

UNIVERSIDAD SAN FRANCISCO DE QUITO

Colegio de Ciencias e Ingenierías

**Implementación de una unidad de mezcla en el módulo didáctico
MPS-PA Compact Workstation**

Milene Rallen Muñoz Medina

Cristhian Andrés Marcial Domínguez

Alberto Sánchez, Ph.D., Director de Tesis

Tesis de grado presentada como requisito
para la obtención del título de Ingeniero Eléctrico y Electrónico

Quito, Mayo de 2013

Universidad San Francisco de Quito

Colegio de Ciencias e Ingenierías

HOJA DE APROBACIÓN DE TESIS

Implementación de una unidad de mezcla en el módulo didáctico MPS-PA Compact

Workstation

Milene Rallen Muñoz Medina

Cristhian Andrés Marcial Domínguez

Alberto Sánchez, Ph.D.

Director de la tesis

Ing. Nelson Herrera

Miembro del Comité de Tesis

Omar Aguirre, MSc

Coordinador de Ingeniería Eléctrica y Electrónica

Ximena Córdova, Ph.D.

Decano de la Escuela de Ingeniería del Colegio de Ciencias e Ingeniería

Quito, Mayo de 2013

© DERECHOS DE AUTOR

Por medio del presente documento certifico que he leído la Política de Propiedad Intelectual de la Universidad San Francisco de Quito y estoy de acuerdo con su contenido, por lo que los derechos de propiedad intelectual del presente trabajo de investigación quedan sujetos a lo dispuesto en la Política.

Asimismo, autorizo a la USFQ para que realice la digitalización y publicación de este trabajo de investigación en el repositorio virtual, de conformidad a lo dispuesto en el Art. 144 de la Ley Orgánica de Educación Superior.

Firma: _____

Nombre: Milene Rallen Muñoz Medina

C. I.: 1717742405

Firma: _____

Nombre: Cristhian Andrés Marcial Domínguez

C. I.: 1803933025

Lugar: Quito, Ecuador

Fecha: Mayo, 2013

RESUMEN

En el presente proyecto se realiza una ampliación del módulo didáctico MPS-PA Compact Workstation, el cual consiste en la implementación de una etapa de mezcla que es controlada mediante un PLC independiente. En primer lugar se muestra el diseño del sistema de mezcla utilizando los elementos de control, sensores y actuadores necesarios. Se utiliza un PLC S7-200 (CPU 226) y el módulo análogo EM 235. Este equipo se comunica con una computadora mediante una interfaz Puerto-Puerto (PPI). El MPS-PA utiliza la interfaz EasyPort USB. Finalmente se desarrolla una aplicación que se ejecuta sobre un PC y elaborada en Indusoft, la cual hace uso de las medidas recolectadas desde el EasyPort y el S7-200 para realizar las operaciones de control y monitoreo del proceso. Este proyecto finaliza con la elaboración de hojas guías para prácticas relacionadas con el área de control de procesos y sensores e instrumentación.

ABSTRACT

In this project an extension of the MPS-PA Compact Workstation training module is designed and tested. The extension consists in a mixing stage, controlled by a separate PLC. A S7-200 PLC (CPU 226), an analog module EM 235, and several sensors and actuators are used. The controller communicates through a PPI interface to a central processing computer. The MPS-PA Compact Workstation module uses EasyPort USB interface. The control computer runs an InduSoft application that implements control and monitoring functions. This application uses the measurements that are collected via EasyPort and the S7-200 PLC. The project concludes with the development of Lab WorkSheets related to process control and sensors & instrumentation.

TABLA DE CONTENIDO

RESUMEN	5
ABSTRACT	6
TABLAS	8
FIGURAS	8
INTRODUCCIÓN	11
DESCRIPCIÓN DEL SISTEMA	12
Descripción del sistema FESTO MPS-PA Compact Workstation	12
Descripción del sistema implementado	28
DESARROLLO	33
Desarrollo de ingeniería	33
Selección y procura de instrumentación y accesorios	38
Configuración y puesta en marcha	39
RECOMENDACIONES Y CONCLUSIONES	75
Recomendaciones.....	75
Conclusiones.....	76
REFERENCIAS	78
ANEXO A: CÓDIGOS DE PROGRAMACIÓN	80
ANEXO B: DATASHEETS DE LOS ELEMENTOS UTILIZADOS	87
ANEXO C: HOJAS GUÍAS DE ESTUDIANTE	88

TABLAS

Tabla 1. Elementos adquiridos

Tabla 2. Asignación de pines

Tabla 3. Tags y asignación de pines del EasyPort

Tabla 4. Asignación de tags a los sensores implementados

Tabla 5. Asignación de tags a los actuadores implementados

Tabla 6. Lista de tags sin asignación de dirección

FIGURAS

Figura 1. Tanque

Figura 2. Válvula proporcional

Figura 3. Sensor de presión

Figura 4. Sensor ultrasónico

Figura 5. Sensor de caudal

Figura 6. Sensor de temperatura

Figura 7. Bomba

Figura 8. Válvula de 2 vías

Figura 9. Calentador

Figura 10. PWM del calentador

Figura 11. Sensores capacitivos de proximidad

Figura 12. Sensor flotador para protección del sistema de calentamiento

Figura 13. Sensor flotador para evitar desbordes en el tanque inferior

Figura 14. Tablero de entradas/salidas

Figura 15. Sistema FESTO MPS-PA Compact Workstation

Figura 16. Diagrama de flujo del sistema de control de nivel

Figura 17. Diagrama de flujo del sistema de control de flujo

Figura 18. Diagrama de flujo del sistema de control de presión

Figura 19. Diagrama de flujo del sistema de control de temperatura

Figura 20. Sensor flotador de nivel tipo interruptor

Figura 21. Sensor de caudal

Figura 22. Sensor ultrasónico PING

Figura 23. Plataforma ARDUINO UNO

Figura 24. Bomba de Agua

Figura 25. Diagrama eléctrico

Figura 26. Diagrama de comunicación

Figura 27. Diagrama de conexión del sensor ultrasónico

Figura 28. Diagrama de conexión del sensor de caudal

Figura 29. Configuración de interruptores DIP

Figura 30. Diagrama de conexión del módulo EM 235

Figura 31. Interruptores DIP del cable multimaestro RS232/PPI

Figura 32. Asignación de pines del cable multimaestro RS232/PPI

Figura 33. Conexión PLC S7-200/PC mediante cable multimaestro RS232/PPI

Figura 34. Selección de tipo de CPU en STEP 7-Micro/WIN

Figura 35. Cuadro de diálogo para configuración de parámetros de comunicación en STEP 7

MicroWIN

Figura 36. Interface PG/PC

Figura 37. Configuración de parámetros PPI

Figura 38. Configuración de parámetros de la conexión local

Figura 39. Bloque de Sistema

Figura 40. Establecimiento de la comunicación

Figura 41. Programa en STEP 7 MicroWIN para acceso a variables del PLC S7-200 desde InduSoft Web Studio

Figura 42. Carga de programa en CPU

Figura 43. Conexiones entre el tablero de entradas y salidas con el EasyPort

Figura 44. Creación de nuevo proyecto en InduSoft Web Studio

Figura 45. Acceso a drivers de comunicación

Figura 46. Selección de driver de comunicación

Figura 47. Driver SIPPI

Figura 48. Configuración del driver SIPPI

Figura 49. Configuración de parámetros en la hoja del driver SIPPI

Figura 50. OPC DA 2.05

Figura 51. Creación de tags

Figura 52. Asignación de nombre y tipo de tag

Figura 53. Inserción de símbolos en la pantalla

Figura 54. Vinculación de tags a símbolos

Figura 55. Pantalla de la aplicación

Figura 56. Pantalla inicial de la interfaz gráfica

Figura 57. Escritura de código en InduSoft

INTRODUCCIÓN

En la actualidad la mayoría de los procesos industriales exigen una automatización debido a que las exigencias de producción son cada vez mayores. Esto justifica la importancia de cursos relacionados con control, sensores e instrumentación y sistemas SCADA en la formación de los alumnos de Ingeniería Electrónica.

La Universidad San Francisco de Quito adquirió un módulo didáctico de control de procesos en el 2011. Este módulo permite realizar prácticas de sensores, actuadores y sistemas SCADA, los cuales en conjunto conforman un proceso.

Tomando en cuenta estos antecedentes se decidió que el tema de tesis sea la implementación de una unidad de mezcla en el módulo didáctico MPS-PA Compact Workstation, de manera que se tenga un módulo complementario para ser utilizado, por estudiantes de futuras generaciones, en prácticas de laboratorio de cursos relacionados con control, sensores e instrumentación y sistemas SCADA.

La importancia de este proyecto se encuentra en ampliar el módulo didáctico con un proceso adicional para realizar prácticas de integración de sistemas de control. La extensión que será construido, puede ser utilizado en varias de las asignaturas de la carrera de Ingeniería Electrónica para prácticas de laboratorio, ya que éste integra varias ramas, etapas y niveles de ingeniería, pudiendo ser usada desde las clases introductorias de la carrera hasta las clases más avanzadas, entre la que se incluyen: Control Automático, Control Inteligente, Sistemas SCADA, Sensores e Instrumentación, entre otras. Este proceso complementario permitirá a los estudiantes implementar redes industriales, sistemas SCADA, y control distribuido.

DESCRIPCIÓN DEL SISTEMA

Descripción del sistema FESTO MPS-PA Compact Workstation

El sistema FESTO MPS-PA Compact Workstation fue desarrollado y diseñado con propósitos de entrenamiento en el campo de la automatización. Este sistema combina 4 lazos cerrados haciendo uso de sensores tanto digitales como analógicos y de actuadores. Los controladores o PLC's pueden utilizarse de manera individual o en cascada para realizar los lazos de control correspondiente a nivel, caudal, presión y temperatura.

Componentes básicos del sistema.

Los componentes básicos que el sistema posee y su respectiva función se presentan a continuación:

Tanques graduados.

Estos tanques rectangulares cuentan con una capacidad de 10 L y pueden ser utilizados tanto para fluidos como para sólidos. Para la descarga, cada tanque tiene un agujero con rosca de conexión en la parte inferior del recipiente. En la parte lateral poseen agujeros con rosca de conexión los cuales sirven para las entradas y sensores. También se proporciona agujeros para el montaje de un elemento de calentamiento y el sensor de temperatura. La tapa tiene varias aberturas, de las cuales una está destinada para el agitador. En el sistema, los tanques se encuentran conectados entre sí, de forma que el líquido refrigerante puede ser transferido de un tanque a otro de acuerdo al proceso que se desee llevar a cabo.

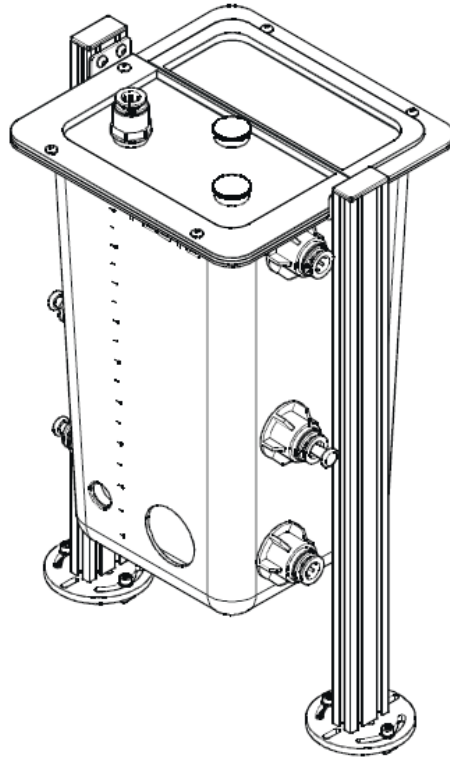


Figura 1. Tanque, Festo Didactic GmbH & Co. KG, 2006.

Válvula proporcional.

La válvula proporcional ayuda a controlar con mayor facilidad el flujo de gases neutros o líquidos. Se puede utilizar como un elemento final de control a distancia o en lazos cerrados de control. Este tipo de válvula es de accionamiento directo 2/2 vías. El pistón de la válvula se levanta de su asiento en función de la corriente de la bobina del solenoide y libera el flujo de la conexión 1 a la 2. Una vez que la válvula se encuentra desenergizada, se cierra por medio de un muelle de reposición. La señal de control se convierte en una señal de PWM, de manera que mediante ésta se controla el pistón de la válvula. La frecuencia de la señal PWM puede cambiarse de acuerdo al tipo de válvula que se utiliza.

Sensor análogo ultrasónico.

Su principio de funcionamiento se basa en la generación de ondas acústicas y su detección a través de la reflexión por medio de un objeto.

Durante un breve periodo, un generador de ultrasonidos actúa emitiendo un pulso de energía ultrasónica, el cual es inaudible para el oído humano. Después el pulso emitido es reflejado a través de un objeto ubicado dentro de cierto rango y rebota hacia el receptor.

En condiciones de fábrica, el incremento de la distancia a medir entre el sensor y el objeto produce un aumento en la señal de salida. Tomando en cuenta que en la medición de nivel en un tanque, la distancia entre el sensor y la superficie del líquido disminuye, la configuración de la señal de salida fue invertida, de tal forma que a menor distancia mayor es la señal de salida del sensor. A más de esto, el rango de medición del sensor se encuentra ajustado al tanque.



Figura 4. Sensor ultrasónico, Festo, 2013.

Sensor de caudal.

El líquido transparente que fluye en la dirección de la flecha, adopta un movimiento centrífugo a través del cuerpo en rotación que se encuentra en la cámara de medición y posteriormente es dirigido sobre el rotor de tres paletas. La velocidad del rotor es proporcional al caudal y se detecta sin contacto por medio del sistema opto-electrónico infrarrojo que tiene incorporado (diodo y fototransistor).

El amplificador integral suministra una señal constante de onda cuadrada, por lo cual el nivel de la señal depende del voltaje de alimentación aplicada (8 - 24 V DC). El rotor posee una configuración especial que evita la dispersión de las potenciales burbujas de gas presentes en el líquido pero las transporta con el mismo.

La posición de montaje es arbitraria. La dirección del flujo es indicada mediante una flecha ubicada en la carcasa del sensor. En su entrada, el sensor posee un filtro protector.



Figura 5. Sensor de caudal, Helmich, 2008.

Sensor de temperatura.

El sensor de temperatura comprende un termómetro de resistencia hecho de platino y un inserto de medición intercambiable. Consiste en un tubo protector, una cabeza de conexión y el inserto de medición. La absorción o suministro de calor a través del sensor se debe evitar.

En el gráfico que se muestra a continuación, el número 1 corresponde al tornillo de ajuste(x2), el número 2 la vaina protectora y el número 3 al termoelemento.

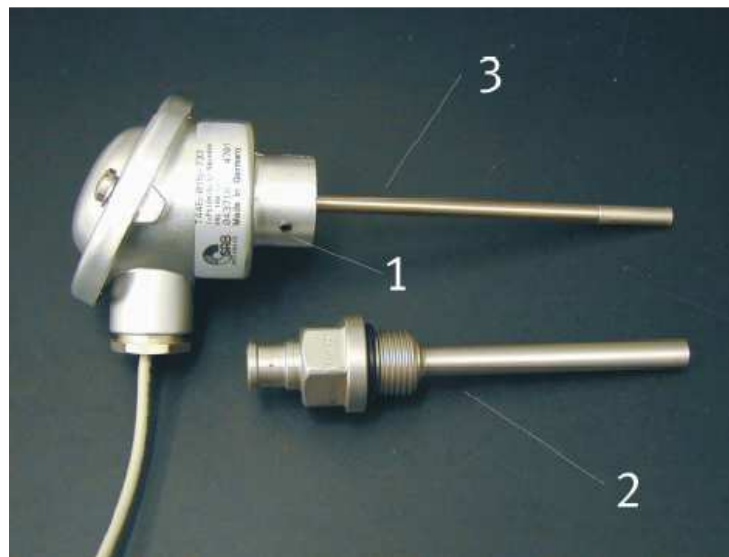


Figura 6. Sensor de temperatura, Helmich, 2008.

Bomba.

En el sistema de control, la bomba es utilizada como equipo de control y es la encargada de entregar el fluido del tanque de depósito a través del sistema de tuberías.

La bomba es accionada por el controlador de un motor y un relé. Con una salida digital, es posible cambiar de control digital binario a control analógico variable de 0 a 24 V. En el control digital binario, la bomba es encendida/apagada con una salida adicional. En el control

analógico, la tensión de accionamiento del canal de la señal de salida analógica es la que ajusta la velocidad de la bomba de 0 a 10 V.



Figura 7. Bomba, Helmich, 2008.

Válvula de 2 vías.

La válvula de bola 2-vías se abre y se cierra mediante un accionamiento giratorio neumático.

El equipo controlado consiste en una válvula de bola de latón con accionamiento rotativo.

Una válvula de 5/2 vías solenoide es montada sobre la brida de accionamiento giratorio. El flujo de fluido que va desde el tanque superior hacia el tanque inferior es controlado con la válvula de bola del disco rotatorio.



Figura 8. Válvula de 2 vías, Helmich, 2008.

Calentador.

El calentador posee un microcontrolador interno, el cual permite su control.

El control del calentador presenta dos modos de operación: análogo o binario. La unidad de calentamiento funciona mediante una fuente de alimentación AC de 230V. Si se utiliza el modo digital de operación, el calentador se enciende y apaga por medio de una entrada de 24V. En el caso de que el modo de operación empleado sea el análogo, la potencia de calefacción puede ser ajustada infinitamente a través de una señal de 0 a 10V.



Figura 9. Calentador, Helmich & ADIRO, 2008.

En el caso particular de la estación MPS-PA Compact Workstation, el control se lo realiza únicamente a través de la activación de una salida binaria. Para utilizar el calentador como un elemento continuo de salida se debe emplear modulación de ancho de pulsos, conocido como PWM. Básicamente esta modulación consiste en el envío de una secuencia de pulsos, la cual tiene un periodo constante T como se muestra en la figura 9. El control del calentador se

realiza a través del tiempo de encendido t_1 y el de apagado t_2 . En el gráfico Q representa la potencia térmica.

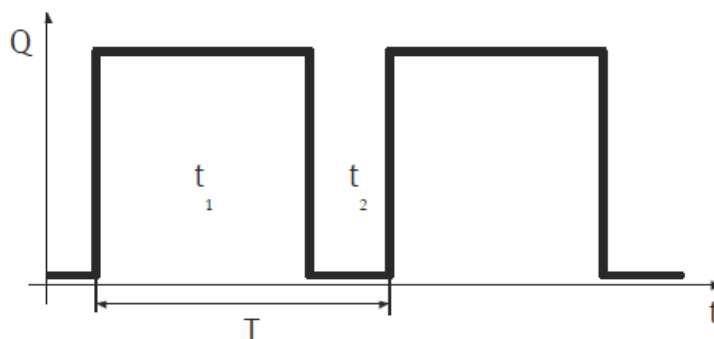


Figura 10. PWM del calentador, Helmich & ADIRO, 2008.

Sensor capacitivo de proximidad (interruptor).

El principio operacional de este sensor se basa en la evaluación de un cambio en la capacitancia de un capacitor en un circuito resonante RC. Si un material se acerca al sensor de proximidad, la capacitancia del capacitor aumenta. Esto lleva a un cambio en la respuesta de vibración del circuito RC, el cual es evaluado posteriormente. El cambio en la capacitancia depende principalmente de la distancia, dimensiones y de las constantes dieléctricas de los respectivos materiales. El sensor de proximidad tiene una salida PNP. El interruptor es designado en la forma de un contacto normalmente abierto. La conexión de la carga se efectúa entre la señal de salida del sensor de proximidad y tierra. La salida de este sensor es una señal binaria de 24V. El sistema FESTO MPS-PA Compact Workstation tiene 2 sensores capacitivos de proximidad en la parte trasera del tanque inferior que permiten monitorear el nivel del líquido refrigerante.



Figura 11. Sensores capacitivos de proximidad, Helmich, 2008.

Sensor flotador como protección al sistema de calentamiento (interruptor).

Este sensor flotador es un interruptor que puede ser utilizado como una protección para el calentador. El sensor puede ser montado en dos posiciones distintas, pero de manera particular se encuentra en la posición normalmente abierta en el tanque del módulo. El interruptor se enciende en el nivel seguro de líquido. En el nivel seguro de agua, el calentador se encuentra totalmente sumergido en el líquido, de forma que éste (el calentador) se puede encender únicamente si se encuentra rodeado por completo de líquido.

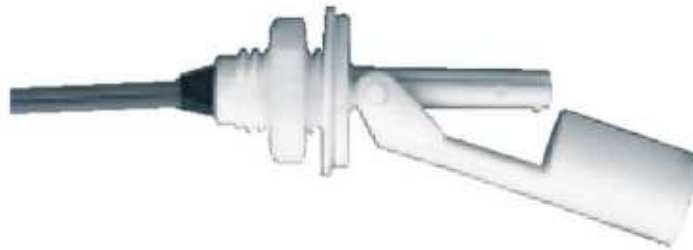


Figura 12. Sensor flotador para protección del sistema de calentamiento, Festo Didactic GmbH & Co. KG, 2006.

Sensor flotador para el control de desbordamiento en el tanque inferior.

Este sensor de flujo es utilizado para tanques planos o en espacios restringidos y su diseño permite exclusivamente un montaje vertical.

El medio a ser medido presiona el flotador hacia arriba y acciona un interruptor a partir de una posición específica.



Figura 13. Sensor flotador para evitar desbordes en el tanque inferior, Festo Didactic GmbH & Co. KG, 2006.

Tablero de entradas/salidas.

El tablero de conexión sirve como una interfaz para las señales analógicas y digitales de entrada y salida. Todas las señales analógicas se convierten en una señal que varía de 0 a 10 V y son aplicadas al terminal analógico. Se pueden tener como máximo 8 entradas binarias y 8 salidas binarias por estación. las cuales son aplicadas al terminal de entrada/salida. Esto asegura compatibilidad con EasyPort, SimuBox, EduTrainer y tableros de PLC.

