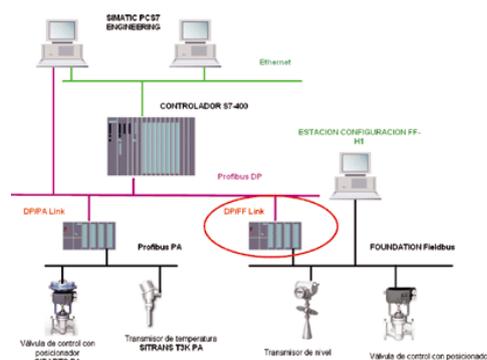
### 2.5.1 Bus de Campo Propietario

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. 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. (Villajulca, 2010)

### 2.5.2 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)

En la actualidad existen muchas opciones. Algunos fabricantes han desarrollado sistemas basados en la tecnología disponible (como la interfaz RS-232 o 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 y HART. (Villajulca, 2010)



**Figura 5 Bus de campo abierto PROFIBUS**

Fuente: (Villajulca, 2010)

### 2.5.3 Buses existentes en el mercado

BUS DE CAMPO	TOPOLOGÍA	MEDIO FÍSICO	VELOCIDAD	DISTANCIA SEGMENTO	NODOS POR SEGMENTO	ACCESO AL MEDIO
P-NET	Anillo	Par trenzado apantallado	768 Kbps	1.200 m	125	Paso de testigo Maestro/esclavo
PROFIBUS	Bus lineal Anillo Estrella Árbol	Par trenzado apantallado Fibra óptica	Hasta 12Mbps	Hasta 9'6 Km y 90 Km	125	Paso de testigo Maestro/esclavo
WORLDFIP	Bus lineal	Par trenzado apantallado Fibra óptica	Hasta 1 Mbps y 5Mbps	Hasta 3 Km y 20 Km	64	Arbitro de bus
HART	Bus lineal	Cable 2 hilos	1'2Kbps	3.000 m	30	Maestro/esclavo
MODBUS	Bus lineal	Par trenzado	Hasta 19'2Kbps	1 Km	248	Maestro/esclavo
INTERBUS-S	Anillo	Par trenzado	500 Kbps	400 m	256	Paso de testigo
BITBUS	Bus lineal	Par trenzado Fibra óptica	Hasta 1'5Mbps	Hasta 1.200m	29	Maestro/esclavo
CAN	Bus lineal	Par trenzado	Hasta 1 Mbps	Hasta 1.000m	127-64	CSMA/CD con arbitraje de bit
SDS	Bus lineal	Cable de 4 hilos	Hasta 1 Mbps	500 m	64	CSMA
DEVICENET	Bus lineal	Par trenzado	Hasta 500 Kbps	Hasta 500 m	64	CSMA/CDMA
CONTROLNET	Bus lineal Árbol Estrella	Coaxial Fibra óptica	5 Mbps	Hasta 3.000m	48	CTDMA
SERIPLEX	Bus lineal	Cable 4 hilos apantallado	98 Kbps	1.500m	300	Maestro/esclavo
AS-i	Bus lineal Árbol - Estrella	Cable 2 hilos	167 Kbps	Hasta 200 m	32-62	Maestro/esclavo
LON WORKS	Bus Anillo Libre	Par trenzado Fibra óptica Red eléctrica Coaxial Radio Infrarrojos	Hasta 1'25 Mbps	Hasta 2.700 m	64	CSMA/CA
ARCNET	Bus Estrella	Par trenzado Fibra óptica Coaxial	2'5 Mbps	122 m	255	Paso de testigo
M-BUS	Bus lineal	Cable 2 hilos	Hasta 9'6 Kbps	1.000 m	250	Arbitro de bus
UNI-TELWAY	Bus lineal	Par trenzado apantallado	Hasta 19'2Kbps	20 m	Hasta 28	Maestro/esclavo
COMPOBUS/S	Bus lineal	Cable de 2 ó 4 hilos	Hasta 750 Kbps	Hasta 500 m	32	Maestro/esclavo

A continuación se muestra una tabla comparativa entre los buses de campo de más utilización en la industria, haciendo referencia a sus características técnicas más importantes. (Serna Salazar, 2011)

**Figura 6 Buses de Comunicación**

Fuente: (Serna Salazar, 2011)

## **2.6 Protocolo de comunicación**

### **2.6.1 Hart**

Este protocolo proporciona comunicación digital bidireccional con dispositivos de campo inteligentes mientras conserva la compatibilidad y familiaridad de los tradicionales sistemas 4-20 mA. Su protocolo utiliza la norma Bell 200 que permite la superposición simultánea a niveles bajos de una señal de comunicaciones digital (el "1" lógico es representado por 1.200 Hz, mientras el "0" lógico corresponde a 2.200 Hz) en la parte superior de la señal analógica 4-20 mA. Ya que HART fue diseñado para ampliar las comunicaciones con los instrumentos de medición y control que tradicionalmente se comunicaban con señales de 4-20 mA, es aplicable a todo tipo de industrias de proceso. Por tratarse de una superposición a un sistema existente, HART ofrece una solución sin ningún riesgo para poder gozar de los beneficios que resultan de una comunicación más amplia con los dispositivos inteligentes (Rosado, 2003)

### **2.6.2 Foundation Fieldbus.**

Su principal objetivo es la sustitución del cableado asociado a los elementos aislados tales como aquellos dispositivos funcionando con tecnología 4 a 20 mA por un bus capaz de proporcionar una compatibilidad con ellos mediante la inclusión en el dispositivo de un pequeño interfaz. Es intrínsecamente seguro, es decir, dispone de capacidad para ser empleado en entornos agresivos (riesgo de explosión, agentes químicos, etc.). Esta tecnología emplea las capas 1, 2 y 7 del modelo OSI. Las señales del bus se transmiten por un par trenzado con una amplitud de 1V de pico a pico moduladas sobre una tensión continua de valores entre 9 y 32 V (incluye la alimentación de los dispositivos en la misma línea), con una velocidad de transferencia de 31,25 kbit/s y con una impedancia equivalente de 50 ohmios (es necesario incluir terminadores en la línea). Permite transmisión gestionada por un dispositivo que otorga el control del bus (paso de testigo), y también transferencias asíncronas para mensajes con prioridad. (Rosado, 2003)

### **2.6.3 Interbus**

Es una red de sensores / accionadores distribuidos para sistemas de fabricación y control de procesos continuos. Es un sistema con método de comunicación maestro-

esclavo, abierto de alta prestaciones, de topología en anillo. Interbus no está respaldado por los grandes fabricantes de autómatas. Sin embargo, alrededor de 700 desarrolladores de dispositivos de campo lo soportan. Un sistema basado en Interbus está compuesto por una tarjeta de control, instalada en un PC industrial o en un autómata programable que comunica con un conjunto de dispositivos de entrada/salida. (Rosado, 2003)

#### **2.6.4 Profibus.**

La base para el desarrollo de este protocolo fue un proyecto de investigación de varias empresas y cinco institutos de investigación alemanes. Actualmente, Profibus en sus 3 versiones FMS, DP y PA son estándar europeo EN50170 desde 1996, aunque sus actividades comenzaron alrededor de 1987. Existen más de dos millones de dispositivos Profibus instalados, con aproximadamente 250 fabricantes de productos Profibus en todo el mundo. (Rosado, 2003)

#### **2.6.5 Compobus.**

Es el principal impulsor de este bus. Compobus D es el nombre que Omron emplea para su particularización de DeviceNet, donde existe fácil integrabilidad con cualquier dispositivo Omron, y además, es posible emplear otros elementos DeviceNet. Compobus se está orientado a la sustitución del cableado tradicional y los terminales múltiples. (Rosado, 2003)

#### **2.6.6 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). (Mejía Herrera, 2016)

#### **2.6.7 Versiones del modbus**

Existen dos versiones principales: MODBUS RTU para la comunicación serial convencional, y MODBUS/TCP, que emplea Ethernet como medio físico de transmisión. 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. (Cevallos, Noviembre 2007)

**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). (Cevallos, Noviembre 2007)

### 2.6.8 ModBus RTU características

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, Noviembre 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).

### **2.6.9 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. (Cevallos, Noviembre 2007)

#### **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: (Mejía Herrera, 2016)

### 2.6.10 Mapa de direcciones Modbus

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

**Salidas digitales (direcciones 00001-09999):** 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. (Mejía Herrera, 2016)

#### Tabla 2.

#### Estructura de los datos en Modbus

Tipo de Datos	Tipo de Acceso	Rango de Memoria
Salidas Digitales	Escritura	00001-09999
Entradas Digitales	Escritura / Lectura	10001-19999
Entradas Analógicas	Escritura	30001-39999

Registros de memoria

Escritura / Lectura

40001-49999

Fuente: (Mejía Herrera, 2016)

## 2.7 Sistema SCADA (Supervisory Control and Data Acquisition)

SCADA es un sistema de control, supervisión y adquisición de datos que consiste en un HMI para monitorear, supervisar además, incorpora un autómata programable para realizar el control de un proceso industrial. La comunicación se realiza mediante buses especiales o redes de área local. Todo esto se ejecuta normalmente en tiempo real, y están diseñados para dar al operador de planta la posibilidad de supervisar y controlar dichos procesos. Los sistemas SCADA han eliminado la necesidad de estar físicamente vigilando justando los componentes del proceso, una red de sensores transmite información del estado de los componentes a una sala de operadores que deciden si hay que realizar alguna modificación sobre el proceso. Muchas veces esta toma de decisiones está apoyada por una unidad central que descarga el operario de tareas repetitivas dejándole actuar sobre el sistema a muy alto nivel. (Serna Salazar, 2011)

### 2.7.1 Funciones y requisitos

Entre las funciones de un sistema SCADA es posible enumerar:

- Crea paneles de alarma, que exigen la presencia del operador para reconocer una parada o situación de alarma, con registro de incidencias.
- Generación de históricos de señal de planta, creando informes, avisos y documentación en general.102 N° 25 - julio - diciembre / 2011
- Ejecución de programas, que modifica la ley de control, o incluso anular o modificar las tareas asociadas al autómata bajo ciertas condiciones.
- Posibilidad de programación numérica, que permite realizar cálculos aritméticos de elevada resolución sobre la CPU del computador.

Adicionalmente, son requisitos necesarios para la implementación de SCADA:

- Deben ser sistemas de arquitecturas abiertas, capaces de crecer o adaptarse según las necesidades de la empresa.

- Deben comunicarse con total facilidad y de forma transparente al usuario con el equipo de planta y LAN de la empresa.
- Los programas deberán ser sencillos de instalar, sin excesivas exigencias de hardware, y fáciles de utilizar con interfaces amigables para el usuario (sonido, imágenes, pantallas táctiles). (Serna Salazar, 2011)

## **2.7.2 Componentes de hardware del SCADA**

### **2.7.2.1 Master Terminal Unit (MTU)**

Se trata de un computador principal el cual supervisa y recoge la información del resto de las subestaciones, estos computadores suelen ser un PC el cual soporta el HMI (Human Machine Interface). Las funciones principales del MTU son:

- Interrogar en forma periódica a los RTU, y les transmite consignas siguiendo usualmente un esquema maestro- esclavo.
- Actúa como interfaz al operador, incluyendo la presentación de información de variables en tiempo real, la administración de alarmas, y la recolección y presentación de información histórica.
- Puede ejecutarse software especializado que cumpla funciones específicas asociadas al proceso supervisado por el SCADA. (Serna Salazar, 2011)

### **2.7.2.2 Remote Terminal Unit, RTU**

Estos procesadores están situados en los nodos estratégicos del sistema gestionando y controlando subestaciones del sistema, reciben señales de los sensores de campo, y comandan los elementos finales de control ejecutando el software de la aplicación SCADA. Se encuentran en un nivel intermedio de automatización, a nivel superior está el MTU y a un nivel inferior los distintos instrumentos de campo que son los que ejercen la automatización física del sistema, control y adquisición de datos. Estos computadores no tienen que ser PC ya que la necesidad de soportar un HMI no es tan grande a este nivel suelen ser computadores industriales, aunque en sistemas muy complejos puede haber subestaciones en formato HMI. Una tendencia son los PLC (en función de E/S a gestionar) con la capacidad de funcionar como RTU gracias a un

nivel de integración mayor y CPU con mayor potencia de cálculos. (Serna Salazar, 2011)

### **2.7.3 Red de comunicación del SCADA**

En este nivel se gestiona la información que los instrumentos de campo envían a la red de computadores desde el sistema. El tipo de bus utilizado en las comunicaciones puede ser muy variado según las necesidades del sistema y del software escogido para implementar el SCADA, ya que no todos las aplicaciones de software (así como los instrumentos de campos como PLC) pueden trabajar con todos los tipos de bus. (Serna Salazar, 2011)

