



Universidad del Azuay

Facultad de Ciencia y Tecnología

Escuela de Ingeniería Electrónica

Diseño e Implementación de un sistema S.C.A.D.A. para
el control de temperatura y llenado de un tanque.

Trabajo de graduación previo a la obtención del título de
Ingeniero Electrónico

Autores:

Diego Fernando Bravo Chicaiza
Edgar Mauricio Vanegas Tenesaca

Director

Ing. Hugo Torres Salamea.

Cuenca – Ecuador

2010

DEDICATORIA

Dedicado con mucho cariño para mi abuelita Mercedes, mi madre Luz y mi hermano Kleber, a quienes admiro y respeto por su don de perseverancia. Por estar día tras día brindándome su apoyo incondicional, gracias a ellos hoy puedo culminar una etapa más de mi vida.

Diego Fernando Bravo.

DEDICATORIA

Quiero dedicar este trabajo a todas las personas que siempre me alentaron a no decaer en este periodo de mi vida y durante todo este tiempo estuvieron a mi lado incondicionalmente, mis padres Marianita y el viejito Jimmy (welo) ya que con sus buenos ejemplos y amor siempre estuvieron apoyándome, a mi amada esposa Jessy por tenerme tanta paciencia, comprensión y por los días que se sacrifico a mi lado también al tío Juan chico siendo la persona especial que nunca falta y no puedo olvidarme de mi querido cuñadito Miguel por su ayuda y su ávidos consejos. Pero en especial este trabajo es para la fuente de mi inspiración que siempre está a mi lado a mi pequeño **DUMANY (TONY)**.

Que mi Dios les pague.

Edgar Vanegas T.

AGRADECIMIENTOS

Agradezco a Dios, por darme fortaleza y valor de lograr las metas que me he trazado.

Al Ing. Hugo Torres, ya que con su acción generosa ha brindado un gran aporte para la culminación de este trabajo.

Al los ingenieros Leonel Pérez y Santiago Orellana, por aportar con sus conocimientos.

Por último, un agradecimiento especial para un buen amigo, Edgar Loján que contribuyo de manera especial en el desarrollo de este trabajo.

Agradezco a DIOS por todos los días de trabajo y a mi amada Jessy que siempre me apoya.

También quiero agradecer al Ing. Hugo Torres por haber contribuido en este trabajo y guiarnos desinteresadamente

De igual manera a los ingenieros Leonel Pérez y Santiago Orellana, por su colaboración en la revisión de nuestro trabajo de graduación.

Diego.

Edgar.

RESUMEN

Existe en el país un gran desarrollo del cultivo en invernaderos pero uno de los problemas más importantes a los que se enfrentan este tipo de técnica es el control de su temperatura, que en nuestro país en su mayor parte, es realizado en forma manual lo que involucra mayores costos a largo plazo.

El presente trabajo de graduación presenta la implementación de un invernadero automatizado dentro del cual es posible obtener las condiciones artificiales del clima y con ello cultivar en condiciones optimas con resultados de eficiencia y calidad en los productos, además de tener la producción en todo el año mediante el control de Autómatas Programables y la visualización de todos los procesos mediante un computador personal que podrá ser manipulado por personal debidamente especializado.

ABSTRACT

Today in the country exists a big development of cultivation in winter quarters, but the most important problem this technique has to face is the temperature control, which in our country is made in handy form, giving us as a result greater expensive in a long time.

This graduated work show you the implementation of an automation winter quarters in which is possible to obtain the artificial conditions of the winter and with them to cultivate in very best obtaining as a result products with efficiency and quality, moreover we could have the opportunity to produce these products all year troughs the control of the programming automates and the visualization of all processes through a personal computer which could be manipulated by specialized personal.

ÍNDICE DE CONTENIDOS

Dedicatoria	ii
Agradecimientos	iv
Resumen	v
Abstract	vi
Índice de Contenidos	vii
Índice de Gráficos	xi
Índice de Tablas	xvii
Índice de Anexos	xvii

INTRODUCCION	1
---------------------	----------

CAPITULO 1: SENSORES

1.1. Definición	3
1.2. Características	4
1.3. Clasificación	4
1.3.1. Pasivos	5
1.3.2. Activos	6
1.3.3. Analógicos	6
1.3.4. Digitales	7
1.4. Sensores de temperatura	8

1.5. Detectores	10
1.6. Detector de nivel	11
1.7. Tipos de salida	12

CAPÍTULO 2: PROGRAMA STEP7 PARA EL PLC SIEMENS S7-200.

2.1. Introducción.	15
2.2. Procedimiento básico con STEP7	16
2.2.1. Instalación del programa Step7	17
2.2.2. Arrancar el SIMATIC	17
2.2.3. Crear el proyecto	18
2.2.4. Programación en KOP	20
2.3. Configuración del sistema de automatización S7-200	21
2.3.1. Resumen breve	21
2.3.2. Cómo configurar la comunicación utilizando el cable PC/PPI	23
2.3.3. Cómo establecer la comunicación con la CPU S7-200	25
2.3.4. Cómo cambiar los parámetros de comunicación de la CPU	27
2.4. Aplicación al proyecto monográfico	28

CAPÍTULO 3: SISTEMA SCADA.

3.1. Introducción a los sistemas S.C.A.D.A.	41
3.1.1. Prestaciones	42
3.1.2. Requisitos	43
3.1.3. Modulo de un SCADA	43
3.2. Conceptos Asociados a un sistema S.C.A.D.A.	44

3.2.1.	Tiempo Real	45
3.2.2.	Hardware en sistemas de supervisión	46
3.2.3.	Tarjetas de expansión.	47
3.2.4.	La estructura abierta	47
3.3.	Introducción a LabView.	48
3.3.1.	Filosofía de LabView	49
3.3.2.	Panel Frontal y Diagrama de Bloques	50
3.3.3.	Tarjetas de adquisición	51
3.3.4.	La estructura de Labview	52
3.4.	Aplicación al proyecto monográfico	52

CAPITULO 4: CONTROL PID DE LA TEMPERATURA.

4.1.	Introducción	67
4.2.	Estructura de control PID.	69
4.2.1.	Acción de control proporcional P.	70
4.2.2.	Acción de control integral I.	70
4.2.3.	Acción de control proporcional-integral PI.	71
4.2.4.	Acción de control proporcional-derivativa PD.	72
4.2.5.	Acción de control proporcional-integral-derivativa PID.	72
4.3.	Métodos clásicos de ajuste de Ziegler and Nichols	73
4.4.	Método de Oscilación	73
4.5.	Método Basado en la Curva Reacción.	75
4.6.	Modificaciones de los esquemas de control PID	77
4.7.	Aplicación al proyecto monográfico	78

CAPITULO 5: INSTALACION Y PRUEBAS DEL SISTEMA DE CONTROL.

5.1. Calibración del sensor de temperatura en diferentes ambientes	92
5.2. Ubicación del sensor de temperatura y del tanque en la maqueta.	92
5.3. Configuración del software	93
5.4. Verificación del correcto funcionamiento del sistema	95
COCLUSIONES	99
RECOMENDACIONES	100
BIBLIOGRAFIA	101
ANEXOS	103

INDICE DE GRAFICOS

Gráfico1. Funcionamiento de un sensor pasivo	5
Gráfico2. Funcionamiento de un sensor activo	6
Gráfico3. Curva de sensor análogo	7
Gráfico4. Sensores Análogo	7
Gráfico5. Tipos de encapsulados del IC LM35	10
Gráfico6. Detector tipo flotante	13
Gráfico7. Amplificador de ganancia 1 a 10	14
Gráfico8. Placa de amplificador con ganancia 1 a 10	14
Gráfico9. Administrador Simatic	18
Gráfico10. Pantalla Principal de STEP7-Micro/WIN	20
Gráfico11. Programación en KOP	21
Gráfico12. Comunicación entre CPU y PC por cable PP	22
Gráfico13. Configuración de CPU de la serie S7-200	22
Gráfico14. Configuración de la comunicación entre PC y la CPU 224	23
Gráfico15. Ajuste de Interface entre PC y CPU	24

Gráfico16. Configuración de la velocidad de comunicación	24
Gráfico17. Configuración del puerto de comunicación	25
Gráfico18. Ventana de comunicación	26
Gráfico19. Configuración de comunicación	27
Gráfico20. Configuración de parámetros de comunicación	28
Gráfico21. Partes externas de un PLC de la gama S7-200	29
Gráfico22. Programa de llenado de nivel de tanque	30
Gráfico23. Programación SERVIDOR de S7-200	31
Gráfico24. Configuración de Asistente de Ethernet	31
Gráfico25. Configuración Posición de Modulo	32
Gráfico26. Configuración de Dirección IP y Mascara de Subred	32
Gráfico27. Configuración del número de conexiones	33
Gráfico28. Configuración de enlace de s7-200	34
Gráfico29. Configuración de protección CRC	34
Gráfico30. Dirección de comunicación para servidor	35
Gráfico31. Finalización de configuración del SERVIDOR	35
Gráfico32. Bloque de Servidor en programa principal	36

Gráfico33. Configuración de Enlaces en cliente	37
Gráfico34. Lectura desde el servidor al cliente	37
Gráfico35. Escritura desde el cliente al servidor	38
Gráfico36. Configuración de protección CRC	38
Gráfico37. Dirección de comunicación para cliente	39
Gráfico38. Finalización de configuración del CLIENTE	39
Gráfico39. Bloque de Cliente en programa principal	40
Gráfico40. Esquema básico de un SCADA	42
Gráfico41. Módulos o software de un SCADA	44
Gráfico42. Pantallas de software Labview	48
Gráfico43. Pantalla para configura la comunicación OPC	53
Gráfico44. Nombre del canal de comunicación OPC	54
Gráfico45. Selección del dispositivo para la comunicación OPC	54
Gráfico46. Parámetros de Configuración	55
Gráfico47. Identificador Maestro OPC	55
Gráfico48. Canal de comunicación ya establecido	56
Gráfico49. Nombre de dispositivo a usar el canal	56

Gráfico50. Modelo del dispositivo declarado en el canal de comunicación	57
Gráfico51. Identificación del dispositivo en el canal de comunicación	57
Gráfico52. Configuración de tiempo de muestreo	58
Gráfico53. Dispositivo configurado en el canal S7-200	58
Gráfico54. Propiedades de Tag	59
Gráfico55. Variables declaradas en OPC	60
Gráfico56. Configuración OPC Cliente	60
Gráfico57. Ventana de inicio de software Labview	61
Gráfico58. Creación de nuevo proyecto en Labview	61
Gráfico59. Creación de nuevo Servidor en Labview	62
Gráfico60. Modificación de tiempos de muestreo en Labview	62
Gráfico61. Creación de la variable en Labview	63
Gráfico62. Propiedades de una variable en Labview	63
Gráfico63. Ubicación de la variable entre OPC y Labview	64
Gráfico64. Creación de nuevo proyecto en Labview	64
Gráfico65. Creación de nuevo VI en Labview	65
Gráfico66. Variable de S7-200 funcionando en Labview	65

Gráfico67. Diagrama de bloques de un control PID	68
Gráfico68. Estructura de un control PID	69
Gráfico69. Respuesta de la planta con ganancia crítica	75
Gráfico70. Respuesta al escalón de la planta	76
Gráfico71. Estructura del sistema en maqueta	78
Gráfico72. Etapa de potencia para manejar el dicroico	79
Gráfico73. Placa Física de potencia para activación de calefacción	79
Gráfico74. Etapa de potencia para temperatura	80
Gráfico75. Asistente de Operaciones para PID	82
Gráfico76. Selección de Asistente PID	82
Gráfico77. Indicador de lazo PID	83
Gráfico78. Parámetros del PID	83
Gráfico79. Configuración de ingresos y salidas en PID	84
Gráfico80. Programación de alarmas del PID	84
Gráfico81. Dirección de memoria para el PID	85
Gráfico82. Nombres a las subrutinas de PID	85
Gráfico83. Finalización del Asistente de Operaciones PID	86

Gráfico84. Bloque de PID en programa principal	86
Gráfico85. Panel de sintonía PID	87
Gráfico86. Configuración del modulo EM235.	88
Gráfico87. Maqueta de proceso.	89
Gráfico88. Llenado de tanque en maqueta.	90
Gráfico89. Maqueta de Invernadero.	90
Gráfico90. Placas de comunicación y de actuadores.	91
Gráfico91. Dicroico que simula perturbación.	91
Gráfico92. Sensor de temperatura en el interior del invernadero.	93
Gráfico93. Pantalla de programa principal en Labview.	94
Gráfico94. Forma de onda del proceso sin parámetros.	95
Gráfico95. Forma de onda con error.	96
Gráfico96. Forma de onda con error y actuador oscilante.	97
Gráfico97. Forma de onda del sistema estable y sin error.	98

INDICE DE TABLAS

Tabla 1. Sensores de temperatura y sus características.	8
Tabla 2. Características de los Fin carrera.	12
Tabla 3. Parámetros de ajuste (método de oscilación).	74
Tabla 4. Parámetros de ajuste (método curva de reacción).	77
Tabla 5. Tabla de valores cálculos para K_p , K_i y K_d .	81
Tabla 6. Tabla de Reacción del Sistema.	97

INDICE DE ANEXOS.

Anexo1: Programa llenado de tanque con la configuración de un cliente.	103
Anexo2: Variables de comunicación entre el PLC S7-200 y Labview.	106
Anexo3: Programación PID y S7-200 como Servidor.	108

Bravo Chicaiza Diego Fernando.

Vanegas Tenesaca Edgar Mauricio.

Ing. Hugo Torres Salamea.

Junio 2010.

Diseño e Implementación de un sistema S.C.A.D.A. para el control de temperatura y llenado de un tanque.

INTRODUCCION

El presente trabajo sugiere la implementación de un invernadero automatizado dentro del cual es posible obtener las condiciones artificiales de clima, y con ello cultivar plantas fuera de estación en condiciones óptimas. Además de tener la oportunidad de mejorar la eficacia y calidad del negocio.

Realizar una aplicación, especialmente diseñada para funcionar sobre ordenadores en el control de producción, proporcionando comunicación con los dispositivos de campo (autómatas programables) y controlando el proceso de forma automática desde la pantalla del ordenador. Además, provee de toda la información que se genera en el proceso productivo a diversos usuarios, tanto del mismo nivel como de otros supervisores dentro de la empresa.

El primer capítulo de nuestro trabajo, hace referencia a la gama de sensores análogos y digitales con sus respectivas características, así como sus ventajas y desventajas en el aspecto industrial.

El capítulo dos será relacionado directamente con el manejo del software Step 7 Microwin que será el que programe al PLC el mismo que llevara a cabo las instrucciones para que funcionen los diferentes sistemas de forma independiente.

Posteriormente el capítulo tres contiene la información de un sistema SCADA que será manejado por medio de un enlace llamado OPC para la comunicación con el computador y por ende será visualizado todo el proceso en el software Labview.

El capítulo cuarto representa la esencia de este trabajo ya que se tendrá que mantener la temperatura interna del invernadero de forma constante ya sea que se modifique la temperatura por factores externos.

El último capítulo será directamente la recopilación de todos los capítulos anteriores sobre una maqueta pero obteniendo los resultados de correcto funcionamiento de todo el proceso del invernadero.

CAPÍTULO 1

SENSORES

1.1. Definición.

“Un sensor es un dispositivo capaz de medir magnitudes físicas o químicas, llamadas variables de instrumentación, y transformarlas en variables eléctricas. Las variables de instrumentación pueden ser por ejemplo: temperatura, intensidad lumínica, distancia, aceleración, etc. Una magnitud eléctrica puede ser una resistencia eléctrica (como en una RTD), una tensión eléctrica (como en un termopar), una corriente eléctrica (como en un fototransistor), etc.”¹

Un sensor se diferencia de un transductor en que el sensor está siempre en contacto con la variable de instrumentación con lo que puede decirse también que es un dispositivo que aprovecha una de sus propiedades con el fin de adaptar la señal que mide para que la pueda interpretar otro dispositivo. Como por ejemplo el termómetro de mercurio que aprovecha la propiedad que posee el mercurio de dilatarse o contraerse por la acción de la temperatura.

Un transductor es un dispositivo capaz de transformar o convertir un determinado tipo de energía en otra diferente a la salida, siendo usado principalmente en la industria para obtener información de entornos físicos o químicos y conseguir

¹BOSH, Robert, Los sensores, Alemania, 2002.

señales o impulsos eléctricos o viceversa. Un sensor es un tipo de transductor que transforma la magnitud que se quiere medir o controlar, en otra, que facilita su medida. Pueden ser de indicación directa (un termómetro de mercurio) o pueden estar conectados a un indicador (posiblemente a través de un convertidor analógico a digital, un computador y un display) de modo que los valores detectados puedan ser leídos por un humano.

1.2. Características.

Entre las características técnicas de un sensor destacan las siguientes:

- “Rango de medida: dominio en la magnitud medida en el que puede aplicarse el sensor.
- Precisión: es el error de medida máximo esperado.
- Offset o desviación de cero: valor de la variable de salida cuando la variable de entrada es nula.
- Linealidad o correlación lineal.
- Sensibilidad de un sensor: relación entre la variación de la magnitud de salida y la variación de la magnitud de entrada.
- Resolución: mínima variación de la magnitud de entrada que puede apreciarse a la salida.
- Rapidez de respuesta: puede ser un tiempo fijo o depender de cuánto varíe la magnitud a medir.
- Derivas: son otras magnitudes, aparte de la medida como magnitud de entrada, que influyen en la variable de salida.
- Repetitividad: error esperado al repetir varias veces la misma medida.”²

1.3. Clasificación.

Aunque es un poco complicado realizar una clasificación única, debido a la gran cantidad de sensores que existen actualmente, las siguientes son las clasificaciones más generales y comunes.