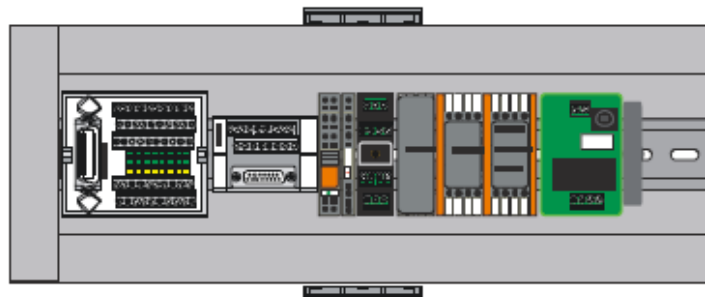


Figura 14. Tablero de entradas/salidas, Helmich, 2008.

A continuación se muestra una figura del sistema completo, sin incluir la etapa de mezcla, puesto que su descripción y presentación se realiza en la siguiente sección.

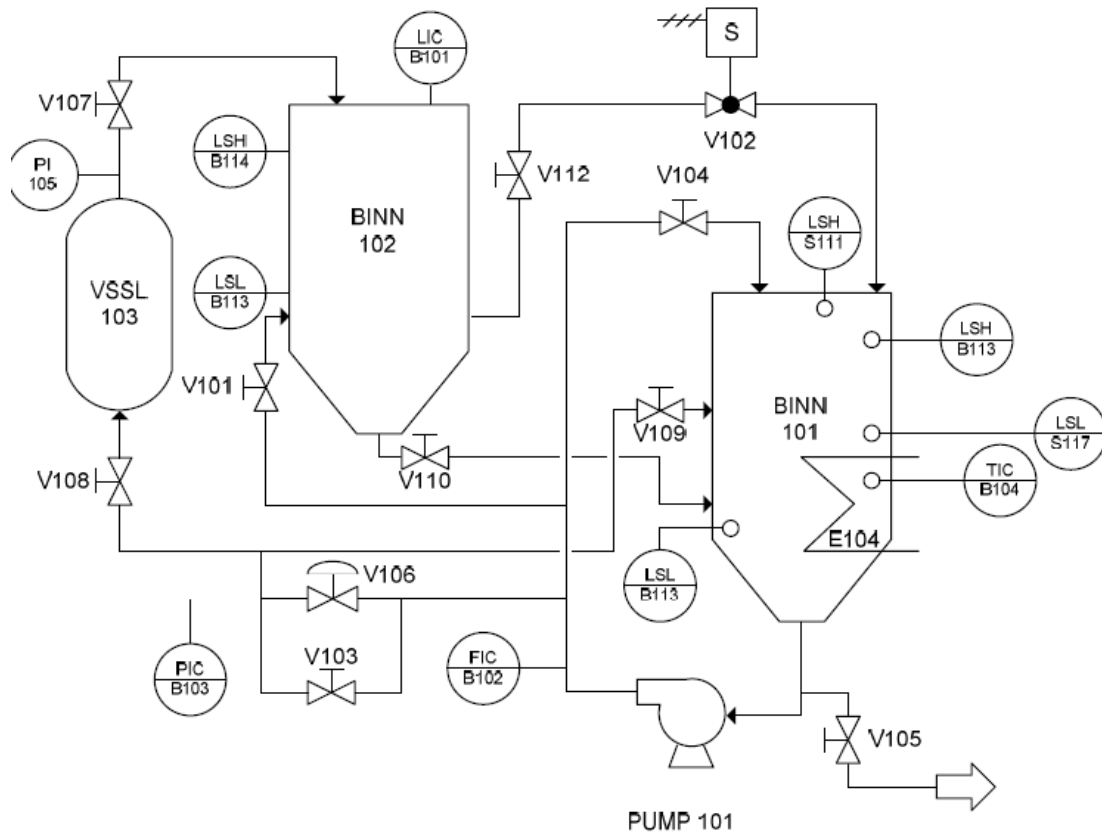


Figura 15. Sistema FESTO MPS-PA Compact Workstation, Helmich & ADIRO, 2008.

Funciones de los lazos de control cerrado.

Función de control de nivel.

El sistema de nivel controlado cumple con la función de regular el nivel de llenado de un líquido en un tanque de depósito. La bomba P101 entrega fluido de un tanque de almacenamiento B101 a otro B102 a través de un sistema de tuberías. El nivel del líquido dentro del tanque B102 se controla con un sensor ultrasónico analógico B101 y se lee como valor real. Este valor real debe mantenerse en un cierto nivel, incluso si ocurren perturbaciones o cambios de set point. La cantidad de fluido que se bombea puede ser un

valor binario o manipulado. Para las perturbaciones se puede abrir/cerrar total o parcialmente la válvula de bola V102 para drenar el fluido desde el tanque superior hacia el inferior o abrir/cerrar la válvula manual V104. En la Figura 16 se presenta el lazo de control de nivel.

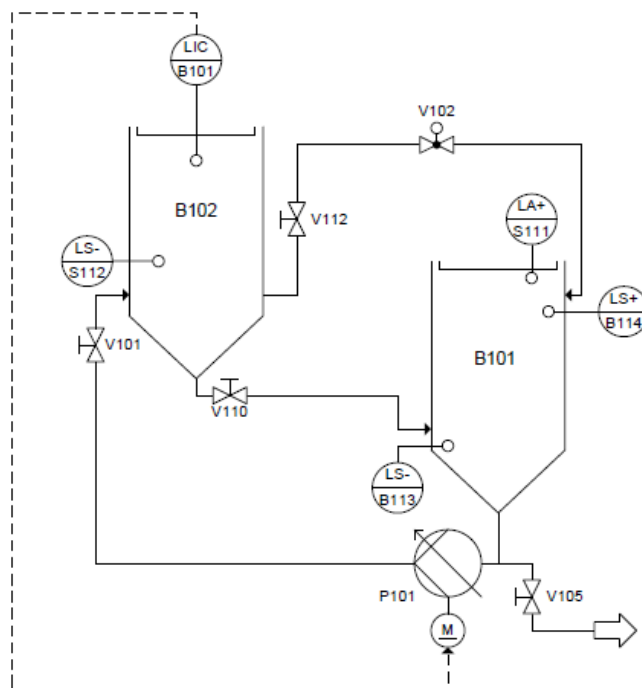


Figura 16. Diagrama de flujo del sistema de control de nivel, Helmich & ADIRO, 2008.

Función de control de flujo.

La bomba suministra un fluido desde el depósito a través de un sistema de tuberías. La velocidad de flujo es detectada por medio de un sensor opto-electrónico de paletas B102 en forma de un valor real. El valor real debe mantenerse en un cierto caudal a pesar de las perturbaciones o cambios de set-point que puedan ocurrir.

Existen dos modos de operación.

- a. *Control del caudal por medio de la bomba P101 como sistema controlado.*

El valor que se manipula es el voltaje de la bomba, el cual establece la velocidad de rotación.

b. *Control del caudal por medio de la válvula proporcional V106 como sistema controlado.*

El valor que se manipula es el voltaje de la bobina de la válvula. La bomba P101 debe operar a velocidad constante.

Para las perturbaciones, se puede abrir o cerrar la válvula manual V104 ya sea parcial o completamente.

La Figura 17 muestra el diagrama de flujo para el control de flujo.

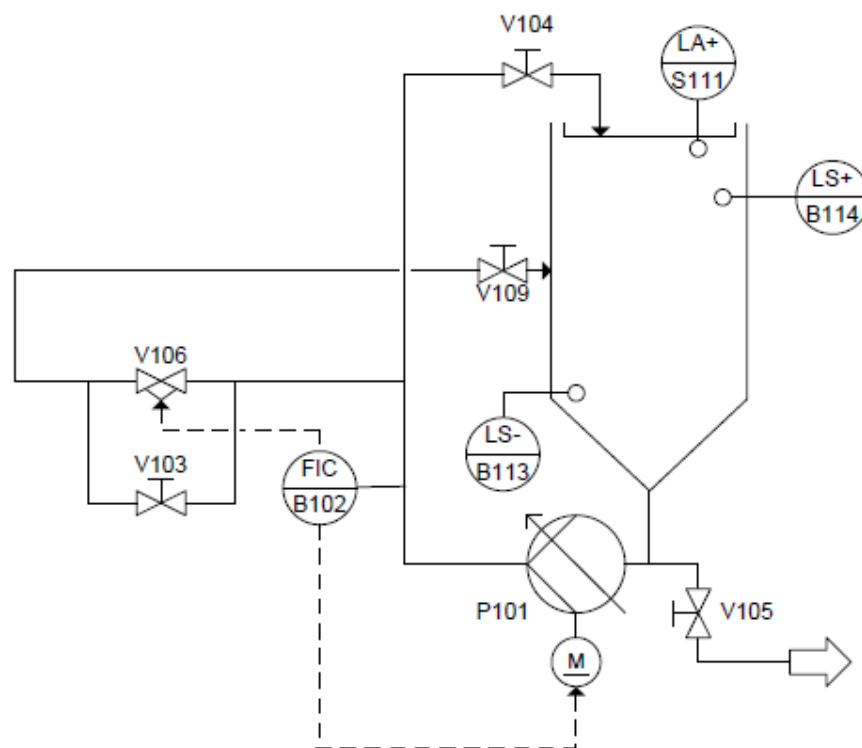


Figura 17. Diagrama de flujo del sistema de control de flujo, Helmich & ADIRO, 2008.

Función de control de presión.

A través de un sistema de tuberías, la bomba P101 entrega líquido desde el tanque de depósito hacia el tanque de presión B103. La presión del aire en el tanque de presión se detecta mediante un sensor piezo resistivo de presión relativa en forma de un valor real.

Dicho valor real debe mantenerse en una cierta presión incluso si se presentan perturbaciones o cambios de set-point. El sistema de control de presión se muestra en la Figura 18.

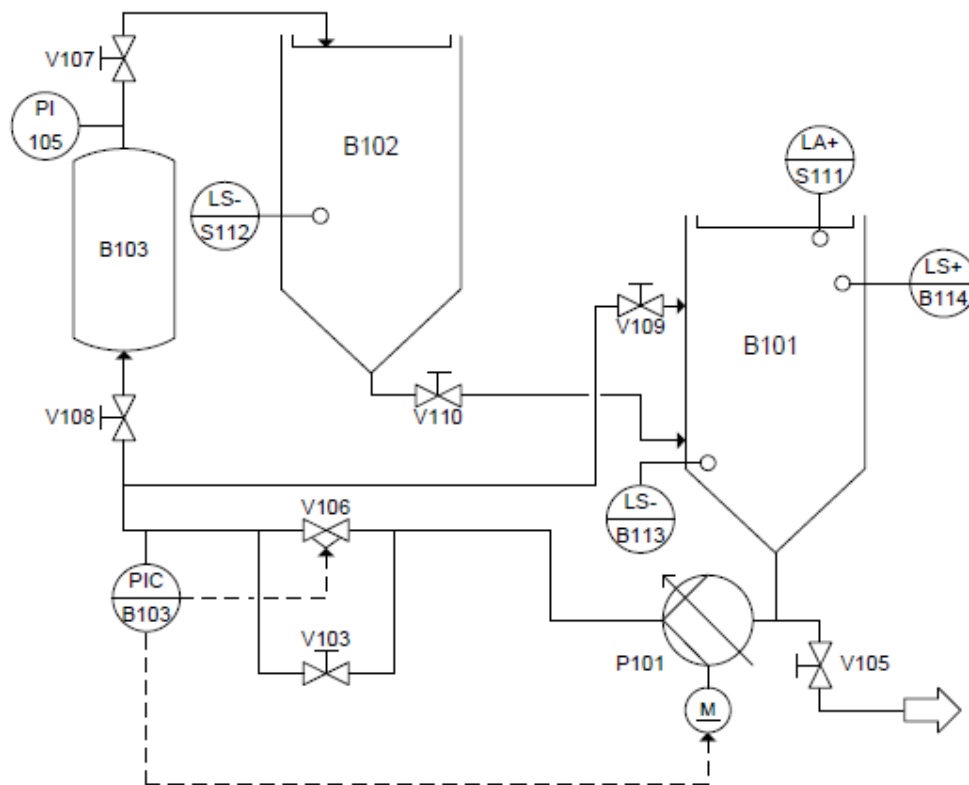


Figura 18. Diagrama de flujo del sistema de control de presión, Helmich & ADIRO, 2008.

Función de control de temperatura.

El tanque B101 tiene incorporado un intercambiador de calor E104. El líquido contenido en este tanque aumenta su temperatura por medio de la unidad de calentamiento y recircula a través de la bomba P101. La temperatura del sistema se mide utilizando un sensor de temperatura B104. El valor medido es real y debe mantenerse en un determinado valor sin importar la presencia de perturbaciones o cambios de set-point. Su respectivo diagrama de flujo se indica en la Figura 19.

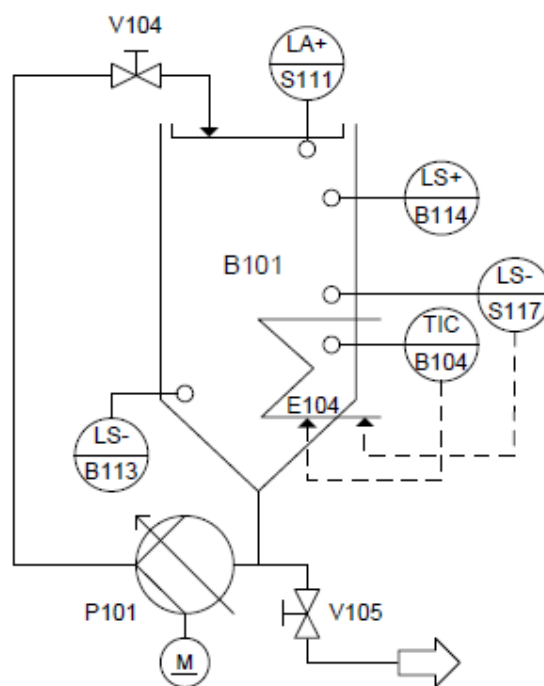


Figura 19. Diagrama de flujo del sistema de control de temperatura, Helmich & ADIRO, 2008.

Descripción del sistema implementado

Esta sección presenta una descripción de todos los elementos utilizados para la implementación de la etapa de mezcla.

En primer lugar fue necesario adquirir otro tanque de depósito igual a los del módulo original. Su respectiva descripción fue realizada en la sección anterior. El resto de elementos utilizados se detallan a continuación.

Componentes del sistema implementado.

Sensor flotador de nivel 1 (Interruptor).

Este sensor se lo utiliza para determinar el nivel bajo en el tanque implementado, se lo monta de manera horizontal, y consiste en un flotador, el cual cierra el circuito cuando el nivel del líquido alcanza o sobrepasa el nivel al cual está instalado el sensor.



Figura 20. Sensor flotador de nivel tipo interruptor.

Sensor flotador de nivel 2 (Interruptor).

Es idéntico al de la figura 14, se lo utiliza como protección para desbordes en el tanque incorporado. El sensor viene incluido en el tanque.

Sensor de caudal.

El sensor de caudal es tipo turbina, al pasar el líquido por el sensor, la turbina gira y como resultado se obtienen una señal en frecuencia directamente proporcional al flujo del líquido.



Figura 21. Sensor de caudal.

Sensor ultrasónico PING.

Este sensor se lo utiliza para medir el nivel de agua en el tanque, lo mide de manera indirecta, ya que lo que está midiendo en realidad es la distancia a la que se encuentra el líquido. Para determinar la distancia este sensor emite una señal ultrasónica, la cual al chocar en una superficie rebota al sensor, y de esta manera mediante el tiempo que se demora en ir y regresar se determina la distancia a la que se encuentra la superficie, en este caso el líquido.



Figura 22. Sensor ultrasónico PING, ROBODACTA, 2011-2013.

ARDUINO UNO.

Esta plataforma de hardware libre, basada en un microcontrolador y un entorno de desarrollo, diseñada para facilitar el uso de la electrónica en proyectos multidisciplinarios. Se utilizó un ARDUINO UNO para procesar las señales enviadas por el sensor de nivel y el sensor de caudal, ya que se necesita una señal análoga que vaya de 0 a 5 V para poder ser procesada por el PLC S7 200. La salida del ARDUINO es una señal PWM, por lo que se debe acondicionar la señal usando un filtro, para de esta manera obtener las señales análogas de 0 a 5V directamente proporcional al nivel y al caudal, que necesita el PLC S7 200.



Figura 23. Plataforma ARDUINO UNO, Arduino.cc, 2013.

Bomba.

Es una bomba que funciona con 24V DC, capaz de bombear hasta 10.8 L/m, se utiliza para desplazar el líquido desde el tanque instalado hasta el tanque resultante.



Figura 24. Bomba de Agua, Aliexpress, 2013.

DESARROLLO

Desarrollo de ingeniería

Proceso.

P&ID.

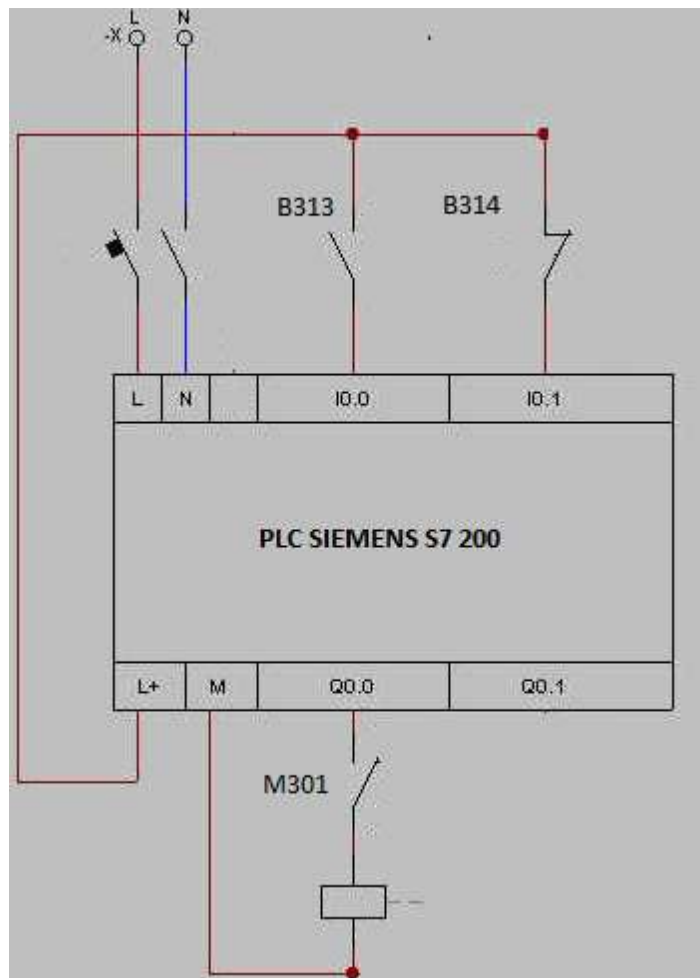
Diagrama eléctrico.

Figura 25. Diagrama eléctrico.

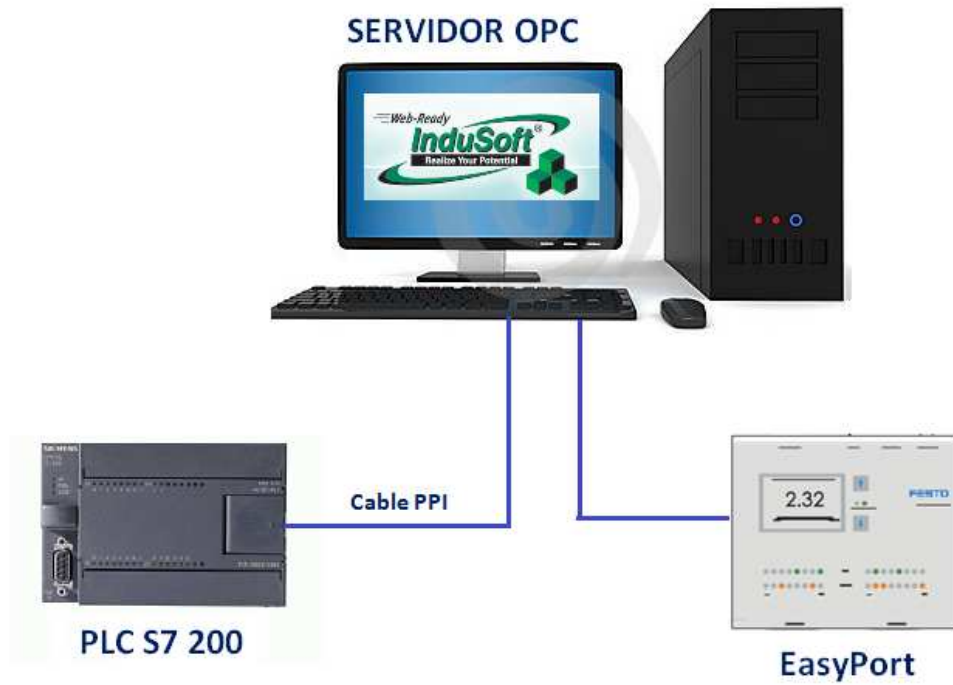
Diagrama de comunicación.

Figura 26. Diagrama de comunicación.

Diagrama mecánico.

Selección y procura de instrumentación y accesorios

Para la selección y procura de la instrumentación y accesorios utilizados en la implementación de la estación de mezcla, fue necesario tomar en cuenta factores importantes como el precio, la calidad, la disponibilidad y la apariencia estética. El primer y segundo punto mencionados son muy importantes debido a que por una parte se deseaba conseguir elementos cuyo costo no sea muy elevados, pero al mismo tiempo se necesitaba garantizar que éstos sean de buena calidad para el correcto funcionamiento de nuestro sistema y para su perdurabilidad en el tiempo. También se debió considerar la disponibilidad de los productos en el mercado local, ya que muchos de los productos que cumplían con las características deseadas se encontraban fuera del país y el importarlos significaba un costo adicional y tiempo. A pesar de esto, en el caso del tanque, su compra se realizó en el exterior, debido a que en Ecuador no se disponía de uno que se ajustara a nuestras necesidades. Otro de los criterios considerados fue la apariencia estética ya que se deseaba que el trabajo se vea lo más profesional posible y para esto se requería que la instrumentación y accesorios tengan buenos acabados. Una vez que toda la instrumentación y accesorios fueron escogidos se elaboró un presupuesto de todos los materiales requeridos. A continuación se presenta en detalle cada uno de los elementos adquiridos con su respectivo costo:

Tabla 1. Elementos adquiridos.