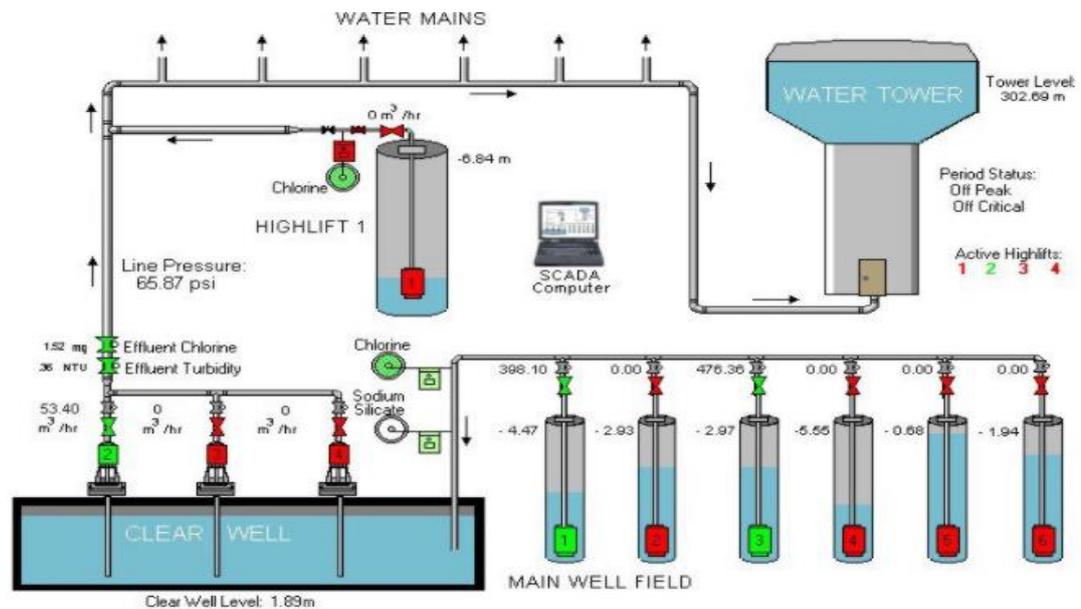
### **2.7.4 Instrumentos de campo.**

Son todos aquellos que permiten realizar la automatización o control del sistema (PLC, controladores de procesos industriales, y actuadores en general) como los que se encargan de la captación de información del sistema (sensores y alarmas). (Serna Salazar, 2011)

## **2.8 HMI (Interfaz Hombre-Máquina) O MMI (Man-Machine Interface)**

Es un mecanismo que le permite a un operador humano interactuar con una máquina o proceso y determinar el estado (prendido / apagado) o magnitud de los dispositivos y/o variables físicas que están presentes en una planta o proceso industrial. Una HMI puede ser tan simple como un interruptor para encender un motor y una lámpara indicadora del estado del mismo, hasta una o varias pantallas desarrolladas en una computadora que llegan a mostrar en la pantalla del monitor .Representaciones esquemáticas de todo el proceso bajo supervisión, incluyendo valores en tiempo real de las variables presentes en ese momento en la planta. Para manejar un sistema SCADA generalmente se recurre a un paquete de software especializado que funciona en la computadora central, por medio del cual se desarrolla una o varias “pantallas” que actúan como una interfaz gráfica entre el hombre y la máquina o el proceso. De esta forma es posible supervisar o cambiar puntos de consigna o reconfigurar dispositivos en el proceso supervisado por medio de acciones ingresadas por el operador en la computadora. Además, estos paquetes tienen opciones que permiten

proveer a un nivel superior administrativo información selecta que se genere en el proceso productivo. (CORRALES, 2007)



**Figura 7. Ejemplo de una HMI de un Sistema SCADA**

Fuente: (CORRALES, 2007)

Programas como el **InTouch** de la Wonderware, **Lookout** de la National Instruments, por mencionar algunos, constituyen plataformas de desarrollo abiertas que facilitan el diseño de las HMI en computadoras. Existen también plataformas propietarias como **RSView**, **WINCC**, que pertenecen a empresas de gran prestigio, lo que garantiza la solides de estos productos. Algunos de estos paquetes de desarrollo incluyen muchas herramientas poderosas que permiten el desarrollo de HMIs de mucho potencial de procesamiento. Si bien en la actualidad los SCADA incluyen tecnologías de comunicación modernas (CORRALES, 2007)

## 2.9 Interface OPC (Comunica el PLC y la computadora).

OPC (OLE for Process Control) de Microsoft es un interfaz con componentes de automatización, proporcionando un acceso simple a los datos. La Fundación OPC está formada por: Siemens, Fisher, Intuitive, OPTO 22, Intellution, Rockwell, etc. Las aplicaciones que requieren servicios, es decir datos, desde el nivel de automatización para procesar sus tareas, los piden como clientes desde los componentes de automatización, quienes a la vez proveen la información requerida como servidores.

La idea básica del OPC está en normalizar el interface entre el servidor OPC y el cliente OPC independientemente de cualquier fabricante particular. Implican la lectura, cambio y verificación de variables de proceso. Mediante estos servicios es posible operar y controlar un proceso. Los servidores OPC apoyan el nexo de tales aplicaciones a cualquier componente de automatización que esté en red por medio de un bus de campo o Ethernet Industrial. (Lozano Castro, 2001)

## **2.10 PLC (Controlador lógico programable)**

Programable Logic Controller (Controlador Lógico Programable) es un dispositivo digital electrónico con una memoria programable para el almacenamiento de instrucciones, permitiendo la implementación de funciones específicas como: lógicas, secuenciales, temporizadas, de conteo y aritméticas; con el objeto de controlar máquinas y procesos. También se puede definir como un equipo electrónico, el cual realiza la ejecución de un programa de forma cíclica. La ejecución del programa puede ser interrumpida momentáneamente para realizar otras tareas consideradas más prioritarias, pero el aspecto más importante es la garantía de ejecución completa del programa principal. Estos controladores son utilizados en ambientes industriales donde la decisión y la acción deben ser tomadas en forma muy rápida, para responder en tiempo real. Los PLC son utilizados donde se requieran tanto controles lógicos como secuenciales o ambos a la vez. (Moreno, 2007)

## **2.11 Instrucciones para la programación en los PLC**

La comunicación Modbus serializa a través de la programación de bloques específicos para éste uso, los mismos que serán especificados a continuación.

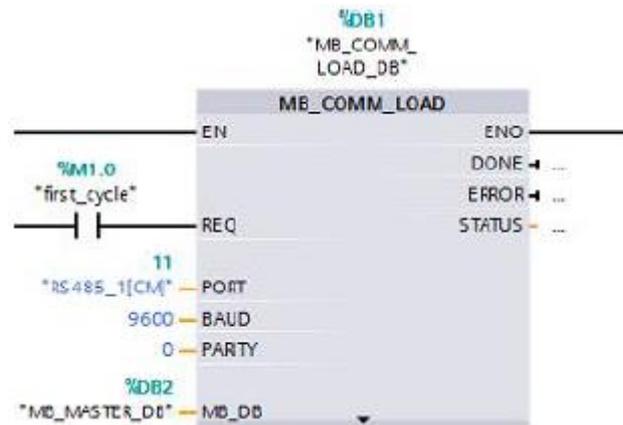
### **2.11.1 BLOQUE MB\_COMM\_LOAD**

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, ajustar los parámetros de la comunicación y parametrizar el enlace con los parámetros 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. (siemens, Controlador programable S7-1200, 2009)

- 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. (siemens, Controlador programable S7-1200, 2009)



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

Fuente: (TIA PORTAL , 2014)

### 2.11.2 BLOQUE 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 (siemens, Controlador programable S7-1200, 2009)

- 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). (siemens, Controlador programable S7-1200, 2009)

### Parámetro REQ

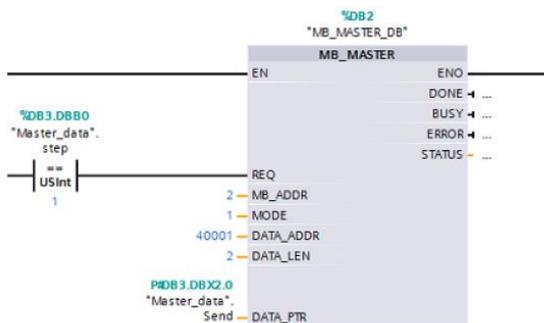
Si el valor de REQ es FALSE (falso), no hay ninguna petición.

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 , 2014)

**Tabla 3**

#### 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 , 2014)

#### 2.11.3 BLOQUE 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. (siemens, Controlador programable S7-1200, 2009)



**Figura 10 Bloque MB\_SLAVE**

Fuente: (TIA PORTAL , 2014)

**Tabla 4**

#### Parámetros del Bloque MB\_SLAVE

Parámetro	Descripción
MB_ADDR	Transfiere la dirección estación MODBUS-RTU.
MB_HOLD_REG	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 , 2014)

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.

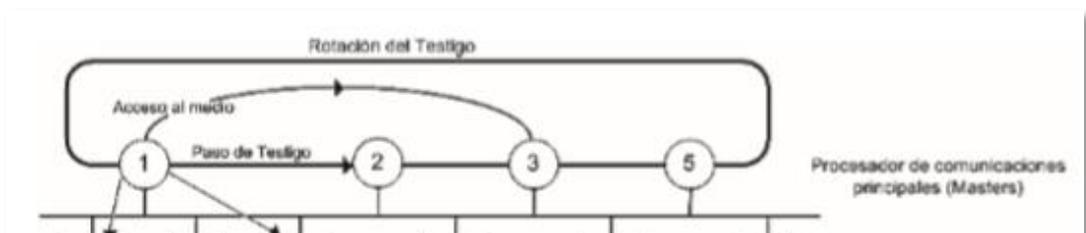
Los parámetros se pueden conmutar con las entradas y salidas de los bloques o mediante el acceso al correspondiente bloque de datos de instancia.

## 2.12 Métodos de acceso al medio

Para la transmisión de datos en redes industriales se ha determinado dos métodos utilizados dependiendo de la comunicación requerida. Existe el método de paso de testigo conocido comúnmente como token bus y el método maestro-esclavo, estos métodos evitan que ocurran colisiones de información. (Guerrero, Yuste, & Martínez, 2010)

### 2.12.1 Token bus

Se desarrolla en topologías en bus, conectadas en estructura de anillo. Básicamente su funcionamiento consiste en rotar una trama de datos (testigo) dentro del anillo en un orden predeterminado y cada dispositivo va a tener su turno de enviar una trama de vuelta para que sea ejecutada la acción, cada dispositivo tiene un tiempo de comunicación máximo 10ms, si el nodo no tiene ningún dato que



**Figura 11 Metodología token bus**

transmitir pasara el siguiente en el anillo y orden correspondiente. Si algún nodo no transmite lo detecta y declara como inactiva. (Bonifas & Flores, 2013)

Fuente: (Bonifas & Flores, 2013)

### 2.12.2 Maestro-esclavo

Se utiliza cuando se tiene una red industrial donde un maestro controla varios esclavos, se caracteriza por que el flujo de datos es unidireccional maestro-esclavo pero la información es bidireccional. El maestro envía y recibe información de los esclavos que permanecen siempre en espera para la comunicación. Este método se utiliza en sistemas de control centralizado o en los niveles bajos de pirámide de comunicación, es decir, en el nivel de campo y control. (Guerrero, Yuste, & Martínez, 2010)



**Figura 12 Métodos de transmisión de datos en una red industrial**

Fuente: (Guerrero, Yuste, & Martínez, 2010)

## 2.13 TIPOS DE TOPOLOGÍAS

### 2.13.1 Bus

Esta topología es la más sencilla y básica, en una línea de bus compartida por todos los nodos de la red. Para poder enviar información los nodos deben preguntar si el bus está libre. Un solo mensaje puede circular en el bus, debido a esto, este tipo de estructura es susceptible a las colisiones de red ya que si emiten dos o más mensajes

al mismo tiempo el sistema colapsa. Es de bajo costo y fácil de implementarla se la utiliza en control de flujos sencillos. (Guerrero, Yuste, & Martínez, 2010)

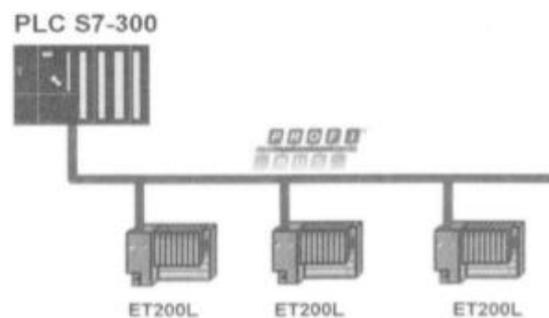
Se las aplica generalmente en redes industriales y redes LAN Ethernet (obsoleto).