² KLINGER, Charles, Sensores Virtuales, Canadá, 2001. <http://www.scribd.com/doc/21905958/Que-es-un-sensor>

- I. Un tipo de clasificación muy básico es diferenciar a los sensores entre
 - Pasivos.
 - Activos.
- II. Según el tipo de señal que proveen a la salida:
 - Digitales.
 - Analógicos.

1.3.1. Sensores Pasivos.

Los sensores pasivos requieren de una fuente de alimentación externa para poder efectuar su función ya que estos examinan la energía emitida por la fuente externa (detectores de calor o simples telescopios, por ejemplo). Aún siendo menos efectivos que los sensores activos, los sensores pasivos gastan menos energía.

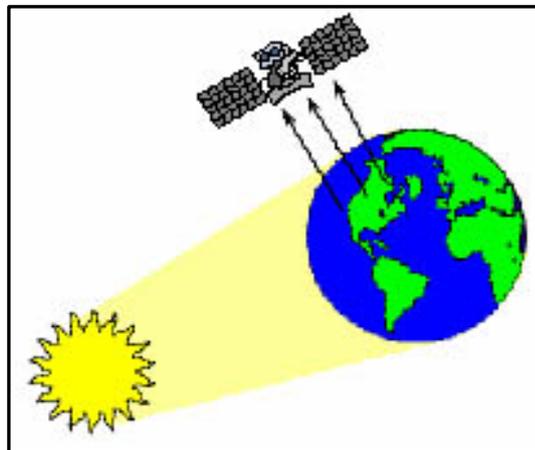


Gráfico1. Funcionamiento de un sensor pasivo.

Fuente: SOTO, Mauro, Sensor de proximidad, México, 2002

<http://www.cce.gov.co/web/guest/wiki/-/wiki/cce>

1.3.2. Sensores Activos.

“Los sensores activos generan la señal de salida sin la necesidad de una fuente de alimentación externa, Un sensor activo emite pulsaciones de energía y examina la energía reflejada o rebotada (radar, por ejemplo).”³

Extremadamente efectivo a corto y medio alcance, pero menos efectivo a largo alcance. Como el sensor proyecta ráfagas controladas de energía, a otros sensores les resulta relativamente fácil localizar una nave que tiene sensores activos.

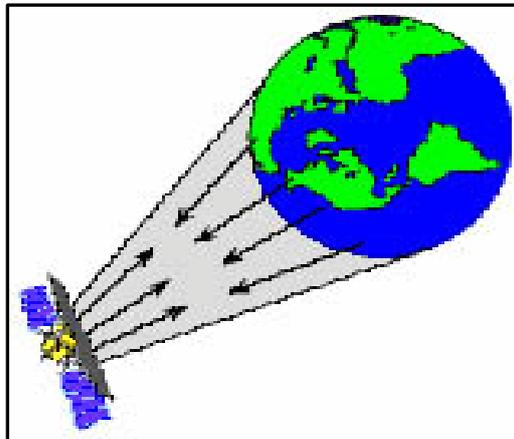


Gráfico2. Funcionamiento de un sensor activo.

Fuente: SOTO, Mauro, Sensor de proximidad, México, 2002

<http://www.cce.gov.co/web/guest/wiki/-/wiki/cce>

1.3.3. Sensores Análogos.

Un sensor analógico es aquel que, como salida, emite una señal comprendida por un campo de valores instantáneos que varían en el tiempo, y son proporcionales a los efectos que se están midiendo como se muestra en el Gráfico 3. Por ejemplo, un termómetro es un dispositivo analógico, la temperatura se mide en grados que pueden tener, en cualquier momento determinado, diferentes valores que son proporcionales a su indicador, o a su salida en caso de un dispositivo electrónico.

³ DIAZ, Henry, Resistance of a wire as a function of temperature, USA, Feb.1999, Pag. 56.

1.4. Sensores de Temperatura.

“A menudo la temperatura se define como aquella propiedad que miden los termómetros. En la práctica existen numerosos tipos de sensores de temperatura o termómetros que, según la aplicación específica, pueden ser los más adecuados. En la Tabla1 se indican algunos tipos de termómetros y sensores de temperatura usuales junto a algunas de sus características más notables.”⁴

Tipo de Sensor	Rango Nominal [°C]	Costo	Linealidad	Características Notables
Termómetro de mercurio	-10 a 300	Bajo	Buena	Simple, lento y de lectura manual
Termoresistencia RTD	-150 a 600	Medio	Alta	Exactitud
Termocupla	-150 a 1500	Bajo	Alta	Requiere referencia de Temperatura
Termistor	-15 a 115	Medio	No lineal	Muy Sensible
Integrado Lineal		Medio	Muy Alta	Fácil conexión a sistemas de toma de datos
Gas	-20 a 100	Medio	Buena	No muy versátil
Diodos	-200 a 50	Bajo	Alta	Bajo costo

Tabla 1. Sensores de temperatura y sus características.

Fuente: ALVAREZ, Juan, Instrumentación Electrónica, Argentina, 2008, Pag9.

En este proyecto utilizaremos un sensor de tipo integrado lineal LM35 ya que los sensores de circuitos integrados resuelven el problema de la linealidad y ofrecen altos niveles de rendimiento, además, relativamente económicos y bastante precisos a temperatura ambiente.

⁴ GIL, Salvador, Física Recreativa, Argentina, 2009.
<http://www.fisicarecreativa.com/guias/sensores-temp.pdf>

Sin embargo, los sensores de IC no tienen tantas opciones de configuraciones del producto o de gama de temperaturas, y además son dispositivos activos, por lo que requieren una fuente de alimentación. Los sensores de IC forman parte de la tendencia hacia los "sensores inteligentes", que son unos transductores cuya inteligencia incorporada facilita las actividades de reducción y análisis de datos que el usuario debe realizar normalmente en el sistema de adquisición de datos.

EL CIRCUITO INTEGRADO LM35

“El LM35 es un sensor de temperatura con una precisión calibrada de 1°C. Puede medir temperaturas en el rango que abarca desde -55° a + 150°C. La salida es muy lineal y cada grado centígrado equivale a 10 mV en la salida.”⁵

Sus características más relevantes son

- Precisión de ~1,5°C (peor caso), 0.5°C garantizados a 25°C.
- No linealidad de ~0,5°C (peor caso).
- Baja corriente de alimentación (60uA).
- Amplio rango de funcionamiento (desde -55° a + 150°C).
- Bajo costo.
- Baja impedancia de salida.

Su tensión de salida es proporcional a la temperatura, en la escala Celsius. No necesita calibración externa y es de bajo costo. Funciona en el rango de alimentación comprendido entre 4 y 30 voltios, como ventaja adicional”⁶.

El LM35 no requiere de circuitos adicionales para su calibración externa cuando se desea obtener una precisión del orden de ± 0.25 °C a temperatura ambiente, y ± 0.75 °C en un rango de temperatura desde 55 a 150 °C. La baja impedancia de

⁵PALAZZESI, Ariel, Amplificadores Operacionales, Argentina, 2009.

<http://www.ucontrol.com.ar/wiki/index.php?title=Portada>

⁶MARTIN, Daniel, Ionitron, España, 2008. <http://www.x-robotics.com/sensores.htm>

salida, su salida lineal y su precisa calibración inherente hace posible una fácil instalación en un circuito de control. Debido a su baja corriente de alimentación (60uA), se produce un efecto de auto calentamiento reducido, menos de 0.1 °C en situación de aire estacionario.

El sensor se encuentra disponible en diferentes encapsulados (Grafico4), pero el más común es el TO-92.

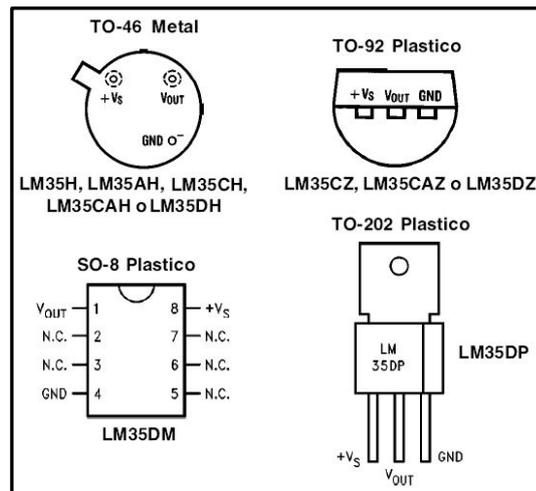


Gráfico5. Tipos de encapsulados del IC LM35.

Fuente: http://www.datasheetcatalog.com/datasheets_pdf/L/M/35

1.5. Detectores.

“Los detectores son dispositivos que indican la presencia de un fenómeno sin proporcionar necesariamente un valor de una magnitud asociada. Una indicación puede producirse solamente cuando el valor de la magnitud alcanza un umbral dado, algunas veces denominado umbral de detección del detector. En determinados campos, el término 'detector' se utiliza para el concepto sensor.”⁷

Las características más importantes de los detectores son:

- “Sensibilidad. Medida de la efectividad de un detector para convertir la muestra en una señal eléctrica medible.

⁷MC CLOY, Introducción a la Robótica, USA, 1982. <http://www.epsj23.net/docs/SENSORES.PDF>

- Linealidad. Rango de masa ó concentración de muestra sobre el cual el detector mantiene una sensibilidad constante sin una desviación.
- El límite de concentración inferior, que es dado por el límite de detección y,
- El límite Superior, definido por un porcentaje de desviación arbitrario de la curva de linealidad, normalmente se toma un 5% de desviación.
- Rango Dinámico Lineal. Rango sobre el cual la sensibilidad del detector es constante.
- Ruido. Es cuantificado por el promedio de la amplitud pico-pico de la señal. .
- Límite de Detección. Está definido como la mínima cantidad de sustancia que puede producir una señal que sea el doble del nivel de ruido.
- Corriente de Fondo. Señal constante de salida generada por el proceso en el que un detector está operativo sin que alguna sustancia pasa a través de él.”⁸

1.6. Detector de Nivel.

Los detectores de nivel en el mercado son muy costosos, de difícil acceso por su electrónica avanzada y de alta precisión.

Los detectores de nivel pueden ser de tipo fijo o de tipo configurables dependiendo si es flotador, desplazador, de presión hidrostática, desplazador, ultrasónico, sondeo se pesa, admitancia, conductancia o capacitancia, etc.

Fin de Carrera.

Son elementos actuadores de conmutación, generalmente provistos de muelles y utilizados en procesos automáticos donde la detección debe ser más robusta.

Las características más importantes de estos dispositivos son mostradas en la siguiente tabla:

⁸ ESCALONA, Iván, Transductores y Sensores en la Automatización Industrial, Mexico, 2007.
<http://www.monografias.com/trabajos48/detectores-cromatografia/detectores-cromatografia2.shtml>

Tensión de corte nominal	V AC/DC (Voltios)
Carga	Amperios
Tipo contacto	1,2... contactos conmutados
Tipo protección	Terminales de tornillo, sellados, etc.
Actuador	Gramos (150-600)
Tipo	Palanca, rodillo, varilla, etc.

Tabla 2. Características de los Fin carrera.

Fuente: ALVAREZ, Juan, Instrumentación Electrónica, Argentina, 2008, Pag13.

En cualquier proceso automático en el que debamos detectar posiciones o estados de este proceso y donde la presión sobre el actuador sea bastante grande. La comprobación se realiza de la misma forma que en los microruptores, es decir midiendo la continuidad en un tester en cada una de las posiciones del final de carrera.

1.7. Tipos de Salida.

Los sensores analógicos entregan como salida un voltaje o una corriente continuamente variable dentro del campo de medida especificado. Los rangos de voltaje de salida son muy variados, siendo los más usuales +10V, +5V, $\pm 10V$, $\pm 5V$ y $\pm 1V$.

La salida por loop de corriente es particularmente adecuada para ambientes industriales por las siguientes razones:

- Permite ubicar sensores en sitios remotos y peligrosos.
- Permite reducir a dos centímetros de alambres por sensor.
- Permite aislar eléctricamente los sensores de los instrumentos DC medición.
- Proporciona mayor confiabilidad puesto que es relativamente inmune a la captación de ruido y la señal no se degrada cuando se transmite sobre largas distancias.

Los sensores digitales entregan como salida un voltaje o una corriente variable en forma DC salidas o pasos discretos de manera codificada, es decir con su valor representado en algún formato de pulsos o palabras, digamos PWM o binario.

1.8. APLICACIÓN AL PROYECTO MONOGRAFICO.

En este proyecto monográfico se propone el control de un llenado de tanque con nivel máximo y mínimo de liquido, así como la estabilización de la temperatura interna del invernadero a 33°C para lo cual utilizaremos los detectores de nivel y sensor de temperatura con la siguiente descripción:

DETECTOR DE NIVEL: Como ya describimos anteriormente los detectores de nivel en el mercado son muy costosos, por eso, por eso en este proyecto utilizaremos un fin carrera adaptado a un flotante de tipo pesca (Grafico5), para simular un detector de nivel flotante de tipo ON-OFF.



Gráfico6. Detector tipo flotante.

Fuente: Autores, Foto en laboratorio de Electrónica U.D.A.

SENSOR DE TEMPERATURA: Los invernaderos por lo general tienden a tener una temperatura ambiente de promedio de 33°C y por las características antes mencionadas del sensor integrado LM35 se asegura al decir que tendríamos una salida de voltaje de 330mV.

CAPÍTULO 2

PROGRAMA STEP7 PARA EL PLC SIEMENES S7-200.

2.1. Introducción.

Con la llegada de los autómatas programables, los llamados PLC, la industria sufrió un impulso importante, que ha facilitado de forma notable que los procesos de producción o control se hayan flexibilizado mucho. Encontramos PLC en la industria, pero también en nuestras casas, en los centros comerciales, hospitalarios, etc. También en nuestras escuelas de formación profesional encontramos frecuentemente autómatas programables. PLC son las siglas en inglés de Controlador Lógico Programable (Programmable Logic Controller).

“El PLC es un dispositivo de estado sólido, diseñado para controlar procesos secuenciales (una etapa después de la otra) que se ejecutan en un ambiente industrial. Es decir, que van asociados a la maquinaria que desarrolla procesos de producción y controlan su trabajo. El PLC es un sistema, porque contiene todo lo necesario para operar, y es industrial, por tener todos los registros necesarios para operar en los ambientes hostiles que se encuentran en la industria.”⁹

Un PLC realiza, entre otras, las siguientes funciones:

- Recoger datos de las fuentes de entrada a través de las fuentes digitales y analógicas.
- Tomar decisiones en base a criterios pre programado.

⁹HORTA, José, Técnicas de Automatización Industrial, México, 1992.
<http://observatorio.cnice.mec.es/modules.php?op=modload&name=News&file=article&sid=504>

- Almacenar datos en la memoria (en caso de tenerla).
- Generar ciclos de tiempo.
- Realizar cálculos matemáticos.
- Actuar sobre los dispositivos externos mediante las salidas analógicas y digitales.
- Comunicarse con otros sistemas externos.

Los PLC se distinguen de otros controladores automáticos, en que pueden ser programados para controlar cualquier tipo de máquina, a diferencia de otros controladores que, solamente, pueden controlar un tipo específico de aparato. Además de poder ser programados, son automáticos, es decir son aparatos que comparan las señales emitidas por la máquina controlada y toman decisiones en base a las instrucciones programadas, para mantener estable la operación de dicha máquina, además se puede modificar las instrucciones almacenadas en memoria, además de monitorizarlas.

2.2. Procedimiento Básico con Step 7-Micro/WIN.

“El paquete de programación STEP 7--Micro/WIN constituye un entorno de fácil manejo para desarrollar, editar y observar el programa necesario con objeto de controlar la aplicación. STEP 7--Micro/WIN provee tres editores que permiten desarrollar de forma cómoda y eficiente el programa de control. Para encontrar fácilmente la información necesaria, STEP 7--Micro/WIN ofrece una completa Ayuda en pantalla y un CD de documentación que incluye una versión electrónica del presente manual, ejemplos de aplicación y otras informaciones de gran utilidad.”¹⁰

STEP 7--Micro/WIN se puede ejecutar en un ordenador (PC), o bien en una programadora de Siemens (p. ej. en una PG 760). El PC o la PG deben cumplir los siguientes requisitos mínimos:

¹⁰ McMILLAN, Gregory K. Tuning and control loop performance: a practitioners's guide. USA, 1994, Pag. 227.

- Sistema operativo: Windows 2000, Windows XP, Vista.
- 350 Mbytes libres en el disco duro (como mínimo).
- Ratón (recomendado)

2.2.1. Instalación del Programa STEP7.

Inserte el CD de STEP 7--Micro/WIN en la unidad de CD--ROM. El asistente de instalación arrancará automáticamente y le conducirá por el proceso de instalación.

Para instalar STEP 7-Micro/WIN 32, siga los siguientes pasos:

- Inserte el CD o el disquete en la correspondiente unidad del PC.
- Haga clic en el botón Inicio para abrir el menú de Windows.
- Haga clic en Ejecutar....
 - Si la instalación se efectúa desde un Disquete: En el cuadro de diálogo Ejecutar, teclee a:\setup y haga clic en el botón Aceptar o pulse la tecla INTRO. Así se inicia la instalación. CD: En el cuadro de diálogo Ejecutar, teclee e:\setup y haga clic en Aceptar o pulse la tecla INTRO. Así se inicia la instalación.
 - Siga las instrucciones que van apareciendo en pantalla hasta finalizar la instalación.
- Al final de la instalación aparece automáticamente el cuadro de diálogo Ajustar interface PG/PC. Haga clic en Cancelar para acceder a la ventana principal de STEP 7-Micro/WIN 32.