Item	Cantidad	P. unitario (USD)	P. total (USD)
Tanque de 10 L	1	393,08	393,08
Sensor de caudal	1	17,19	17,19
Sensor de nivel (Switch)	1	6	6
Cable UTP flexible	3	0,14	0,42
Borneras	2	0,5	1
Taípe	1	0,6	1,2
Lija	2	0,55	1,1
Tornillos Arduino	3	0,04	0,12
Tornillos para soporte	2	0,05	0,1
Llaves Allen STD Pretul	1	1,17	1,17
Sensor ultrasónico PING	1	50	50
Spaguetti 3mm	2	0,46	0,92
Spaguetti 6mm	2	0,76	1,62
Arduino UNO	1	46	46
Bomba	1	48	48
Tapón	1	2	2
Cable UTP rígido	3	0,14	0,42
Piezas de soporte	2	10	20
Cortadora	1	3,5	3,5
Manguera	1	7	7
Tornillos soporte bomba	3	0,1	0,3
Teflón	1	1,25	1,25
Cables de conexión	10	0,2	2
Codo	4	7,83	31,32
T	1	9,43	9,43
Válvula Check	3	7,8	23,4
Tubería	2	13,74	27,48
Uniones	2	7,67	15,34
		Total sin IVA	711,36
		12% IVA	85,3632
		Total	796,7232

Configuración y puesta en marcha

Una vez concluido el diseño y montaje físico de la etapa de mezcla, se procedió con la configuración de las comunicaciones necesarias y la programación del proceso de mezcla.

Procesamiento de señales mediante la plataforma ARDUINO UNO.

Normalmente las señales análogas de los sensores no vienen los rangos adecuados, por lo cual es necesario "acondicionarlas" para obtener salidas de voltaje o corriente en los rangos adecuados para la aplicación en la que serán utilizados. Particularmente, en este proyecto fue necesario procesar tanto la señal del sensor ultrasónico PING como la del sensor de caudal, mediante la utilización de la plataforma ARDUINO UNO. La programación se la realizó en Arduino-1.0.2, software propio de este microprocesador. Para la programación del sensor ultrasónico, se tomó como base un código que aparecía en la lista de ejemplos de Arduino-1.0.2. Este programa permitía leer la distancia entre el sensor y un objeto, a través del monitor serial del software. El algoritmo toma el tiempo que la onda ultrasónica tarda en ir y regresar, y luego se utiliza este dato para calcular la distancia hasta la superficie del líquido, teniendo en cuenta que la velocidad del sonido es conocida. El programa fue modificado para obtener un voltaje de salida entre 0 y 5V, que variara en forma inversamente proporcional con respecto a la distancia medida. Por otra parte, para el sensor de caudal, se utilizó como guía un programa encontrado en internet, pero al igual que para el sensor ultrasónico, fue necesario modificar el código, de manera que a la salida se obtuviera un voltaje con un rango de 0-5V. Finalmente, se incorporaron ambos programas en un solo código, debiendo realizarse algunos ajustes en los tiempos de toma de datos. También se colocaron filtros RC en las salidas de voltaje, para obtener un promedio del PWM a la salida del ARDUINO UNO. El código realizado se presenta en el ANEXO A.

Se presenta una tabla con la asignación de pines de ARDUINO UNO.

Tabla 2. Asignación de pines.

ARDUINO UNO	SENSOR
Pin 2 (DIGITAL)	Señal del sensor de caudal
Pin 7 (DIGITAL)	Señal del sensor ultrasónico
Pin 9 (DIGITAL)	Salida sensor ultrasónico (0-5V)
Pin 10 (DIGITAL)	Salida sensor de caudal (0-5V)

En la Figura 27 y 28 se muestran los diagramas de conexión tanto del sensor ultrasónico como del de caudal:

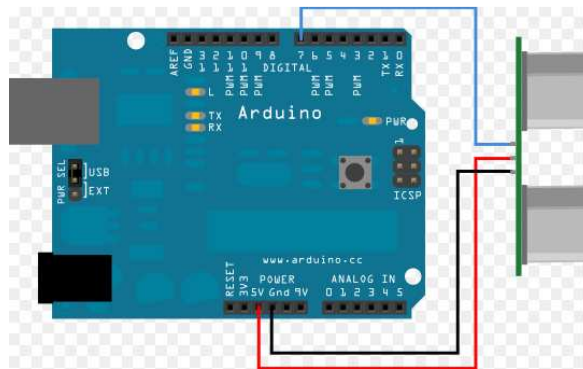


Figura 27. Diagrama de conexión del sensor ultrasónico, Arduino, 2013.

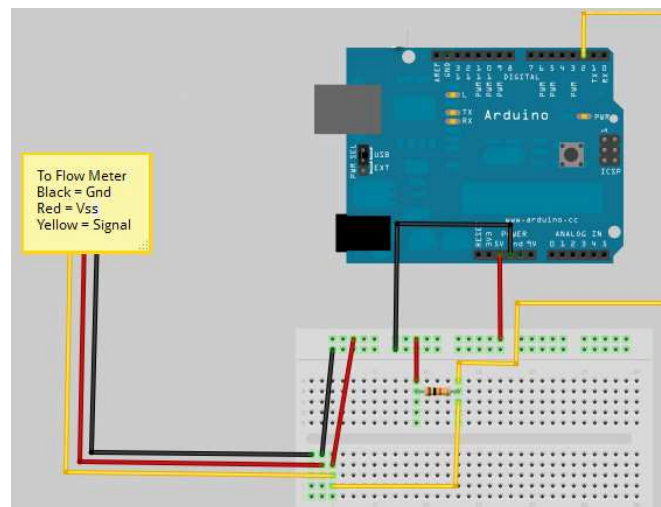


Figura 28. Diagrama de conexión del sensor de caudal, The Makers Workbench, 2013.

Incorporación del módulo de expansión análogo EM 235.

El CPU 226 permite trabajar únicamente con entradas y salidas digitales, razón por la cual se debió utilizar el módulo de expansión EM 235, el cual permite trabajar hasta con 4 entradas análogas y hasta con 2 salidas análogas. En este caso se trabaja únicamente con entradas análogas (0-5V) correspondiente a los sensores ultrasónico y de caudal. El módulo EM 235 puede trabajar con diferentes rangos de corriente o voltaje en sus entradas, ya que los transforma a formato de palabra, entregando a la salida un valor que puede ir desde -32000 a 32000, dependiendo de la configuración de los interruptores. Es así que, considerando el rango de las entradas y escogiendo "Unipolar" como formato de palabra (se obtendrán valores entre 0 y 32000), los interruptores DIP del módulo fueron configurados como se muestra en la Figura 29.

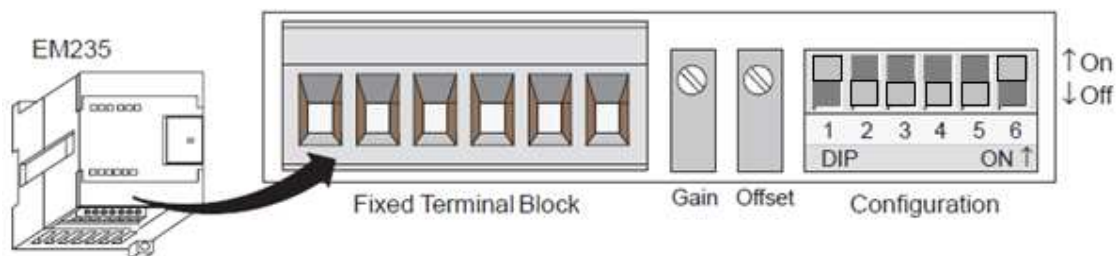


Figura 29. Configuración de interruptores DIP, Pomares, 2009.

En la Figura 30 se muestra el diagrama de conexión del módulo:

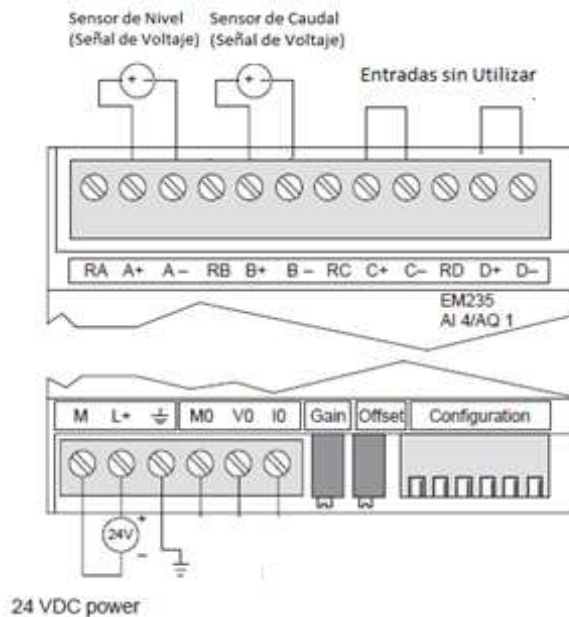


Figura 30. Diagrama de conexión del módulo EM 235, Pomares, 2009.

Configuración de Comunicaciones entre el PLC S7-200 y la computadora.

La comunicación entre el PLC S7-200 y la computadora fue establecida mediante la configuración de parámetros utilizando STEP 7 MicroWIN SP2 V3.1.

Conexión física.

El PLC Siemens S7-200 se conecta físicamente al computador a través del cable multimaestro RS-232/PPI de Siemens. El cable multimaestro sirve como interfaz de RS-232 a RS-485 y permite trabajar a diferentes velocidades de transmisión, según lo que sea requerido. Además de esto, presenta unos interruptores DIP que requieren ser configurados correctamente de acuerdo a la velocidad y modo de transmisión con los cuales se desea operar. Ver Figura 31.



Figura 31. Interruptores DIP del cable multimaestro RS232/PPI, SIEMENS AG, 2008.

En la Figura 32 se muestra la asignación de pines proporcionada por el fabricante:

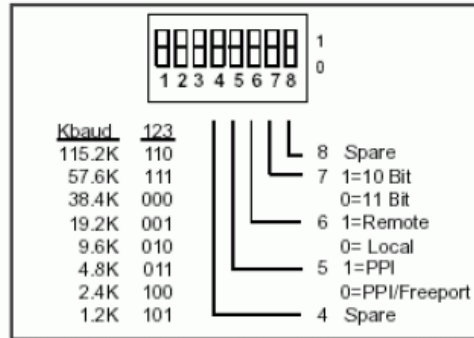


Figura 32. Asignación de pines del cable multimaestro RS232/PPI, SIEMENS AG, 2008.

En este proyecto se escogió una velocidad de transmisión de 9.6 Kbps en modo PPI/Freeport, por lo cual la configuración a utilizarse es 01000000 (Pin 2 en alto). El conector RS-232 (PC) del cable va conectado al puerto de comunicación de la PC (COM), mientras que el conector RS-485 (PPI) del cable se conecta al puerto 0 ó 1 (en este caso al 1) del PLC S7-200, tal como se muestra en la Figura 33.

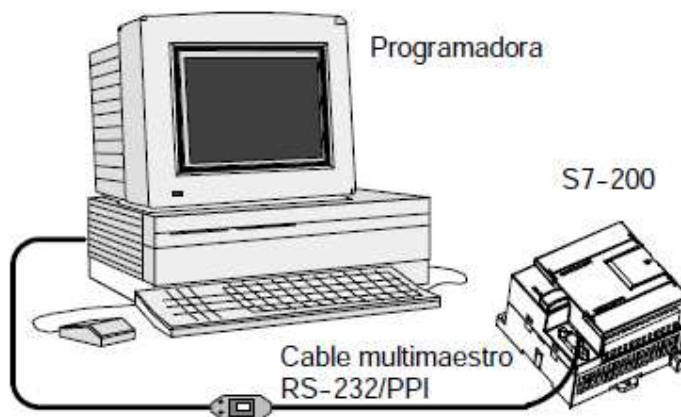


Figura 33. Conexión PLC S7-200/PC mediante cable multimaestro RS232/PPI, SIEMENS

AG, 2008.

Configuración de la comunicación.

Protocolo de comunicación.

Se utiliza el protocolo serial S7 PPI, el cual fue desarrollado por Siemens para el uso en sus equipos de campo.

Configuración del Software STEP 7 MicroWIN.

Para establecer la comunicación entre el PLC S7-200 y la computadora, en primer lugar se requiere configurar ciertos parámetros de comunicación en el software STEP 7 MicroWIN. Entre estos parámetros se encuentra la dirección remota del PLC, la velocidad de transmisión de los datos, la dirección del computador, entre otros.

Procedimiento de configuración de comunicación del S7-200

- Se crea un nuevo proyecto en STEP 7 MicroWIN. Se escoge el tipo de CPU que en este caso es el 226, como se puede apreciar en la Figura 34.

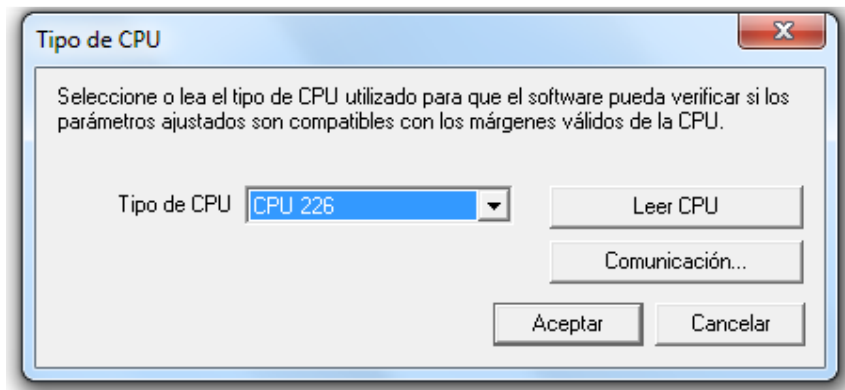


Figura 34. Selección de tipo de CPU en STEP 7-Micro/WIN, STEP 7-Micro/WIN V3.1.

- Después, en la barra de navegación ubicada en la parte izquierda del monitor se selecciona el ícono de comunicación y se configuraran los parámetros de comunicación que aparecen en la ventana "Enlaces de comunicación". Ver Figura 35. En la parte inferior del cuadro, aparece la dirección remota del PLC, cuyo valor estándar es 2. En este caso se conserva este último valor pero cabe mencionar que el mismo puede ir de 0 a 126.

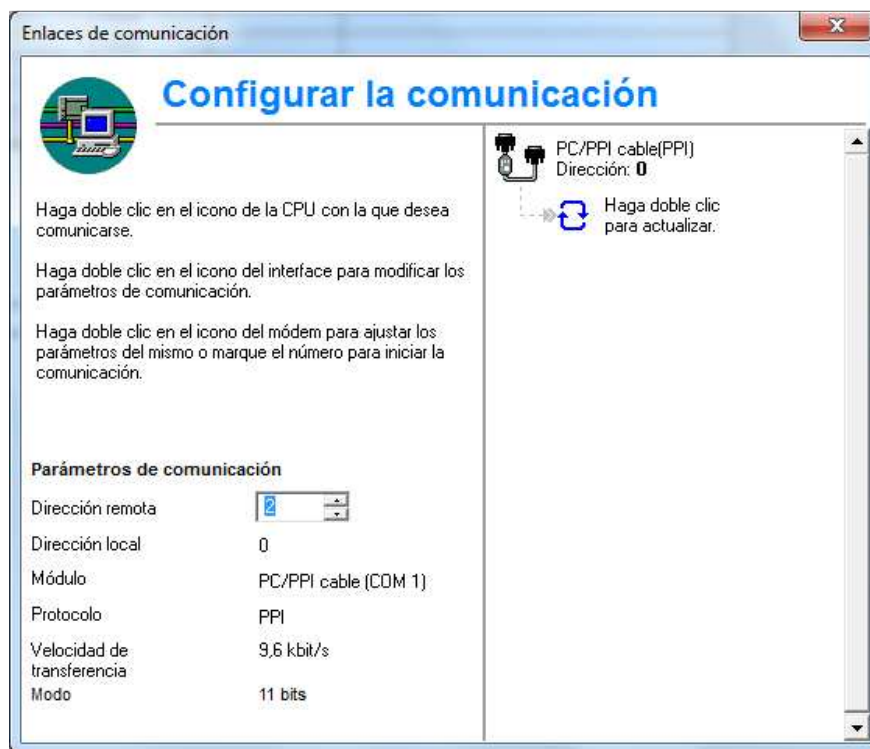


Figura 35. Cuadro de diálogo para configuración de parámetros de comunicación en STEP

7 MicroWIN, STEP 7-Micro/WIN V3.1.

- A continuación para modificar el resto de parámetros de comunicación como la velocidad de transferencia y la dirección de estación de STEP 7 MicroWIN, se selecciona el ícono de la parte superior derecha del cuadro de diálogo y se escoge la opción PC/PPI cable(PPI) como se muestra en la Figura 36.

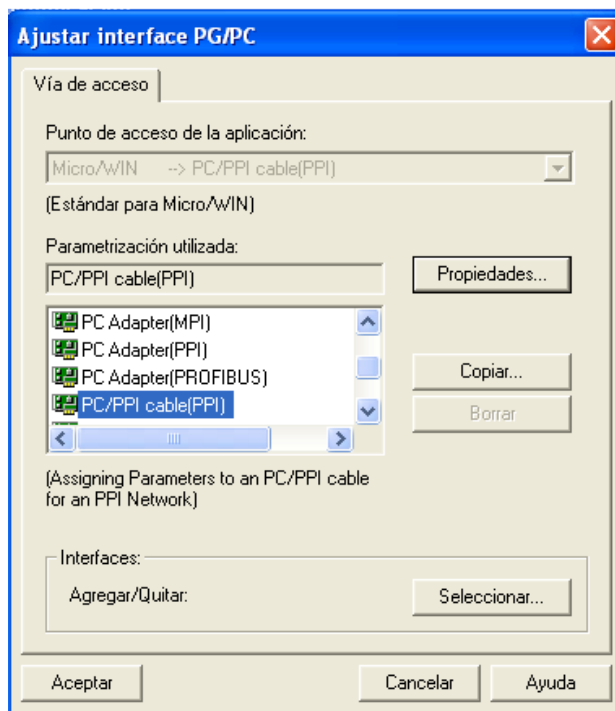


Figura 36. Interfaz PG/PC, STEP 7-Micro/WIN V3.1.

- Posteriormente en "Propiedades", en la pestaña de "PPI" se configura la dirección de la estación. Es necesario considerar que la dirección asignada debe tomar un valor diferente al utilizado para la CPU (es decir diferente a 2). En esta ocasión se asigna el valor 12. Además, se escoge 9.6 kbps como la velocidad de transmisión. El resto de parámetros se mantienen con los valores preestablecidos. Ver Figuras 37 y 38.

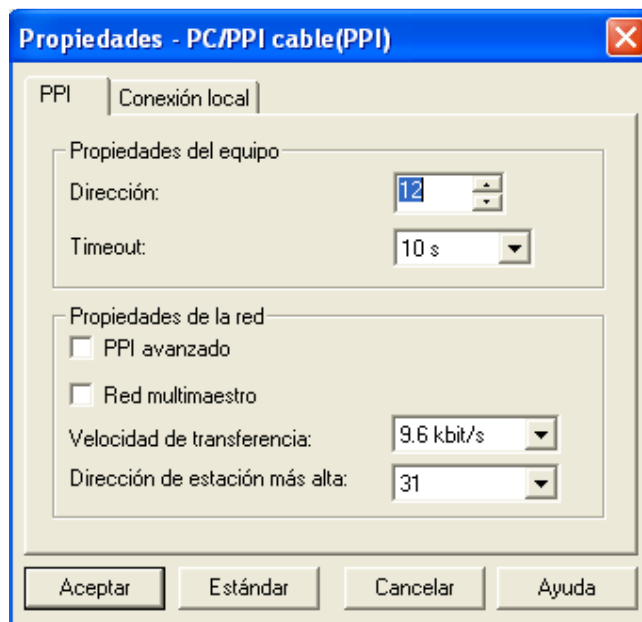


Figura 37. Configuración de parámetros PPI, STEP 7-Micro/WIN V3.1.

- Luego, en la pestaña etiquetada como "Conexión local", se selecciona COM 1.

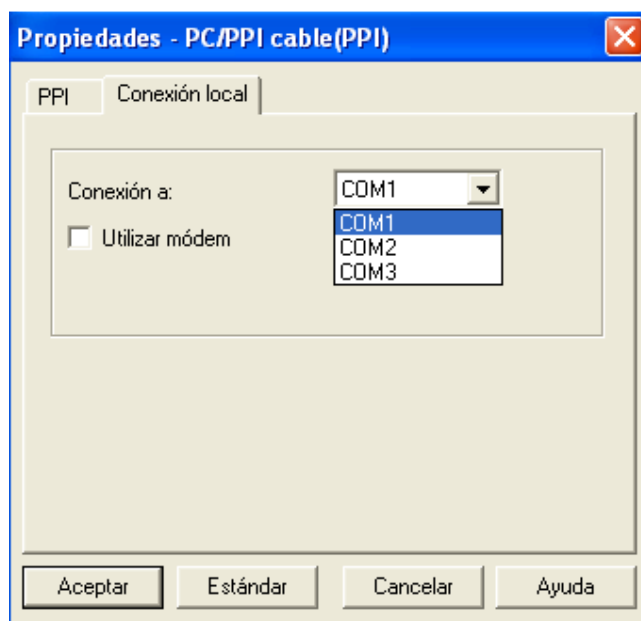


Figura 38. Configuración de parámetros de la conexión local, STEP 7-Micro/WIN V3.1.

- A continuación se selecciona el ícono "Bloque de sistema" y se corrobora que el puerto del PLC que se está utilizando (el 1 en este caso) tenga asignada la dirección remota correspondiente del PLC (en este caso la 2). Esto se aprecia en la Figura 39.