En profibus para una estructura de este tipo se utiliza cables de cobre para la conexión, puede llegar a una distancia de 1200 metros de bus para una buena comunicación (Bonifas & Flores, 2013)



**Figuran 13 Topologías**

Fuente:(Bonifas & Flores, 2013)

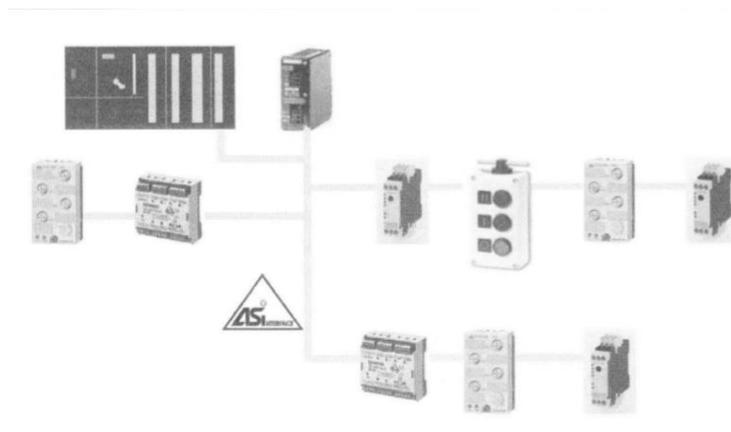


**Figura 14 Profinet**

Fuente:(Bonifas & Flores, 2013)

### 2.13.2 Árbol

Está formado por un grupo de buses conectados entre sí, dando lugar a una estructura arbórea. Al implementa este tipo de topología se logra mayor alcance a diferencia de un bus simple en línea recta, con la desventaja de incremento de la atenuación. Este tipo de red puedes ser utilizada para montar una red en una casa u oficina. (Guerrero, Yuste, & Martínez, 2010)



**Figura 15 Tipología Árbol**

Fuente:(Bonifas & Flores, 2013)

### 2.13.3 Anillo

Es una estructura cerrada en forma de anillo se caracteriza por:

- La información fluye en un único sentido.
- Puede estar circulando más de un mensaje por el anillo.
- Es fácil añadir otro nodo al anillo.
- Tiene una alta velocidad de transmisión.
- El control de esta estructura es sencilla tanto en hardware como software.
- No tiene colisiones

(Guerrero, Yuste, & Martínez, 2010)



**Figura 16 Tipología Anillo**

Fuente:(Bonifas & Flores, 2013)

#### 2.13.4 Estrella

“Todos los puestos de trabajo están conectados a un mismo nodo de la red, llamado concentrador o HUB (repetidor de la información).” (Guerrero, Yuste, & Martínez, 2010)

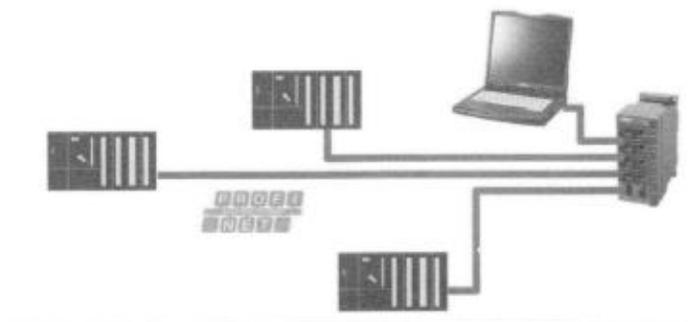
El nodo central es el que controla toda la transmisión e información de la red con la desventaja que si este llega a fallar, cae toda la red. Admite diferentes velocidades de transmisión y a su vez es posible añadir o suprimir nodos, gracias a su estructura facilita su mantenimiento. (Guerrero, Yuste, & Martínez, 2010)

En profibus para esta estructura se utiliza repetidores con un número máximo de estaciones por segmento de bus de 32 incluyendo el repetidor, “Se puede conectar mas segmentos a través de repetidores, considerado cada repetidor como una estación, hasta de 126 estaciones contando con todos los segmento de bus.” (Bonifas & Flores, 2013)



### Figura 17 Repetidor

Fuente:(Bonifas & Flores, 2013)



### Figura 18 Tipología Estrella

Fuente:(Bonifas & Flores, 2013)

## CAPITULO III

### IMPLEMENTACIÓN DE UNA RED MODBUS

#### 3.1 Operación del sistema SCADA

El sistema SCADA lo constituyen de tres PLCs, uno de ellos será el maestro y los dos restantes serán los esclavos (esclavo 1: estación de temperatura y el esclavo 2: es la estación de flujo). La comunicación se realiza mediante un bus de campo bajo el

protocolo Modbus con el método de transmisión de datos Maestro y Esclavo. El HMI está realizado en el software LabVIEW en donde se enciende y apaga el sistema, contiene avisos, almacena un histórico de la variación de la temperatura en el transcurso del tiempo y existe un paro de emergencia en caso de ser necesario.

### **3.1.1 Estación de temperatura (Esclavo 1)**

El primer proceso corresponde al control de temperatura conformado por el elemento primario de control (transmisor de temperatura), un controlador (PLC) y un elemento final de control (niquelina). Desde el HMI el operado digita el porcentaje de calentamiento al que desea que funcione la niquelina, éste dato es enviado al controlador y a la vez el transmisor envía información de la temperatura instantánea al controlador para visualizarla en el HMI.

### **3.1.2 Estación de caudal (Esclavo 2)**

El segundo proceso corresponde al control de caudal donde se tiene un elemento primario de control (flujómetro), un controlador (PLC) y el elemento final de control (bomba). Desde el HMI el operador digita el porcentaje de bombeo para controlar el caudal dentro de la tubería manipulando la velocidad de la bomba mediante el uso de un variador de velocidad conectado a ésta. A la vez que el flujómetro envía la información de flujo instantáneo para visualizarlo en el HMI.

## **3.2 Selección de Hardware**

### **3.2.1 Controlador lógico programable**

En el siguiente proyecto técnico para la elaboración de una red Modbus se utilizó un PLC SIEMENS 1200 CPU 1215 AC/DC/Rly, serie 6ES7 215-1B40-0XB0. El PLC consta de dos entradas Profinet, dos salidas y dos entradas analógicas. Para realizar la comunicación Modbus se usó el módulo de comunicación CM 1241 485/422. Para la conexión física se usó un cable PROFIBUS par trenzado apantallado color morado, tecnología RS 485 y conexión en bus.

### 3.3 Selección de software

#### 3.3.1 Interfaz hombre máquina (HMI)

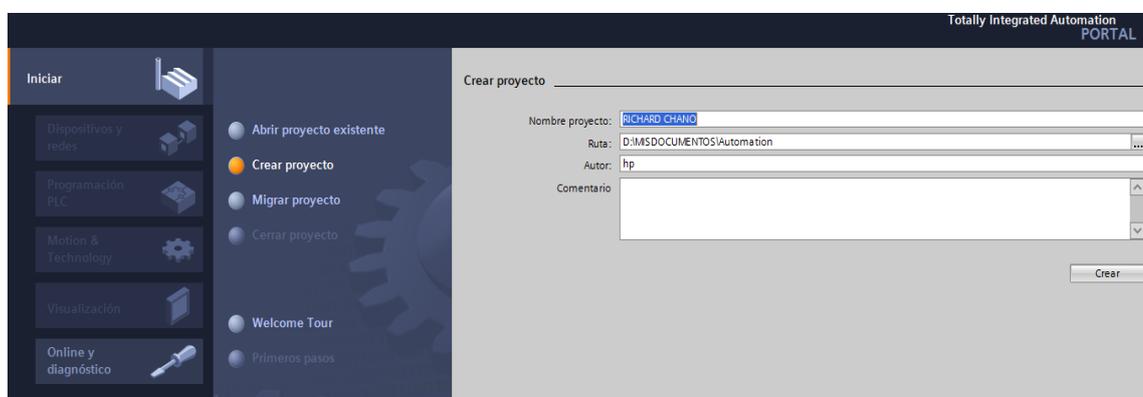
Para el diseño del HMI se utilizó el software Labview de National Instrument, ya que tiene un entorno amigable fácil de comprender.

También utilizamos el software OPC server el cual realiza la comunicación entre el PLC y la computadora

#### 3.4 Creación del proyecto

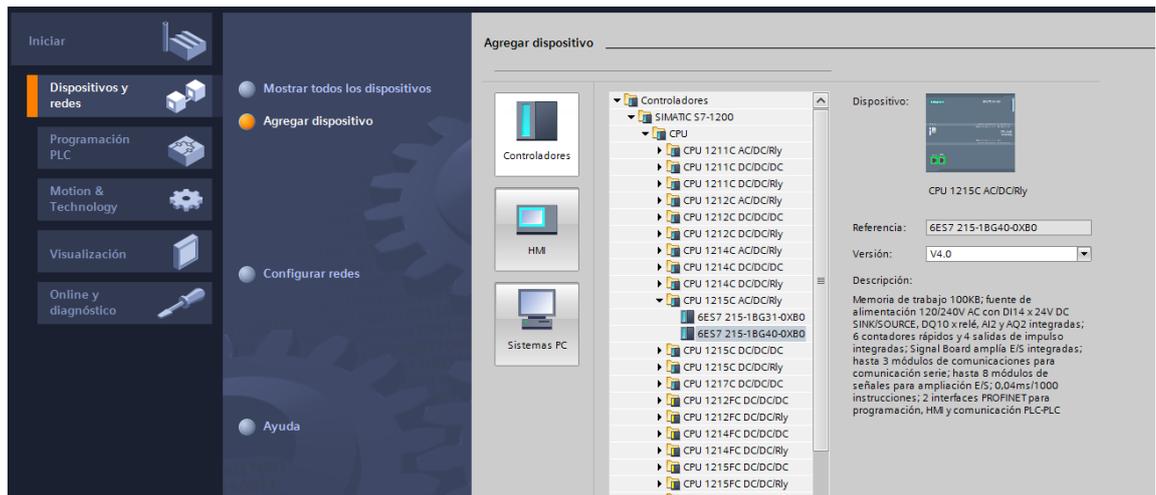
Para la creación del proyecto donde se va a programar el sistema SCADA se realizó los siguientes pasos:

- a) Dar click en crear proyecto, asignar un nombre y presionar crear



**Figura 19 Crear proyecto**

- b) Dar click en dispositivos y redes, presionar en agregar dispositivo, click en controladores y seleccione el modelo del PLC que se usará.

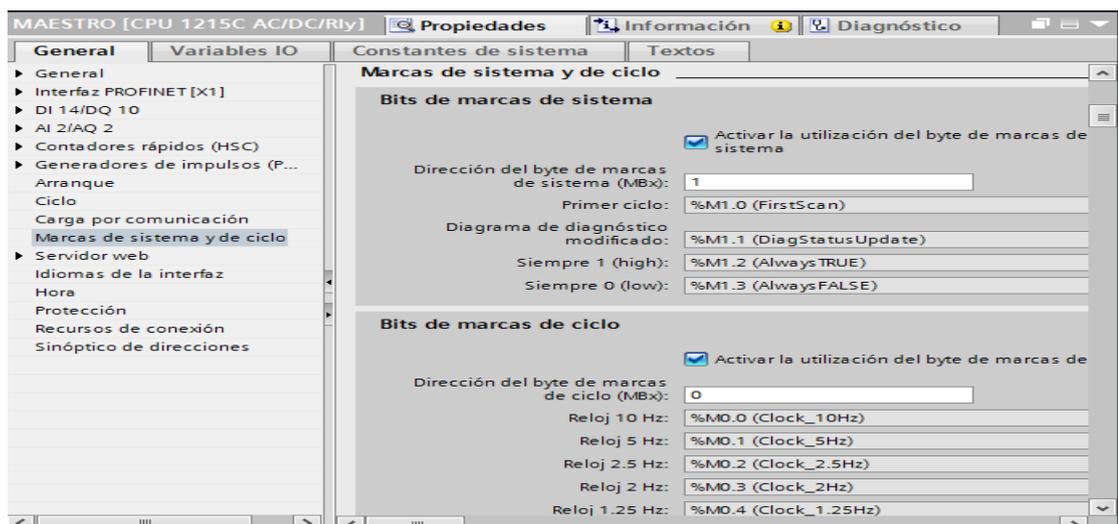


**Figura 20 Seleccionar Dispositivo**

### 3.5 Configuración de dispositivos

Primero se configura el PLC maestro para esto realizar lo siguiente:

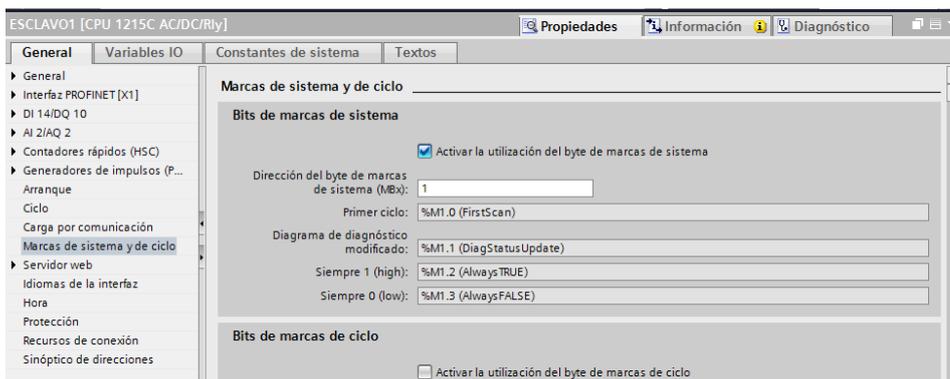
- a) Click en propiedades; dirigirse a marcas de sistema y de ciclo, activar el visto de los bits de marcas de sistema y marcas de ciclo



**Figura 21 Activación de bits de marca de sistema y ciclo del maestro**

Segundo, se configura al Esclavos 1, seguir los siguientes pasos:

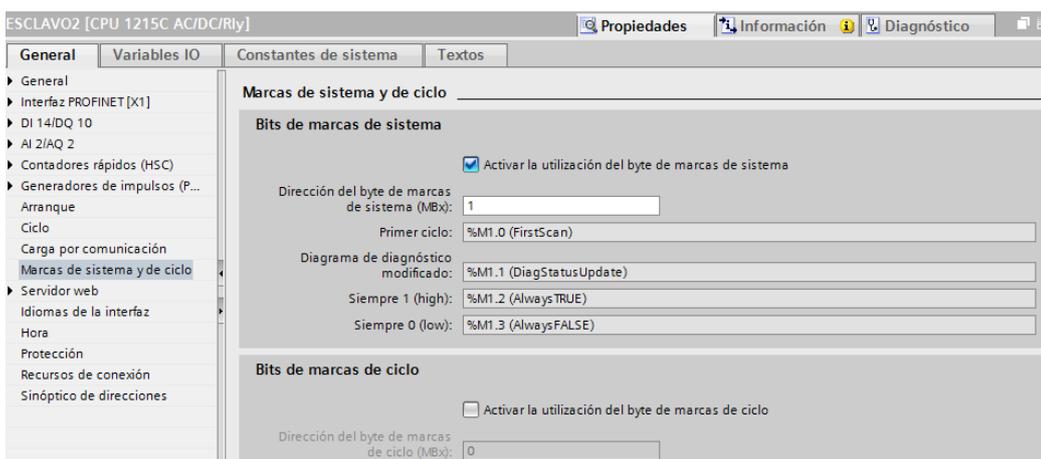
- b) Click en propiedades; diríjase a marcas de sistema y de ciclo, activar el visto en los bits de marcas de sistema