2.2.2. Arrancar el Simatic.

La manera más rápida de iniciar STEP7 es haciendo doble clic en el icono "Administrador SIMATIC". A continuación se abre la ventana del Administrador SIMATIC. Desde allí es posible acceder a todas las funciones instaladas, tanto del

paquete estándar como de los paquetes opcionales. El Administrador SIMATIC es el interface para acceder a la configuración y a la programación que permite:

- Crear proyectos.
- Configurar y parametrizar el hardware.
- Configurar redes de hardware.
- Programar bloques.
- Probar y hacer funcionar los programas.

El acceso a las funciones se ha orientado a los objetos, siendo fácil de aprender. La ventana del administrador Simatic es la siguiente:



Gráfico9. Administrador Simatic.

Fuente: SIEMENS, Manual de Funcionamiento de S7-200, España,1993.
manual_ S7200N_s.pdf. Pag 85.

2.2.3. Crear el proyecto.

Para crear un nuevo proyecto en STEP7-Micro/WIN haga doble click en el icono de STEP7-Micro/WIN o elija los comandos Inicio>SIMATIC>STEP7 Micro/WIN 32V4.0. STEP7-Micro/WIN ofrece una interfaz de usuario cómoda para crear el programa de

control. Las barras de herramientas incorporan botones de método abreviado para los comandos de menú de uso frecuente. Estas barras se pueden mostrar u ocultar.

La barra de navegación comprende iconos que permite acceder a las diversas funciones de programación de STEP7-Micro/WIN. En el árbol de operaciones se visualizan todos los objetos del proyecto y las operaciones para crear el programa de control. Para insertar operaciones en el programa, se puede utilizar el método de arrastrar y soltar desde el árbol de operaciones, o bien hacer doble clic en una operación con objeto de insertarla en la posición actual del cursor en el editor de programas.

El editor de programas contiene el programa y una tabla de variables locales donde se puede asignar nombres simbólicos a las variables locales temporales. Las subrutinas y las rutinas de interrupción se visualizarán en forma de fichas en el borde inferior del editor del programa. Para acceder a las subrutinas, a las rutinas de interrupción o al programa principal, haga clic en la ficha en cuestión.

STEP7-Micro/WIN incorpora los tres editores de programa siguientes: Esquemas de Contactos (KOP), Lista de instrucciones (AWL) y Diagrama de funciones (FUP). En este proyecto programaremos el S7-200 mediante el STEP7-Micro/WIN con Esquemas de Contactos (KOP).

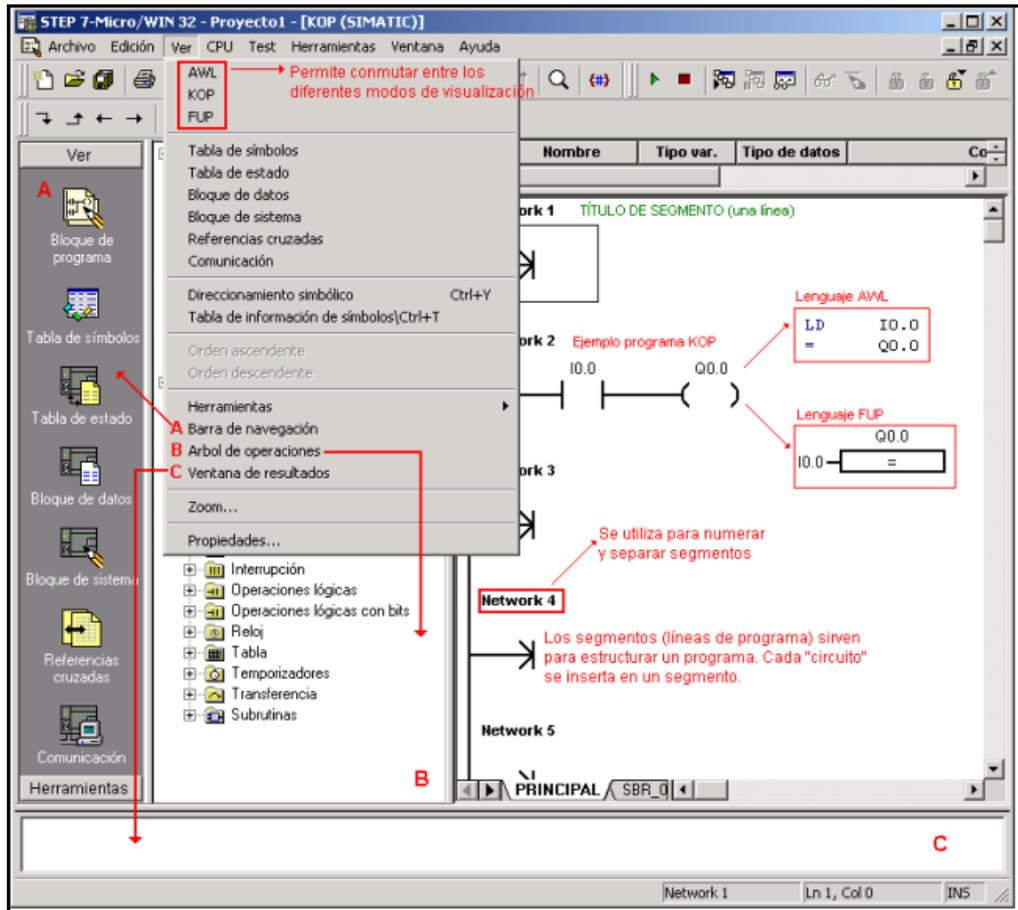


Gráfico10. Pantalla Principal de STEP7-Micro/WIN.

Fuente: SIEMENS, Manual de Funcionamiento de S7-200, España,1993.
manual_ S7200N_s.pdf. Pag 93.

2.2.4. Programación en KOP.

“KOP es la abreviatura alemana de Kontaktplan que significa Esquema de Contactos. KOP es un lenguaje de programación gráfico. La sintaxis de las instrucciones se parece a un esquema de circuitos: Con KOP puede observar cómodamente el flujo de corriente entre conductores a través de condiciones lógicas de entrada que a su vez habilitan condiciones lógicas de salidas.”¹¹

¹¹ SIEMENS – SIMATIC. **Sistema de automatización S7-200.** Manual del Sistema. Alemania, 1.997.

El lenguaje de programación KOP facilita todos los elementos que se necesitan para crear un programa de usuario completo. KOP contiene un amplio juego de instrucciones, ofreciéndole operaciones básicas, así como una gama completa de operandos y su respectivo direccionamiento. Lo mismo se puede decir de la concepción de las funciones y de los bloques de función, que le servirán para estructurar sus programas KOP de modo claro y fácil de comprender.

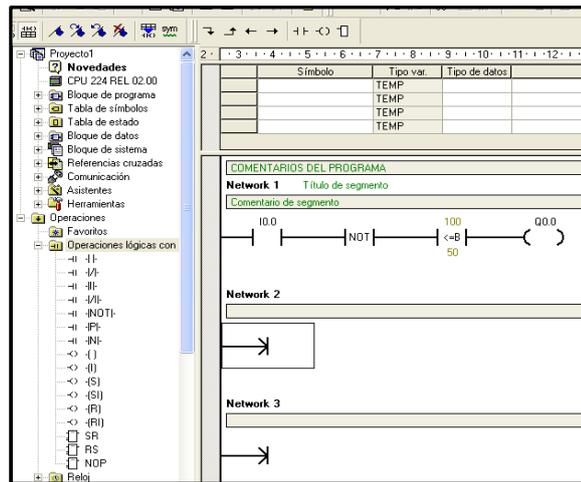


Gráfico11. Programación en KOP.

Fuente: Autores, Programación en el software STEP7.

2.3. Configuración del sistema de Automatización S7-200.

Ahora explicaremos cómo configurar la comunicación entre la CPU S7-200 y el PC utilizando el cable PC/PPI. Esta es una configuración con un solo maestro y sin ningún otro equipo de hardware instalado (como p.ej. un módem o una unidad de programación).

2.3.1. Resumen breve.

El Gráfico11 muestra una configuración física para conectar el PC a la CPU mediante el cable PC/PPI. Para establecer un enlace correcto entre los componentes:

- I. Ajuste los interruptores DIP del cable PC/PPI para determinar la velocidad de transferencia deseada.
- II. Conecte el extremo RS-232 (PC) del cable PC/PPI al puerto de comunicación de su PC (COM1 o COM2) y apriete los tornillos de conexión.
- III. Conecte el otro extremo (RS-485) del cable PC/PPI al interface de comunicación de la CPU y apriete los tornillos de conexión.

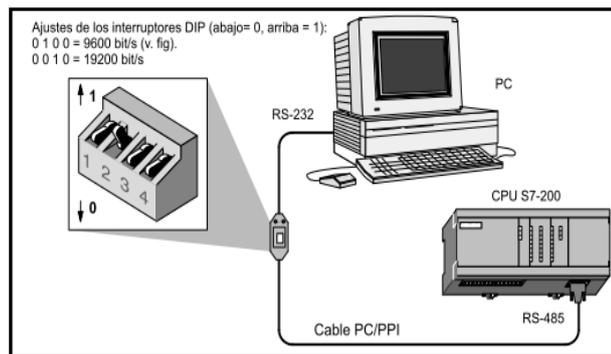


Gráfico12. Comunicación entre CPU y PC por cable PPI.

Fuente: SIEMENS, Manual de Funcionamiento de S7-200, España,1993.
 manual_ S7200N_s.pdf. Pag 109.

- IV. Seguidamente configuraremos el tipo de CPU de la serie S7-200 que para nuestra tesis monográfica será CPU-224.

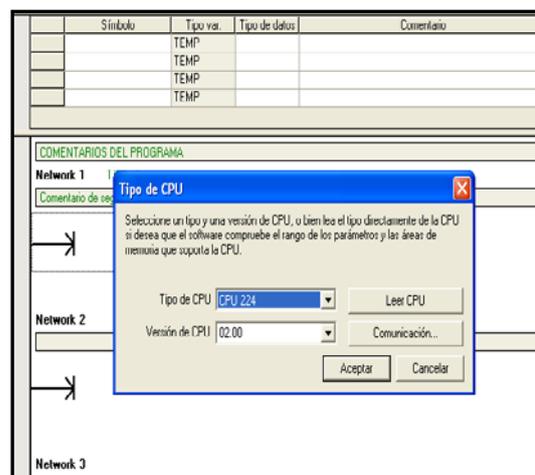


Gráfico13. Configuración de CPU de la serie S7-200.

Fuente: Autores, Programación en el software STEP7.

2.3.2. Como configurar la comunicación utilizando el cable PC/PPI.

Ahora configuraremos los parámetros estándar de su interface, siga los siguientes pasos:

- i. En la ventana de STEP 7-Micro/WIN 32, haga clic en el icono Comunicación o elija el comando de menú Ver > Comunicación. Aparecerá el cuadro de diálogo Configurar comunicación.

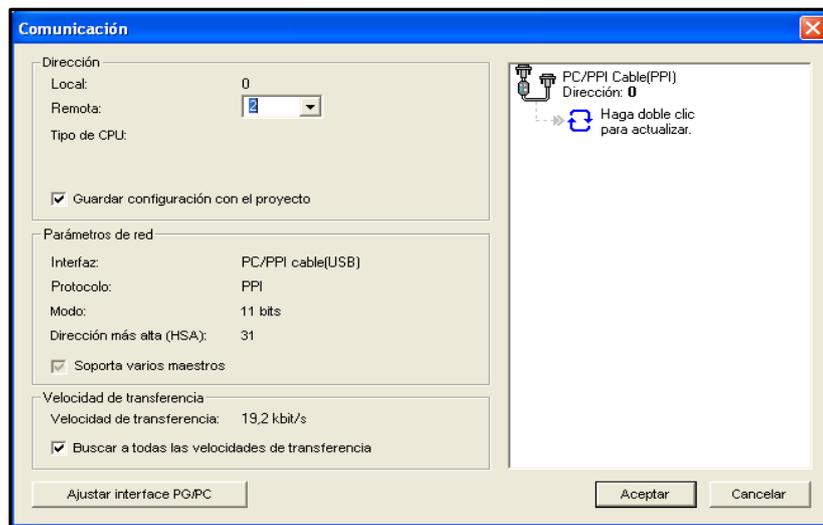


Gráfico14. Configuración de la comunicación entre PC y la CPU 224.

Fuente: Autores, Programación en el software STEP7.

- ii. En el cuadro de diálogo Configurar comunicación, haga doble clic en el icono del cable PC/PPI. Aparecerá el cuadro de diálogo Ajustar interface PG/PC.

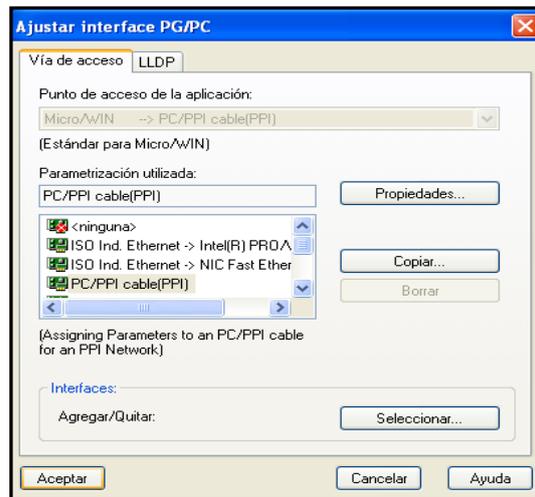


Gráfico15. Ajuste de Interface entre PC y CPU.

Fuente: Autores, Programación en el software STEP7.

- iii. Haga clic en el botón Propiedades para acceder al cuadro de diálogo donde se visualizan las propiedades del interface. Verifique las propiedades. La velocidad de transferencia debe estar ajustada a 9.600 bit/s.

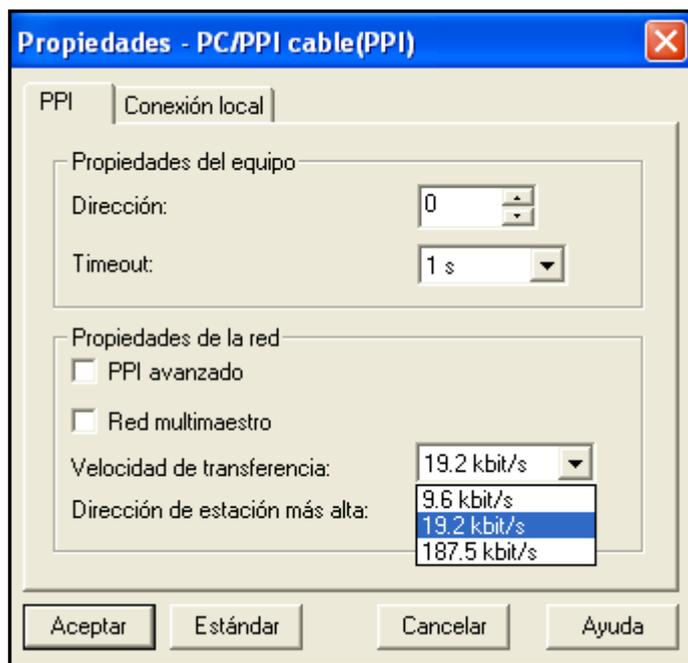


Gráfico16. Configuración de la velocidad de comunicación.

Fuente: Autores, Programación en el software STEP7.

- iv. Haga clic en el botón Conexión local para configurar el puerto por donde se comunicara el automata programable con el PC, que en nuestro caso será por puerto USB.

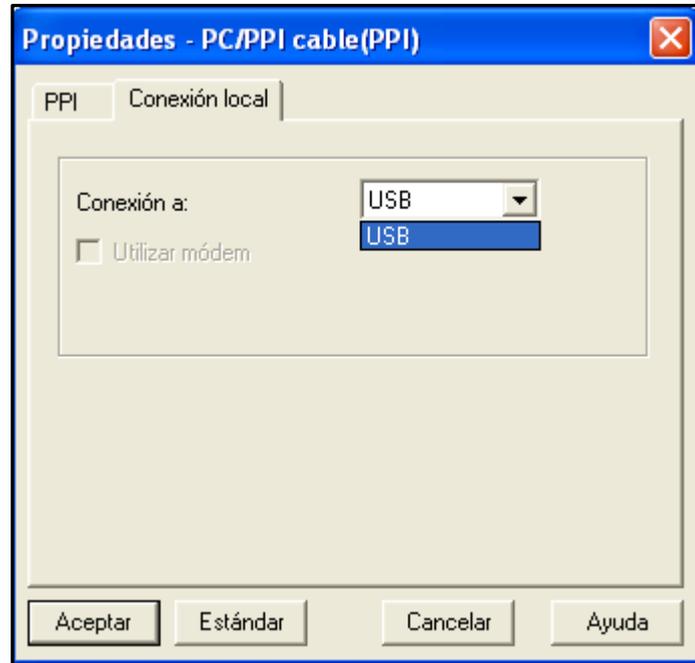


Gráfico17. Configuración del puerto de comunicación.

Fuente: Autores, Programación en el software STEP7.

2.3.3. Como establecer la comunicación con la CPU S7-200.

El siguiente paso será establecer la comunicación con la CPU S7-200 con los siguientes pasos:

- i. En la pantalla de STEP 7-Micro/WIN 32, haga clic en el icono Comunicación o elija el comando de menú Ver > Comunicación. Aparecerá el cuadro de diálogo Configurar comunicación donde se indica que no hay ninguna CPU conectada.