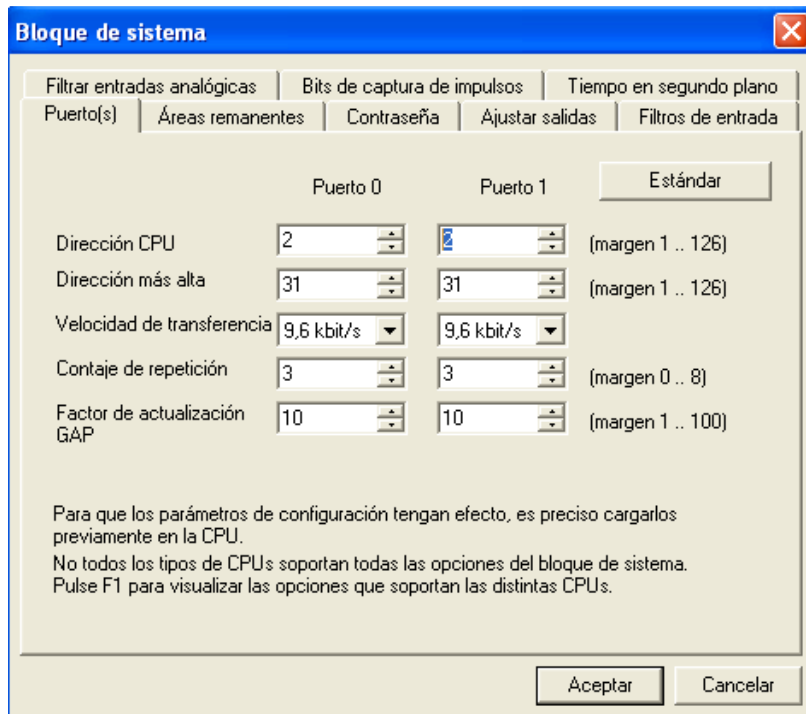


Figura 39. Bloque de Sistema, STEP 7-Micro/WIN V3.1.

- Por último, nuevamente en el ícono "Comunicación", se da doble click en el ícono de actualización. Se espera unos segundos hasta que se establezca la comunicación, tal como se indica en la Figura 40.

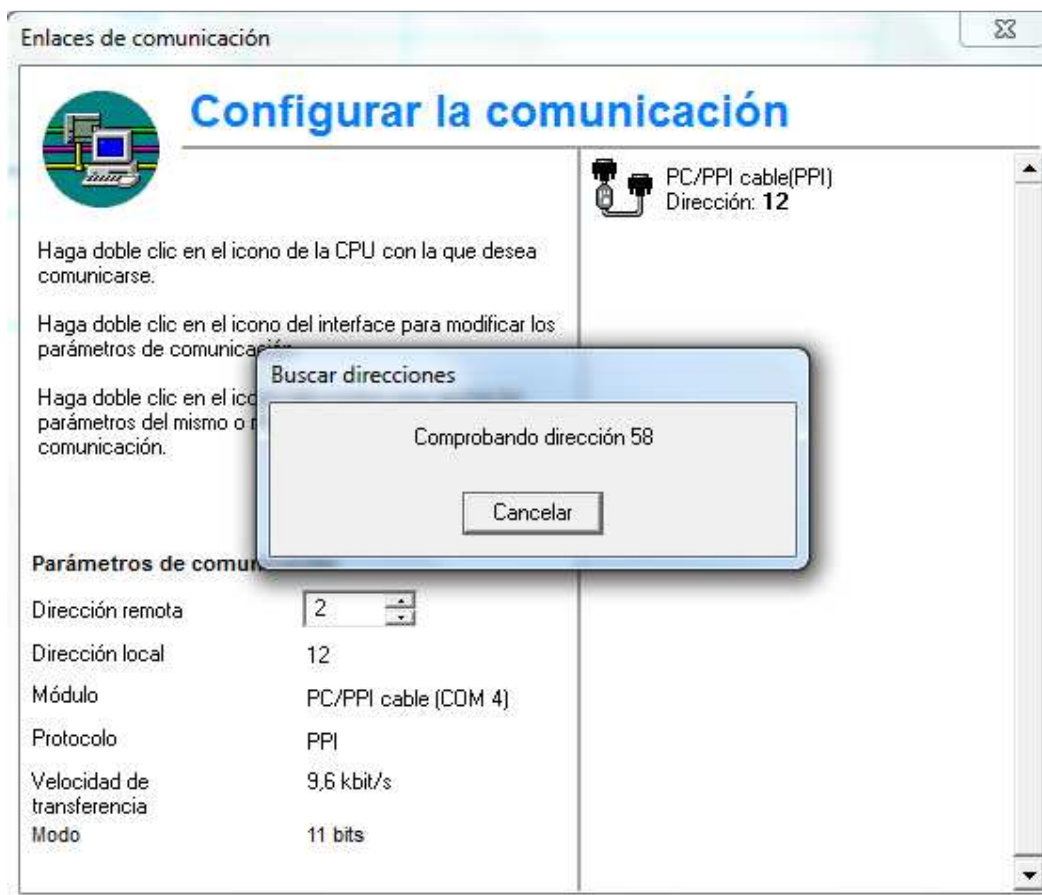


Figura 40. Establecimiento de la comunicación, STEP 7-Micro/WIN V3.1.

Programación del PLC S7-200 en STEP 7 MicroWIN.

En este proyecto, se trabaja con variables análogas y digitales. Las variables análogas son accesibles desde el software HMI InduSoft Web Studio, de manera indirecta, razón por la cual se creó un programa en STEP 7 MicroWIN, de tal forma que dichas variables son almacenadas en la memoria volátil del PLC, y puedan ser leídas a través de InduSoft desde dicha memoria.

Como se explicó anteriormente, el módulo análogo EM 235 convierte los valores que recibe de los sensores a un formato de palabra o "word", razón por la cual se utilizó la nomenclatura

“AIW” para leer dichos valores y de igual manera se usaron bloques de transferencia de palabras (MOV_W) para moverlos y guardarlos en un espacio de memoria con formato de palabra (VW). Con respecto al número que se le asigna a AIW (Analog Input Word) y a VW (Volatile Word), éste corresponde al byte inicial que se le está asignando para almacenar el valor de la entrada análoga. Las variables en formato de palabra tienen un tamaño de 2 bytes, por eso, como se puede observar en el programa, a la primera entrada análoga se le asigna un espacio desde el byte 0 y a la siguiente desde el byte 2 (AIW2) y lo mismo ocurre en el caso de VW.

El programa realizado se indica en la Figura 41.

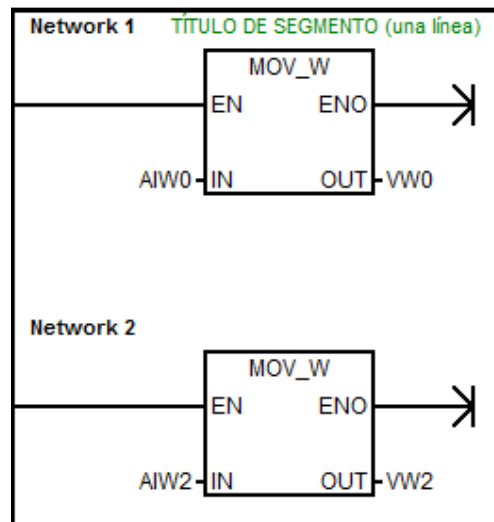


Figura 41. Programa en STEP 7 MicroWIN para acceso a variables del PLC S7-200 desde InduSoft Web Studio, STEP 7-Micro/WIN V3.1.

Una vez que la comunicación fue establecida, siguiendo los pasos descritos en la sección anterior, se procede con la descarga del programa al PLC, seleccionando en "Archivo" la opción "Cargar en CPU", tal como se muestra en la Figura 42.

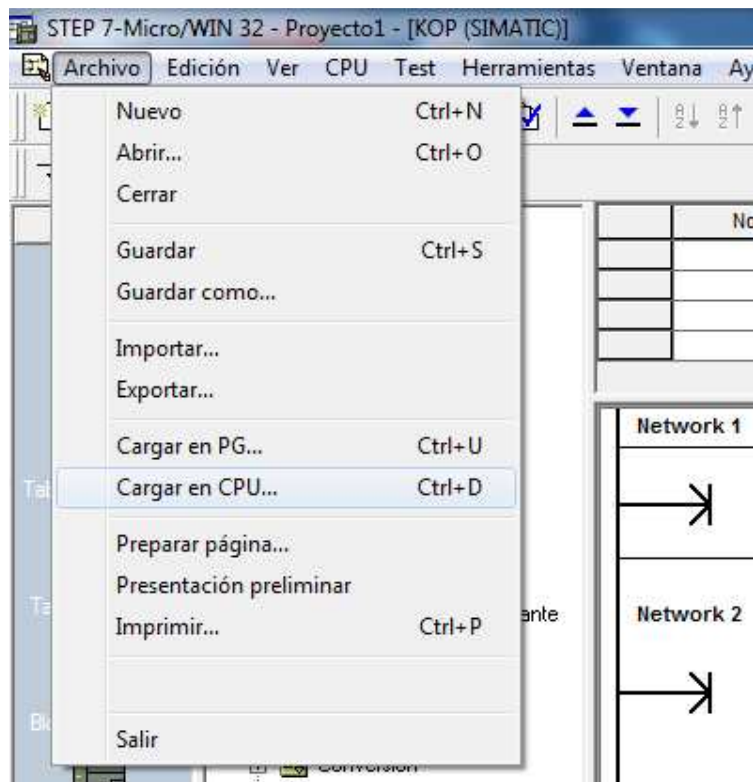


Figura 42. Carga de programa en CPU, STEP 7-Micro/WIN V3.1.

Configuración de Comunicaciones entre el EasyPort USB y la computadora.

La comunicación entre el Easyport y la computadora fue establecida mediante el EzOPC, el cual es un software propio de FESTO, el cual estaba incorporado con la compra del equipo, y para la manipulación de los sensores y actuadores se utilizó el InduSoft Web Studio v7.1.

Conexión Física.

Para la conexión física el principal equipo que se utilizará será el dispositivo Easyport de FESTO. Este trabaja a 24VDC. Posee 16 salidas digitales de 24VDC, 2 salidas análogas de 0-10VDC o de -10 a 10VDC con una resolución de 12bits. Tiene 16 entradas digitales de 24VDC, 4 entradas análogas de 0-10 VDC o -10 a 10VDC con una resolución de 12bits. Tiene dos interfaces de comunicación por serial o por USB ambos aislados eléctricamente. Utiliza un protocolo ASCII, a 115.2 kBaud, 8 bits sin paridad con 1 bit de parada (8N1).

El EasyPort tiene dos interfaces SysLink para leer 8 entradas digitales y escribir 8 salidas digitales, una entrada Sub-D para leer 4 entradas análogas y escribir 2 salidas análogas.

Las interfaces SysLink utilizan una conexión IEEE488 de 24 pines en los que se encuentran las salidas y entradas digitales y 0VDC y 24VDC. La entrada Sub-D para datos análogos tiene 15 pines en los que están las 4 entradas análogas y las 2 salidas análogas, 0VDC y +10VDC que es el voltaje de referencia. La conversión análogo-digital tiene una resolución de 12bits y la frecuencia de muestreo es 0.5kHz.

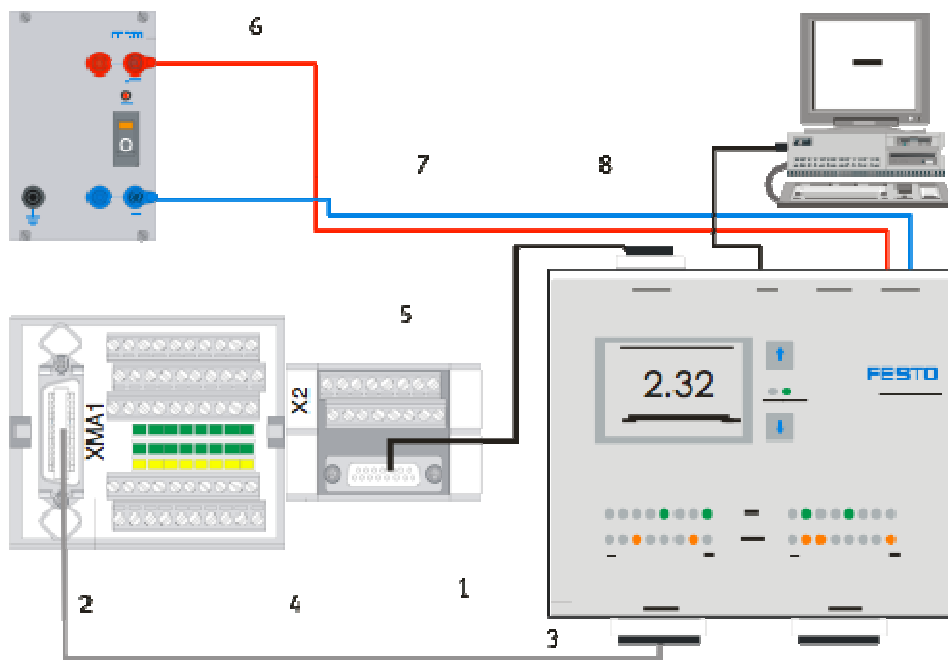


Figura 43. Conexiones entre el tablero de entradas y salidas con el EasyPort, Helmich, 2008.

A continuación se detalla la lista de entradas y salidas, su tag, la asignación de pines y su descripción:

Tabla 3. Tags y asignación de pines del EasyPort.

Tag	EasyPort	Descripción
Entradas Digitales		
B102	I0	Sensor de caudal (Frecuencia de 0 a 1000Hz)
S111	I1	Interruptor flotador, overflow del tanque 101
S112	I2	Interruptor flotador, nivel bajo tanque 102
B113	I3	Sensor capacitivo nivel mínimo tanque 101
B114	I4	Sensor capacitivo nivel máximo tanque 101
S115	I5	Interruptor de la válvula de bola V102 cerrado
S116	I6	Interruptor de la válvula de bola V102 abierto
No Usado	I7	
Salidas Digitales		
M102	Q0	Abrir válvula de bola ON/OFF V102

Tag	EasyPort	Descripción
E104	Q1	Prender calentador del tanque 101
K1	Q2	Relé selecciona si la bomba es binaria o análoga
M1	Q3	Prender bomba P101 binario
M106	Q4	Prender válvula proporcional
No Usado	Q5	
No Usado	Q6	
No Usado	Q7	
Entradas Análogas		
B101	AI 0	PV variable del proceso, nivel en tanque 102
B102	AI 1	PV variable del proceso, caudal en el sistema
B103	AI 2	PV variable del proceso, presión en el sistema
B104	AI 3	PV variable del proceso, temperatura en el sistema
Salidas Análogas		
P101	AQ 0	Manipular actuador, bomba P101
V106	AQ 1	Manipular actuador, válvula proporcional V106

Configuración de la Comunicación.

OPC (Object linking and embedding for process control) diseñado para crear un puente entre software basado en Windows y hardware de control de procesos. Un servidor OPC para un dispositivo provee con los mismos métodos al cliente OPC para que pueda acceder a sus datos. Por lo que cada fabricante desarrolla su propio servidor OPC para que el nuevo dispositivo pueda ser utilizado por cualquier software, para que posteriormente los desarrolladores del software creen un cliente OPC para tener acceso al software. La tecnología Microsoft OLE es utilizada por los servidores OPC para comunicarse con los clientes, esta tecnología permite intercambio en tiempo real de datos entre software y hardware.

Los datos son transmitidos desde y hacia el EasyPort por medio de comandos de escritura y lectura, con dirección e individuales. En el EasyPort existen distintos niveles de acceso para ejecutar estos comandos, el nivel más bajo es el interpretador de comandos y puede ser utilizado directamente desde un programa terminal (por ejemplo el HyperTerminal). Luego está el control ActiveX que define las funciones de acceso y es la interfaz de programación del Easyport. Y en el nivel superior está el EzOPC el cual es un servidor OPC que establece comunicaciones con las aplicaciones. Como se dijo antes el interpretador de comandos conoce todos los comandos que puede ser ejecutado por el Easyport, se puede conectar directamente por un programa terminal. El control ActiveX es una interfaz para programar el Easyport y puede ser incorporado a proyectos como un objeto COM (component object model), el único pre-requisito es que el software pueda utilizar puertos COM, incluyendo lenguajes de programación como Visual Basic y C++. Y finalmente está el servidor EzOPC el que puede ser escrito y leído por cualquier cliente OPC compatible con la versión OPC 2.x. En la interfaz gráfica el controlador virtual es el corazón del EzOPC, representa la central que conecta ambos lados de la comunicación, en este caso el EasyPort con el software InduSoft.

Para entender un poco mejor el sistema se detalla un poco el modo de uso del interpretador de comandos, como fue mencionado con anterioridad el objeto COM utiliza el protocolo 8N1 a 115.2kbps. Los comandos consisten de una letra de comando (operador) y la dirección del recurso (operando). El comando de display (D) se utiliza para leer los recursos y el comando de modificación (M) para modificar recursos. Así mismo existen tipos de recursos, entrada (E), salidas (A), timer (T), contador (C). Y estos pueden ser de tipo bit, Byte (B), Word (W). Por ejemplo si se utiliza el comando M, recurso A, modulo 1, palabra 0 y el bit 4, lo ponemos en 1. MA1.0.4 = 1. En este se puede ver que la palabra 0 es la que define las salidas digitales y contiene 16 bits. Para el caso de comandos para señales análogas, se pueden utilizar dos

rangos de -10 a 10VDC y de 0 a 10VDC, el rango de los datos en decimal será de 0 a 32767. Por ejemplo DEW1.2 la respuesta es EW1.2 = xxxx (0000-7FF8), este lee la entrada en el canal 0 en un módulo EasyPort con dirección 1.

El servidor EzOPC facilita el manejo de comandos, ya que se evita trabajar directamente con los mismos.

Integración con HMI y programación del proceso de mezcla.

Finalmente, una vez que tanto la comunicación del PLC S7-200 como la del EasyPort USB con el ordenador fueron establecidas independientemente, se procedió a integrar ambas mediante InduSoft Web Studio v7.1, incorporando la visualización de todos los valores de los sensores en un HMI y posteriormente se programó un proceso de mezcla.

Configuración del SIEMENS Port to Port Interface (SIPPI) en InduSoft Web Studio.

Para la visualización de las variables del PLC S7-200 en un HMI se utiliza el software InduSoft Web Studio, el cual permite crear una interfaz gráfica entre el controlador y el usuario. Para la lectura de valores de los sensores implementados, es necesaria la instalación de un driver de comunicación, el cual se encuentra disponible en InduSoft. A continuación se describe el procedimiento a seguir:

- En el programa InduSoft Web Studio, se crea un nuevo proyecto y en el asistente se escoge la plantilla y la resolución, como se indica en la Figura 44.

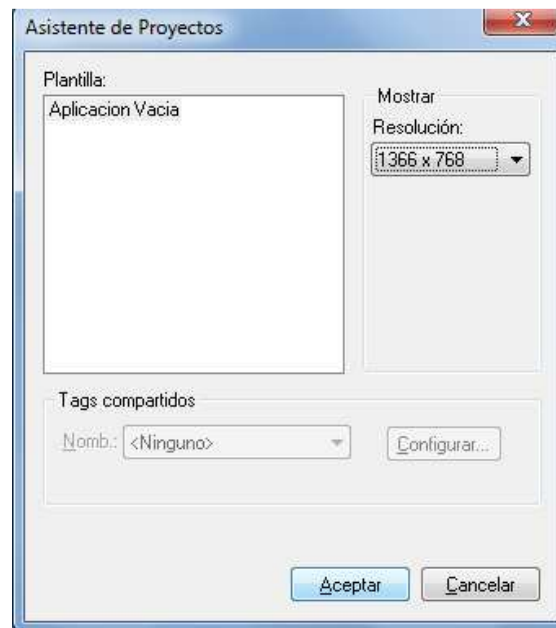


Figura 44. Creación de nuevo proyecto en InduSoft Web Studio, InduSoft Web Studio v7.1.

- A continuación, en explorador de proyectos se selecciona la pestaña inferior de comunicaciones y con un click derecho sobre la carpeta "Drivers" se escoge "Agregar/Eliminar drivers". Ver Figura 45.

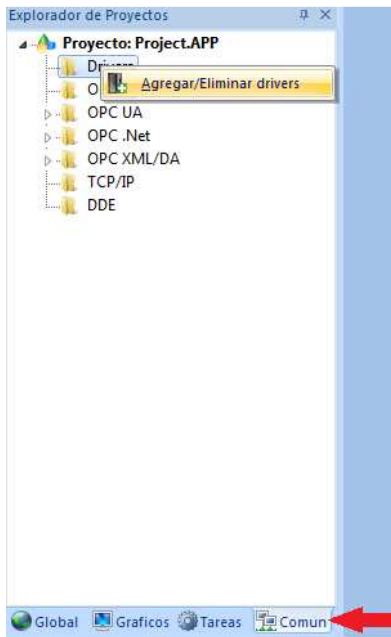


Figura 45. Acceso a drivers de comunicación, InduSoft Web Studio v7.1.

- Debido a que en este caso se está trabajando con un PLC S7-200, en el cuadro de diálogo de drivers de comunicación se escoge el SIPPI.

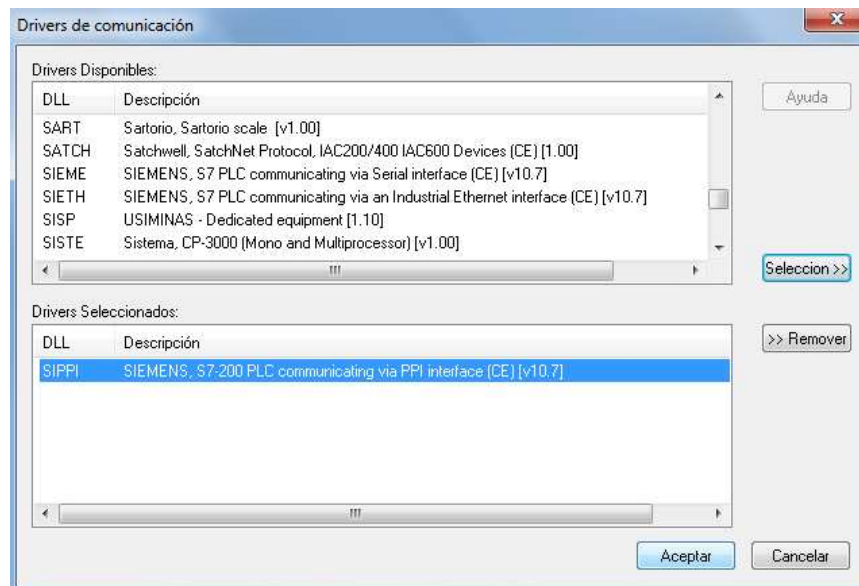


Figura 46. Selección de driver de comunicación, InduSoft Web Studio v7.1.