**Figura 22 Activación de bits de marca de sistema del esclavo 1**

Tercero, configurar al Esclavos 2. Se realiza los mismos pasos ejecutados para el esclavo 1, los mismos que se detallan a continuación:

- c) Click en propiedades, diríjase a marcas de sistema y de ciclo, activar el visto de los bits de marcas de sistema

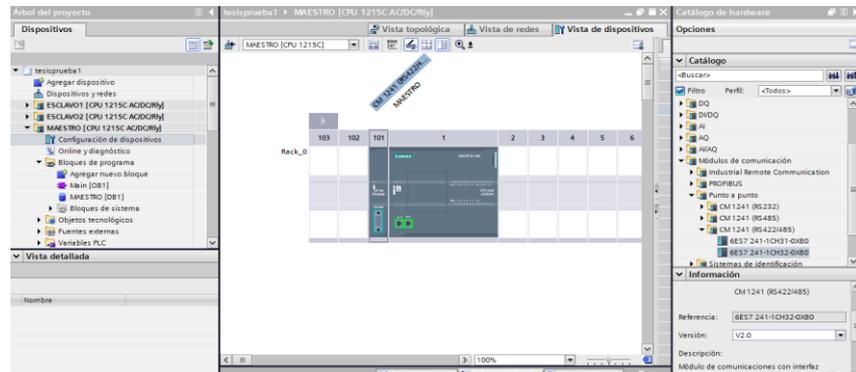


**Figura 23 Activación de los bits de marca de sistema del esclavo 2**

### 3.6 Creación de la red Modbus

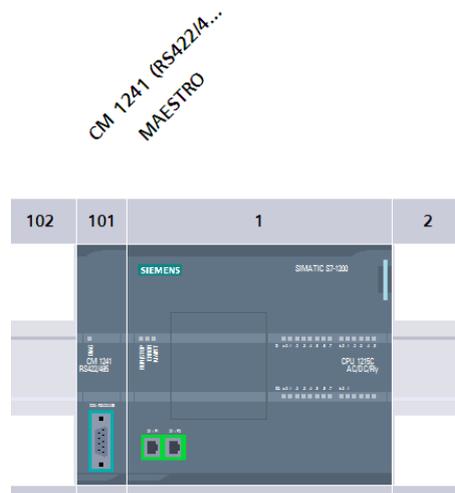
Para la creación de la red modbus se utilizó el módulo de comunicación CM 1241 RS (422/485), serie 6ES7241-1CH32-0XB0 para agregarlos siga los siguientes pasos:

Click en configuración de dispositivos, buscar en el catálogo, click en módulos de comunicación, agregar según la serie del dispositivo CM 1241 RS485/422. Guíese en la figura 24.



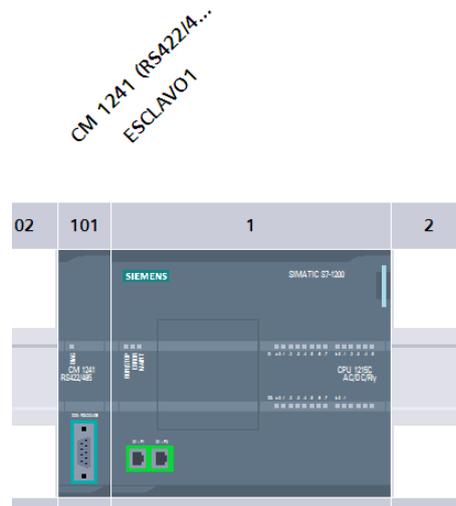
**Figura 24 Selección del módulo CM 1241.**

El PLC maestro y el CM 1241 agregados se visualizan en la figura 25.



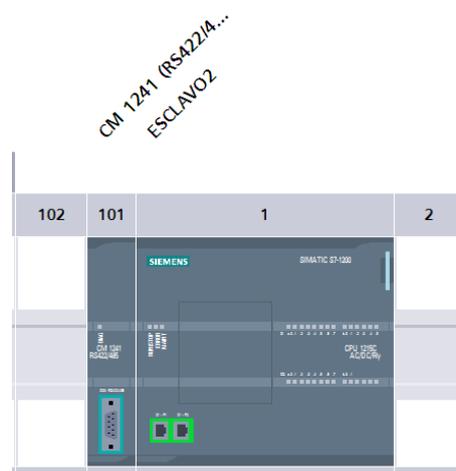
**Figura 25 Maestro y CM 1241**

El PLC esclavo 1 y el CM 1241 agregados se visualizan en la figura 26.



**Figura 26 Esclavo 1 y CM 1241**

El PLC esclavo 1 y el CM 1241 agregados se visualizan en la figura 27.

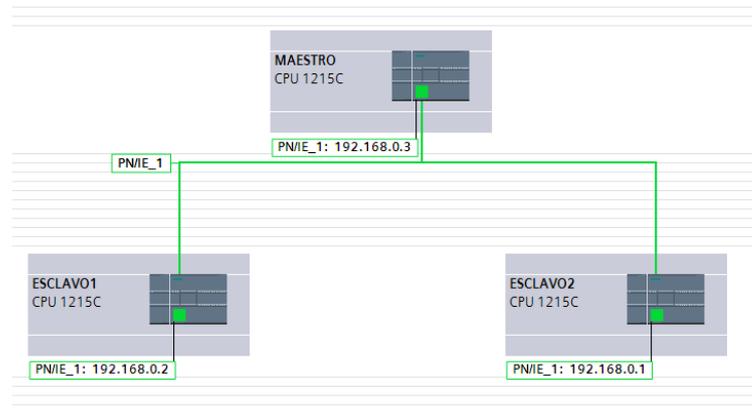


**Figura 27 Esclavo 2 y CM 1241**

### 3.7 Creación de la red profinet

Conecte los tres PLCs en red mediante la Interfaz profinet y asigne las siguientes direcciones IP. Al PLC 1: 192.168.0.1, al PLC 2: 192.168.0.2, al PLC 3: 192.168.0.3, a la PC: 192.168.0.10. Como se muestra en la figura 28.

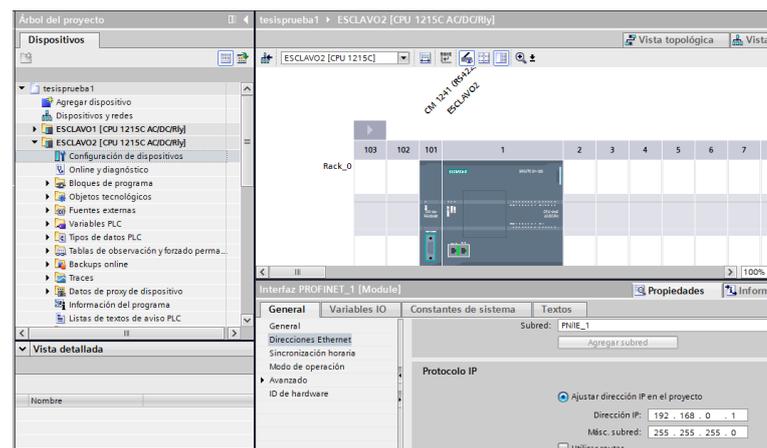
Nota: Las direcciones IP de los dispositivos deben ser diferentes.



**Figura 28 Red profinet del proyecto técnico.**

Para asignar la dirección IP al PLC 1 siga los siguientes pasos:

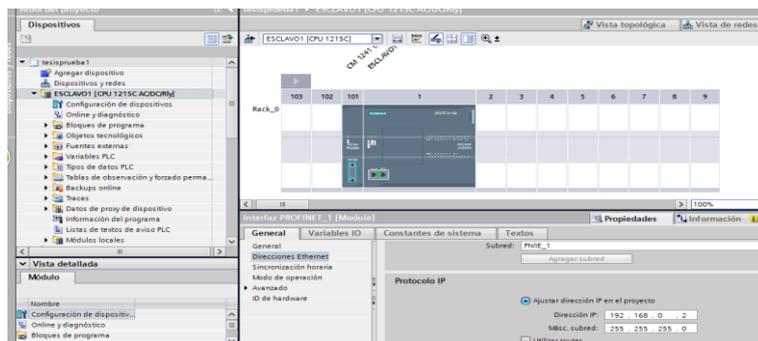
Dar click en configuración de dispositivo / propiedades / general / direcciones Ethernet / ajustar dirección IP en el proyecto.



**Figura 29 Asignación IP al PLC 1**

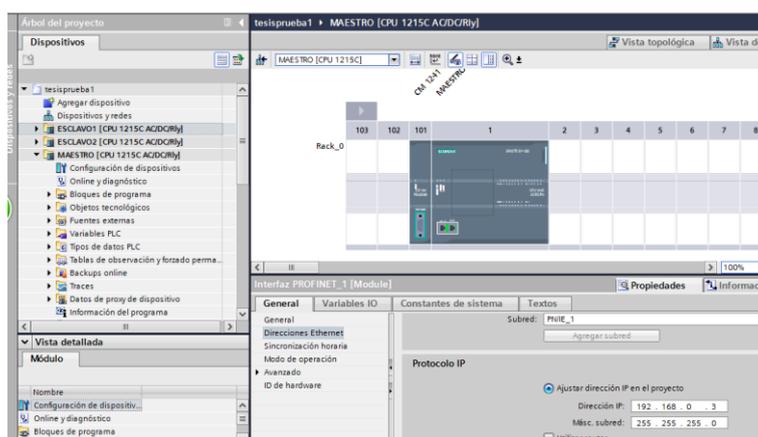
Seguir los mismos pasos para asignar la dirección IP al PLC 2 Y PLC 3

Dar click en configuración de dispositivo / propiedades / general / direcciones Ethernet / ajustar dirección IP en el proyecto.



**Figura 30 Asignación IP al PLC 2**

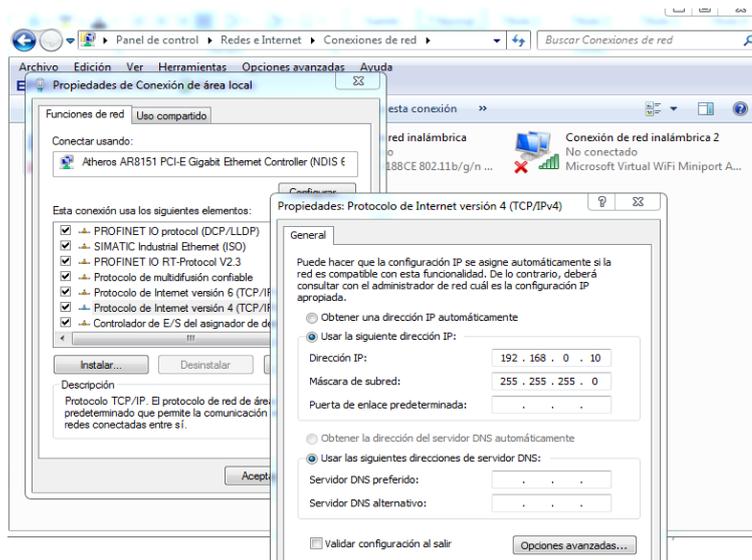
La figura 30 muestra la asignación IP al PLC 3.



**Figura 31 Asignación IP al PLC 3**

Para asignar la dirección IP a la PC siga los siguientes pasos:

Diríjase a Redes e internet / conexiones de red/ conexión de área local / protocolo de internet versión 4(TCP/IPv4).



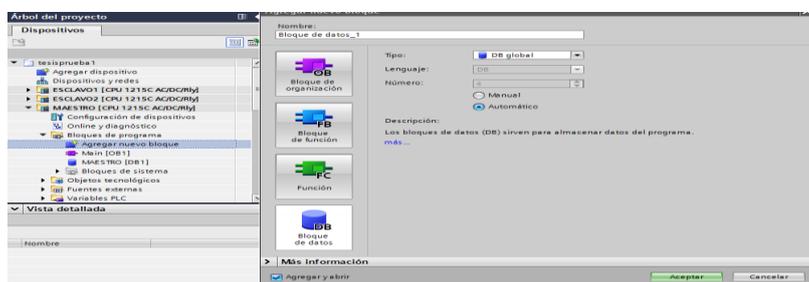
**Figura 32 Asignación IP de la PC**

### 3.8 Programación del PLC maestro

Este es el encargado de solicitar las peticiones o enviar datos a los dispositivos esclavos cada 200 ms. Es necesario crear un bloque de datos para intercambiar los datos entre el maestro y los esclavos.