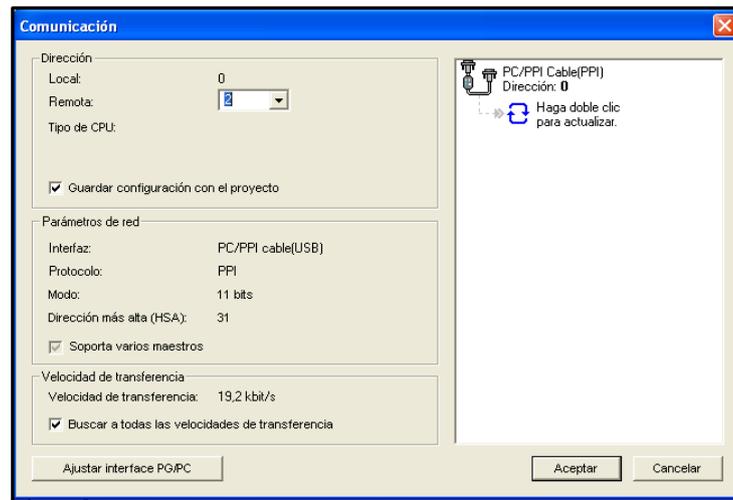


Gráfico18. Ventana de comunicación.

Fuente: Autores, Programación en el software STEP7.

- ii. En el cuadro de diálogo Configurar comunicación, haga doble clic en el icono Actualizar. STEP 7-Micro/WIN 32 comprueba si hay CPUs S7-200 (estaciones) conectadas.
- iii. Por cada estación conectada aparecerá un icono de CPU en el cuadro de diálogo Configurar comunicación
- iv. Haga doble clic en la estación con la que desea establecer la comunicación. Como podrá apreciar, los parámetros de comunicación visualizados en el cuadro de diálogo corresponden a la estación seleccionada
- v. Así queda establecido el enlace con la CPU S7-200.

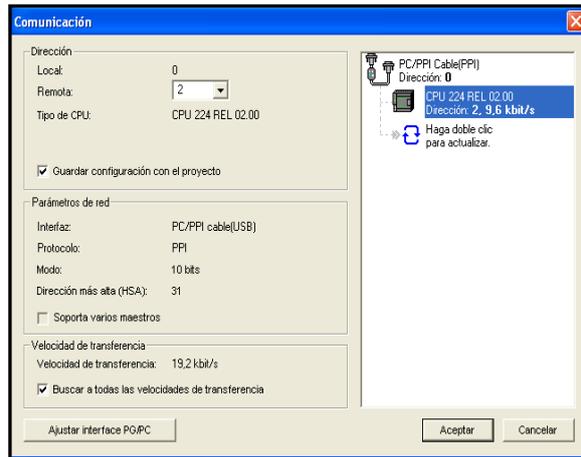


Gráfico19. Configuración de comunicación.

Fuente: Autores, Programación en el software STEP7.

2.3.4. Como cambiar los parámetros de comunicación de la CPU.

Tras haber establecido un enlace con la CPU S7-200 puede verificar o cambiar los parámetros de comunicación de la CPU mediante los siguientes pasos:

- i. En la barra de navegación, haga clic en el icono Bloque de sistema o elija el comando de menú Ver > Bloque de sistema.
- ii. Aparecerá el cuadro de diálogo Bloque de sistema. Haga clic en la ficha Puerto(s). El ajuste estándar de la dirección de estación es 2 y el de la velocidad de transferencia es de 9,6 kbit/s.

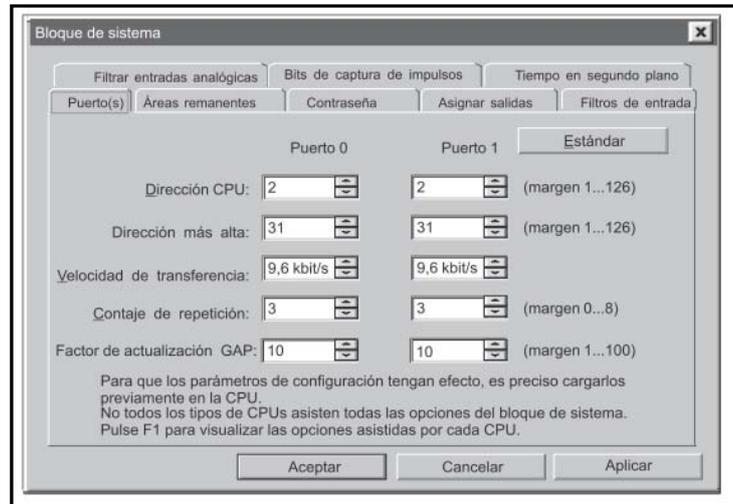


Gráfico20. Configuración de parámetros de comunicación.

Fuente: Autores, Programación en el software STEP7.

- iii. Haga clic en Aceptar para conservar dichos parámetros. Si desea modificar la parametrización, efectúe los cambios deseados, haga clic en el botón Aplicar y, por último, en el botón Aceptar.
- iv. En la barra de herramientas, haga clic en el botón Cargar en CPU para cargar los cambios en la CPU.
- v. Así se adopta la parametrización deseada para la comunicación.

2.4. APLICACIÓN AL PROYECTO MONOGRAFICO.

En este proyecto utilizaremos los PLC de la marca Siemens de la gama S7-200 la CPU 224 el cual programaremos mediante un software STEP 7-Micro/WIN.

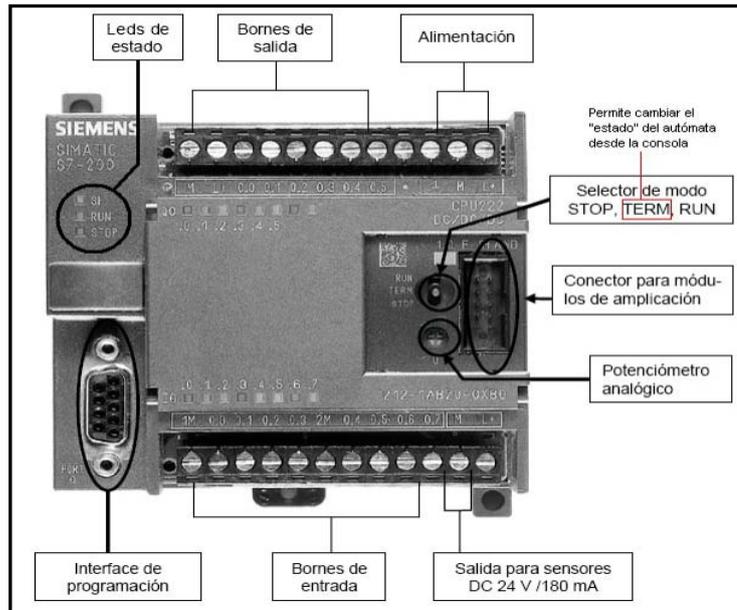


Gráfico21. Partes externas de un PLC de la gama S7-200.

Fuente: SIEMENS, Manual de Funcionamiento de S7-200, España,1993.
manual_ S7200N_s.pdf. Pag 120.

Una vez realizado la instalación de STEP 7-Micro/WIN 32, así como la creación de un nuevo proyecto con todas las configuraciones detalladas anteriormente en este capítulo realizaremos el programa para el control del llenado de nivel de tanque ya que este será un programa digital, y el control de temperatura interna del invernadero se desarrollara en el capítulo 4.

Este tanque tendrá 2 detectores de tipo flotante descritos en el Capitulo 1 los mismos que serán conectados a las entradas I1 e I2 en donde I1 será el detector de nivel mínimo e I2 será el detector de nivel máximo.

Cuando el nivel sea menor que sensor I1 (mínimo) se activara automáticamente un actuador que prendera una bomba de agua que será el encargado de llenar el tanque de agua hasta que llegue a accionar I2 (máximo), cuando esto sucede se apaga la bomba hasta que manualmente se haga la descarga del nivel por medio de una llave simulando el riego de invernadero.

El vaciado del tanque será total ya que cuando I2 se desactive el PLC no realizara ninguna acción solo hasta que el nivel sea menor a I1.

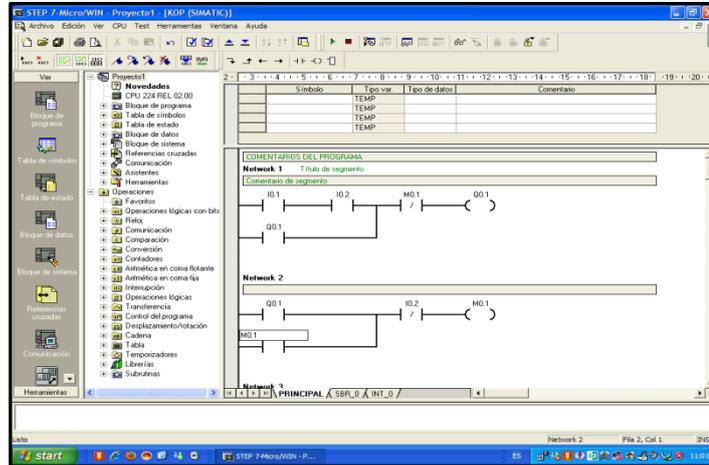


Gráfico22. Programa de llenado de nivel de tanque.

Fuente: Autores, Programación en el software STEP7.

El S7-200 soporta la comunicación como servidor y cliente, es decir, es posible

- configurar en el S7-200 un enlace cliente, es decir, el S7-200 crea el enlace S7 de forma activa.
- configurar en el S7-200 un enlace servidor, es decir, el S7-200 participa de forma pasiva en la creación del enlace S7. El cliente remoto crea el enlace S7 con el S7-200. Éste dispone de los datos como servidor.

La configuración de la comunicación S7 para el S7-200 se realiza con el asistente de Ethernet del STEP 7-MicroWIN.

Para realizar la configuración de la comunicación S7-200 como servidor, se procederá según los siguientes pasos:

PASO 1: Estando en microwin abrir HERRAMIENTAS > ASISTENTE DE Ethernet

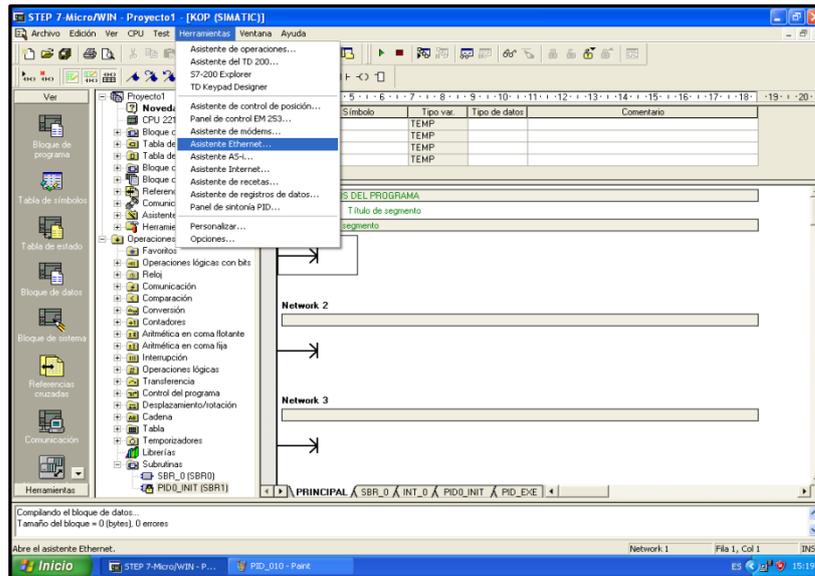


Gráfico23. Programación SERVIDOR de S7=200.

Fuente: Autores, Programación en el software STEP7.

PASO 2: El primer cuadro nos permite aceptar la configuración dando click en Siguiente.

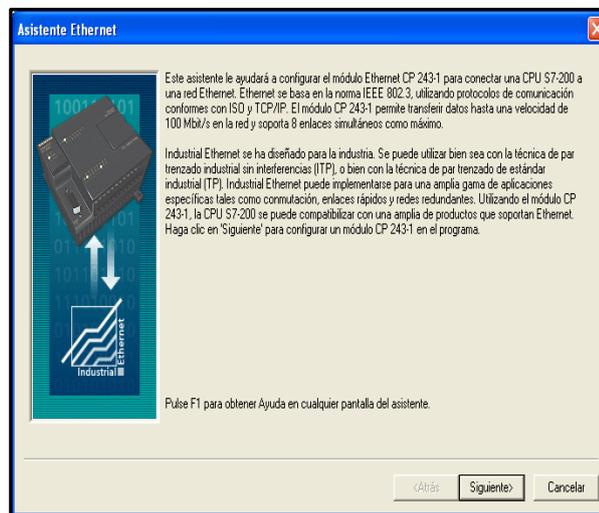


Gráfico24. Configuración de Asistente de Ethernet.

Fuente: Autores, Programación en el software STEP7.

PASO 3: Ahora tenemos que configurar la posición del módulo ya que se puede hacer en forma automática (LEER MODULOS) o de forma manual para luego dar click en el icono SIGUIENTE.

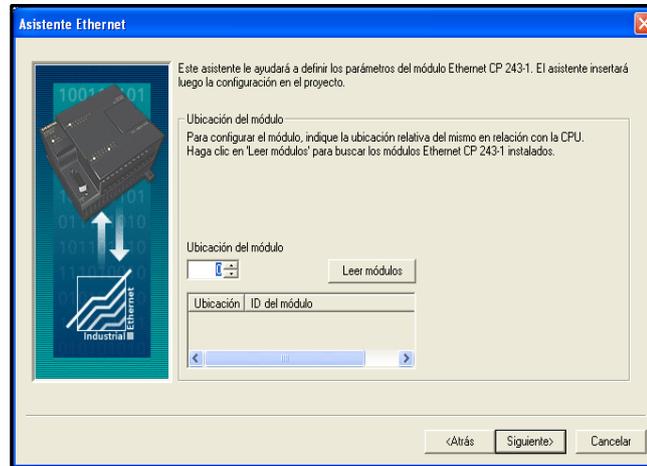


Gráfico25. Configuración Posición de Módulo.

Fuente: Autores, Programación en el software STEP7.

PASO 4: Ingresar una sola dirección IP y la correspondiente Mascara de Subred, además pulsar en siguiente para aplicar los ajustes

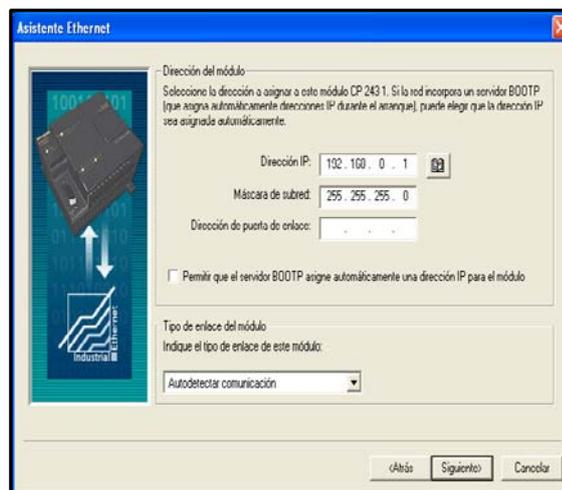


Gráfico26. Configuración de Dirección IP y Mascara de Subred.

Fuente: Autores, Programación en el software STEP7.

PASO 5: En el siguiente cuadro de diálogo usted debe poner el número de conexiones S7 que es configurado. Más de una conexión del S7-200 usted puede leer datos de la comunicación y usted puede escribir datos al compañero de comunicación. Luego debemos dar click en el botón "Siguiente" para seguir con la configuración de la conexión S7-200.

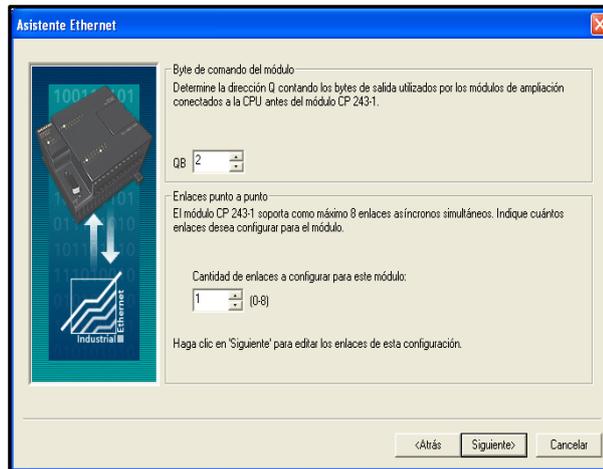


Gráfico27. Configuración del número de conexiones.

Fuente: Autores, Programación en el software STEP7.

PASO 6: La conexión es configurada por el primer s7-200 como la conexión de Servidor y el segundo s7-200 es el cliente para esta conexión de servidor.

TSAP. Usted entra 10.00 para TSAP remoto, para luego usted activar la función "Aceptar todas las conexión peticiones de enlaces" y se pulsa el botón "Aceptar".

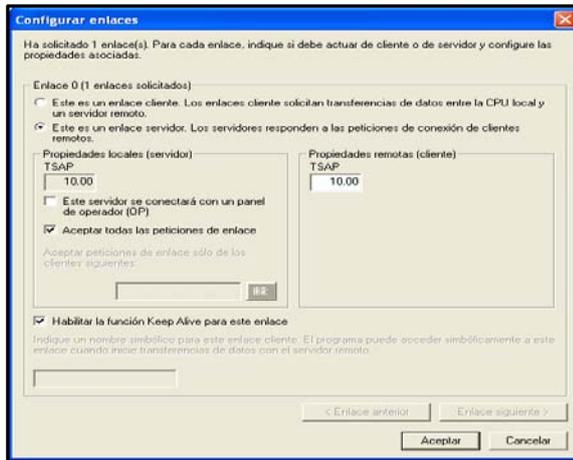


Gráfico28. Configuración de enlace de s7-200.