- Después de seleccionar el driver apropiado, con un click derecho sobre la carpeta "SIPPI" se escoge la opción "Configuración", como se muestra en la Figura 47.

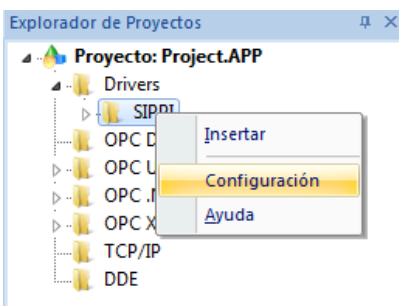


Figura 47. Driver SIPPI, InduSoft Web Studio v7.1.

- En "Puerto Serie" se selecciona COM1 y en el resto de campos se mantienen los parámetros predeterminados, verificando que la velocidad de transmisión sea 9600 bps. Ver Figura 48.

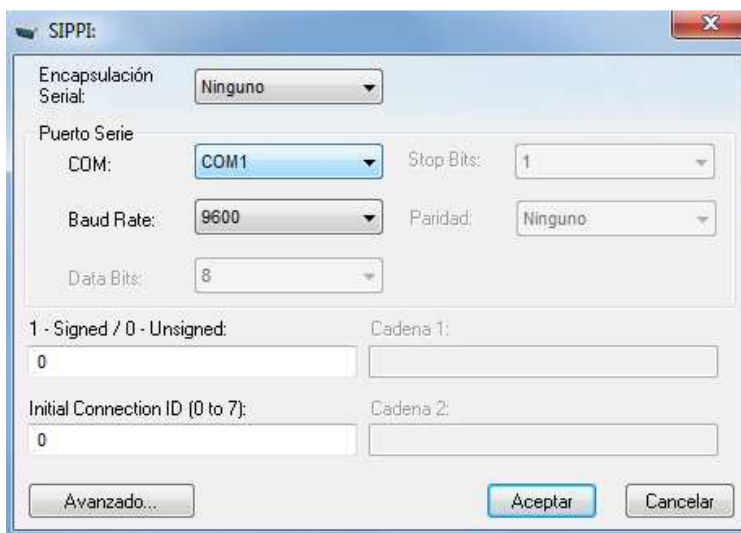


Figura 48. Configuración del driver SIPPI, InduSoft Web Studio v7.1.

- Después se accede a la hoja del driver abriendo la carpeta "SIPPI". Los campos de la parte superior se llenaron tal como se ilustra en la Figura 49. Es muy importante tener en cuenta que aquí es donde se configuran los parámetros para la lectura de las variables del PLC S7-

200. Se tienen 5 parámetros: Nombre de Tag, Estación, Direcciones I/O, Acción y Escanear. En el primer recuadro se asigna el nombre de un tag a una entrada o salida del PLC. El nombre del tag se escoge de una lista preexistente o se puede crear uno con el nombre que se desee. Al crear el tag se debe considerar si la entrada/salida es digital o análoga, ya que de acuerdo a esto se escoge su tipo. En el caso de las entradas/salidas digitales se escoge booleano mientras que para las entradas análogas se selecciona el tipo entero (ya que éstas van a tomar únicamente valores entre 0 y 32768 sin decimales). En "Estación" se escribe la dirección remota del PLC que en este caso es 2, tal como se puede apreciar en los parámetros de comunicación en STEP 7 MicroWIN configurados anteriormente. En el tercer recuadro se asigna una dirección a cada tag, la cual debe concordar con las conexiones físicas realizadas. Para las entradas digitales se utiliza la letra "I" seguida del número de byte y especificando a continuación el bit (precedido por un punto), como por ejemplo I0.0, la cual hace referencia a una entrada digital ubicada en el byte 0 bit 0. Para las salidas digitales la nomenclatura utilizada es la misma descrita para las entradas pero en lugar de "I" se utilizó "Q". En el caso de las entradas análogas, se utiliza la dirección de memoria volátil (VW) que le fue asignada previamente en el programa realizado en STEP 7 MicroWIN. Los dos últimos parámetros se mantienen como los predeterminados. La tabla queda tal como se indica en la Figura 49.

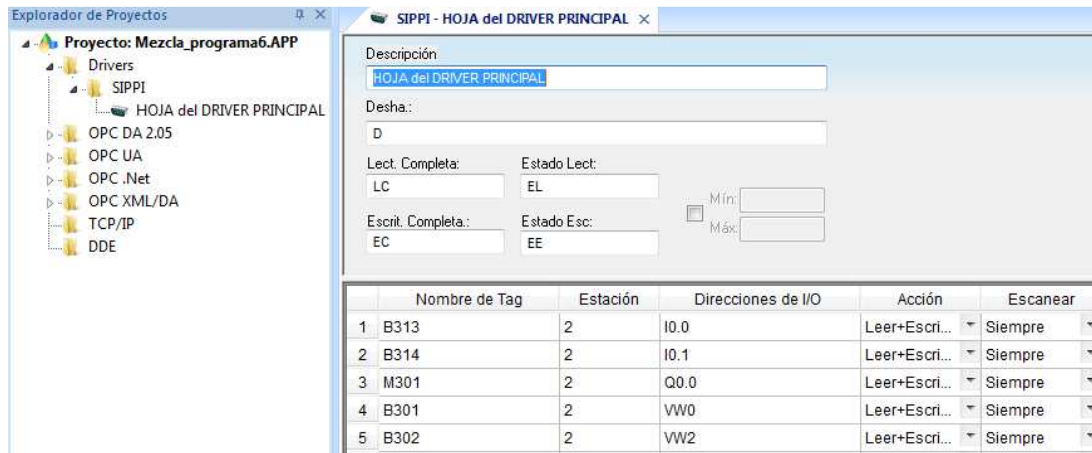


Figura 49. Configuración de parámetros en la hoja del driver SIPPI, InduSoft Web Studio v7.1.

En la Tabla 4 se presentan los sensores y el respectivo tag asignado a cada uno de ellos.

Tabla 4. Asignación de tags a los sensores implementados.

SENSORES PLC S7-200			
Digitales		Análogos	
Sensor	Tag	Sensor	Tag
Interruptor nivel alto tanque 2	B314	Sensor ultrasónico tanque 2	B301
Interruptor nivel bajo tanque 2	B313	Sensor de caudal 2	B302

-En la parte implementada se tiene únicamente un actuador que es una salida digital y ésta va conectada al PLC S7-200:

Tabla 5. Asignación de tags a los actuadores implementados.

ACTUADORES PLC S7-200	
Digital	
Actuador	Tag
Bomba 2	M301

Configuración del OPC DA 2.05 en InduSoft Web Studio.

El software Indusoft posee la opción de utilizar el cliente OPC DA 2.05. El OPC DA se utiliza para datos en tiempo real y no datos históricos. Hay tres atributos asociados con este tipo de cliente, el valor, la calidad del valor y un tiempo de control. Estos atributos deben ser respondidos por el servidor OPC al haber una petición, sino tiene un tiempo de control el servidor lo crea.

Para agregar el cliente OPC, en la pestaña de comunicación de InduSoft se seleccionó la carpeta "OPC DA 2.05" y se pudo identificar inmediatamente el servidor y revisar sus tags por defecto.

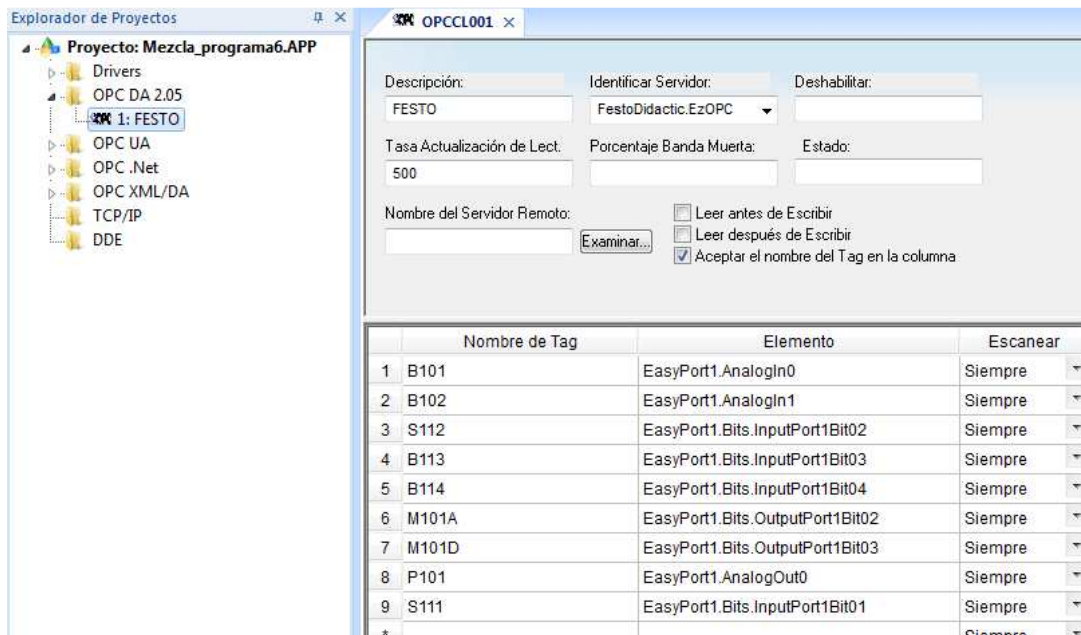


Figura 50. OPC DA 2.05, InduSoft Web Studio v7.1.

Se mantuvieron únicamente los tags que iban a ser utilizados en la programación de la interfaz gráfica.

Creación de la interfaz gráfica (HMI).

Para realizar la interfaz gráfica se utilizó el software InduSoft Web Studio v7.1, el cual es un programa que contiene una colección de herramientas de automatización y que mediante la utilización de bloques permite desarrollar HMIs, sistemas SCADA, entre otras aplicaciones. A través de este software, se desarrolló un sistema SCADA para el monitoreo y control eficiente del módulo.

InduSoft permite insertar gráficos de todo tipo, puesto que posee una librería de imágenes prediseñadas tales como (bombas, válvulas, interruptores, etc), así como permite ingresar botones, texto, etc.

A todos los elementos visuales ingresados se les puede relacionar con cada sensor del sistema, los cuales son leídos como tags en InduSoft.

En la interfaz gráfica se pueden visualizar todos los datos que se consideren importantes/críticos para nuestro sistema. Además de los valores provenientes de los sensores, se pueden visualizar otras variables "internas" que se definen en el código de programación. A continuación se detallan los pasos seguidos para la creación del HMI del sistema completo:

- En primer lugar, es necesaria la creación de tags que no están direccionados a una entrada/salida física del PLC. Para esto se selecciona la pestaña "Global" y con un click derecho sobre la carpeta "Etiquetas del Proyecto" se escoge la opción "Insertar Tag", como se indica en la Figura 51.

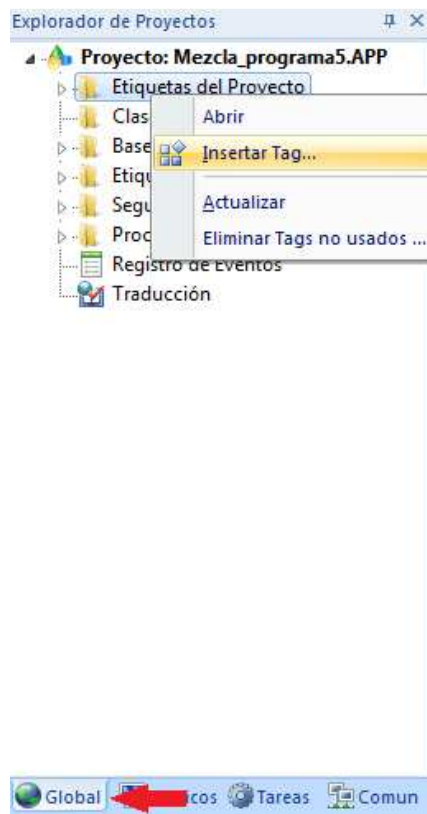


Figura 51. Creación de tags, InduSoft Web Studio v7.1.

- En la ventana emergente, se le asigna un nombre al nuevo tag y se escoge el tipo. En el caso de las variables que pueden tomar dos valores (on/off), se selecciona la opción booleano, mientras que para las variables con punto flotante se escoge el tipo real. Ver Figura 52.



Figura 52. Asignación de nombre y tipo de tag, InduSoft Web Studio v7.1.

En la Tabla 6 se enumeran todos los tags creados que no poseen una dirección y se especifica su tipo.

Tabla 6. Lista de tags sin asignación de dirección.

TAGS CREADOS (SIN DIRECCIONAMIENTO)	
Nombre del Tag	Tipo
contador	Entero
Stop	Booleano
ModoAutomatico	Booleano
LitrosMedidosTanque2	Real
LitrosMedidosTanque3	Real
LitrosInicioTanque2	Real
LitrosInicioTanque3	Real
LitrosTransferidosTanque2	Real
LitrosTransferidosTanque1	Real
LitrosDeseadosTanque2	Real
LitrosDeseadosTanque1	Real
Boton	Booleano
LitrosTanque1	Real
Caudal1	Real
Caudal2	Real
AB314	Booleano
AS111	Booleano

- A continuación, en la pestaña "Gráficos", se crea una nueva pantalla escogiendo "Insertar" al dar click derecho sobre la carpeta "Pantallas". Una vez creada la pantalla en blanco, se escoge la opción "Símbolos" para la inserción de objetos, como se aprecia en la Figura 53. Se abre la carpeta "Símbolos de Sistema" y se agregan objetos para visualizar una representación gráfica del sistema con los valores considerados pertinentes.

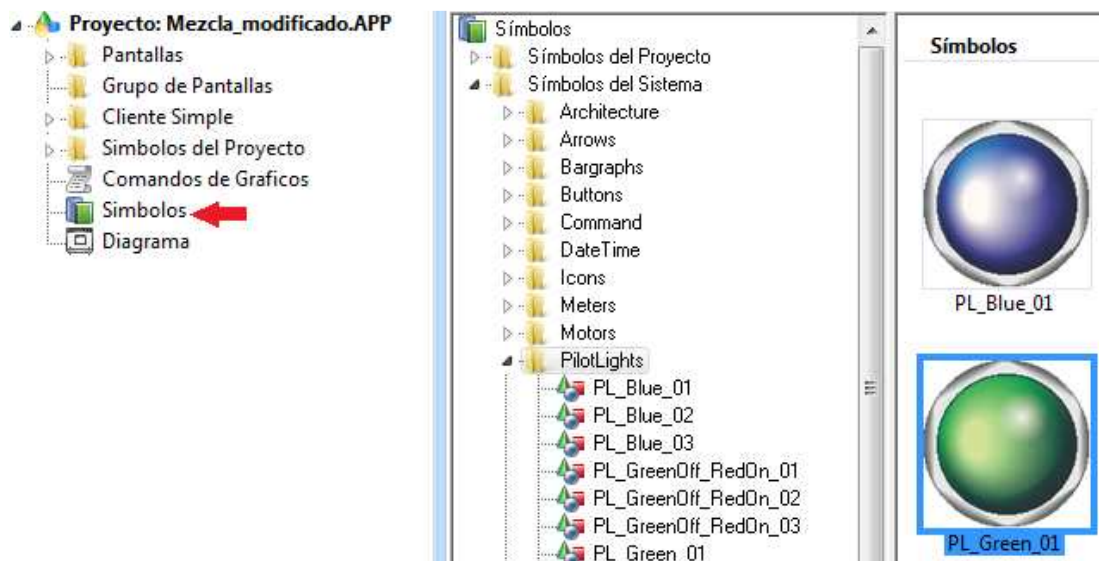


Figura 53. Inserción de símbolos en la pantalla, InduSoft Web Studio v7.1.

- A los símbolos introducidos en la pantalla, se les asigna un tag, de acuerdo al sensor o al valor de alguno de los tags sin dirección al cual se requiere asociarlo. En la Figura 54 se presenta a continuación, se le asigna el tag "B314" a una luz verde, que se encenderá cuando el sensor esté en "on". Este tag corresponde al interruptor de nivel alto del tanque 2, tal como se indica en la tabla 4.

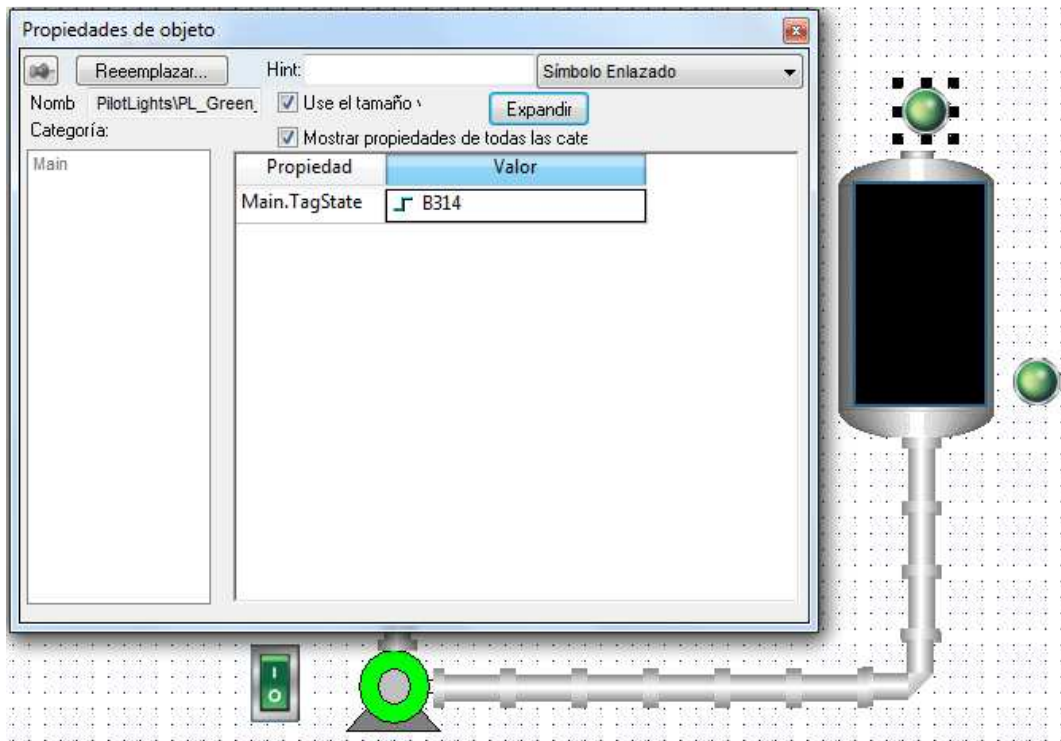


Figura 54. Vinculación de tags a símbolos, InduSoft Web Studio v7.1.

- Por último, después de insertar los objetos en la pantalla, se procede a ejecutar la aplicación, en la cual se puede observar la medición de los distintos sensores y accionar los actuadores desde los botones incorporados en la pantalla. Ver Figura 55.

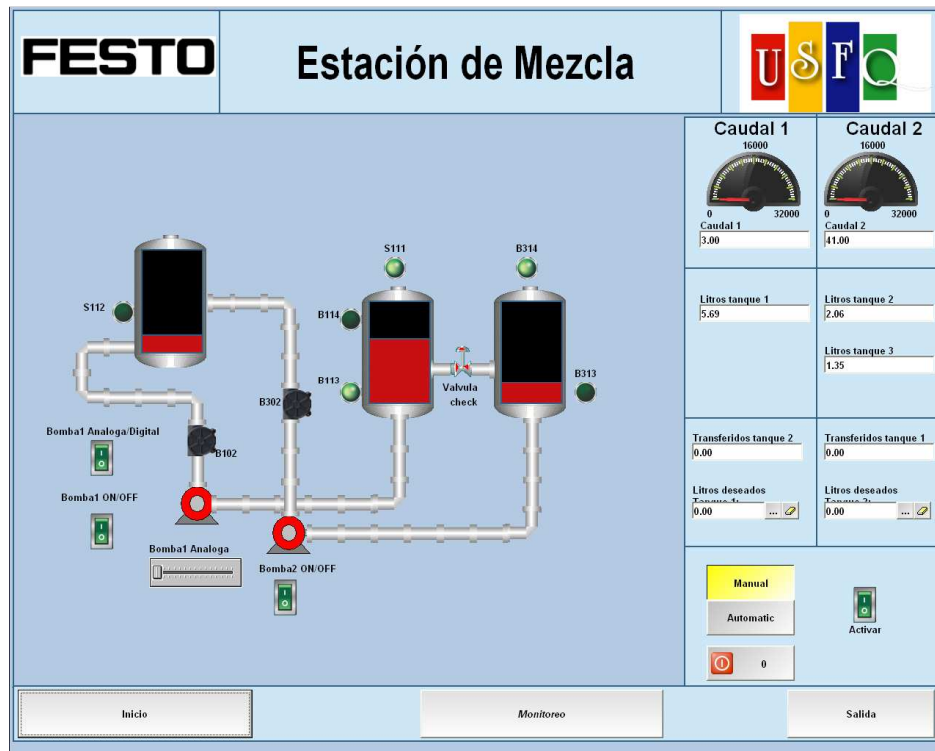


Figura 55. Pantalla de la aplicación, InduSoft Web Studio v7.1.

Para el presente proyecto, además de la pantalla principal, se desarrolló una pantalla de introducción y se muestra en la Figura 56.



Figura 56. Pantalla inicial de la interfaz gráfica, InduSoft Web Studio v7.1.

Programación del proceso de mezcla.