Para crear el bloque siga los siguientes pasos:

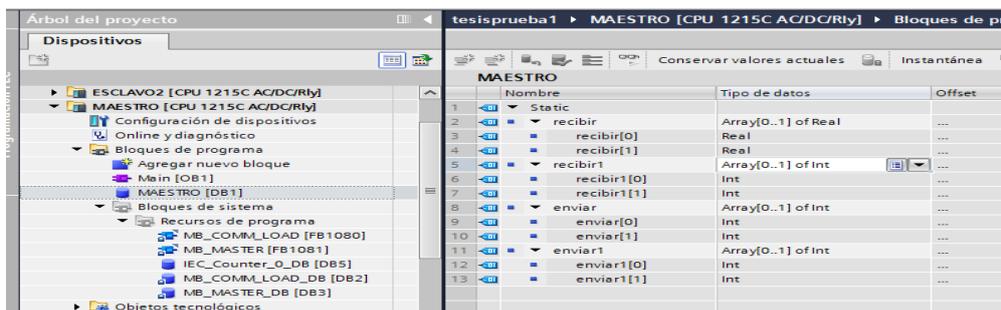
Dar click en bloques de programa / agregar nuevo bloque / bloque de datos / aceptar



**Figura 33 Creación del bloque**

Doble click en el bloque creado del maestro / escribir los arrays. El bloque de datos de la figura 34, tiene cuatro datos tipo array llamados recibir y enviar que pertenecen al esclavo 1, recibir1 y enviar1 pertenecen al esclavo 2, los cuales realizarán la

comunicación entre el maestro y esclavo 2. Para crear los datos tipo array siga los siguientes pasos:



**Figura 34 Bloque de Datos del Maestro**

Los bloques de sistema que contiene el PLC maestro se muestra en la siguiente figura.



**Figura 35 Bloque del sistema del Maestro**

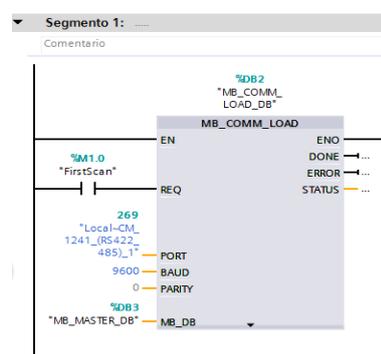
El maestro solicita o envía información a los esclavos mediante una secuencia es decir, el maestro realiza una petición de lectura o escritura a un esclavo a la vez. Por tal motivo se crea un contador de 0 a 3 para cumplir con este requerimiento. El contador es ascendente, alcanza un valor de conteo máximo de 3 porque el cuarto sirve como reseteo del contador.

**Tabla 5**

**Función de lectura y escritura que realiza el maestro**

Valor del Contador		Bits de marca interna		de marca 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

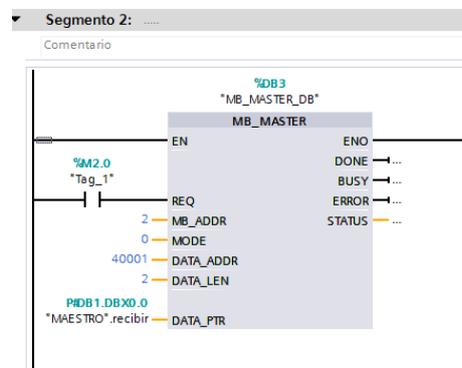
**En el segmento 1:** el bloque MB\_COMM\_LOAD\_DB crea el bus para establecer la comunicación ModBus RTU del maestro con los esclavos, se ejecuta una sola vez gracias al bit de marca de sistema FirstScan que corresponde al bit de marca interna %M1.0. La velocidad de transmisión para la comunicación del protocolo modbus es 9600 baudios. La paridad = 0, representa que no tiene. 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, como se muestra en la figura 36.



**Figura 36 Configuración del Maestro Modbus RTU**

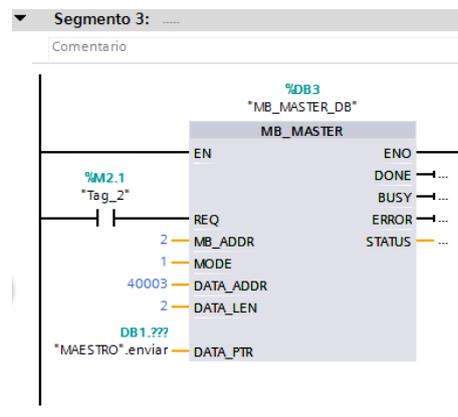
**El segmento 2:** Se crea el canal de comunicación entre el maestro y el esclavo 1 para recibir datos. La dirección (MB\_ADDR) es 2 porque en una red Modbus el maestro tiene la dirección 1. El MODE es 0 porque se realiza la operación de lectura;

se inicia en la dirección 40001 porque corresponde a los registros. En DATA\_PTR, se especifica que datos del bloque creado en el maestro se va a emplear en este caso “maestro.recibir”. La entrada REQ del bloque MB\_MASTER\_DB 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. Como se muestra en la figura 37.



**Figura 37 Instrucción para recibir datos del esclavo 1**

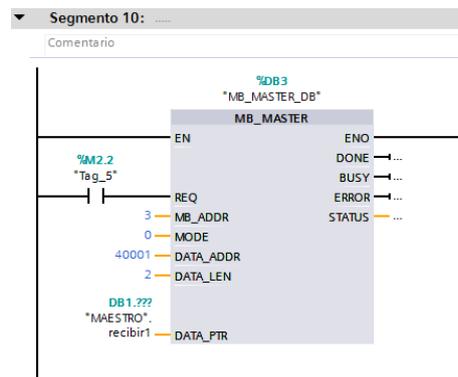
**En el segmento 3:** Se escribe datos al esclavo 1 mediante la dirección 40003, el dato enviar del bloque de datos del maestro. Para esto la entrada MODE es igual a 1, como se muestra en la figura 38.



**Figura 38 Instrucción para escribir datos al esclavo 1**

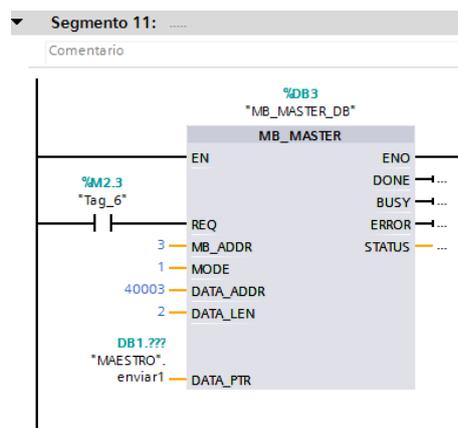
**El segmento 10:** Se crea el canal de comunicación entre el maestro y el esclavo 2 para recibir datos. La dirección (MB\_ADDR) es 3. El MODE es 0 porque se realiza la operación de lectura se inicia en la dirección 40001 porque corresponde a los registros,

En DATA\_PTR, se especifica que datos del bloque creado en el maestro se va a emplear, en este caso “**maestro.recibir1**”. Como se muestra en la figura 39.



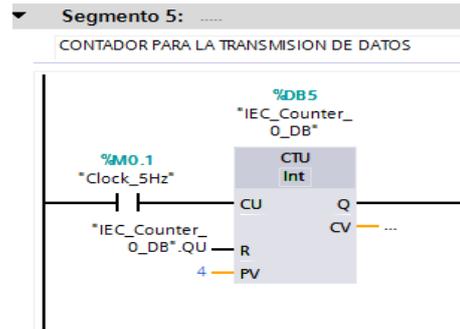
**Figura 39 Instrucción para recibir datos del esclavo 2**

**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. Como se muestra en la figura 40.



**Figura 40 Instrucción para escritura de datos del esclavo 2**

**En el segmento 5:** creamos el contador para la transmisión de datos. El intervalo de conteo es 200ms debido a que tiene conectada en la entrada CU, la marca de ciclo %M0.1 que genera una señal de reloj con una frecuencia de 5 Hz. Como se muestra en la figura 41



**Figura 41 Contador para la secuencia de comunicación Modbus**

En el segmento 6: se crea para la lectura que realiza el maestro al esclavo 1. Se ejecuta cuando se activa la marca %M2.0. Como se muestra en la figura 42.



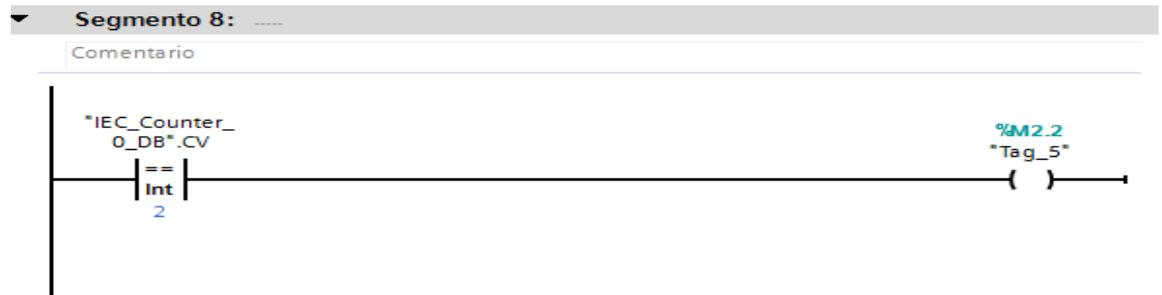
**Figura 42 Bit que activa la lectura del esclavo 1**

En el segmento 7: se crea para La escritura que realiza el maestro al esclavo 1. Se ejecuta cuando se activa la marca %M2.1. Como se muestra en la figura 43



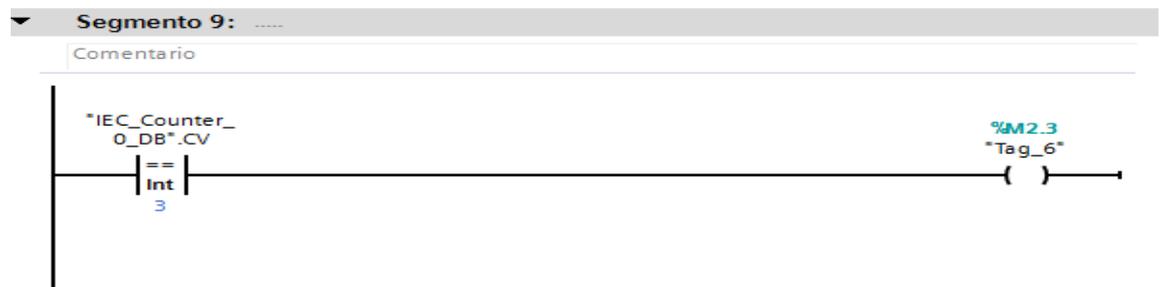
**Figura 43 Bit que activa la escritura del esclavo 1**

**En el segmento 8:** se crea para la lectura que realiza el maestro al esclavo 2. Se ejecuta cuando se activa la marca %M2.2. Como se muestra en la figura 44.



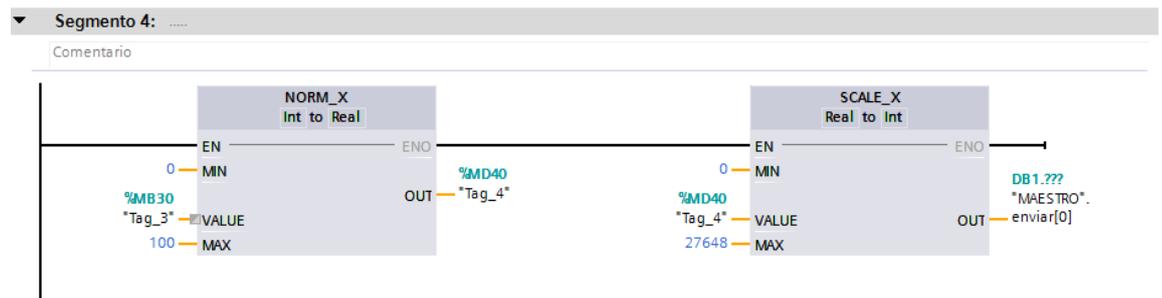
**Figura 44 Bit que activa la lectura del esclavo 2**

**En el segmento 9:** se crea para la escritura que realiza el maestro al esclavo 2. Se ejecuta cuando se activa la marca %M2.3. Como se muestra en la figura 45



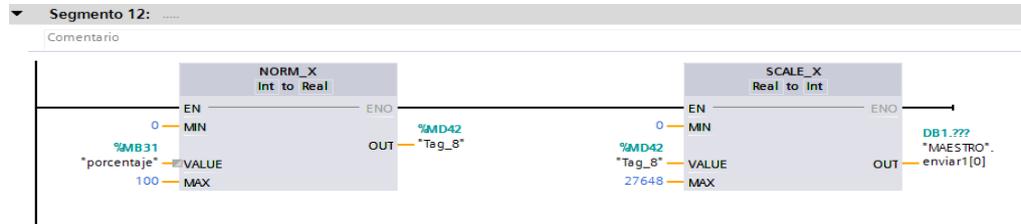
**Figura 45 Bit que activa la escritura del esclavo 2**

**En el segmento 4:** el maestro envía desde el set point de temperatura en un rango de 0 a 10 Vdc enviados desde el HMI hacia la niquelina. Empleando las funciones `NORM_X` y `SCALE_X`. Como se muestra en la figura 46



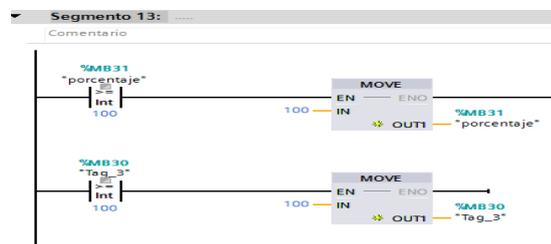
**Figura 46 Escalamiento de valores a ser enviados al esclavo 1**

**En el segmento 12:** el maestro envía desde el set point de porcentaje de flujo en un rango de 0 a 10 V al variador de frecuencia para controlar la bomba. Empleando las funciones NORM\_X y SCALE\_X. Como se muestra en la figura 47



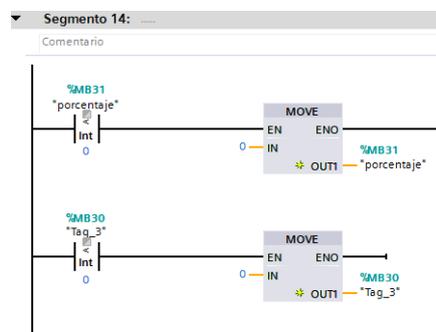
**Figura 47 Escalamiento de valores a ser enviados al esclavo 2**