Fuente: Autores, Programación en el software STEP7.

PASO7: Sabemos que la configuración del módulo en lo posterior puede ser cambiada, entonces usted selecciona el ajuste con la protección CRC en donde usted puede especificar el tiempo de comunicación en Keep Alive con un intervalo con el tiempo de falta de respuesta. Dar click sobre el botón "Siguiente".

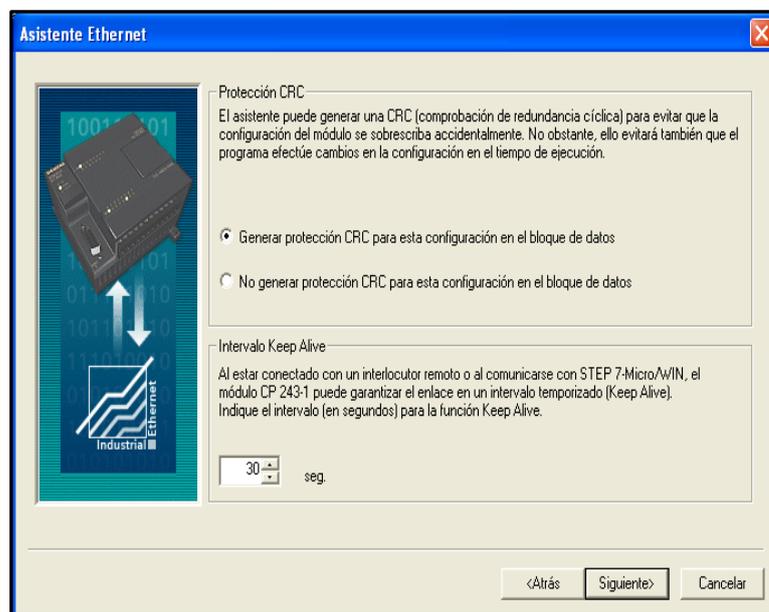


Gráfico29. Configuración de protección CRC.

Fuente: Autores, Programación en el software STEP7.

PASO 8: Seleccionar una sección de área libre para establecer la comunicación, o a su vez dar click en “Proponer Dirección” para que el asistente ponga la dirección.

NOTA: se debe tener en cuenta que la dirección que se proponga en esta configuración no debe ser utilizada dentro del programa general ya que puede ser sobre escrita en el mismo sin dar ningún aviso de ERROR.



Gráfico30. Dirección de comunicación para servidor.

Fuente: Autores, Programación en el software STEP7.

PASO9: Pulsamos finalizar y tendremos ya configurado el SERVIDOR.

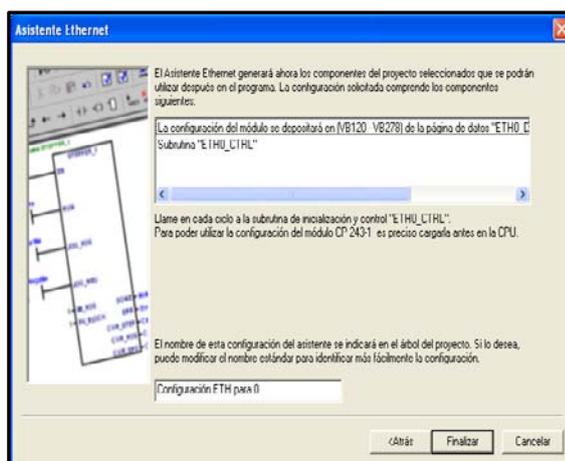


Gráfico31. Finalización de configuración del SERVIDOR.

Fuente: Autores, Programación en el software STEP7.

Ahora tendremos en el programa principal una subrutina de control para servidor en nuestro caso llamada ETH0_CTRL que será utilizada en la comunicación con el S7-200 cliente.

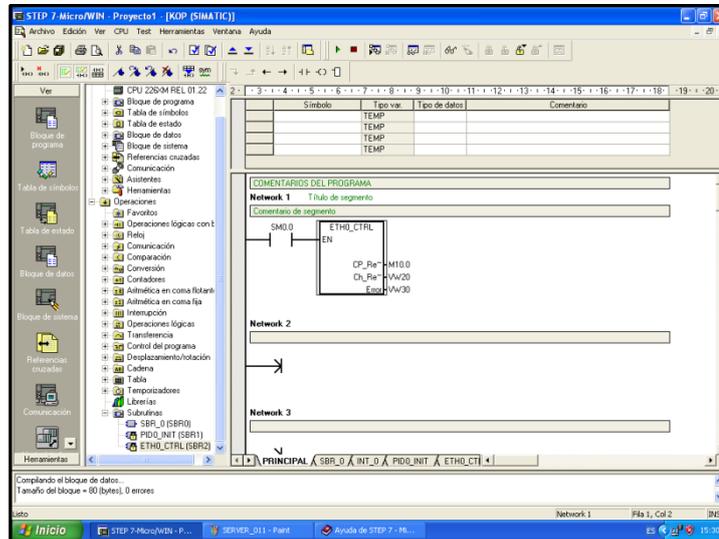


Gráfico32. Bloque de Servidor en programa principal.

Fuente: Autores, Programación en el software STEP7.

Una vez configurado la parte de SERVIDOR realizaremos el complemento del mismo que será el CLIENTE que deberá ser configurado de la siguiente manera:

Desde el paso 1 hasta el paso 5 será la misma configuración realizada en la configuración del servidor con la única excepción que en el paso 4 la dirección IP será diferente para nuestro caso será 192.168.13.2 y desde el paso 6 se explica a continuación.

PASO 6: Para el parámetro remoto usted entra en la dirección de del servidor (192.168.13.1). La conexión S7 es bien definida con el TSAPs. La conexión de cliente es unilateral configurada en el S7-200. Usted debe dar click en el botón "Transferencia de Datos".



Gráfico33. Configuración de Enlaces en cliente.

Fuente: Autores, Programación en el software STEP7.

PASO 7: Usted tiene que escoger el icono nueva transferencia en donde deberá configurar Leer o escribir los datos del servidor, cuantos bytes se deben leerse o escribir en el servidor (nuestro proyecto será 1) además de las direcciones donde se van a guardar los datos tanto en el cliente como en el servidor y un nombre para la transferencia cuando lee y otro cuando escribe.

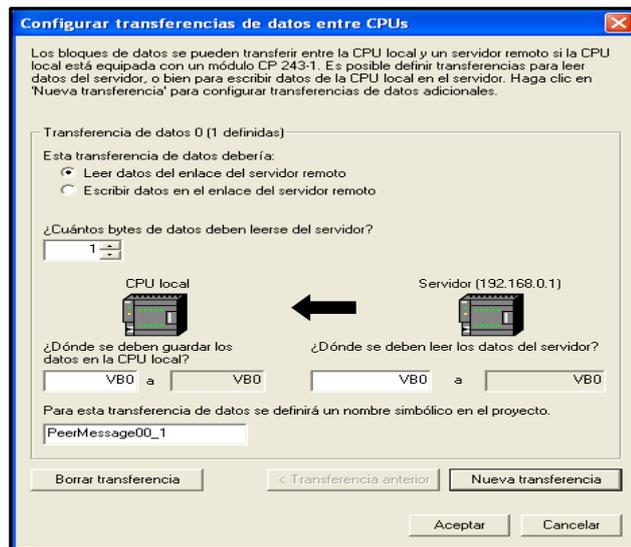


Gráfico34. Lectura desde el servidor al cliente.

Fuente: Autores, Programación en el software STEP7.

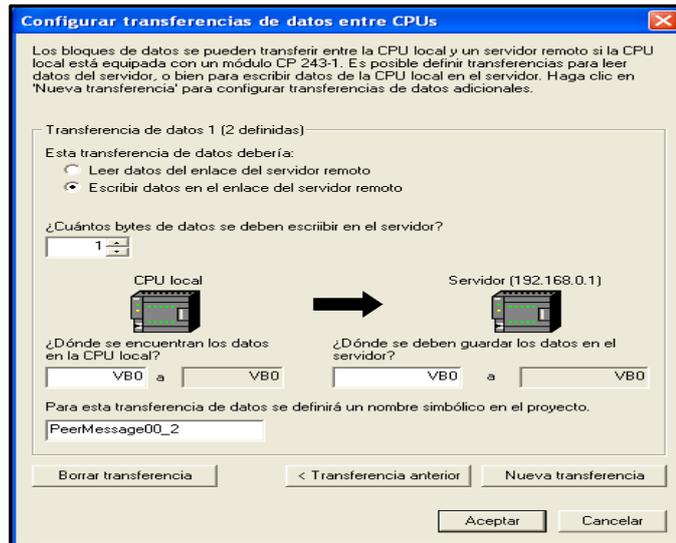


Gráfico35. Escritura desde el cliente al servidor.

Fuente: Autores, Programación en el software STEP7.

PASO7: Como lo explicamos anteriormente la configuración del módulo en lo posterior puede ser cambiada, entonces usted selecciona el ajuste con la protección CRC y Damos click sobre el botón "Siguiente".

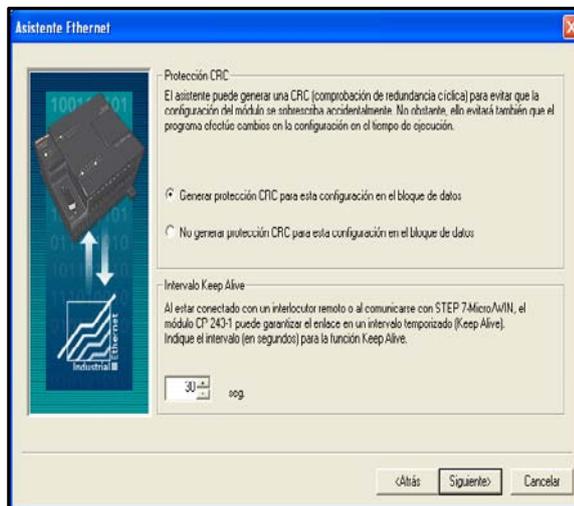


Gráfico36. Configuración de protección CRC.

Fuente: Autores, Programación en el software STEP7.

PASO 8: Seleccionar una sección de área libre para establecer la comunicación diferente a la seleccionada en el servidor, o a su vez dar click en “Proponer Dirección” para que el asistente ponga la dirección.

NOTA: se debe tener en cuenta que la dirección que se proponga en esta configuración no debe ser utilizada dentro del programa general ya que puede ser sobre escrita en el mismo sin dar ningún aviso de ERROR.

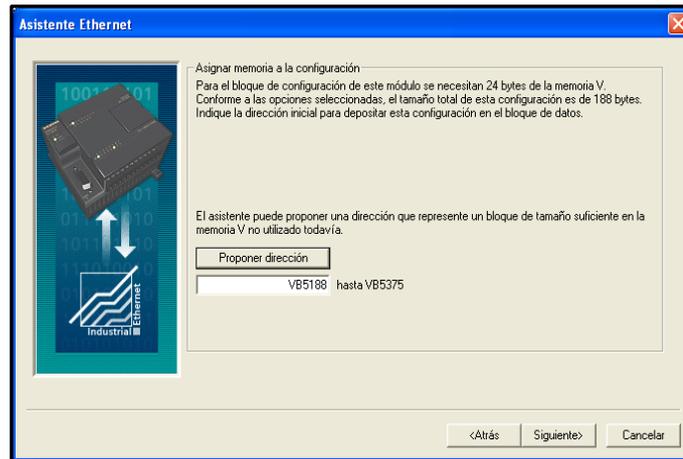


Gráfico37. Dirección de comunicación para cliente.

Fuente: Autores, Programación en el software STEP7.

PASO9: Pulsamos finalizar y tendremos ya configurado el CLIENTE.

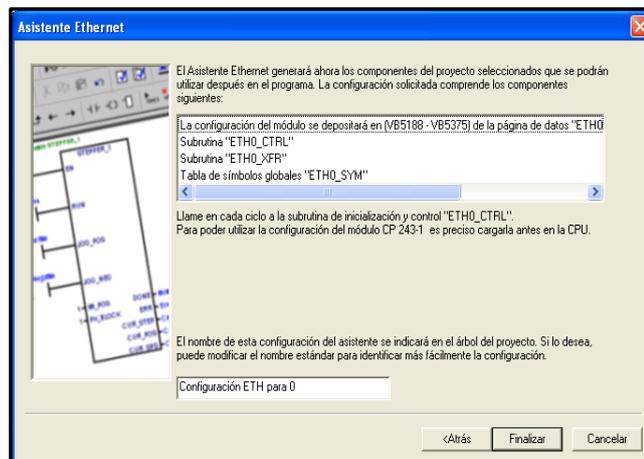


Gráfico38. Finalización de configuración del CLIENTE. **Fuente:** Autores, Programación en el software STEP7.

Ahora tendremos en el programa principal una subrutina de control para el cliente en nuestro caso llamada ETH0_XFR que será utilizada en la comunicación con el s7-200 Servidor.

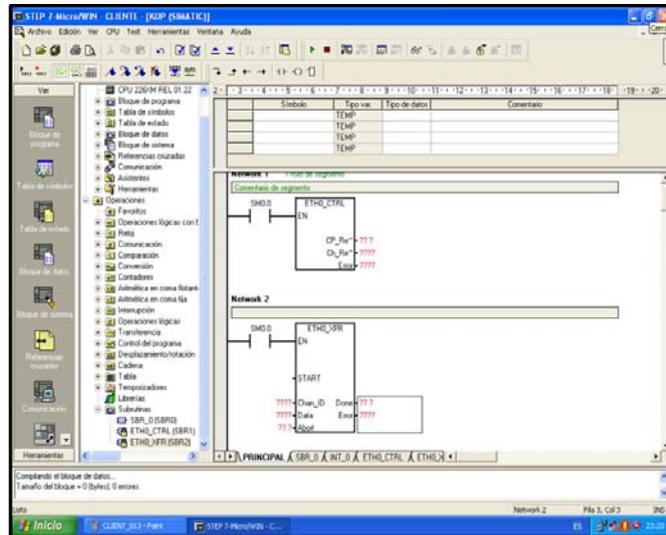


Gráfico39. Bloque de Cliente en programa principal.

Fuente: Autores, Programación en el software STEP7.

El programa de llenado de tanque con la configuración de un cliente esta adjunto en el **ANEXO1**.

CAPÍTULO 3

SISTEMA SCADA.

3.1. Introducción al Sistema S.C.A.D.A.

“SCADA viene de las siglas de *"Supervisory Control And Data Adquisición"*, es decir: adquisición de datos y control de supervisión. Se trata de una aplicación software especialmente diseñada para funcionar sobre ordenadores en el control de producción, proporcionando comunicación con los dispositivos de campo (controladores autónomos, autómatas programables, etc.) y controlando el proceso de forma automática desde la pantalla del ordenador. Además, provee de toda la información que se genera en el proceso productivo a diversos usuarios, tanto del mismo nivel como de otros supervisores dentro de la empresa: control de calidad, supervisión, mantenimiento, etc.”¹²

En este tipo de sistemas usualmente existe un ordenador, que efectúa tareas de supervisión y gestión de alarmas, así como tratamiento de datos y control de procesos. La comunicación se realiza mediante buses especiales o redes LAN. Todo esto se ejecuta normalmente en tiempo real, y están diseñados para dar al operador de planta la posibilidad de supervisar y controlar dichos procesos.

Los programas necesarios, y en su caso el hardware adicional que se necesite, se denomina en general sistema SCADA.

¹²LANZILLOTTA, Analia, Definición de SCADA, España, 2005
<http://www.mastermagazine.info/termino/6614.php>

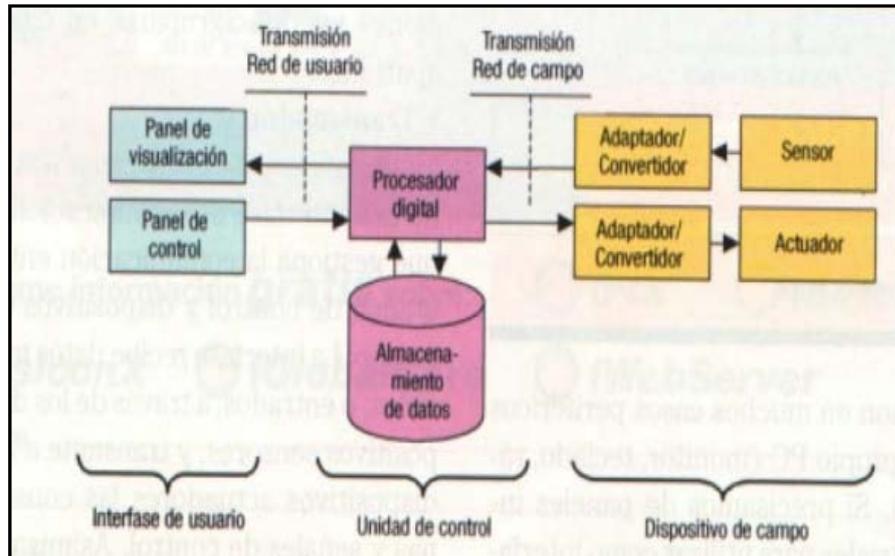


Gráfico40. Esquema básico de un SCADA.

Fuente: GONZALES, Víctor, Aplicaciones para la supervisión y el control de producción, España, 2003

www.esi2.us.es/clases/1ioi_api/documentos/Tema_11/Tema_11.pdf.