Una vez concluida la elaboración de la pantalla, se procedió con la escritura del código para conseguir un proceso de mezcla. El programa consiste en obtener una mezcla en el tanque 3, la cual debe estar compuesta por una cantidad de líquido proveniente del tanque 1 y otra proveniente del tanque 2. En teoría, el líquido contenido en cada uno de los tanques de suministro, debería ser distinto, pero en este caso debido a que este no es un proceso industrial real, se utiliza un solo líquido, de manera que éste pueda recircular. Pero el proceso de mezcla en sí, es recreado satisfactoriamente. La lógica de funcionamiento del programa se basa en las medidas de los dos sensores ultrasónicos (B101 y B301), ya que esto es lo más recomendable cuando se trabaja con mediciones de líquidos. En la pantalla elaborada se incluyeron dos recuadros de introducción de valores numéricos, los cuales permiten

especificar en litros, la cantidad que se desea transferir desde cada uno de los tanques de distribución. Considerando que se tiene un sensor ultrasónico únicamente en uno de los tanques de suministro (tanque 2) y otro en el tanque de mezcla (tanque 3), se elaboró un programa secuencial, en el cual primero se bombea el líquido del tanque 2 (se activa la bomba 2) y una vez que se ha transferido el volumen deseado, se procede con la activación de la bomba 1, para entregar la cantidad especificada desde el tanque 1. En el programa también se incluyó un botón para escoger el modo de operación pudiendo ser éste manual o automático. En ambos modos se debe introducir la cantidad a ser transferida desde cada tanque de suministro, y acto siguiente se debe activar un botón para dar inicio a la transferencia de líquido. En el caso del modo manual, la mezcla se realiza una sola vez, ya que cuando se ha cumplido con las transferencias deseadas de cada líquido, el botón de inicio se desactiva. Si se desea repetir el proceso de mezcla, ya sea con nuevas cantidades o con las ya introducidas, se debe activar el botón de inicio nuevamente. En el modo automático, el proceso es cíclico, es decir se repite cada cierto periodo. Al igual que en el modo manual, se requiere activar el botón de inicio para dar paso a la ejecución del proceso. La fase de mezcla se repite hasta que el botón de inicio sea desactivado por el usuario o hasta que uno de los tanques de suministro se quede sin líquido. Tanto para el modo manual como para el automático, se utilizó un contador que empieza a funcionar desde el momento en que se activa el botón de inicio. Este contador permite registrar el nivel inicial del tanque 2, antes de que el líquido sea bombeado hacia el tanque 3. En ambos modos, el contador se inicializa en 0 al final del proceso de mezcla, pero en el caso del modo manual, también se manda a apagar el botón de inicio para salir del lazo y detener su ejecución, mientras que en el modo automático, el botón de inicio permanece encendido todo el tiempo, de tal forma que al reiniciar el contador, se ejecuta nuevamente el lazo. Por otra parte, también se consideraron

acciones de seguridad en las líneas de programación, como por ejemplo el apagado de la respectiva bomba si es que alguno de los tanques de suministro está vacío. Asimismo se incluyó un botón de "shutdown" de la planta, a través del cual se apagan las bombas y se resetean los valores medidos por los sensores.

La unidad de volumen del programa es litros, por lo cual se tuvo que determinar una ecuación lineal de conversión, para tener la equivalencia en litros de los valores leídos por cada sensor ultrasónico. En el caso del sensor ultrasónico PING (ubicado en el tanque 2), se leen valores entre 0 y 32000, siendo 0 el valor cuando el tanque está vacío y 32000 cuando se alcanza el nivel máximo (32 cm). Por su parte, el sensor ultrasónico de Festo, proporciona valores entre 0 y 32768. Para obtener la ecuación de conversión, se tomaron datos del valor medido por el sensor cada 0.5 litros y se elaboró una tabla. Con estos datos se realizó un gráfico en Excel y se hizo una regresión lineal para obtener la ecuación. De esta manera, se obtuvo una ecuación para cada sensor ultrasónico y se las incluyó en el código de programación.

En InduSoft Web Studio, en la pestaña "Gráficos", se tiene una sección en la que se puede programar utilizando los tags creados. El código fue escrito en lenguaje Visual Basic, en la ventana de "Comandos de Gráficos", como se aprecia en la Figura 57.

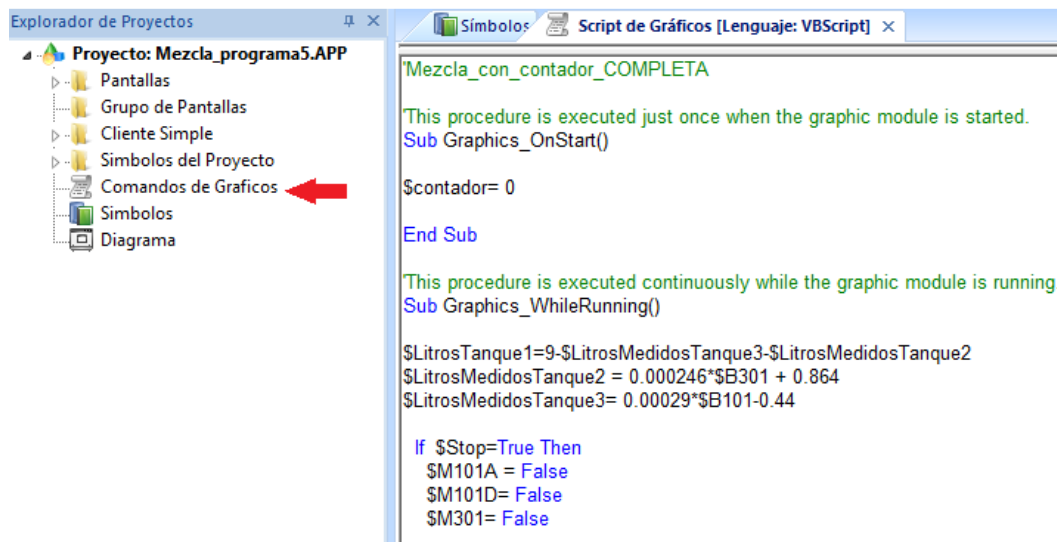


Figura 57. Escritura de código en InduSoft, InduSoft Web Studio v7.1.

El correspondiente código de programación se presenta en el ANEXO A.

RECOMENDACIONES Y CONCLUSIONES

Recomendaciones

- Debido a los altos costos de los sensores industriales, y al limitado presupuesto asignado para la elaboración de la tesis, se utilizaron los sensores de nivel y caudal que se detallaron en el cuerpo de la tesis, que a pesar de su buen funcionamiento, no son industriales y podrían presentar problemas de acuerdo a las condiciones que se presenten, es por esto recomendable que en un futuro, se reemplacen estos sensores por los mismos o similares a los que incorpora la planta FESTO.
- En el proceso de mezcla, para evitar que el líquido fluya de regreso a los tanques una vez que ha sido bombeado al tanque resultante, se utilizan válvulas manuales, sería recomendable que se reemplacen estas válvulas por válvulas automáticas, de manera que el proceso se automatice totalmente, esto no se lo puedo realizar por el alto costo que tienen este tipo de válvulas.
- Hubiese sido interesante automatizar el proceso de mezcla utilizando el PLC S7 300 313C (incorporado en el módulo), a pesar de que es un buen PLC en cuanto a Robustez, sus características básicas en cuanto a comunicación es un gran limitante, así que sería recomendable, reemplazar el PLC por uno de mejores características como por ejemplo al PLC S7 300 313C 2DP, el cual permito realizar la comunicación por medio de PROFIBUS, o adquirir un módulo Ethernet adicional, de manera que se pueda comunicar con otros PLC.
- Para programar el PLC S7 300 313C (incorporado en el módulo), es necesario utilizar un cable MPI, exclusivo de SIEMENS, para programar se utilizó un cable genérico, pero no en

recomendable ya que no es robusto y podría fallar dependiendo de varios factores, por lo que sería recomendable que se adquiriera inmediatamente el cable original de SIEMENS.

- Para poder realizar un HMI con el PLC S7 300 313C, se debe adquirir el software propio de SIEMENS, el cual es WinCC, y para poder utilizar Indusoft se requiere la librería PRODAVE, ambos software excedían el presupuesto, es por esto que no se utilizó el PLC incorporado en el módulo, es recomendable que se adquiriera el software propio de SIEMENS WinCC o la librería PRODAVE.
- El PLC S7 300 313C, permite solamente comunicarse solamente mediante el protocolo MPI, sería recomendable que se realice la misma estación de mezcla pero esta vez con una diferente arquitectura, comunicando directamente 2 PLC's pero esta vez utilizando el protocolo MPI.

Conclusiones

- Se logró la etapa de mezcla en el módulo FESTO, de tal manera que se pueda ingresar una receta y se obtenga la cantidad del líquido resultante en el tanque 3, esta etapa se la realizó con un presupuesto relativamente bajo, ya que si se hubiese comprado ya armado, el precio hubiese ascendido considerablemente.
- A pesar de no utilizar sensores industriales para medir el nivel y el caudal en la parte implementada, se obtuvieron resultados muy precisos al momento de medir y manipular el nivel y el caudal.

- Al momento de realizar el proceso de mezcla, una vez que está el líquido en el tanque resultante, el líquido regresa por las tuberías donde llegaron al tanque resultante.

- El PLC S7 300 313C que está incorporado en el módulo FESTO, fue realmente un inconveniente al momento de desarrollar el proyecto, ya que por sus características limitadas no se lo pudo manipular de la mejor manera, en tres etapas, primeramente en la programación, ya que necesita el cable MPI, lo cual fue solucionado, luego al momento de comunicar solo se lo puede realizar mediante el uso de un protocolo exclusivo de SIEMENS, el cual se denominada MPI, luego al momento de realizar el HMI solamente se lo puede realizar si se posee software exclusivo de SIEMENS.

- La salida que se encontró para desarrollar la etapa de mezcla, fue utilizando un servidor OPC, usando por una parte el PLC S7 200, y por otra el EasyPort propio de FESTO, a pesar de que las acciones y la visualización en el HMI, se retrasaban, el proceso no se veía afectado, ya que no se debían procesar demasiadas variables.

- El EasyPort, fue una gran ayuda ya que fue mediante este sistema por el cual se pudo concluir con el objetivo de la tesis, que fue crear una etapa de mezcla dentro del módulo FESTO.

- La plataforma Arduino fue de gran ayuda al momento de procesar las señales, ya que esta fue la mejor manera que se encontró para que la señales puedan ser leídas por el PLC.

REFERENCIAS

- Aliexpress. (2013). *Bomba de agua imagen*. Obtenido el 1 de Mayo 2013 de <http://www.aliexpress.com/item/DC40-2470-DC-24V-Brushless-Magnetic-Drive-Centrifugal-Water-Pump-Submersible-CPU-Cooling-Amphibious-650L-H/580302909.html>
- ARDUINO. (2013). *Diagrama de conexión del sensor ultrasónico imagen*. Obtenido el 10 de Abril 2013 de <http://arduino.cc/en/Tutorial/Ping?from=Tutorial.UltrasoundSensor>
- ARDUINO. (2013). *ARDUINO UNO imagen*. Obtenido el 10 de Abril 2013 de <http://arduino.cc/en/Main/ArduinoBoardUno>
- Festo. (2013). *Ultrasound sensor image*. Obtenido el 02 de octubre 2012 de <http://www.festo-didactic.com/int-en/learning-systems/process-automation/edukit-pa/sensor,ultrasound.htm?fbid=aW50LmVuLjU1Ny4xNy4xOC4xMTE4LjY1ODg>
- Festo Didactic GmbH & Co. KG. (2001, 2004, 2006 & 2009). *Collection of Datasheets*.
- Gantt, C. (2010). *Tutorial: Reading Water Flow rate with Water Flow Sensor*. Seedstudio. Obtenido el 20 de enero 2013 de <http://www.seedstudio.com/forum/viewtopic.php?f=4&t=989&p=3632#p3632>
- Helmich, J. & ADIRO. (2008). *MPS@PA Compact Workstation Manual*. Adiro Automatisierungstechnik GmbH, 73734 Esslingen, Germany.
- Helmich, J. (2008). *Module 2: Commissioning a basic closed-loop control system MPS@PA Compact Workstation*.
- InduSoft, Ltd. (2007). *VBScript Reference Manual for InduSoft Web Studio*.
- Murillo, P. (2011). Tutorial Arduino # 0003 – Entrada Analógica y Salida PWM. *Arduiteka*. Obtenido el 12 de enero 2013 de <http://www.arduteka.com/2011/11/tutorial-arduino-0003-entrada-analogica-y-salida-pwm/>
- Pomares, J. (2009). *Manual de programación de Arduino*. Universitat d'Alacant.
- ROBODACTA. (2011-2013). *PING sensor ultrasónico imagen*. Obtenido el 15 de noviembre 2012 de http://www.robodacta.mx/index.php?dispatch=products.view&product_id=30711

SIEMENS AG. (2008). *Manual del sistema de automatización S7-200*. Bereich Automation and Drives, Postfach 4848, D 90327.

Siemens Energy & Automation, Inc. (1999). *SIMATIC S7-200 Data Sheet for EM231, EM232, and EM23*. Obtenido el 24 de febrero 2013 de <http://fa.jonweb.net/Siemens/images/Article/Technical/s7200analog.pdf>

The Makers Workbench. (2013). *Diagrama de conexión del sensor de caudal imagen*. Obtenido el 10 de Abril 2013 de <http://themarkersworkbench.com/content/tutorial/reading-liquid-flow-rate-arduino>

ANEXO A: CÓDIGOS DE PROGRAMACIÓN

**CÓDIGO DEL PROCESAMIENTO DE SEÑALES DEL SENSOR ULTRASÓNICO PING Y DEL
SENSOR DE CAUDAL REALIZADO EN ARDUINO 1.0.2**

```

const int pingPin = 7;          // Pin asignado a la señal del sensor ultrasónico.
const int analogOutPin = 9;    // Pin asignado a la salida analógica (voltaje) que se obtiene como
                                // resultado de nuestro programa.

int outputValue = 0;          // Declaración e inicialización de la variable de salida outputValue.

int sensorflujo = 2;          // Pin asignado a la señal del sensor de flujo. Cabe mencionar que este pin
                                // corresponde a la interrupción externa #0.
const int sensorflujoOutPin = 10; // Pin de salida (analógica) de voltaje (0-5V).

volatile int NumPulsos;       // Declaración de variable que se utilizará como contador de flancos
                                // ascendentes. Se utiliza volatile debido a que se tienen secciones
                                // de código asociadas a interrupciones (rutina de servicio de interrupción).

int outputValue1 = 0;         // Declaración e inicialización de la variable de salida outputValue1.
int Caudal=0;                 // Declaración e inicialización de la variable de salida Caudal.

long duration, cm;           // Se establecen variables para la duración del ping y para la distancia.

void rpm ()                   // Esta es la función a la que se invoca cuando la interrupción tiene lugar (rutina de
                                // interrupción de servicio).
{
  NumPulsos++; // Mide los flancos ascendentes y descendentes de la señal del sensor.
}

void setup()
{
  pinMode(sensorflujo, INPUT); // Se inicializa el pin digital 2 como una entrada.
  Serial.begin(9600);          // Inicialización de la comunicación serial.

  attachInterrupt(0, rpm, RISING); // Se enlaza la interrupción a la función rpm. El primer parámetro
                                    // indica el # de interrupción externa.
                                    // El segundo parámetro es el nombre de la función a la que se llama
                                    // cuando se efectúa la interrupción.
                                    // Con "RISING" se indica que la interrupción debe ser disparada cuando
                                    // el pin pase de valor alto (HIGH) a bajo (LOW).

```

```

}
void loop ()
{

    NumPulsos = 0;           //Inicialización de la variable NumPulsos.

    // NOTA: El pin "pinPing" (7) funciona como entrada o salida.

    // "pinPing" COMO SALIDA
    pinMode(pingPin, OUTPUT); // Indica que el "pinPing" se está utilizando como salida.
    digitalWrite(pingPin, LOW); // Se envía un pulso en bajo (LOW) de corta duración para asegurar un
                                // pulso en alto "limpio".

    delayMicroseconds(2);    // Duración del pulso en bajo de 2 microsegundos.

    digitalWrite(pingPin, HIGH); // Se envía un pulso en alto (HIGH) de 2 o más microsegundos.
    delayMicroseconds(5);     // Duración del pulso en alto de 5 microsegundos.
    digitalWrite(pingPin, LOW); // Se envía un pulso en bajo.

    // "pinPing" COMO ENTRADA
    pinMode(pingPin, INPUT); // Se utiliza el "pinPing" como una entrada, para leer la señal del PING.
    duration = pulseIn(pingPin, HIGH); // Se almacena en "duration" la duración de un pulso en alto. Su
                                        // duración es el tiempo (en microsegundos)
                                        // desde el envío de la señal del PING hasta la recepción de su eco
                                        // después de rebotar en un objeto.

    sei();                       // Habilita las interrupciones.
    delay (250);                 // Retardo de 250 milisegundos.
    cli();                       // Deshabilita las interrupciones.

    cm = microsecondsToCentimeters(duration); // Se convierte el tiempo a una distancia en cm.
    Caudal = (NumPulsos *100/ 7.5);           // Conversión de frecuencia a L/min (Pulse frequency x
                                              // 60) / 7.5Q = flow rate in L/min

    pinMode (analogOutPin, OUTPUT); // Indica que el "analogOutPin" es utilizado como salida.
    outputValue = map(cm, 4, 32, 255, 0); // Se toma el valor almacenado en la variable cm (0-32) y se lo
                                           // convierte a un valor entre 0 y 255.
    analogWrite(analogOutPin, outputValue); // El valor almacenado en la variable outputValue se envía al

```

```

// pin "analogOutPin" (9), obteniéndose a la salida un voltaje
// que puede tomar valores entre 0 y 5V.

pinMode (sensorflujoOutPin, OUTPUT); // Indica que el "sensorflujoOutPin" es utilizado como
// salida.
outputValue1 = map(Caudal, 0, 620, 0, 255); // Se toma el valor almacenado en la variable Caudal
// y se lo convierte a un valor entre 0 y 255.
analogWrite(sensorflujoOutPin, outputValue1); // El valor almacenado en la variable outputValue1 se
// envía al pin "sensorflujoOutPin" (10), obteniéndose
// a la salida un voltaje que puede tomar valores entre 0 y 5V.

Serial.println(cm); // Se imprime en el monitor serial, el valor almacenado en la variable cm.
Serial.print("cm"); // Se imprime "cm" en el monitor serial.

Serial.println (Caudal); //Imprime el caudal calculado en L/min.
Serial.print ("L/min"); //Imprime las unidades del caudal "L/min".
}

long microsecondsToCentimeters(long microseconds)
{
// La velocidad el sonido es 340 m/s o 29 centímetros por microsegundo.
// Tomando en cuenta que la señal viaja una distancia de ida y de vuelta, se debe dividir la distancia
// viajada total para 2 y así se obtiene la distancia a la que se encuentra el objeto.
return microseconds / 29 / 2; // Conversión de microsegundos a cm
// (microsegundos*cm/microsegundo*1/2).
}

```

CÓDIGO DEL PROCESO DE MEZCLA REALIZADO EN INDUSOFT WEB STUDIO

'Este procedimiento se ejecuta sólo una vez cuando el módulo gráfico se inicia

Sub Graphics_OnStart()

\$contador= 0 'Se inicializa la variable contador

End Sub

'Este procedimiento se ejecuta continuamente mientras el módulo gráfico está corriendo

Sub Graphics_WhileRunning()

'Cálculo de caudal

\$Caudal1 = \$B102/4000

\$Caudal2 = \$B302/1196

'Estado sensores de nivel alto

\$AB314 = Not \$B314

\$AS111 = Not \$S111

'Ecuaciones de conversión a litros

\$LitrosMedidosTanque2 = 0.00024748*\$B301 + 0.3438

\$LitrosMedidosTanque3= 0.0002929*\$B101-0.3299

'Ecuación para obtener los litros del tanque 1

\$LitrosTanque1=9.1-\$LitrosMedidosTanque3-\$LitrosMedidosTanque2

'Para no tener valores negativos en los litros del tanque 3

If \$LitrosMedidosTanque3<0 Then

\$LitrosMedidosTanque3= 0

End If

'Botón de "shutdown"

If \$Stop=True Then

\$M101A = False

\$M101D= False

\$M301= False

\$Boton= False

ElseIf \$Stop= False Then

'Si el tanque 2 tiene líquido todavía, se ejecuta el lazo ya sea en modo manual o automático

If \$B301>2000 Then

'Si el tanque 1 tiene líquido todavía, se ejecuta el lazo ya sea en modo manual o automático

If \$LitrosTanque1>0.5 Then

If \$ModoAutomatico= False Then 'Modo Manual

If \$Boton= True Then 'Activa el modo manual, dando inicio al contador

\$contador= \$contador+1 'El contador se incrementa

If \$contador=1 Then 'Cuando el contador es igual 1 se almacena el valor medido por cada uno
 \$LitrosInicioTanque2= \$LitrosMedidosTanque2 'de los sensores ultrasónicos en litros, es decir
 \$LitrosInicioTanque3= \$LitrosMedidosTanque3 'se registra el número de litros iniciales en los
 'tanques 2 y 3

End If

If \$contador>1 Then 'Cuando el contador es mayor 1 se inicia la transferencia de líquido

'Ecuaciones para calcular la cantidad de litros que se está transfiriendo desde cada tanque

\$LitrosTransferidosTanque2= \$LitrosInicioTanque2-\$LitrosMedidosTanque2

\$LitrosTransferidosTanque1= \$LitrosMedidosTanque3- \$LitrosInicioTanque3-

\$LitrosTransferidosTanque2

If \$LitrosTransferidosTanque2<\$LitrosDeseadosTanque2 Then 'La bomba 2 se activa
\$M301= True 'si la cantidad de litros transferidos es
'menor a la ingresada en la aplicación

ElseIf \$LitrosTransferidosTanque2>=\$LitrosDeseadosTanque2 Then 'Cuando la cantidad
\$M301= False 'de litros transferidos es igual o mayor a
'la deseada la bomba 2 se apaga

If \$LitrosTransferidosTanque1<\$LitrosDeseadosTanque1 Then 'Una vez que se ha
\$M101D= True 'transferido la cantidad de litros
'deseados desde el tanque 2, se procede
'con la transferencia desde el tanque 1,
'siguiendo la misma lógica del tanque 2

Elseif \$LitrosTransferidosTanque1>=\$LitrosDeseadosTanque1 Then 'Si se cumple con
\$M101D= False 'la cantidad de transferencia especificada
\$contador=0 'para el tanque 1, entonces la bomba 1
\$Boton= False 'se apaga, el contador se inicializa y
'para de contar

End If

End If

End If

End If

Elseif \$ModoAutomatico= True Then 'Modo Automático

If \$Boton= True Then 'Activa el modo automático, dando inicio al contador

\$contador= \$contador+1 'Empieza el conteo

If \$contador=300 Then 'Se toma la cantidad de litros de los tanques 2 y 3 en el instante
\$LitrosInicioTanque2= \$LitrosMedidosTanque2 'en el que el contador es igual a 300
\$LitrosInicioTanque3= \$LitrosMedidosTanque3

End If

If \$contador>300 Then 'Cuando el contador es mayor 300 se entra al lazo que permite la
'transferencia de líquido

'Ecuaciones para calcular la cantidad de litros que se está transfiriendo desde cada tanque

\$LitrosTransferidosTanque2= \$LitrosInicioTanque2-\$LitrosMedidosTanque2

\$LitrosTransferidosTanque1= \$LitrosMedidosTanque3- \$LitrosInicioTanque3-

\$LitrosTransferidosTanque2

If \$LitrosTransferidosTanque2<\$LitrosDeseadosTanque2 Then 'Si los litros transferidos
\$M301= True 'desde el tanque 2 son menores a los
'especificados, se bombea líquido

```

ElseIf $LitrosTransferidosTanque2>=$LitrosDeseadosTanque2 Then 'Si los litros
    $M301= False                                     'transferidos llegan al valor deseado
                                                    'entonces se desactiva la bomba 2
If $LitrosTransferidosTanque1<$LitrosDeseadosTanque1 Then 'Se empieza a bombear
    $M101D= True                                     'líquido desde el tanque 1, siempre y cuando se
                                                    'haya alcanzado la cantidad de transferencia
                                                    'deseada en el tanque 2. La bomba 1 se
                                                    'mantiene activa si los litros transferidos son
                                                    'menores a los especificados para el tanque 1
ElseIf $LitrosTransferidosTanque1>=$LitrosDeseadosTanque1 Then 'Al llegar a la
    $M101D= False                                     'cantidad de transferencia solicitada, la bomba
    $contador=0                                       '1 se detiene y se inicializa el contador, con lo
                                                    'cual se ejecuta nuevamente el lazo cuando el
                                                    'contador alcanza un valor mayor a 300
End If
End If
End If

ElseIf $Boton= False Then 'Si se apaga el botón que permite entrar en el lazo de modo automático
    $contador=0                                       'entonces el contador es inicializado y las bombas se desactivan
    $M301= False
    $M101D= False
End If
End If

'Si el tanque 1 está vacío, se desactivan las bombas
ElseIf $LitrosTanque1<=0.5 Then
    $M301= False
    $M101D= False
End If

'Si el tanque 2 está vacío, se desactivan las bombas
ElseIf $B301<=2000 Then
    $M301= False
    $M101D= False
End If
End If
End Sub

'Este procedimiento es ejecutado sólo una vez cuando el módulo gráfico se cierra
Sub Graphics_OnEnd()
End Sub

```

ANEXO B: DATASHEETS DE LOS ELEMENTOS UTILIZADOS

ANEXO C: HOJAS GUÍA DE ESTUDIANTE

PRÁCTICA 1: Configuración de Comunicaciones entre un PLC S7-200 y una computadora con HMI

Objetivo:

Establecer la comunicación entre un PLC S7-200 y una computadora, mediante la configuración de parámetros utilizando STEP 7 MicroWIN SP2 V3.1. Crear un HMI del tanque 2 y sus sensores y actuadores, utilizando InduSoft Web Studio v7.1.