**En el segmento 13:** en este segmento se programan los límites, set point  $\leq 100$ . Como se muestra en la figura 48



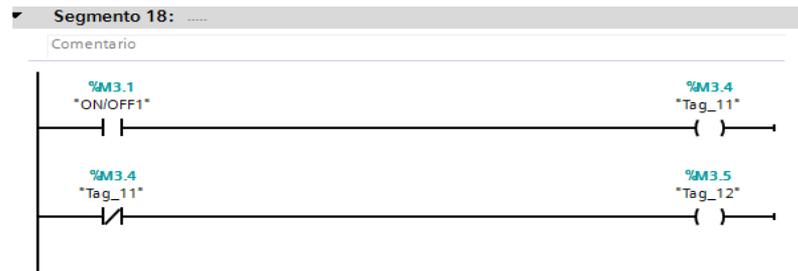
**Figura 48 No permite valores mayores a 100**

**En el segmento 14:** set point  $\geq 0$ . Como se muestra en la figura 49



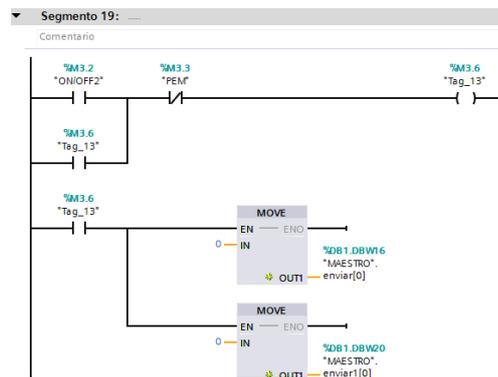
**Figura 49 No permite valores menores a 0**

**En el segmento 18:** La %M3.1 encenderá y apagará el sistema. Como se muestra en la figura 50



**Figura 50 Encendido y apagado**

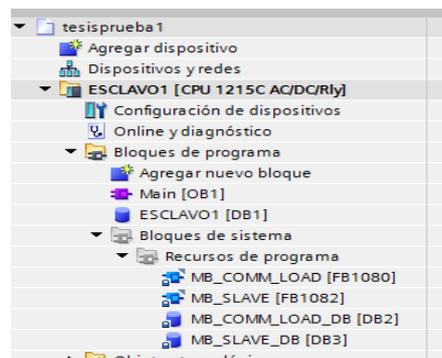
En el segmento 19: se programa el paro de emergencia recibido desde el HMI. Como se muestra en la figura 51.



**Figura 51 PEM y Reinicio**

### 3.9 Programación del esclavo 1:

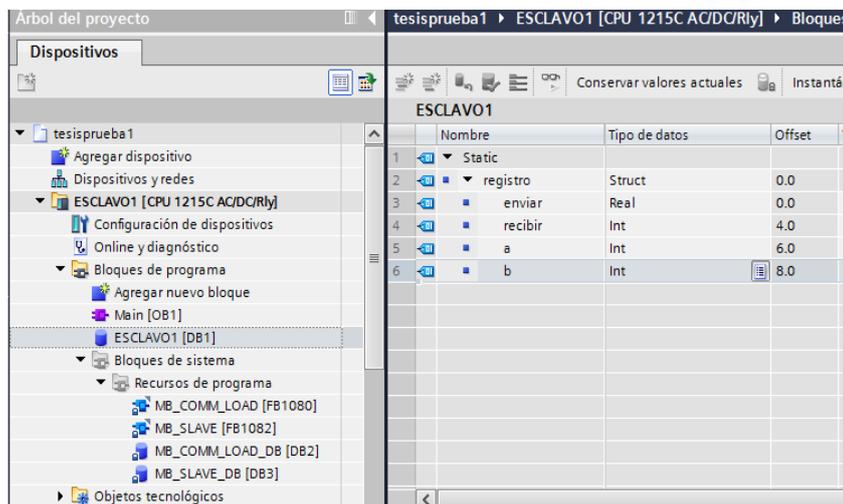
Este es el encargado de recibir y enviar datos al maestro. En el esclavo 1 el bloque de sistema queda como se muestra en la figura 52.



**Figura 52 Bloques de sistema del esclavo 1**

Además, es necesario crear un bloque de datos para intercambiar los mismos entre el esclavo 1 y el maestro. Contiene 4 elementos necesarios para el intercambio de datos entre el esclavo 1 y el maestro. Para crear los datos tipo estructura siga los siguientes pasos:

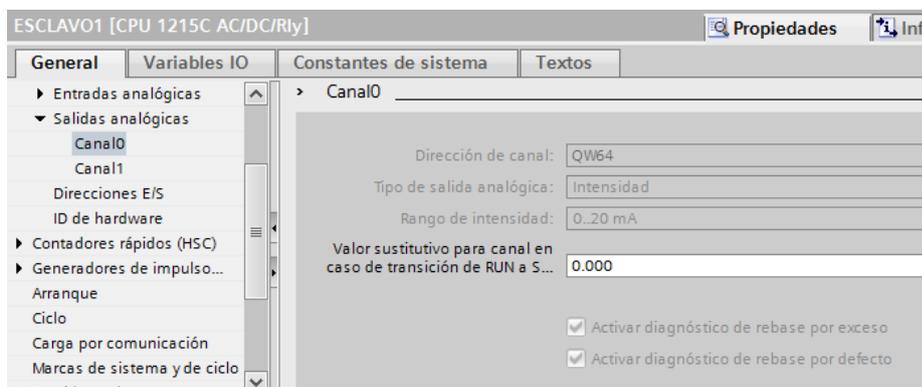
Doble click en el bloque creado del esclavo 1 / escribir la estructura. Como se muestra en la figura 53.



**Figura 53 Bloque de datos del esclavo 1**

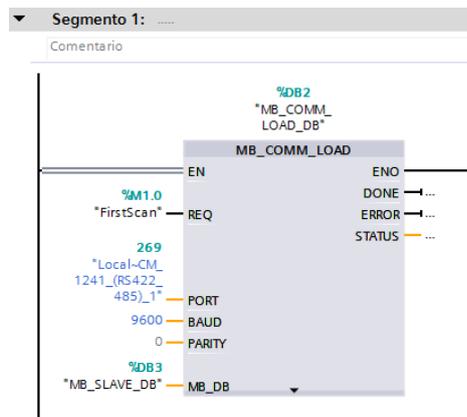
En los PLCs utilizados (CPU 1215 C) ya tiene incorporado dos salidas analógicas cuya información verificamos en la siguiente figura 54:

Clic en general / salidas analógicas / canal



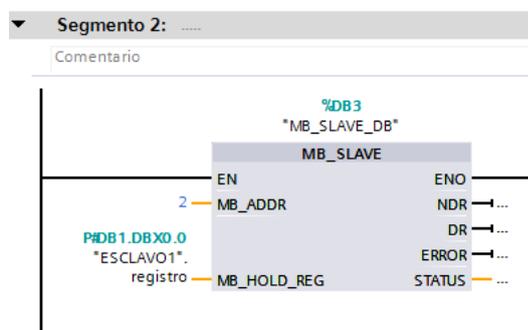
**Figura 54 Dirección de salida analógica del canal 0**

**En el segmento 1:** se encuentra el bloque MB\_COMM\_LOAD\_DB. Sirve para configurar la comunicación ModBus RTU .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. 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. Como se muestra en la figura 55.



**Figura 55 Configuración del esclavo 1**

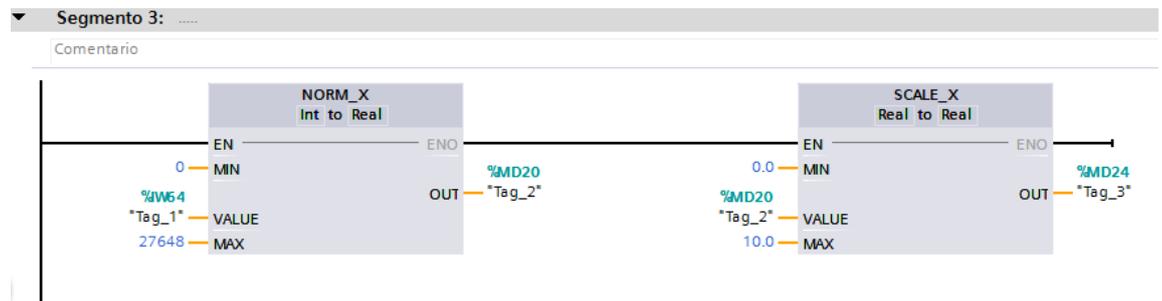
**En el segmento 2:** Se emplea la instrucción MB\_SLAVE\_DB para asignar la dirección del esclavo, éste bloque de datos es necesario para el intercambio de datos entre el esclavo 1 y el maestro. Como se muestra en la figura 56.



**Figura 56 Asignación de dirección del esclavo 1**

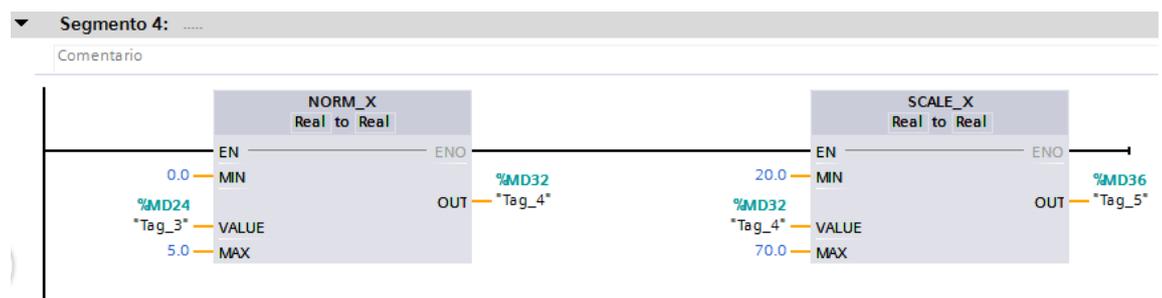
**En el segmento 3:** El esclavo lee por la entrada analógica cero, que viene integrada en el mismo PLC, la señal acondicionada del sensor de temperatura. Escala el valor de

0 a 27648 a un valor de 0 a 10. Este valor se guarda en la marca %MD24. Como se muestra en la figura 57.



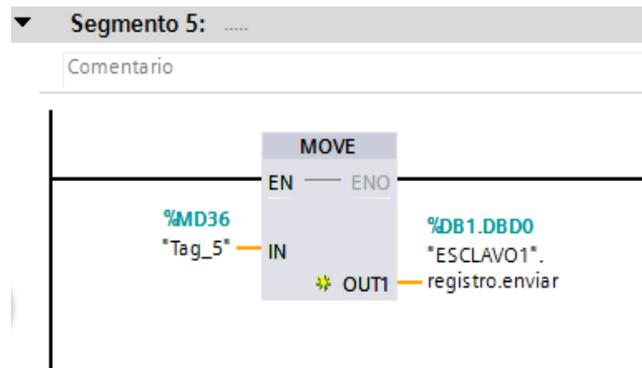
**Figura 57 Escalamiento de la entrada analógica 0 del esclavo 1**

**En el segmento 4. :** 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. Como se muestra en la figura 58.



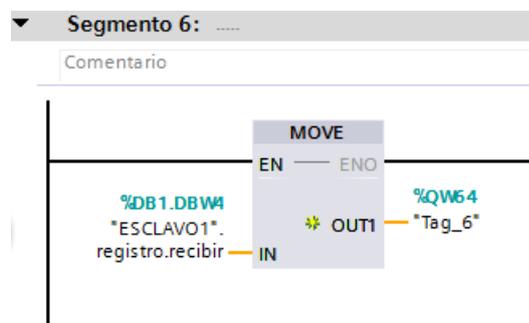
**Figura 58 Escalamiento para convertir a grados centígrados**

**En el segmento 5:** El valor presente en la marca interna %MD36 es enviado mediante el bloque de datos creado en el esclavo 1 al maestro. Como se muestra en la figura 59



**Figura 59 Valor de temperatura enviado al maestro**

**En el segmento 6:** El esclavo 1 recibe los datos del maestro mediante el bloque de datos “ESCLAVO1.registro.recibir” este dato es movido a la dirección %QW64 que corresponde al canal de salida analógico cero. Como se muestra en la figura 60.



**Figura 60 Recepción de datos desde el maestro**

### 3.10 Programación del esclavo 2

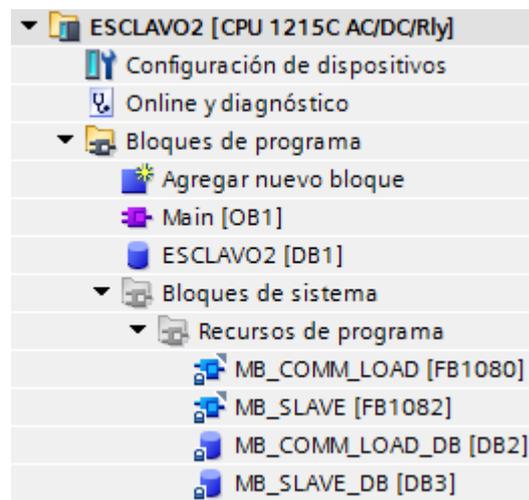
Además, es necesario crear un bloque de datos para intercambiar información entre el esclavo 2 y el maestro. Contiene 4 elementos necesarios para el intercambio de datos. Para crearlos siga los siguientes pasos:

Doble clic en el bloque del esclavo 2 / escribir la estructura. Como se muestra en la figura 61.