3.1.1. Prestaciones.

“Un paquete SCADA debe estar en disposición de ofrecer las siguientes prestaciones:

- Posibilidad de crear paneles de alarma, que exigen la presencia del operador para reconocer una parada o situación de alarma, con registro de incidencias.
- Generación de históricos de señal de planta, que pueden ser volcados para su proceso sobre una hoja de cálculo.
- Ejecución de programas, que modifican la ley de control, o incluso anular o modificar las tareas asociadas al autómatas, bajo ciertas condiciones.
- Posibilidad de programación numérica, que permite realizar cálculos aritméticos de elevada resolución sobre la CPU del ordenador.”¹³

¹³ HENTEN, *Greenhouse climate management an optimal control approach*. The Netherlands 1994, Pag 306.

Con ellas, se pueden desarrollar aplicaciones para ordenadores (tipo PC, por ejemplo), con captura de datos, análisis de señales, presentaciones en pantalla, envío de resultados a disco e impresora, etc.

Además, todas estas acciones se llevan a cabo mediante un paquete de funciones que incluye zonas de programación en un lenguaje de uso general (como C, Pascal, o Basic), lo cual confiere una potencia muy elevada y una gran versatilidad. Algunos SCADA ofrecen librerías de funciones para lenguajes de uso general que permiten personalizar de

3.1.2. Requisitos.

“Un SCADA debe cumplir varios objetivos para que su instalación sea perfectamente aprovechada:

- Deben ser sistemas de arquitectura abierta, capaces de crecer o adaptarse según las necesidades cambiantes de la empresa.
- Deben comunicarse con total facilidad y de forma transparente al usuario con el equipo de planta y con el resto de la empresa (redes locales y de gestión).
- Deben ser programas sencillos de instalar, sin excesivas exigencias de hardware, y fáciles de utilizar, con interfaces amigables con el usuario.”¹⁴

3.1.3. Módulos de un SCADA.

“Los módulos o bloques software que permiten las actividades de adquisición, supervisión y control son los siguientes:

¹⁴COSTA, Antonio, Un panorama sobre sistemas SCADA, Brasil, 1995
<http://isa.uniovi.es/~felipe/files/infindII/documentos/scadas.pdf>

- Configuración: permite al usuario definir el entorno de trabajo de su SCADA, adaptándolo a la aplicación particular que se desea desarrollar.
- Interfaz gráfico del operador: proporciona al operador las funciones de control y supervisión de la planta. El proceso se representa mediante sinópticos gráficos almacenados en el ordenador de proceso y generados desde el editor incorporado en el SCADA o importados desde otra aplicación durante la configuración del paquete.
- Módulo de proceso: ejecuta las acciones de mando pre programado a partir de los valores actuales de variables leídas.
- Gestión y archivo de datos: se encarga del almacenamiento y procesado ordenado de los datos, de forma que otra aplicación o dispositivo pueda tener acceso a ellos.
- Comunicaciones: se encarga de la transferencia de información entre la planta y la arquitectura hardware que soporta el SCADA, y entre ésta y el resto de elementos informáticos de gestión. ¹⁵

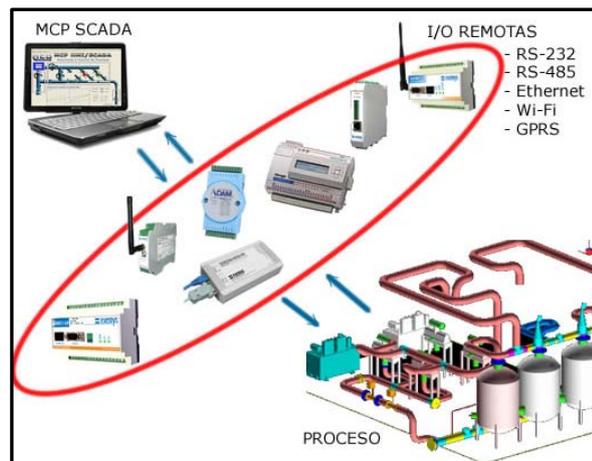


Gráfico41. Módulos o software de un SCADA.

Fuente: BALCELLS Josep, ROMERAL José, Autómatas Programables, México, 1998. www.mcpscada.com.ar/MCP_automatismo.jpg&imgrefurl=http

3.2. Conceptos asociados a un sistema S.C.A.D.A.

“En casa y en la oficina, el ordenador personal continúa con su progreso. El PC se ha establecido en un gran número de campos. Los componentes hardware y

¹⁵ORTEGA, Mario, sistemas de control, México, 2003.
http://www.esi2.us.es/~mortega/clases/1ioi_api/documentos/Tema_11/Tema_11.pdf, año 2009

software están siendo cada vez más potentes y más rentables. Es lógico, por tanto, que la industria quiera tomar provecho de este hecho, para reducir costos e incrementar la productividad.

Ciertas tareas industriales están actualmente en manos de los ordenadores desde hace tiempo: desde emplear la tecnología Windows cuando se manejan pedidos/o se ajustan parámetros de maquinaria hasta preparar o visualizar datos prácticamente de cualquier tipo.”¹⁶

No hay que sorprenderse entonces, que los especialistas en automatización y los usuarios estén pensando ahora en qué forma se pueden transferir al PC otras tareas, para poder llegar a un mayor ahorro. Más recientemente un gran número de simuladores de PLC (controladores lógicos programables) por software ha aparecido en el mercado, que están ayudando a transferir el control de tareas al disco duro y presentan una automatización más efectiva en costes en una simple pieza de hardware (el PC).

3.2.1. Tiempo real.

“La capacidad en tiempo real se refiere a la capacidad del ordenador en programas de procesamiento de datos para que siempre esté listo para procesar y proporcionar los resultados dentro de un tiempo especificado. En este contexto "estrictamente en tiempo real" significa que un sistema reacciona a los eventos externos dentro de un tiempo especificado en un 100% de los casos. Además si se habla de “tiempo real” el sistema debe responder en tiempos concretos también en un 100% de los casos. Si, de otra forma, los tiempos concretos de reacción pueden superarse en ciertos casos, como en sistemas no críticos, hablamos de tiempo real suave”.¹⁷

¹⁶ MORENO, Romualdo, Sistema SCADA para una planta de producción, España, 2008.
http://www.ibtta.org/files/PDFs/Mengotti_Pedro.pdf

¹⁷ MARLIN, Thomas E. Process control: designing processes and control systems for dynamic performance. Mc Graw-Hill, 1975, Pag. 145.

3.2.2. Hardware en sistemas de supervisión.

“El hecho es que las tareas automatizadas de control, visualización y computación pueden ser efectuadas por PLCs (conectados en red mediante los módulos adecuados) mejor que con sistemas exclusivos de control basados en PC. Lo que finalmente es práctico, no obstante, depende de un gran número de factores y la mayoría deben ser considerados individualmente para cada proyecto de automatización.”¹⁸

Así, por ejemplo, los actuales conocimientos y preferencias del usuario pueden jugar un mayor papel que la pura potencia del ordenador. Los factores cruciales, no obstante, son los atributos de capacidad en tiempo real y las propiedades de seguridad que hasta ahora han sido fuertemente asociadas con el PLC, aunque el PC también puede disponer de la característica de capacidad en tiempo real. Un sistema de control es inconcebible sin capacidad en tiempo real. Es común en sistemas de control por ordenador tener que elegir, según las características del sistema a supervisar, entre el PLC o el PC. Se debe elegir aquel hardware que mejor se adapte a las necesidades del sistema a supervisar.

Los controladores lógicos programables, en la mayoría de los casos, están diseñados específicamente para ser empleados en ambientes industriales exigentes y han sido continuamente desarrollados de forma que sus sistemas operativos en tiempo real representan su mayor virtud. Ellos son y seguirán siendo, no obstante, la primera elección para todo control de tareas críticas o extremas por su rendimiento y simpleza, en los que un PC podría estar simplemente sobrecargado debido al trabajo que le pueden suponer otras tareas de ámbito común, como la gestión y visualización de datos, accesos a periféricos, bases de datos, etc.

¹⁸ DANFOSS, Drivers and Controls, Alemania, 1998.
<http://www.monografias.com/trabajos11/sisco/sisco.shtml>, año 1999

Si, además del control de tareas, se necesita un procesamiento de datos, trabajo en red o visualización (una aplicación SCADA), un sistema basado en PC debe ser tomado en consideración.

En cuanto a sistemas operativos, Windows, por ejemplo, no es estrictamente un sistema operativo en tiempo real como el de un PLC, pero puede actuar de forma suficientemente rápida para aplicaciones "suaves" en tiempo real, gracias a su arquitectura de micro-kernel.

3.2.3. Tarjetas de expansión.

Como el sistema operativo sólo puede proporcionar respuestas suaves en tiempo real lo más simple es emplear extensiones hardware para las tareas críticas (placas de expansión PC) y soluciones software para el resto de tareas. Esto nos lleva a una compatibilidad con futuros sistemas operativos y una solución totalmente factible actualmente. Estas tarjetas de expansión asumen las tareas críticas en tiempo real que el ordenador (PC) no puede atender, se está hablando de tarjetas que incorporan DSPs (Procesadores de Señales Digitales) o micro controladores y que aportan una ayuda a la anterior "sobrecarga" mencionada para los ordenadores (PC).

3.2.4. La estructura abierta.

Aún no se ha establecido un estándar para poseer extensiones compatibles en tiempo real de sistemas operativos. De una forma estrictamente determinante, los sistemas estándar actuales deben ser modificados de forma general, así que la principal ventaja de un sistema basado en PC su estructura abierta puede llegar a ser un inconveniente. No obstante, la estructura abierta, permite a la empresa o el desarrollador más libertad en la elección de la herramienta adecuada para el análisis, diseño y programación del sistema SCADA. La solución comienza a ser propietaria nuevamente (cada empresa ofrece su solución) y la conversión a futuras generaciones de sistemas operativos lo hace más difícil.

3.3. Introducción a LABVIEW.

“*LabVIEW* es una herramienta diseñada especialmente para monitorizar, controlar, automatizar y realizar cálculos complejos de señales analógicas y digitales capturadas a través de tarjetas de adquisición de datos, puertos serie y GPIBs (Buses de Intercambio de Propósito General).

Es un lenguaje de programación de propósito general, como es el Lenguaje C o Basic, pero con la característica que es totalmente gráfico, facilitando de esta manera el entendimiento y manejo de dicho lenguaje para el diseñador y programador de aplicaciones tipo SCADA.

Incluye librerías para la adquisición, análisis, presentación y almacenamiento de datos, GPIB y puertos serie. Además de otras prestaciones, como la conectividad con otros programas, por ejemplo de cálculo, y en especial MatLab. “¹⁹

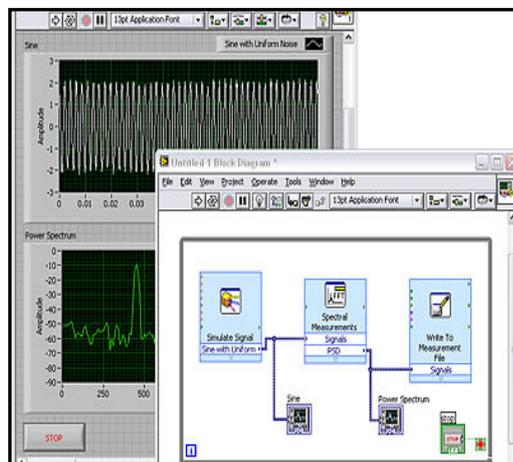


Gráfico42. Pantallas de software Labview.

Fuente: RONCANCIO, Antonio, Tutorial de Labview, España, 2001.
www.mcpscada.com.ar/MCP_automatismo.jpg&imgrefurl=http

¹⁹ MORALES, Jaime, Consultorías Industriales en Automatización, Colombia, 2007.
<http://blogs.ua.es/industriales/2010/02/06/diseño-de-aplicaciones-scada-con-labview>.

Está basado en la programación modular, lo que permite crear tareas muy complicadas a partir de módulos o sub-módulos mucho más sencillos. Además estos módulos pueden ser usados en otras tareas, con lo cual permite una programación más rápida y provechosa.

También ofrece la ventaja de “debugging” en cualquier punto de la aplicación. Permite la posibilidad de poner “break points”, ejecución paso a paso, ejecución hasta un punto determinado y se puede observar como los datos van tomando valores a medida que se va ejecutando la aplicación. Además también lleva incorporado generadores de señales para poder hacer un simulador.

3.3.1 Filosofía de LABVIEW.

LabVIEW es un lenguaje completamente gráfico, y el resultado de ello es que es totalmente parecido a un instrumento, por ello a todos los módulos creados con LabVIEW se les llama VI (Instrumento Virtual).

Existen dos conceptos básicos en LabVIEW: el Front Panel (Panel Frontal) y el Block diagram (Diagrama de Bloque).

El Panel Frontal es el interfaz que el usuario está viendo y puede ser totalmente parecido al instrumento del cual se están recogiendo los datos, de esta manera el usuario sabe de manera precisa cual es el estado actual de dicho instrumento y los valores de las señales que se están midiendo, El diagrama de bloques es el conexionado de todos los controles y variables, que tendría cierto parecido al diagrama del esquema eléctrico del instrumento. LabVIEW tiene la característica de descomposición modular ya que cualquier VI que se ha diseñado puede convertirse fácilmente en un módulo que puede ser usado como una sub-unidad dentro de otro VI. Esta peculiaridad podría compararse a la característica de procedimiento en los lenguajes de programación estructurada.

“Es un sistema abierto, en cuanto a que cualquier fabricante de tarjetas de adquisición de datos o instrumentos en general puede proporcionar el driver de su producto en forma de VI dentro del entorno de LabVIEW. También es posible programar módulos para LabVIEW en lenguajes como C y C++, estos módulos son conocidos como Sub-VIs y no se difieren a los VI creados con LabVIEW salvo por el interfaz del lenguaje en el que han sido programados. Además estos Sub-VIs son muy útiles por ejemplo en el campo de cálculos numéricos complejos que no se encuentran incluidos en las librerías de LabVIEW”.²⁰

3.3.2 Panel frontal y diagrama de bloques.

Se podría decir que en cualquier VI existen dos caras bien diferenciadas: El Panel Frontal y el Diagrama de Bloques.

El Panel Frontal es la cara que el usuario del sistema está viendo cuando se está monitorizando o controlando el sistema, o sea, el interfaz del usuario. Este contiene controles e indicadores y existe una gran variedad de ellos, pero además incluso se pueden diseñar controles e indicadores personalizados, lo cual permite tener una amplia gama de dichos controles e indicadores.

Un control puede tomar muchas formas, y muchas de estas formas el dibujo real usado en instrumentos reales. Otros son estrictamente conceptos digitales o analógicos. Pero todos los controles tienen una forma visual que indican al usuario cual es el estado de dicho control en el instrumento real.

Es muy importante en un sistema SCADA que el usuario no tenga que interpretar nada, sino que todo le sea claro y conciso, las interpretaciones pueden dar lugar a

²⁰ LÁZARO, Manuel – FERNÁNDEZ, Joaquín del Río. LabVIEW 7.1 Programación Gráfica para el Control de Instrumentación. España, 2005, Pag. 32.

falsas actuaciones y, por consiguiente, podrían existir lamentables errores. Además, dos usuarios podrían interpretar de manera diferente cualquier evento.

El Diagrama de Bloques del VI sería la cara oculta del Panel Frontal, una cara que el usuario del sistema no puede ver. En ella están todos los controles e indicadores interconectados, pareciéndose mucho a un diagrama de esquema eléctrico. Esta cara es mucho menos conceptual que el Panel Frontal y para el usuario sería muy difícil entenderla.

Todos los módulos están interconectados, mediante líneas de conexión, por donde circulan los diferentes datos o valores del VI., de esta manera se logra que el VI funcione como un conjunto de elementos, módulos y sub-módulos.

3.3.3 Tarjetas de adquisición.

“Las tarjetas DAQ son tarjetas insertables que permiten la entrada y salida de datos del computador a otros aparatos, donde se conectan sensores, y actuadores, para interactuar con el mundo real. Los datos que entran y salen pueden ser señales digitales o análogas, o simplemente conteos de ocurrencias digitales, tanto de entrada, como de salida.

Las tarjetas se comportan como si fueran un puerto más en el computador, y poseen todo un protocolo y sistema de manejo, por lo que entender cada tarjeta, como su funcionamiento, al igual que cualquier instrumento, requiere de tiempo y cuidado.”²¹

Para aplicaciones de alta velocidad y tiempo real, se requiere de hardware especial ósea tarjetas inteligentes, que se programen, y transfieran los datos a memoria, ya

²¹ VELASCO, Hugo, Introducción a Labview, Colombia, 2000.
<http://www.monografias.com/proyecto-integrador-digital/proyecto-integrador-digital.shtml>

sea por rutinas de DMA (acceso directo a memoria), o por rutinas de interrupciones al procesador.

Las tarjetas como cualquier otro periférico, requiere de sus parámetros de programación, y hasta protocolos de comunicación, por lo que se requiere de un software Driver que maneje lo bajo de programación, y deje en la superficie, la posibilidad de programar aplicaciones con los beneficios de dichas tarjetas, de una forma sencilla.

La configuración se hace a través del programa anexo a LabVIEW, NI-DAQ o bien por programaciones especiales para rutinas específicas como MATLAB.

3.3.4 La estructura de Labview.

“Las instrucciones de control permiten a un programa ejecutar un código de forma condicional o repetirlo cierto número de veces. En Labview estas instrucciones son estructuras que encierran en su interior el código al que afectan. Acbe destacar las nuevas estructuras introducidas en las últimas versiones como EVENT y TIMED LOOP, así como las tradicionales FOR y WHILE.”²²

3.4. APLICACIÓN AL PROYECTO MONOGRAFICO.

Una vez revisado los conceptos principales para un sistema SCADA nos dirigimos directamente a la configuración del enlace que comunicara al S7-200 con el software Labview para visualización del proceso, para esto necesitamos el “enlace” que será el software OPC.