Procedimiento

Conexión Física

El PLC Siemens S7-200 se conecta físicamente a un computador a través del cable multimaestro RS-232/PPI de Siemens. El cable multimaestro sirve como interfaz de RS-232 a RS-485 y permite trabajar a diferentes velocidades de transmisión, según lo que sea requerido. Los interruptores DIP se deben configurar de acuerdo a la velocidad y modo de transmisión con los cuales se desea operar.



Figura 1. Interruptores DIP del cable multimaestro RS-232/PPI

En la Figura 2 se muestra la asignación de pines proporcionada por el fabricante:

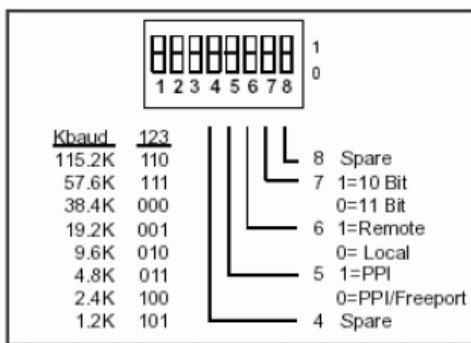


Figura 2. Asignación de pines del cable multimaestro RS-232/PPI

En este caso se trabajará a una velocidad de transmisión de 9.6 Kbps y en modo PPI/Freeport, por lo cual la configuración a utilizarse es 01000000 (Pin 2 en alto).

Los pasos para realizar la conexión son los siguientes:

- 1.- Se une el conector RS-232 (PC) del cable al puerto de comunicación de la PC (COM).
- 2.- Se une el conector RS-485 (PPI) del cable al puerto (0 ó 1) del PLC S7-200.
- 3.- Por último se debe verificar que la secuencia de los interruptores DIP sea 01000000.

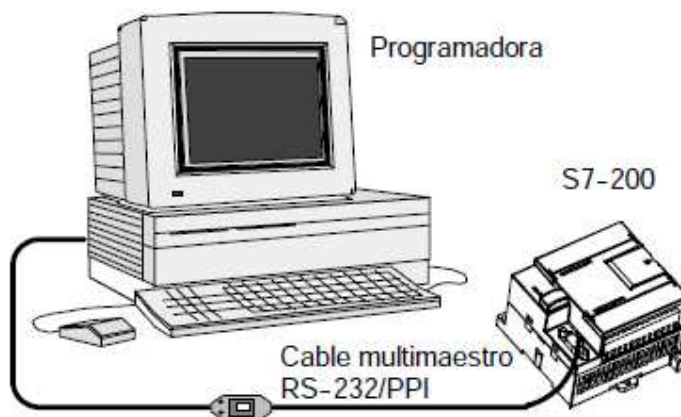


Figura 3. Conexión entre PLC S7-200 y computadora mediante cable multimaestro RS-232/PPI

Configuración de la comunicación

a) Protocolo de Comunicación

Se utiliza el protocolo serial S7 PPI, el cual fue desarrollado por Siemens para el uso en sus equipos de campo.

b) Configuración del Software STEP 7-Micro/WIN

Antes de la creación del HMI es necesario establecer la comunicación entre el PLC S7-200 y la computadora, a través de la configuración de parámetros en el software STEP 7-Micro/WIN. Estos parámetros comprenden la dirección remota del PLC, la velocidad de transmisión de los datos, la dirección del computador, entre otros. A continuación se detallan los pasos a seguir para la configuración de la comunicación:

- 1.- Se abre STEP 7-Micro/WIN y se crea un nuevo proyecto. Sobre el nombre del proyecto se hace click derecho para escoger el tipo de CPU y se selecciona el 226:

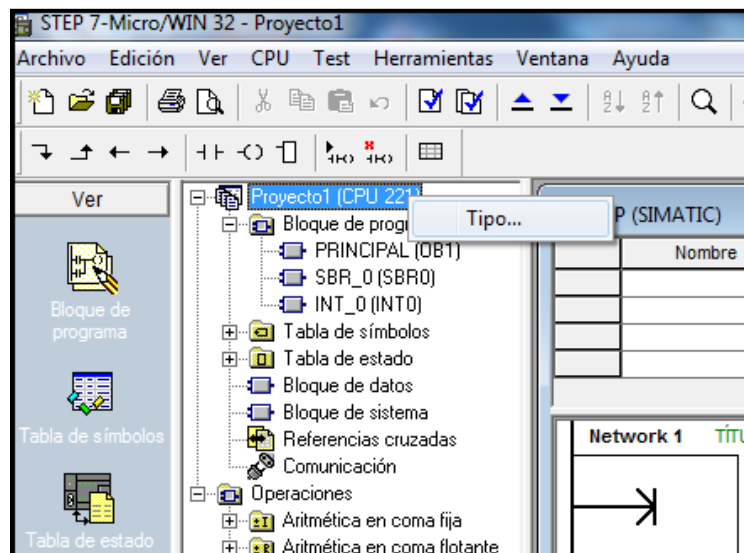


Figura 4. Creación de nuevo proyecto y selección de CPU en STEP 7-Micro/WIN

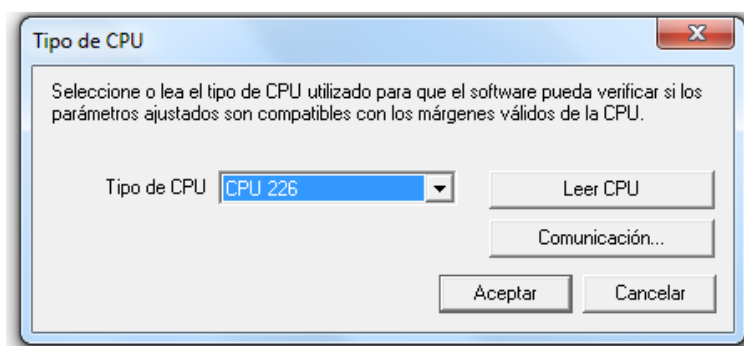


Figura 5. Selección de tipo de CPU en STEP 7-Micro/WIN

2.- En la barra de navegación ubicada en la parte izquierda del monitor se selecciona el ícono de "Comunicación". A continuación aparecerá un cuadro de diálogo en donde se deberán configurar los parámetros de comunicación. En la parte inferior del cuadro, aparece la dirección remota del PLC, cuyo valor estándar es 2.

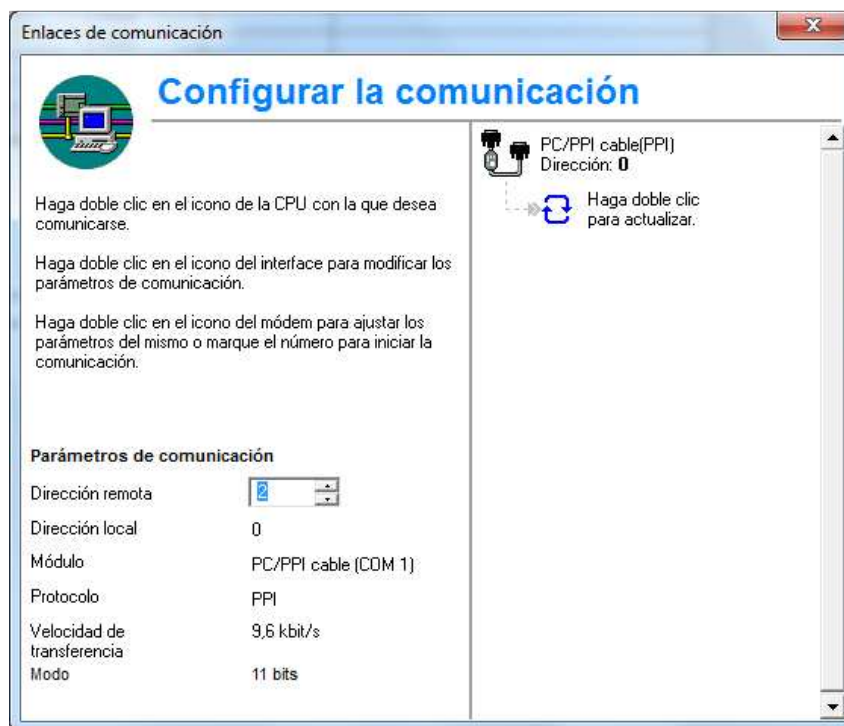


Figura 6. Cuadro de diálogo para configuración de parámetros de comunicación en STEP 7-Micro/WIN

3.- A continuación se modifican el resto de parámetros de comunicación. Se da doble click en el ícono de la parte superior derecha del cuadro de diálogo, se escoge la opción PC/PPI cable PPI y se selecciona el botón "Propiedades". En la pestaña de "PPI" se configura la dirección de la estación con el valor 12. Es necesario considerar que la dirección asignada debe tomar un valor diferente al utilizado para la CPU (es decir diferente a 2). En la parte inferior se escoge 9.6 kbps como la velocidad de transmisión. El resto de parámetros se mantienen con los valores preestablecidos.

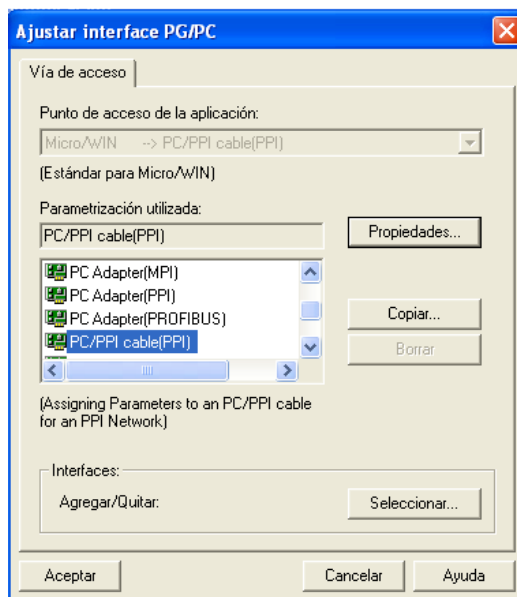


Figura 7. Interface PG/PC



Figura 8. Configuración de parámetros PPI

4.- Posteriormente, en la pestaña etiquetada como "Conexión local" se selecciona COM1.



Figura 9. Configuración de parámetros de la conexión local

5.- A continuación se selecciona el ícono "Bloque de sistema" y se corrobora que el puerto del PLC que se está utilizando (0 ó 1) tenga asignada la dirección remota correspondiente del PLC (2).

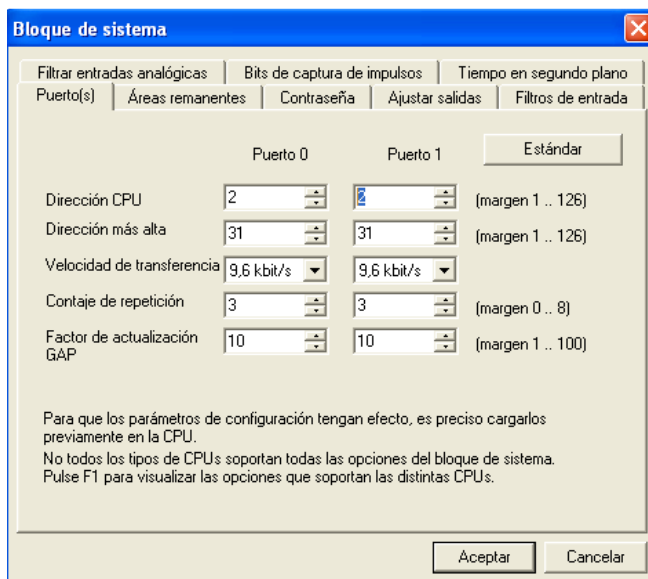


Figura 10. Bloque de Sistema

6.- Nuevamente en el ícono "Comunicación", se hace doble clic en la parte superior derecha para actualizar. Se espera unos segundos hasta que se establezca la comunicación.

7.- Una vez que la comunicación ha sido establecida, se selecciona la sección bloque de programa y se escribe el siguiente programa:

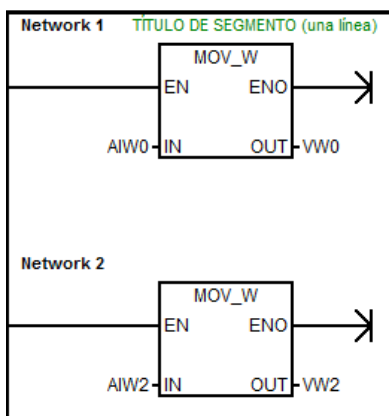


Figura 11. Programación escalera

8.- Por último se procede con la descarga del programa al PLC. Se selecciona en "Archivo" la opción "Cargar en CPU", tal como se muestra en la Figura 12:

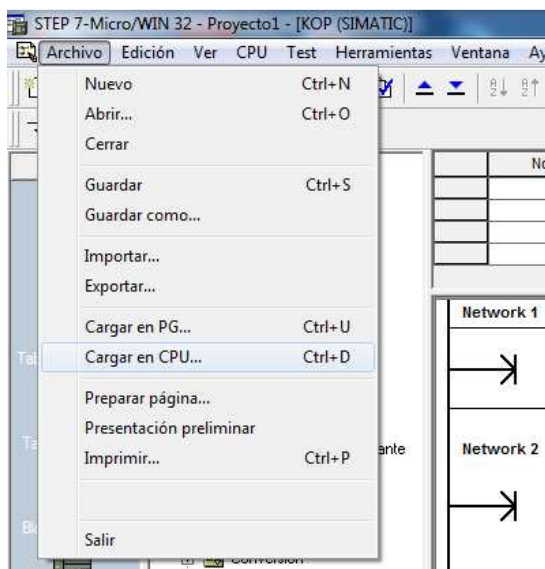


Figura 12. Carga de programa en CPU

c) Configuración del SIPPI driver en InduSoft Web Studio

Para la visualización de las variables del PLC S7-200 en la computadora se utilizará el software InduSoft Web Studio, el cual permite crear una interfaz gráfica entre el controlador y el usuario (HMI). Los pasos a seguir son los siguientes:

1.- Se abre el programa InduSoft Web Studio, se crea un nuevo proyecto y se le asigna un nombre. En asistente de proyectos se selecciona la plantilla aplicación vacía y se escoge la resolución.

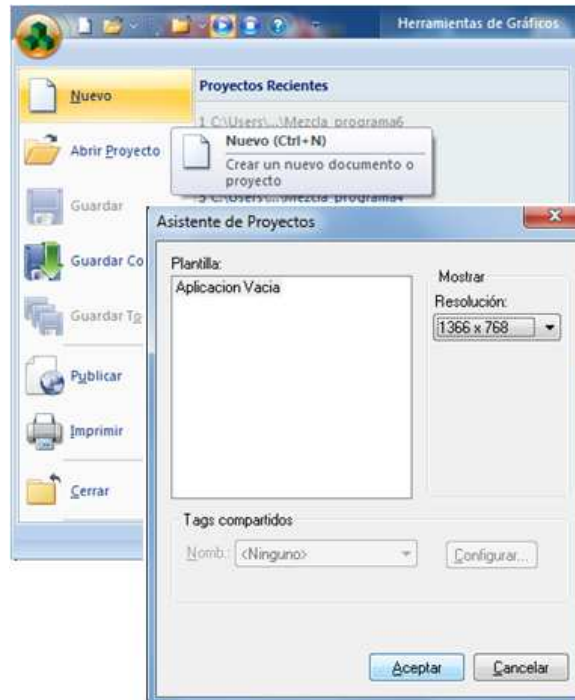


Figura 13. Creación de nuevo proyecto en InduSoft Web Studio

2.- A continuación, en explorador de proyectos se selecciona la pestaña inferior de comunicaciones, se da click derecho sobre la carpeta "Drivers" y se escoge "Agregar/Eliminar drivers":

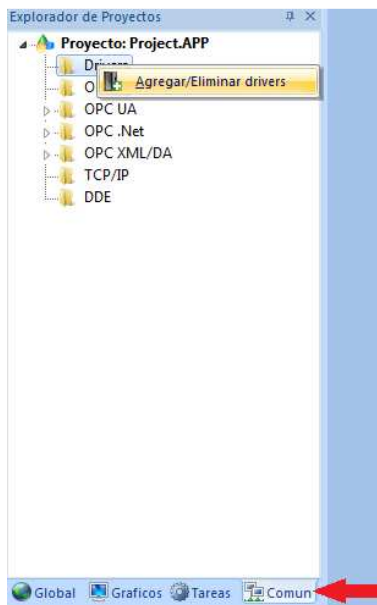


Figura 14. Acceso a drivers de comunicación

3.- En la ventana "Drivers de comunicación" se selecciona el driver SIPPI.

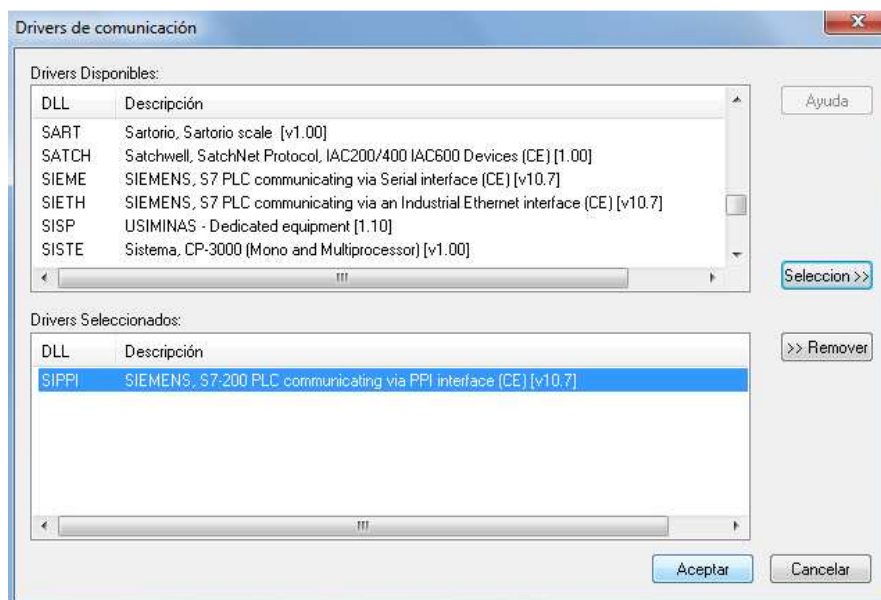


Figura 15. Selección de driver de comunicación

4.- Luego, se da click derecho sobre la carpeta SIPPI y se va a "Configuración". En "Puerto Serie" se escoge COM1 y el resto de campos se mantienen como los predeterminados.