ESCLAVO2				
	Nombre	Tipo de datos	Offset	
1	Static			
2	registro	Struct	0.0	
3	enviar	Real	0.0	
4	recibir	Int	4.0	
5	a	Int	6.0	
6	b	Int	8.0	

**Figura 61 Bloque de datos del esclavo 2**

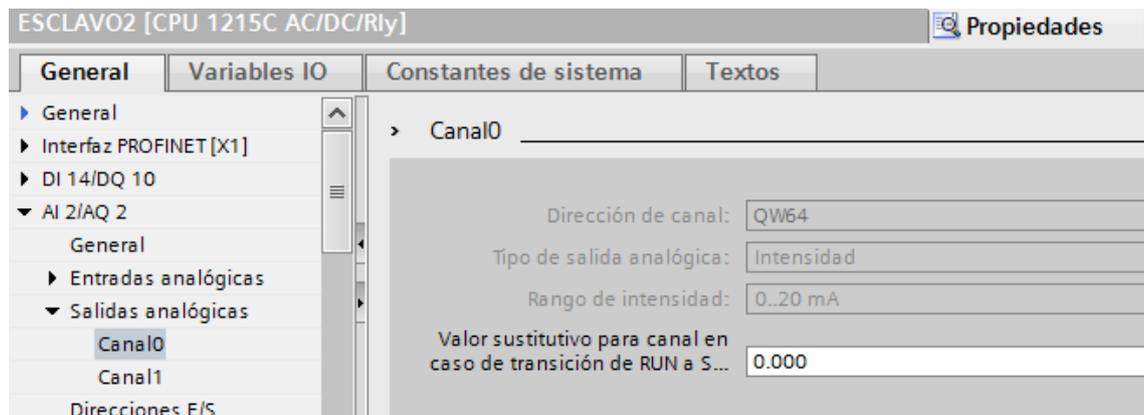
En el esclavo dos el bloque de sistema se muestra en la figura 62.



**Figura 62 Bloque de sistema del esclavo 2**

En los PLCs utilizados (CPU 1215 C) tiene incorporado dos salidas analógicas cuya información se verifica en la siguiente figura 63:

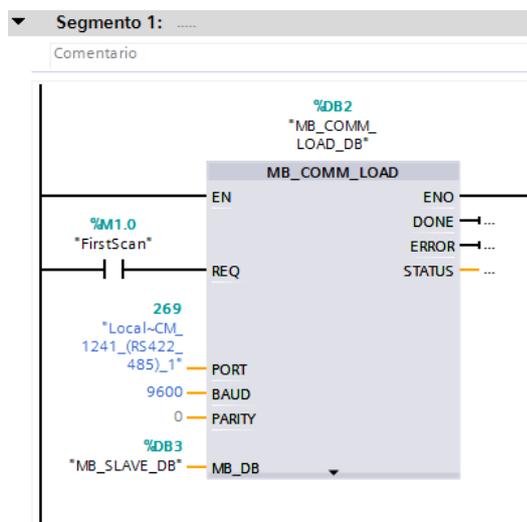
Clic en general / salidas analógicas / canal



**Figura 63 Salida analógica del esclavo 2**

**En el segmento 1:** El bloque MB\_COMM\_LOAD\_DB, sirve para configurar la comunicación ModBus RTU. Se ejecuta una sola vez gracias al bit de marca de sistema FirstScan que corresponde al bit de marca interna %M1.0. La velocidad de comunicación es 9600 baudios y la paridad debe ser 0 que representa ninguna.

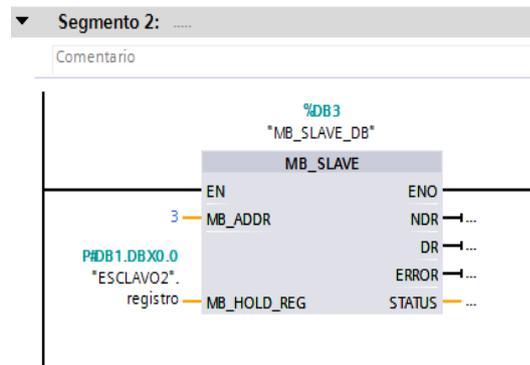
En la entrada del bloque de sistema denominada MB\_DB, se especifica la función mediante la cual el esclavo recibe o envía datos a maestro. Como se muestra en la figura 64.



**Figura 64 Configuración del esclavo 2**

**En el segmento 2:** Se emplea la instrucción MB\_SLAVE\_DB. En esta instrucción se asigna la dirección del esclavo 2 y el bloque de datos necesario

para el intercambio de datos entre el esclavo 2 y el maestro. Como se muestra en la figura 65.

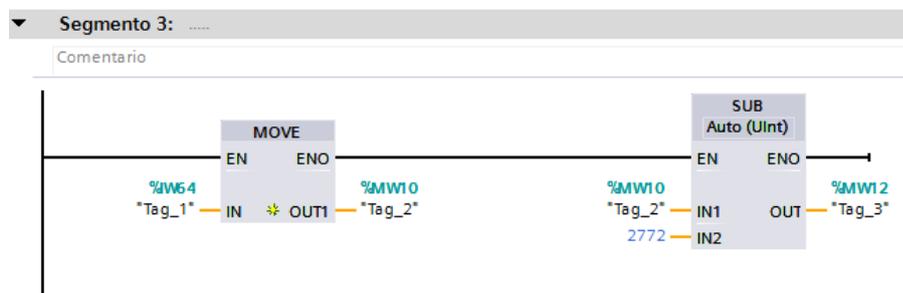


**Figura 65 Asignación de dirección del esclavo 2**

El esclavo 2 lee desde la entrada analógica cero, la señal del transmisor caudal para conocer los galones por minuto que pasan por la tubería. Para construir la ecuación que define la variación del caudal se usa la fórmula de la pendiente.

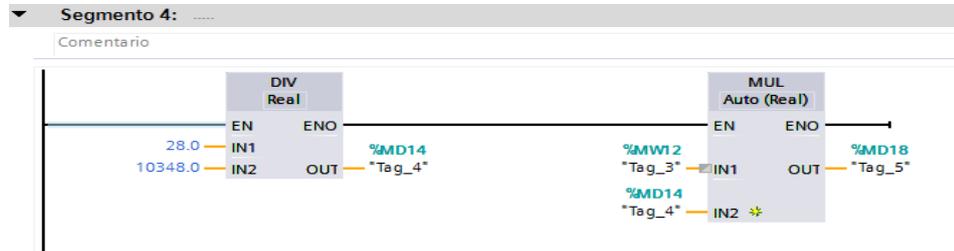
$$(1) \quad y = \frac{28}{10348}(x - 2772)$$

**En 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 %MW10. Se realiza primero la resta donde X es la entrada del sensor y Y es la variable buscada. Como se muestra en la figura 66.



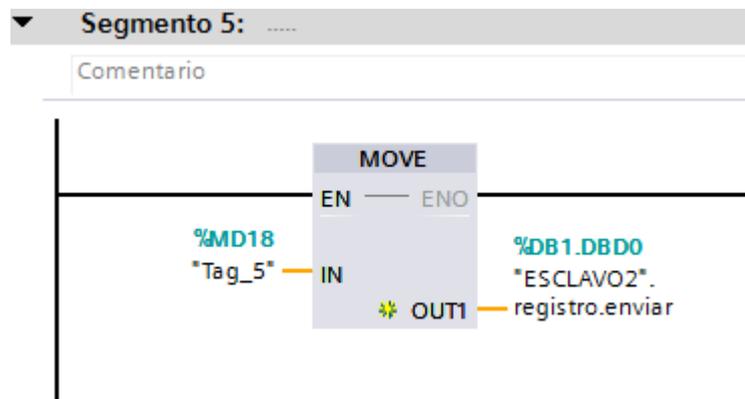
**Figura 66 Resta la entrada del sensor de caudal**

**En el segmento 4:** Se realiza la operación división de la ecuación 1 ( $28 / 2772$ ) y luego se multiplica por el valor del segmento 3. La respuesta de esta ecuación se guarda en la marca %MD18. Como se muestra en la figura 67.



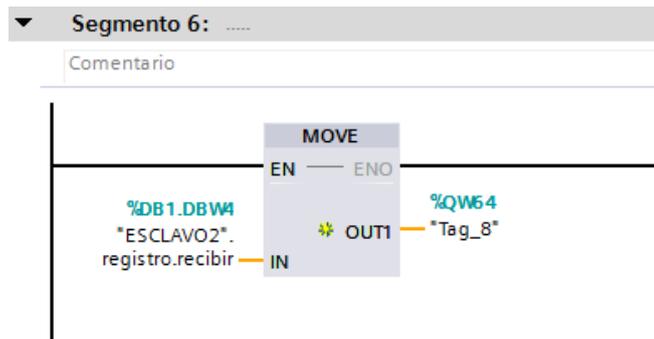
**Figura 67 Resultado de la ecuación**

**En el segmento 5:** El valor presente en la marca interna %MD18 es enviado mediante el bloque de datos creado en el esclavo 2 al maestro. Como se muestra en la figura 68.



**Figura 68 Valor de caudal enviado al maestro**

**En el segmento 6:** El esclavo 2 recibe los datos del maestro mediante el bloque de datos "ESCLAVO2.registro.recibir". Este dato es movido a la dirección %QW64 que corresponde al canal de salida analógico cero. Como se muestra en la figura 69.



**Figura 69 Recepción de datos desde el maestro**

### 3.11 HMI

#### 3.11.1 NI OPC Server

Para crear esta interfaz se utilizó el software NI OPC Server para comunicar el PLC maestro con el Software labview.

Creamos un bloque donde están los tags que utilizaremos en labview.

Tag Name	Address	Data Type	Scan Rate	Scaling	Desc
A	MD38	Float	100	None	
B	MD46	Float	100	None	
caudal	MB31	Byte	100	None	
on/off1	M3.1	Boolean	100	None	
on/off2	M3.2	Boolean	100	None	
pem	M3.3	Boolean	100	None	
temperatura	MB30	Byte	100	None	

**FIGURA 70 Tags para el HMI**

### 3.11.2 HMI

La figura 70 muestra el HMI programado

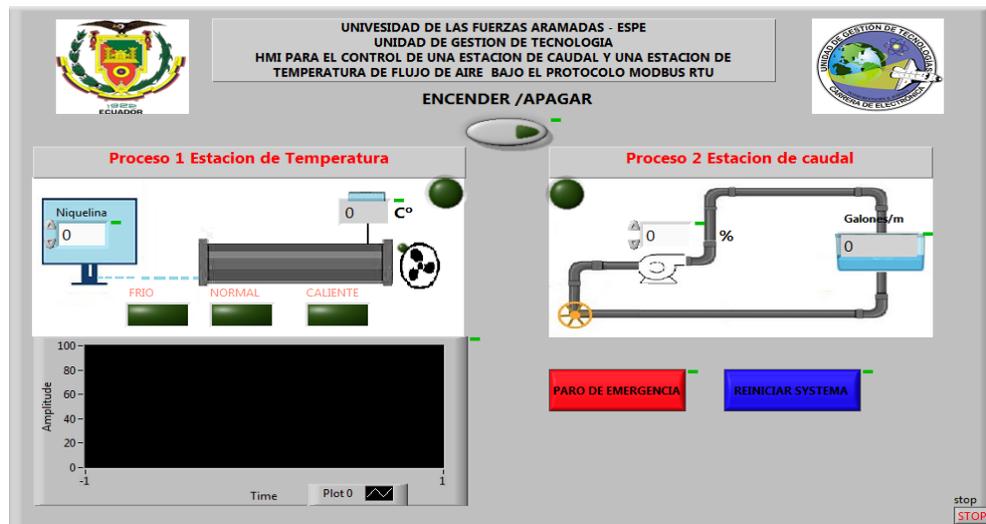


Figura 71 HMI en Labview

La programación se realizó usando la sentencia FOR LOOP de labview como se muestra la figura 71.

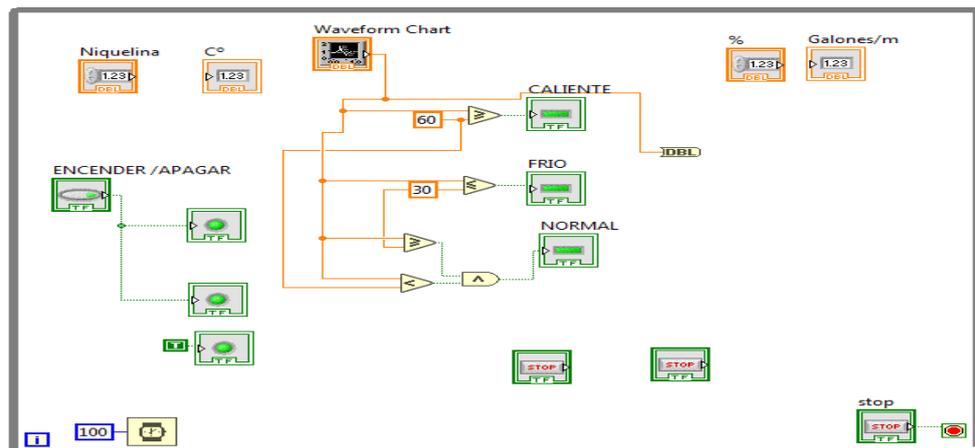


Figura 72 Programación del HMI

### 3.12 Diagrama de conexiones eléctricas de los PLC's

Conexiones eléctricas esclavo 1

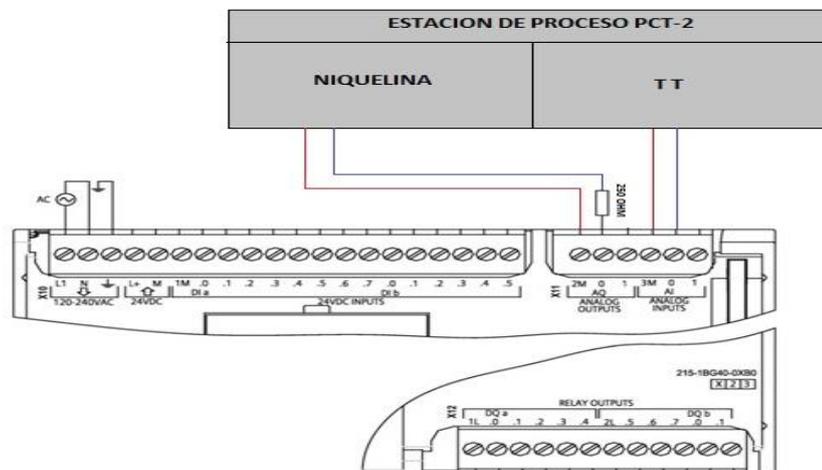


Figura 73 Esquema de conexiones de la PCT-2.