El OLE para el Control de Procesos (OPC), corresponde a un conjunto de especificaciones basadas en los estándares de Microsoft (COM, DCOM, OLE Automation y ActiveX) que cubren los requerimientos de comunicación industrial

²² RIVAS, Lizbeth, Técnicas emergentes para la automatización, Colombia, 2006.
<http://books/concepto+de+estructura+de+labview&source>, año 2009

entre aplicaciones y dispositivos, especialmente en lo que se refiere a la atención al tiempo real.

La configuración del OPC tanto como SERVIDOR así como CLIENTE se realiza de la siguiente manera:

PASO 1a: Una vez ejecutado el OPC server damos un click en crear un nuevo canal de comunicación, que será el medio por el cual nos comunicaremos entre el s7-200 y el computador.

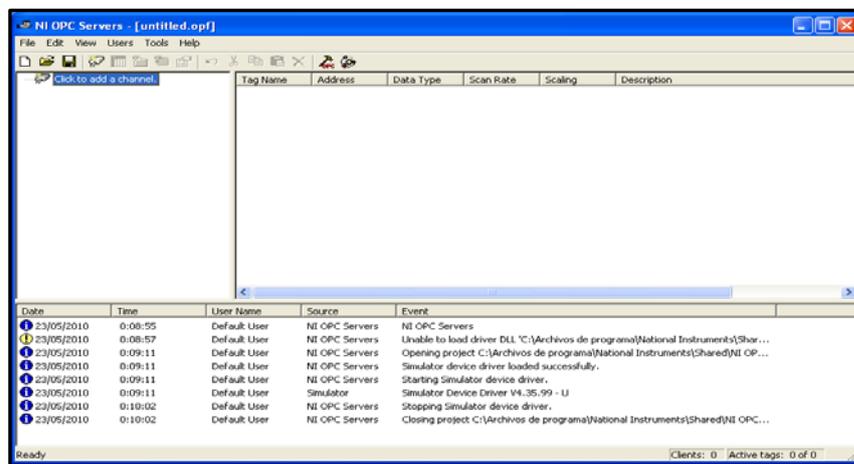


Gráfico43. Pantalla para configura la comunicación OPC.

Fuente: Autores, Programación en el software NI-POC.

PASO 1b: poner un nombre al canal de comunicación para poder identificar en caso de que se comunique con varios dispositivos, en nuestro caso se S7-200.

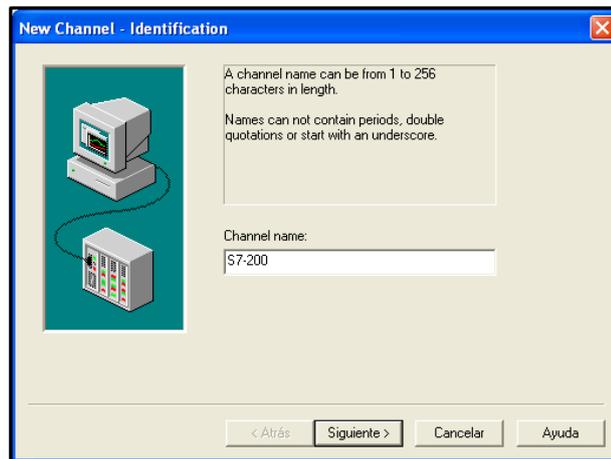


Gráfico44. Nombre del canal de comunicación OPC.

Fuente: Autores, Programación en el software NI-POC.

PASO 1c: Debe seleccionar el dispositivo que va a comunicarse, de acuerdo a marca y modelo del mismo.



Gráfico45. Selección del dispositivo para la comunicación OPC.

Fuente: Autores, Programación en el software NI-POC.

PASO 1d: Ahora tendremos que configurar los parámetros de comunicación de acuerdo a lo programado en el PLC s7-200, como la velocidad de transmisión, número de bits de transmisión, bit de paridad, reporte de errores.

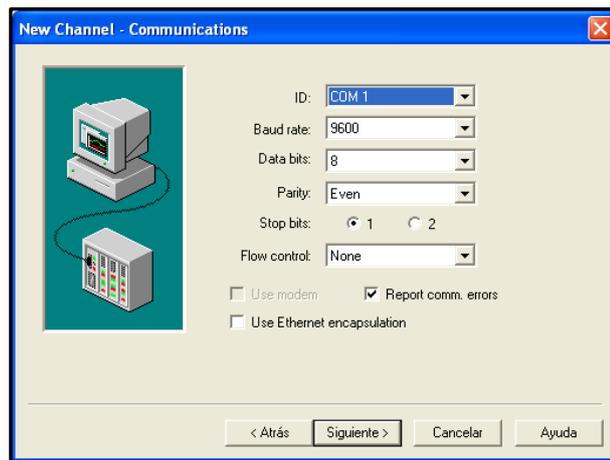


Gráfico46. Parámetros de Configuración.

Fuente: Autores, Programación en el software NI-POC.

PASO 1e: Ahora configuraremos el Identificador Maestro por el cual se va a comunicar y no podrá ser utilizado por ningún otro dispositivo.

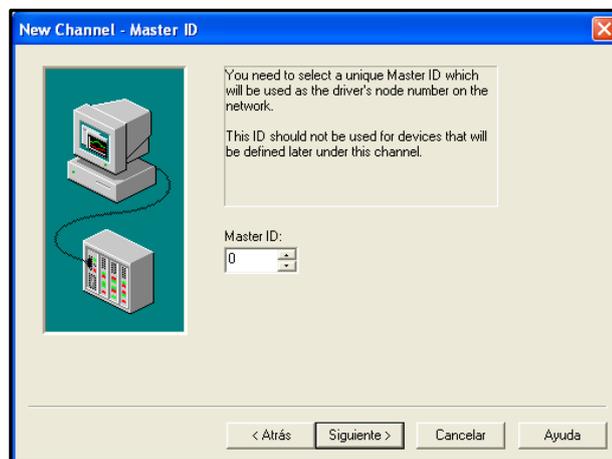


Gráfico47. Identificador Maestro OPC.

Fuente: Autores, Programación en el software NI-POC.

PASO 1f: Damos click en finalizar y queda configurado el canal de comunicación.

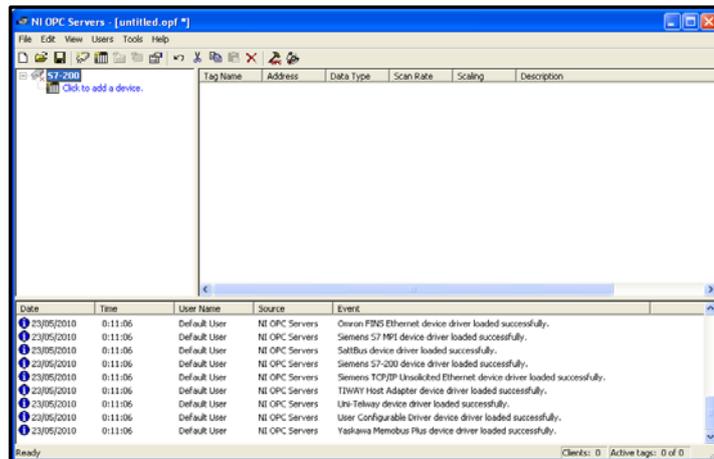


Gráfico48. Canal de comunicación ya establecido.

Fuente: Autores, Programación en el software NI-POC.

PASO 2a: Seguidamente daremos un nombre que identifique al dispositivo que utilizara este canal ya establecido.

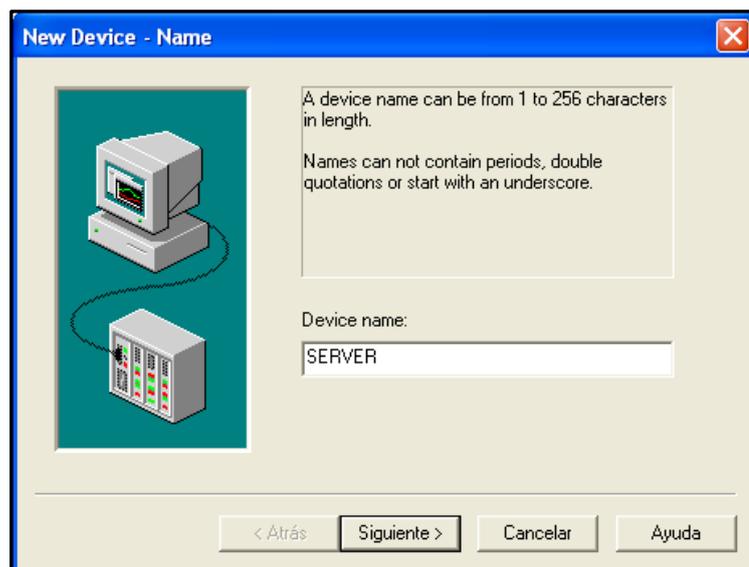


Gráfico49. Nombre de dispositivo a usar el canal.

Fuente: Autores, Programación en el software NI-POC.

PASO 2b: Se debe configurar un modelo de dispositivo que ya se declaro anteriormente en la configuración del canal de comunicación, para nuestro proyecto está declarado como S7-200.

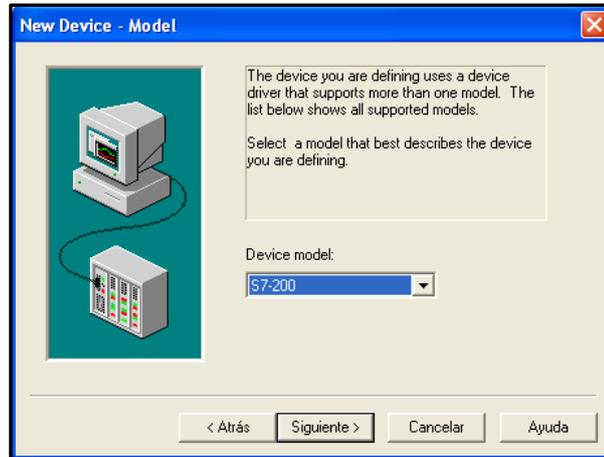


Gráfico50. Modelo del dispositivo declarado en el canal de comunicación.

Fuente: Autores, Programación en el software NI-POC.

PASO 2c: Ahora se configurara el número del dispositivo para que sea identificado dentro del canal de comunicación.

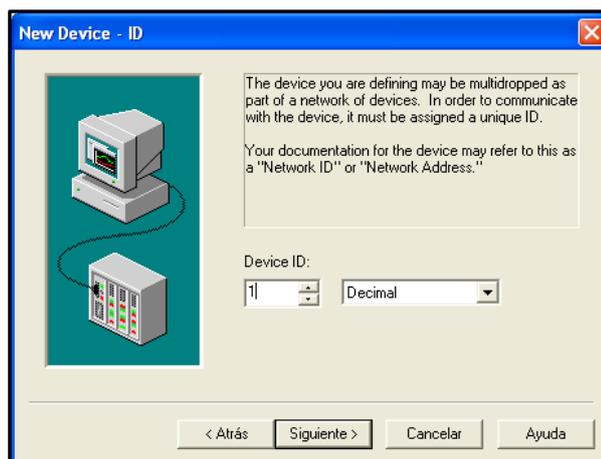


Gráfico51. Identificación del dispositivo en el canal de comunicación.

Fuente: Autores, Programación en el software NI-POC.

PASO 2d: La siguiente ventana será la configuración del tiempo de muestreo en donde se obtendrá los datos dentro del computador.

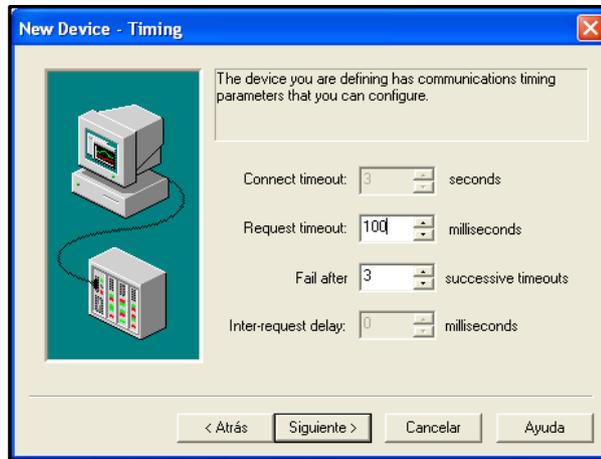


Gráfico52. Configuración de tiempo de muestreo.

Fuente: Autores, Programación en el software NI-POC.

PASO 2e: Damos click en finalizar y queda configurado el dispositivo dentro del canal de comunicación.

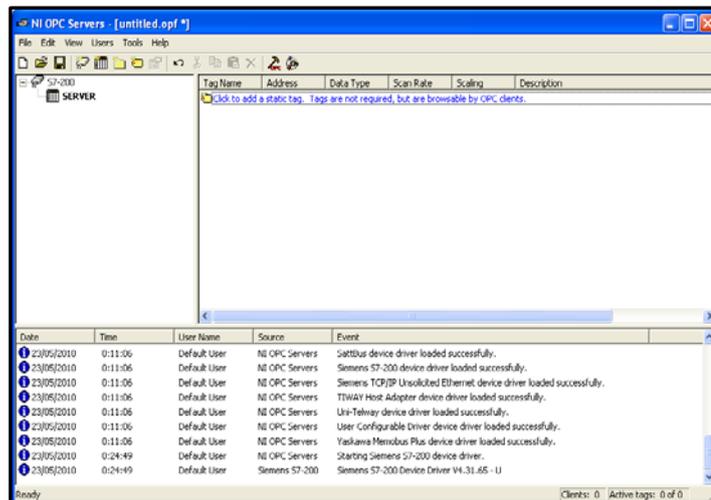


Gráfico53. Dispositivo configurado en el canal S7-200.

Fuente: Autores, Programación en el software NI-POC.

Para poder reconocer cada uno de los dispositivos dentro de canal de comunicación lo que debemos hacer es configurarlos por su nombre de la siguiente manera.

PASO 3a: En cada uno de los dispositivos de este canal damos click derecho e ingresamos en propiedades de Tag en donde podemos cambiar el nombre y la dirección que se desea acceder para leer o escribir.

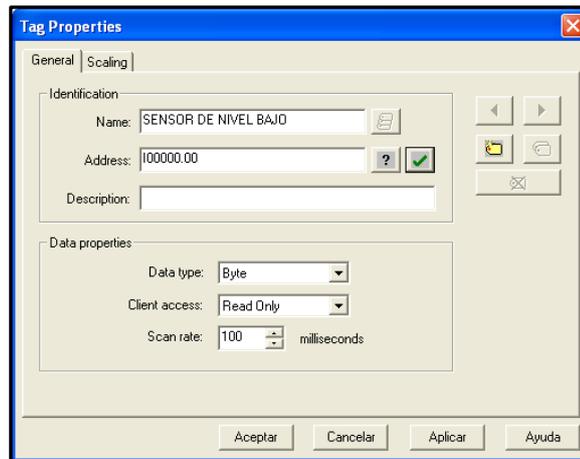


Gráfico54. Propiedades de Tag.

Fuente: Autores, Programación en el software NI-POC.

PASO 3b: Dar click en aceptar y se tendrá todos los dispositivos por el nombre de la variable que desea manipular.

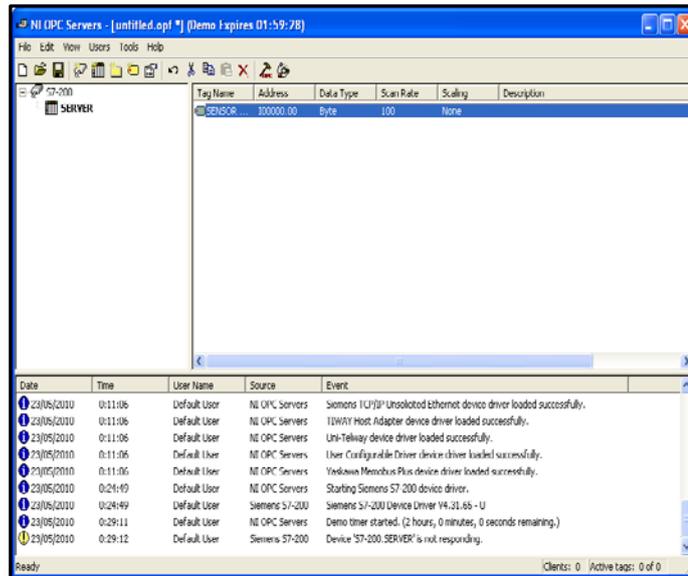


Gráfico55. Variables declaradas en OPC.

Fuente: Autores, Programación en el software NI-POC.

PASO 4: Una vez configurado el Servidor damos paso al Cliente que será encargado de comunicarse con el canal del SERVIDOR que se configuro en el PASO 2.