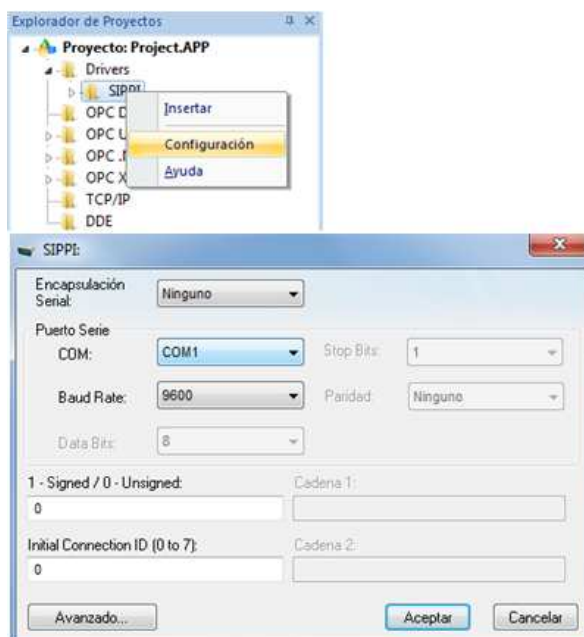


Figura 16. Configuración del driver SIPPI

5.- A continuación se accede a la hoja del driver. Se da doble click en la carpeta SIPPI y se abre la hoja. Los campos de la parte superior se llenan tal como se ilustra en la Figura 17. En "Nombre de tag" es necesario crear etiquetas para introducirlas en este campo. En el caso de las entradas/salidas digitales se escoge el tipo booleano mientras que para las entradas/salidas análogas se selecciona el tipo entero. En "Estación" se escribe la dirección remota del PLC que es 2. En el tercer recuadro se asigna una dirección a cada tag, la cual debe concordar con las conexiones físicas que se realizan. Para las entradas digitales se utiliza "I" seguida del número de byte y especificando el bit precedido por un punto, como por ejemplo I0.0. Para las salidas digitales la nomenclatura utilizada es la misma descrita para las entradas pero en lugar de "I" se utiliza "Q". En el caso de las entradas análogas, se utiliza la dirección de memoria volátil (VW) que le fue asignada previamente en el programa STEP 7 MicroWIN. Los dos últimos parámetros se dejan como los predeterminados.

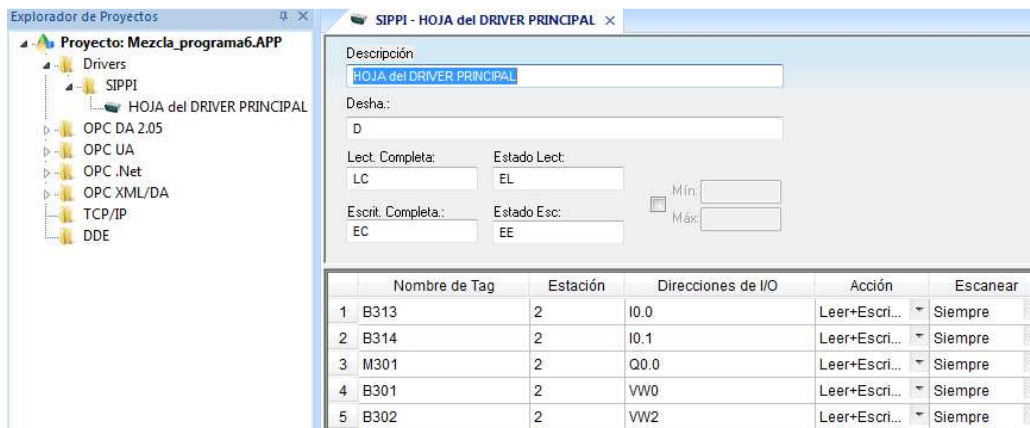


Figura 17. Configuración de parámetros en la hoja del driver SIPPI

6.- Después, se escoge la pestaña "Gráficos", se da click derecho sobre la carpeta "Pantallas" y se escoge "Insertar". En la ventana "Atributos de pantalla" se presiona el botón "Aceptar".

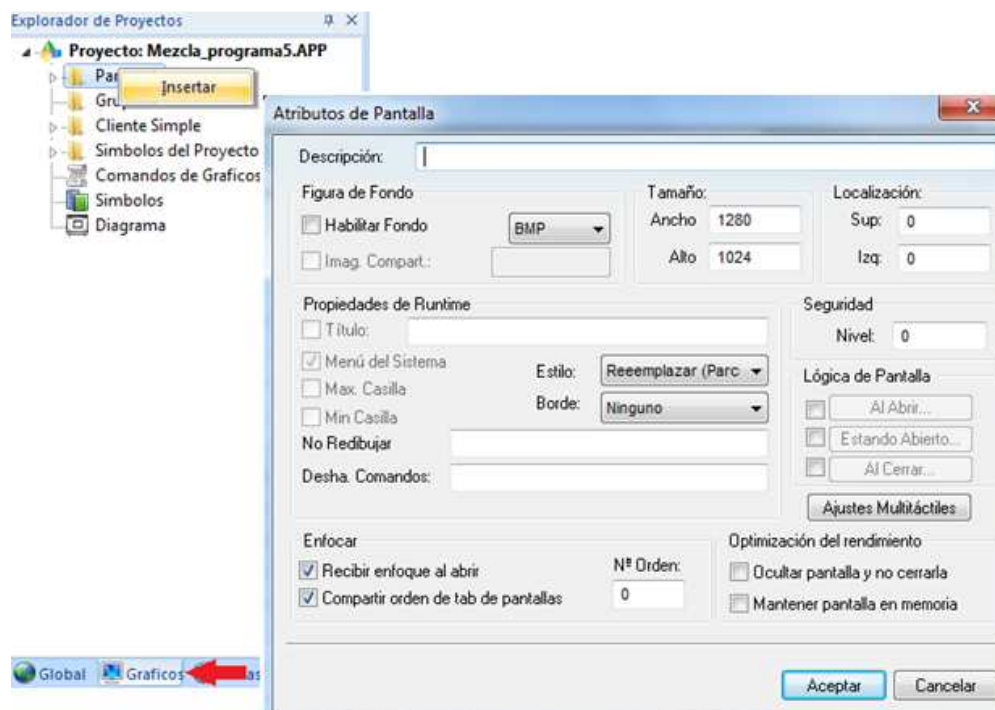


Figura 18. Creación de pantallas

7.- Una vez que se tiene creada la pantalla en blanco, se escoge la opción "Símbolos", se da click sobre la carpeta "Símbolos de Sistema" y se selecciona alguna de las subcarpetas de acuerdo al símbolo que se desee insertar en la pantalla.

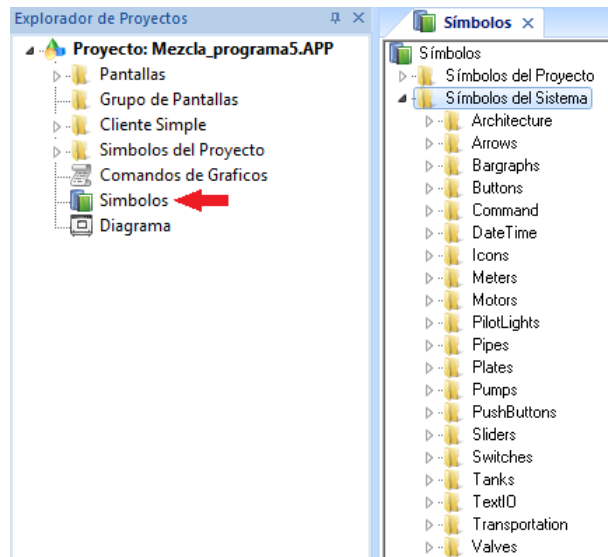


Figura 19. Inserción de símbolos en la pantalla

8.- A los símbolos insertados en la pantalla, se les asigna un tag. En la Figura 20 se le asigna el tag "B314" a una luz. Este tag corresponde a un sensor de nivel ("switch") del tanque.

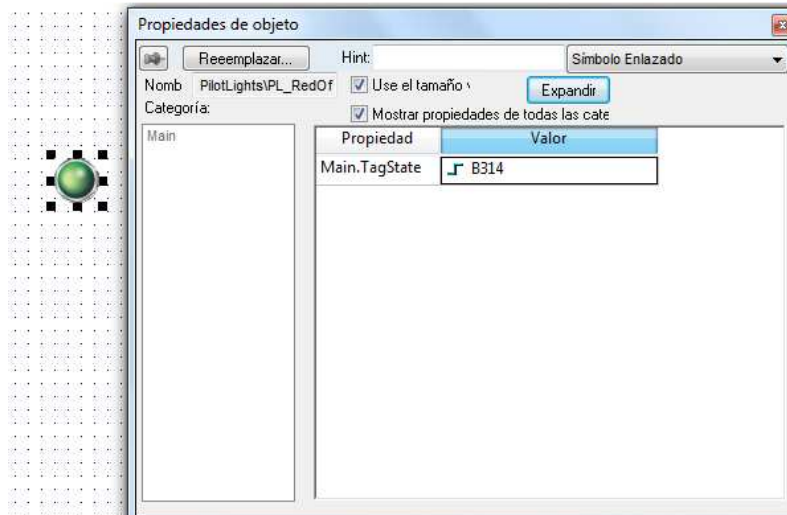


Figura 20. Vinculación de tags con símbolos

9.- Finalmente, una vez que se han insertado todos los objetos que forman parte de la planta y se les ha asignado el tag respectivo, se procede a ejecutar la aplicación, presionando el botón "play" ubicado en la parte superior de la pantalla.

Práctica

En base a las guías dadas, elabore un HMI del tanque 2 y sus sensores. Al ejecutar la aplicación se deben visualizar los valores leídos por los sensores y se debe tener un interruptor que permita encender y apagar la bomba desde dicha aplicación.

PRÁCTICA 2: Configuración de Comunicaciones entre el EasyPort y una computadora con HMI

Objetivo:

Establecer la comunicación entre el EasyPort del módulo FESTO Compact Workstation y una computadora y crear un HMI del módulo con sus sensores y actuadores, utilizando InduSoft Web Studio v7.1.

Procedimiento

Todos los sensores y actuadores del módulo se encuentran ya conectados así que lo primero que se debe hacer es realizar la comunicación entre el EasyPort y la computadora, mediante el EzOPC, el cual es un software propio de FESTO, el mismo que está incorporado con la compra del equipo.

Conexión Física

Para la conexión física se utilizará el dispositivo EasyPort de FESTO. Este trabaja a 24VDC. Posee 16 salidas digitales de 24VDC, 2 salidas análogas de 0-10VDC o de -10 a 10VDC con una resolución de 12bits. Tiene 16 entradas digitales de 24VDC, 4 entradas análogas de 0-10 VDC o -10 a 10VDC con una resolución de 12bits. Tiene dos interfaces de comunicación por serial o por USB ambos aislados eléctricamente. Utiliza un protocolo ASCII, a 115.2 kBaud, 8 bits sin paridad con 1 bit de parada (8N1).

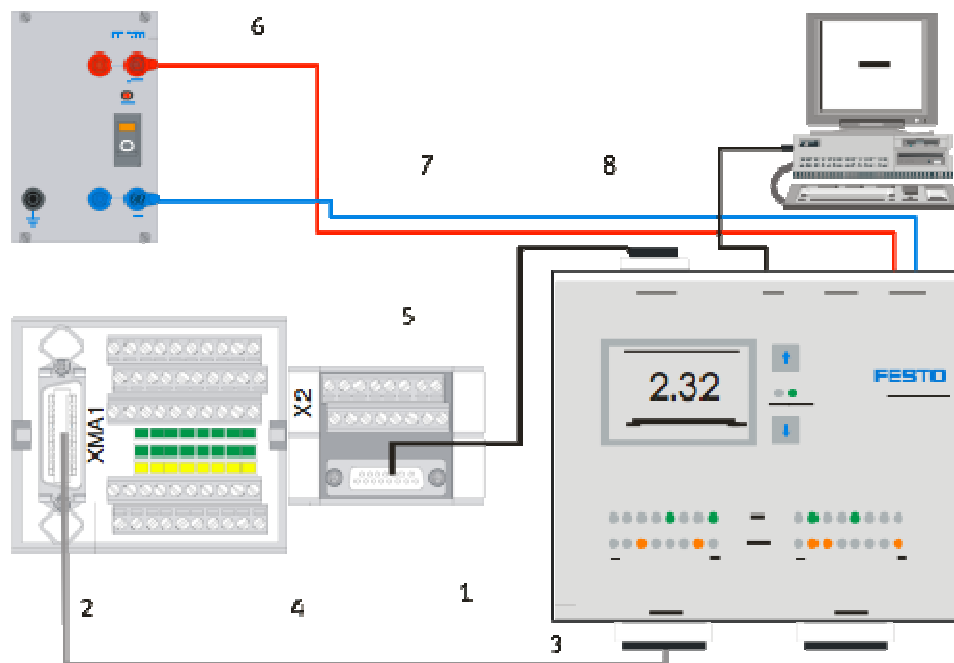


Figura 1. Conexiones entre el tablero de entradas y salidas con el EasyPort, Helmich, 2008.

A continuación se detalla la lista de entradas y salidas, su tag, la asignación de pines y su descripción:

Tag	EasyPort	Descripción
Entradas Digitales		
B102	I0	Sensor de caudal (Frecuencia de 0 a 1000Hz)
S111	I1	Interruptor flotador, overflow del tanque 101
S112	I2	Interruptor flotador, nivel bajo tanque 102
B113	I3	Sensor capacitivo nivel mínimo tanque 101
B114	I4	Sensor capacitivo nivel máximo tanque 101
S115	I5	Interruptor de la válvula de bola V102 cerrado
S116	I6	Interruptor de la válvula de bola V102 abierto
No Usado	I7	
Salidas Digitales		
M102	Q0	Abrir válvula de bola ON/OFF V102
E104	Q1	Prender calentador del tanque 101
K1	Q2	Relé selecciona si la bomba es binaria o análoga

Tag	EasyPort	Descripción
M1	Q3	Prender bomba P101 binario
M106	Q4	Prender válvula proporcional
No Usado	Q5	
No Usado	Q6	
No Usado	Q7	
Entradas Análogas		
B101	AI 0	PV variable del proceso, nivel en tanque 102
B102	AI 1	PV variable del proceso, caudal en el sistema
B103	AI 2	PV variable del proceso, presión en el sistema
B104	AI 3	PV variable del proceso, temperatura en el sistema
Salidas Análogas		
P101	AQ 0	Manipular actuador, bomba P101
V106	AQ 1	Manipular actuador, válvula proporcional V106

Tabla 1. Tags y asignación de pines del EasyPort.

Configuración del OPC DA 2.05 en InduSoft Web Studio.

El software Indusoft posee la opción de utilizar el cliente OPC DA 2.05. El OPC DA se utiliza para datos en tiempo real y no datos históricos. Hay tres atributos asociados con este tipo de cliente, el valor, la calidad del valor y un tiempo de control. Estos atributos deben ser respondidos por el servidor OPC al haber una petición, sino tiene un tiempo de control el servidor lo crea.

Para agregar el cliente OPC, en la pestaña de comunicación de InduSoft se selecciona la carpeta "OPC DA 2.05" y se identifica inmediatamente el servidor y sus tags por defecto.

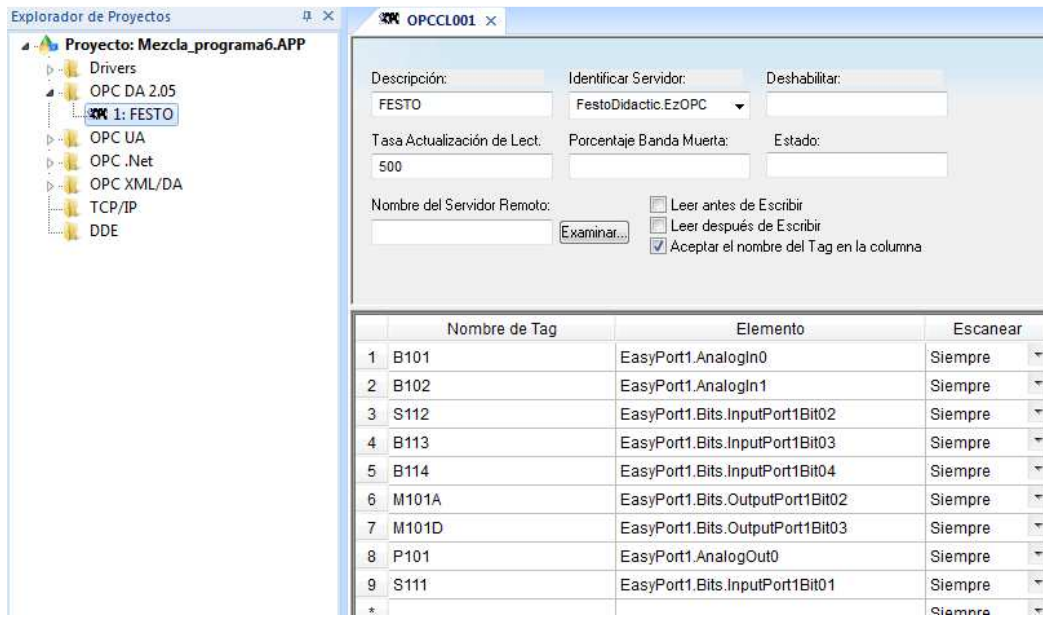


Figura 50. OPC DA 2.05, InduSoft Web Studio v7.1.

Mantenga únicamente los tags que serán utilizados en la programación de la interfaz gráfica.

Práctica

Una vez designados los tags a utilizarse, en base a las guías dadas y a la práctica número 1, elabore un HMI del módulo FESTO Compact Workstation. Al ejecutar la aplicación se deben visualizar los valores leídos por los sensores y se debe tener un interruptor que permita encender y apagar la bomba desde dicha aplicación.

PRÁCTICA 3: Programación de un proceso de Mezcla con HMI

Objetivo:

Integrar en un HMI las aplicaciones elaboradas en las prácticas 1 y 2 y programar un proceso de mezcla, utilizando InduSoft Web Studio v7.1.

Procedimiento

Para esta práctica es necesario haber realizado las 2 prácticas anteriores (1 y 2). A continuación se detallan los pasos a seguir:

1.- Se realizan las conexiones físicas requeridas para la comunicación del PLC S7-200 con la computadora. De igual manera se conecta el EasyPort vía usb a la computadora y se siguen los pasos para establecer la comunicación.

2.- Se abre la aplicación realizada en InduSoft del PLC S7-200 (práctica 1) y en la pestaña de comunicaciones se repite lo indicado en la práctica 2.

3.- Se crea una pantalla, en la que se incorporen todos los sensores y actuadores del sistema completo y se asignan los tags correspondientes a cada objeto.

4.- A continuación se escoge "Comandos de Gráficos" y en la ventana que aparece en el lado derecho del monitor se puede escribir el código de programación. El lenguaje de programación es Visual Basic.

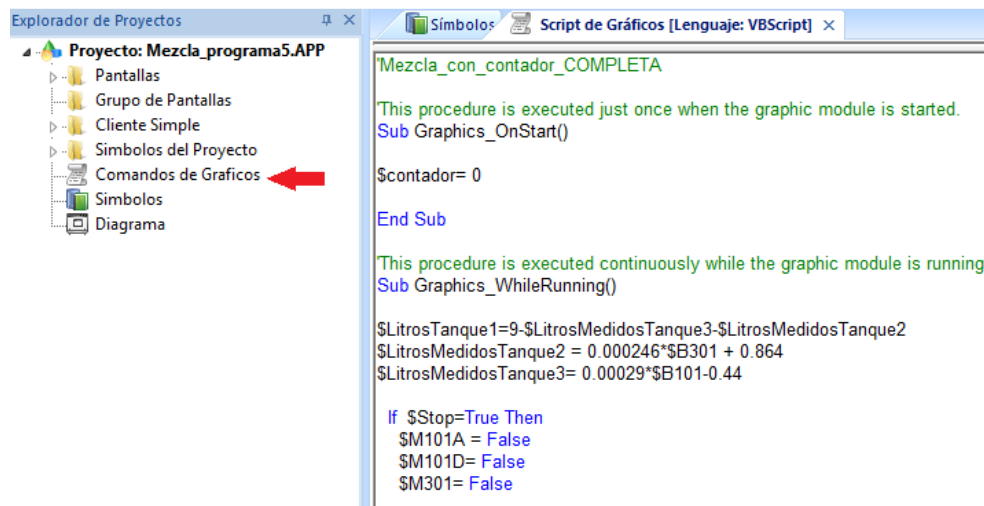


Figura 1. Escritura de programas en InduSoft

5.- Para crear tags que no están direccionados a una entrada/salida física del PLC, se selecciona la pestaña "Global" y se da click derecho en las carpeta "Etiquetas del Proyecto" y se escoge la opción "Insertar Tag":

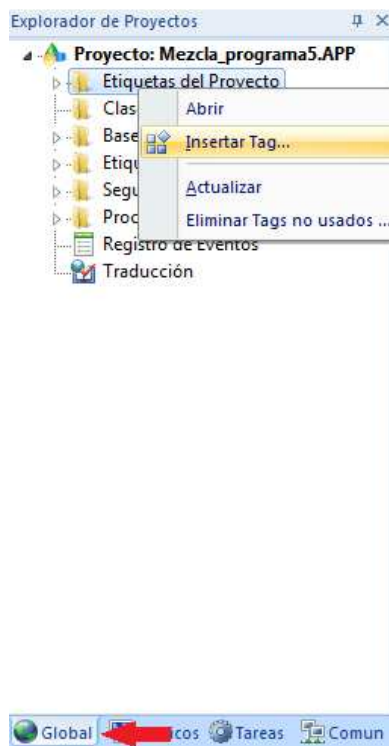


Figura 2. Creación de tags

6.- En el cuadro de diálogo "Nuevo Tag", en el primer recuadro se la asigna un nombre al tag. En el campo "Tipo" se selecciona booleano en caso de que se trate de una variable que va a tomar dos valores (on/off), mientras que si se trata de una variable con punto flotante, se escoge la opción real.



Figura 20. Asignación de nombre y tipo de tag

7.- Para visualizar la lista de todos los tags existentes se abre la carpeta "Etiquetas del Proyecto" y se selecciona "Vista Hoja de datos" como se muestra en la siguiente figura:

	Nombre	Matriz	Tipo	Descripción	Ambito
1	B101	0	Entero		Servidor
2	B102	0	Entero		Servidor
3	S112	0	Booleano		Servidor
4	B113	0	Booleano		Servidor
5	B114	0	Booleano		Servidor
6	M101A	0	Booleano		Servidor
7	M101D	0	Booleano		Servidor
8	P101	0	Entero		Servidor

Figura 21. Lista de tags

Práctica

En base a las guías dadas, elabore un HMI que integre los sensores y actuadores conectados al PLC S7-200 y al EasyPort. Programar en la ventana de VBScript un proceso de mezcla automático del sistema. Los 2 tanques inferiores son los que suministran el líquido a través de las bombas mientras que el tanque superior es el que almacena la mezcla. La aplicación creada debe permitir ingresar la cantidad de líquido (en litros) que se desea transferir desde cada uno de los tanques de suministro. Para mayor información con respecto a la escritura de programas en VBScript de InduSoft remitirse a "VBScript Reference Manual for InduSoft Web Studio".