Conexiones eléctricas del esclavo 2

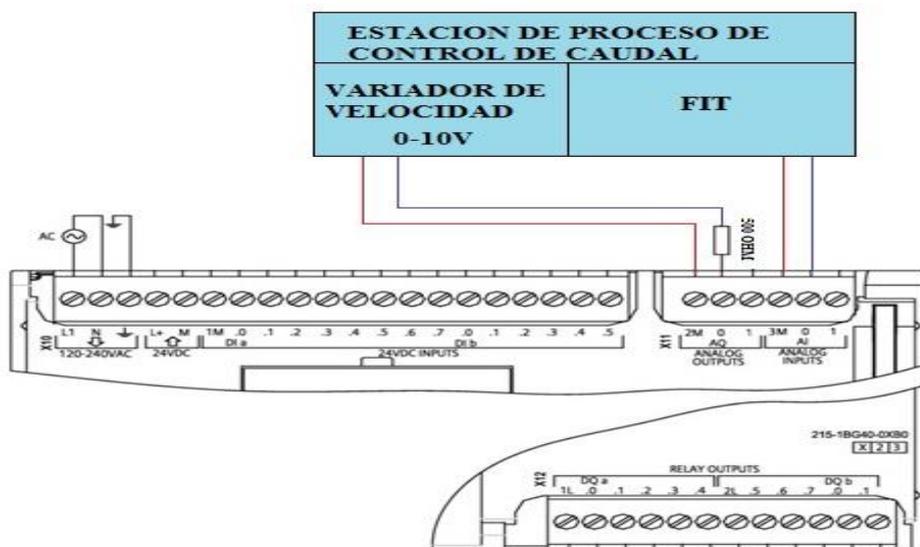


Figura 74 Esquema de conexión de la estación de caudal.

## CAPITULO IV

### CONCLUSIONES Y RECOMENDACIONES

#### 4.1 Conclusiones

- Se comprobó que exista comunicación entre el elemento primario de control, el controlador y el elemento final de control en cada una de las estaciones enviando y recibiendo información.
- Se configuró y habilitó la red modbus mediante la programación realizada en TIA PORTAL con el lenguaje ladder.
- Se adquirió la respuesta de los esclavos en el bloque MB\_SLAVE y a través del bloque MB\_COM\_LOAD se envió la información al maestro para ser visualizados en el HMI.
- Se realizó un HMI para receptor los datos del operador y enviarlos al PLC maestro el mismo que los redirige a los respectivos esclavos a través del bloque MB\_COM\_LOAD.

#### 4.2 Recomendaciones

- Cuando conecte el cable PROFIBUS a los CM 1241 asegúrese de que las resistencias en el esclavo 1 y 2 en estado ON y en el maestro en estado OFF debido a la configuración tipo bus que se construyó en este proyecto.
- Compile y quite los atributos de los bloques tanto maestro como los esclavos para que exista comunicación.
- Cuando inicie el sistema y cargue los PLCs asegúrese q todos los leds estén de color verde eso quiere decir que está funcionando correctamente
- Una vez terminada la programación en ladder primero debe cargar la información a los esclavos y al último al maestro

## REFERENCIAS BIBLIOGRÁFICA

- Bonifas, L. F., & Flores, V. M. (2013). Diseño e implementación de un módulo didáctico para una red profibus DP. Quito.
- Cevallos. (Noviembre 2007). Protocolo Modbus. Quito.
- Chano Pillajo , R. E. (2019).
- CORRALES, L. (12 de 2007). Obtenido de <http://bibdigital.epn.edu.ec/bitstream/15000/10020/2/PARTE%202.pdf>
- Echeverría, G. (15 de Octubre de 2013). IMPLEMENTACIÓN DE UNA RED PROFIBUS DP EN UN SISTEMA AUTOMATIZADO. Pereira. Obtenido de IMPLEMENTACIÓN DE UNA RED PROFIBUS DP EN UN SISTEMA AUTOMATIZADO.
- Francesch Saavedra, L. G. (2015). Obtenido de <https://pirhua.udep.edu.pe/handle/123456789/2416>
- Guerrero, V., Yuste, R., & Martínez, L. (2010). Comunicaciones industriales. S.A. MARCOMBO.
- Lozano Castro, C. (22 de 03 de 2001). Obtenido de <http://www.uco.es/grupos/eatco/automatica/ihm/descargar/scada.pdf>
- Marín Iturrarte, A. (2015). Obtenido de <http://academica-e.unavarra.es/bitstream/handle/2454/6711/577960.pdf?sequence=1>
- Mejía Herrera, D. J. (2016). IMPLEMENTACIÓN DE UNA RED MODBUS RTU CON PLC S7-1200 PARA EL CONTROL Y MONITOREO DE DOS ESTACIONES DE PROCESOS. LATACUNGA: ESPE.
- Moreno, M. (19 de 6 de 2007). Obtenido de <http://www.microautomacion.com/capacitacion/Manual061ControladorLgicoProgramablePLC.pdf>
- Ojeda Sosa, F. R. (2017). Obtenido de <http://tesis.ucsm.edu.pe/repositorio/handle/UCSM/6532>

- Rosado. (22 de 4 de 2003). Obtenido de [https://www.uv.es/rosado/courses/sid/Capitulo3\\_rev0.pdf](https://www.uv.es/rosado/courses/sid/Capitulo3_rev0.pdf)
- Salinas, S. (10 de 11 de 2011). Obtenido de <https://www.dspace.espol.edu.ec/bitstream/123456789/14620/10/10%20CAPITULO%203%20%283.3%29.pdf>
- Serna Salazar, C. A. (10 de 12 de 2011). Obtenido de <http://revistasum.umanizales.edu.co/ojs/index.php/ventanainformatica/article/viewFile/126/184>
- Server, N. O. (2016).
- siemens. (11 de 2009). Controlador programable S7-1200. Obtenido de <https://w5.siemens.com/spain/web/es/industry/automatizacion/simatic/Documents/S71200-MANUAL%20DEL%20SISTEMA.PDF>
- siemens. (02 de 02 de 2015). [www.infopl.net](http://www.infopl.net). Obtenido de [http://www.infopl.net/files/descargas/siemens/infoPLC\\_net\\_guia2.pdf](http://www.infopl.net/files/descargas/siemens/infoPLC_net_guia2.pdf)
- siemens. (22 de 1 de 2019). Hoja de datos 6ES7241-1CH32-0XB0. Obtenido de <https://support.industry.siemens.com/cs/pd/59426?pdtdi=td&dl=en&lc=en-AE>
- TIA PORTAL . (2014).
- Torres Hurtado, J. M. (24 de 1 de 2015). [www.infopl.net](http://www.infopl.net). Obtenido de [http://www.infopl.net/files/documentacion/comunicaciones/infoPLC\\_net\\_introduccion-a-las-redes-de-comunicacion-industrial.pdf](http://www.infopl.net/files/documentacion/comunicaciones/infoPLC_net_introduccion-a-las-redes-de-comunicacion-industrial.pdf)
- Vicente, G. (2016). comunicaciones industriales.
- Villajulca, J. C. (19 de 07 de 2010). [nstrumentacionycontrol.net](http://instrumentacionycontrol.net). Obtenido de <http://instrumentacionycontrol.net/los-buses-de-campo-directo-al-grano/>

HOJA DE LEGALIZACIÓN DE FIRMAS

DEL CONTENIDO DE LA PRESENTE INVESTIGACIÓN  
SE RESPONSABILIZA EL AUTOR

---

CHANO PILLAJO RICHARD ESTALIN

DIRECTOR DE LA CARRERA DE ELECTRÓNICA MENCIÓN  
INSTRUMENTACIÓN Y AVIÓNICA

---

ING. PABLO PILATASIG

Latacunga, Febrero del 2019

## **SESIÓN DE DERECHOS DE PROPIEDAD INTELECTUAL**

Yo, **CHANO PILLAJO RICHARD ESTALIN**, Egresado de la carrera de Electrónica Mención Instrumentación y Aviónica, en el año 2018, con cedula de Ciudadanía No. **1723616130**, Autor del trabajo de Graduación **“IMPLEMENTACIÓN DE UNA RED MODBUS CON UN MAESTRO Y DOS ESCLAVOS UTILIZANDO UN PLC-S7 – 1200 - CPU - 1215C CON MÓDULOS DE COMUNICACIÓN MODBUS 485. PARA PRÁCTICAS DE REDES INDUSTRIALES”**, cedo mis derechos de propiedad intelectual a favor de la Unidad de Gestión de Tecnologías de la Universidad de las Fuerzas Armadas.

Para constancia firmo la presente sesión de propiedad intelectual.

---

**CHANO PILLAJO RICHARD ESTALIN**

**CI: 1723616130**

Latacunga, Febrero del 2019

## HOJA DE VIDA



### **DATOS PERSONALES:**

NOMBRE: Chano Pillajo Richard Estalin  
CEDULA DE CIUDADANÍA: 1723616130  
FECHA DE NACIMIENTO: 06-02-1997  
LUGAR DE NACIMIENTO: Sangolquí  
ESTADO CIVIL: Soltero  
DIRECCIÓN: Valle de los Chillos - Inchalillo  
TELÉFONO: 0995013676  
E-MAIL: richardchano@hotmail.es

### **ESTUDIOS REALIZADOS:**

ESTUDIOS PRIMARIOS: Unidad Educativa Carlos Larco Hidalgo  
ESTUDIOS SECUNDARIOS: Unidad Educativa Juan de salinas  
Bachiller en “Instalaciones, Equipos y Máquinas Eléctricas”.  
UNIVERSITARIOS: Unidad de Gestión de Tecnologías de la Universidad de las Fuerzas Armadas “ESPE”  
Tecnólogo en “Electrónica Mención Instrumentación y Aviónica”

## **IDIOMA EXTRANJERO:**

Dominio del idioma hablado: 50%

Dominio del idioma escrito: 50%

## **CERTIFICACIONES:**

Seis niveles de inglés aprobados – ESPE

Certificado en Prevención de Riesgos Laborables

## **EXPERIENCIA PROFESIONAL O PRÁCTICAS PRE PROFESIONALES:**

Pasante en mantenimiento de instalaciones eléctricas

## **CURSOS Y SEMINARIOS.**

Curso de Seguridad Electrónica (cámaras CCTV, porteros eléctricos, video portero, alarmas de intrusión e incendios, control de accesos, control de asistencia).

Curso de cámaras CCTV, IP y espías.

Certificado en Jornadas Internacionales de Ciencia y Tecnología en Electromecánica

## **DATOS DE INTERÉS**

Arbitro Profesional de la ciudad de Cotopaxi.



## **CARRERA DE ELECTRÓNICA MENCIÓN INSTRUMENTACIÓN & AVIÓNICA**

### **ACTA DE RECEPCIÓN DE TRABAJO DE TITULACIÓN**

En la ciudad de Latacunga, provincia de Cotopaxi siendo (las catorce horas del 15 de febrero del 2019) ante el señor **ING. PABLO PILATASIG** Director de la Carrera de **ELECTRÓNICA MENCIÓN INSTRUMENTACIÓN & AVIÓNICA** de la **UNIDAD DE GESTIÓN DE TECNOLOGÍAS** de la **UNIVERSIDAD DE LAS FUERZAS ARMADAS-ESPE**, comparece que el señor egresado: **CHANO PILLAJO RICHARD ESTALIN**, quien manifiesta hacer la entrega de una impresión del desarrollo del trabajo de graduación titulado **“IMPLEMENTACIÓN DE UNA RED MODBUS CON UN MAESTRO Y DOS ESCLAVOS UTILIZANDO UN PLC-S7 – 1200 - CPU - 1215C CON MÓDULOS DE COMUNICACIÓN MODBUS 485. PARA PRÁCTICAS DE REDES INDUSTRIALES”**. En efecto, verificado por el Director de Carrera que el egresado presenta el trabajo de graduación dentro del plazo estipulado en este reglamento, por lo que el señor Director dispone se levante la presente Acta de Entrega – Recepción, para dar cumplimiento a lo establecido en la forma legal.

Para constancia de lo actuado, firman en la unidad el señor **DIRECTOR DE LA CARRERA DE ELECTRÓNICA MENCIÓN INSTRUMENTACIÓN & AVIÓNICA** y el señor **EGRESADO** de esta carrera.

---

Sr. Chano Pillajo Richard Estalin

**Egresado**

---

Ing. Pablo Pilatasig

**Director de la Carrera**