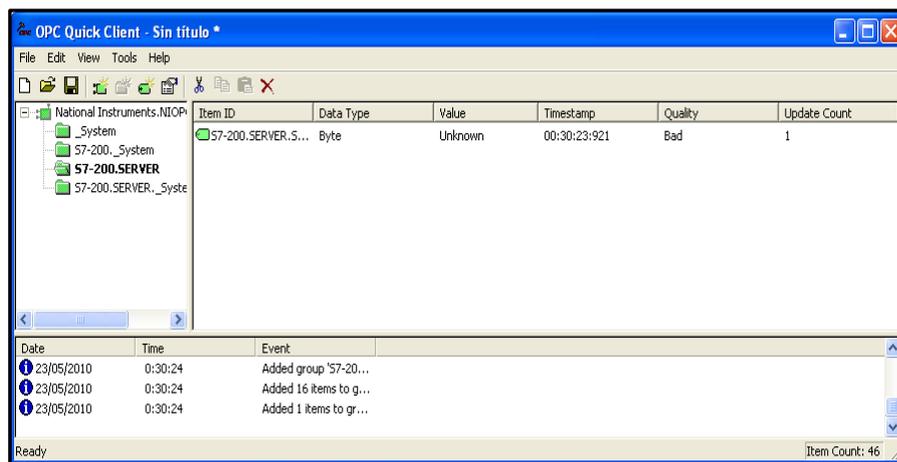


Gráfico56. Configuración OPC Cliente.

Fuente: Autores, Programación en el software NI-POC.

PASO 5a: Luego de realizar todas las configuraciones del OPC debemos iniciar el software Labview y dar click en el icono Empty Project para poder realizar la configuración con el OPC.

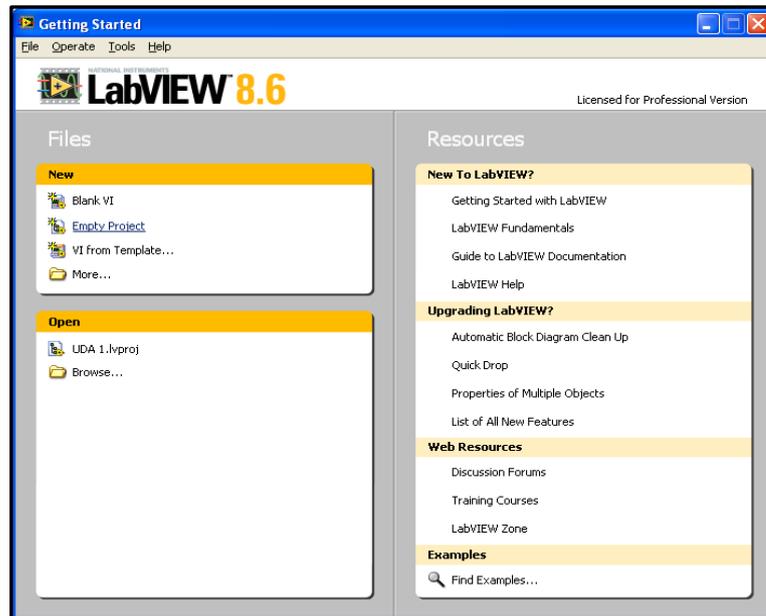


Gráfico57. Ventana de inicio de software Labview.

Fuente: Autores, Programación en el software LABVIEW.

PASO 5b: El siguiente paso será crear un nuevo archivo en el software Labview en donde se pueda obtener las entradas y salidas del servidor.

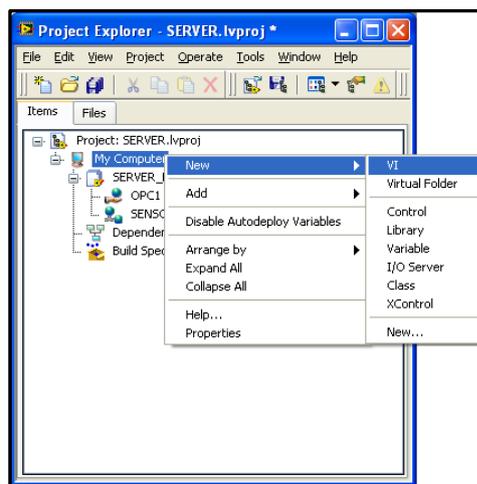


Gráfico58. Creación de nuevo proyecto en Labview.

Fuente: Autores, Programación en el software LABVIEW.

PASO 5c: Ahora seleccionaremos el enlace por el cual se van a comunicar que en nuestro proyecto será OPC, seleccionando OPC Cliente.

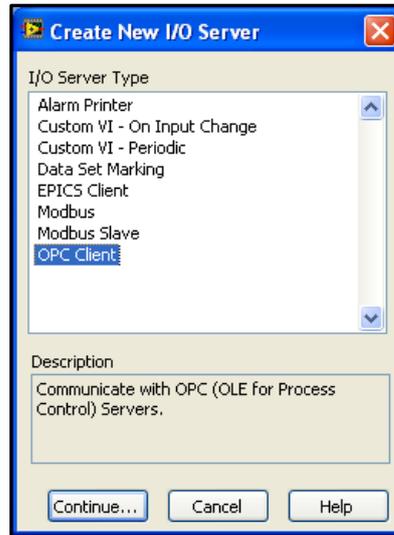


Gráfico59. Creación de nuevo Servidor en Labview.

Fuente: Autores, Programación en el software LABVIEW.

PASO 5d: La siguiente pantalla la tendremos por defecto, por lo que debemos modificar el tiempo de muestreo de acuerdo a las necesidades.

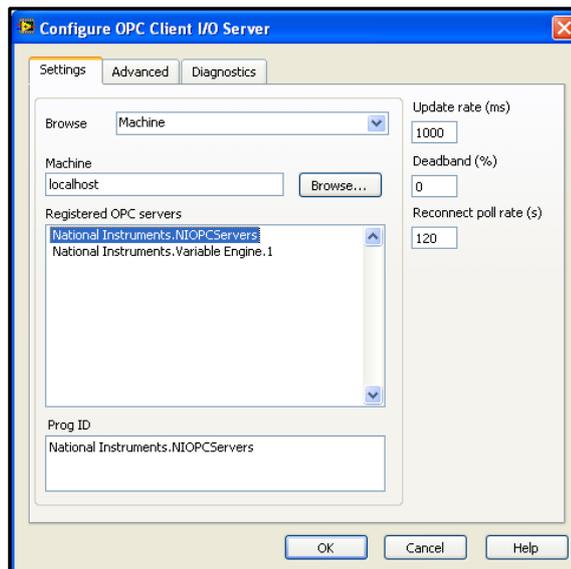


Gráfico60. Modificación de tiempos de muestreo en Labview.

Fuente: Autores, Programación en el software LABVIEW.

PASO 5e: Una vez establecidos los tiempos introducimos la variable a que se quiere visualizar, leer o escribir.

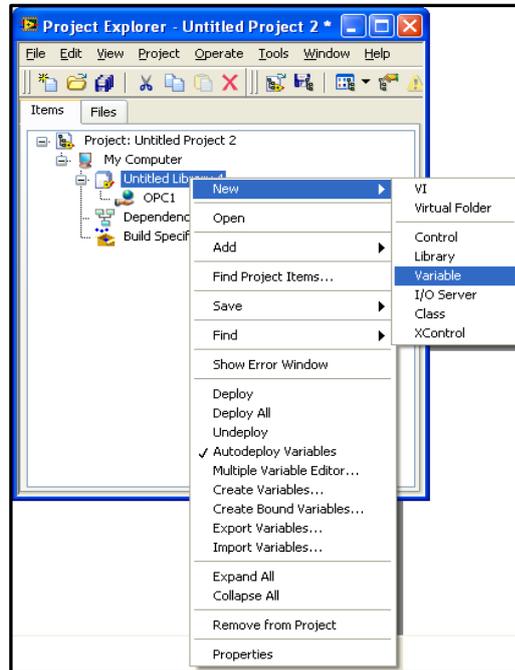


Gráfico61. Creación de la variable en Labview.

Fuente: Autores, Programación en el software LABVIEW.

PASO 5f: Se debe configurar las propiedades de la variable declarada en el paso anterior, para tener el nombre de la variable así como el tipo además de si es de lectura o escritura.

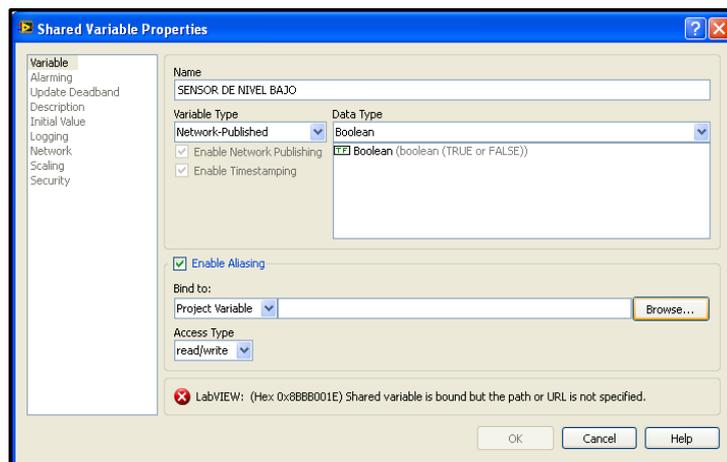


Gráfico62. Propiedades de una variable en Labview.

Fuente: Autores, Programación en el software LABVIEW.

PASO 5g: Seguidamente obtendremos la debida ubicación de la variable dentro del enlace OPC.

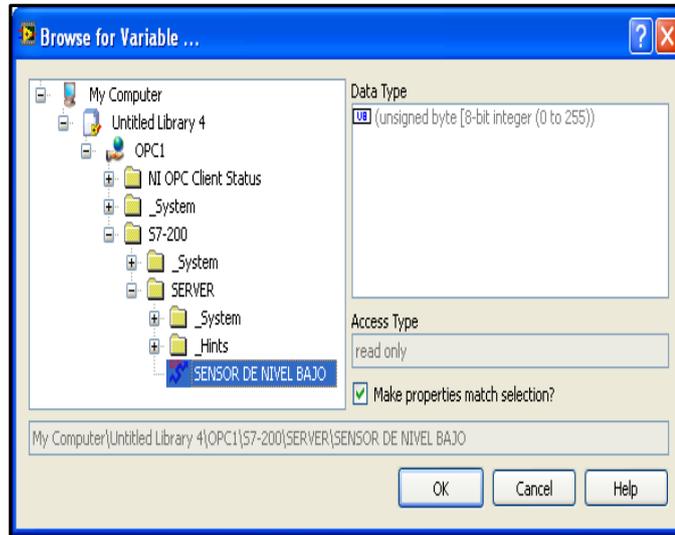


Gráfico63. Ubicación de la variable entre OPC y Labview.

Fuente: Autores, Programación en el software LABVIEW.

PASO 5h: Una vez efectuada la declaración y configurado las propiedades se debe tener una pantalla similar al Grafico64.

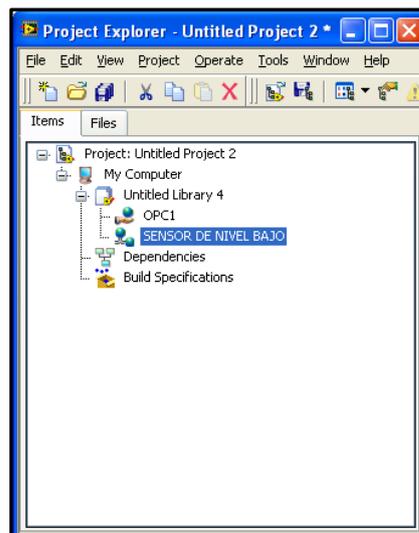


Gráfico64. Creación de nuevo proyecto en Labview.

Fuente: Autores, Programación en el software LABVIEW.

PASO 5i: Un nuevo proyecto debe ser creado mediante un click derecho en el icono my Computer y la opción Nuevo>> VI (Instrumento Virtual).

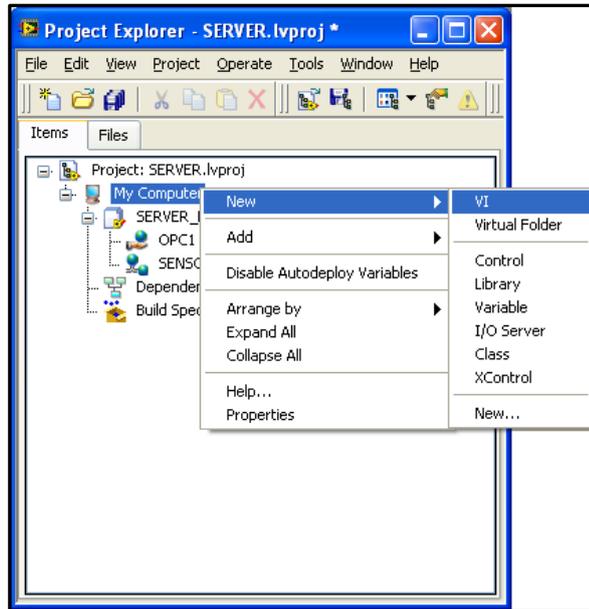


Gráfico65. Creación de nuevo VI en Labview.

Fuente: Autores, Programación en el software LABVIEW.

PASO 5j: Para finalizar lo que se necesita es desde el cuadro de dialogo entre OPC y Labview llevar la variable hacia el Panel de Control de Labview.

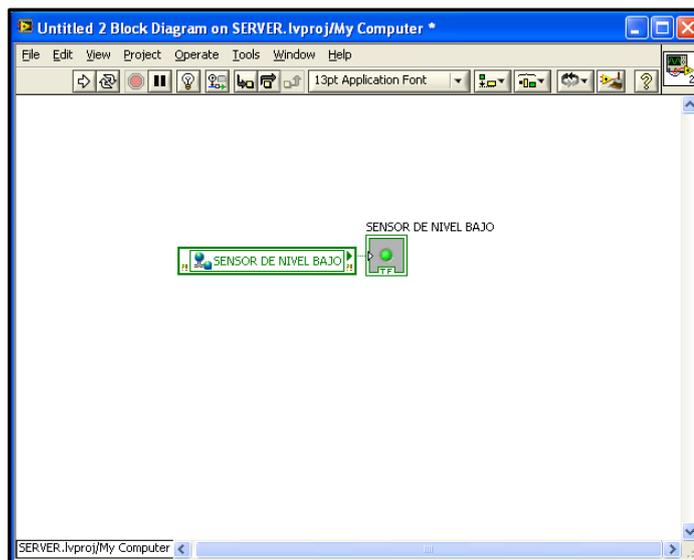


Gráfico66. Variable de S7-200 funcionando en Labview.

Fuente: Autores, Programación en el software LABVIEW.

La programación de las variables que serán comunicados entre el PLC S7-200 y el software Labview puede ser consultada en el Anexo2.

CAPITULO 4

CONTROL PID DE LA TEMPERATURA.

4.1. Introducción.

El control automático desempeña un papel importante en los procesos de manufactura, industriales, navales, aeroespaciales, robótica, económicos, biológicos, etc.

Aquí se debe tratar de encontrar las Raíces del sistema, el cual le dará información importante sobre la dinámica del mismo. El conocimiento del funcionamiento del sistema junto con el análisis de la función de transferencia de lazo abierto y del Lugar de las Raíces darán las bases necesarias para seleccionar el controlador.

Se debe tener en cuenta los siguientes términos:

“Señal de salida: es la variable que se desea controlar (posición, velocidad, presión, temperatura, etc.). También se denomina variable controlada.

Señal de referencia: es el valor que se desea que alcance la señal de salida.

Error: es la diferencia entre la señal de referencia y la señal de salida real.

Señal de control: es la señal que produce el controlador para modificar la variable controlada de tal forma que se disminuya, o elimine, el error.

Planta: es el elemento físico que se desea controlar. Planta puede ser: un motor, un horno, un sistema de disparo, un sistema de navegación, un tanque de combustible, etc.

Proceso: operación que conduce a un resultado determinado.

Sistema: consiste en un conjunto de elementos que actúan coordinadamente para realizar un objetivo determinado.

Perturbación: es una señal que tiende a afectar la salida del sistema, desviándola del valor deseado.

Sensor: es un dispositivo que convierte el valor de una magnitud física (presión, flujo, temperatura, etc.) en una señal eléctrica codificada ya sea en forma analógica o digital. También es llamado transductor. Los sensores, o transductores, analógicos envían, por lo regular, señales normalizadas de 0 a 5 voltios, 0 a 10 voltios o 4 a 20 mA.

Sistema de control en lazo cerrado: es aquel en el cual continuamente se está monitoreando la señal de salida para compararla con la señal de referencia y calcular la señal de error, la cual a su vez es aplicada al controlador para generar la señal de control y tratar de llevar la señal de salida al valor deseado. También es llamado control realimentado.

Sistema de control en lazo abierto: en estos sistemas de control la señal de salida no es monitoreada para generar una señal de control.”²³

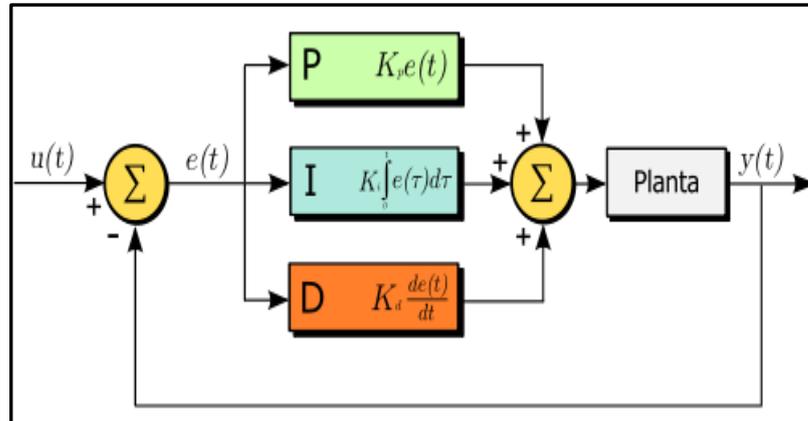


Gráfico67. Diagrama de bloques de un control PID.

Fuente: http://es.wikipedia.org/wiki/Proporcional_integral_derivativo

4.2. Estructura de control PID.

Consideramos el lazo básico de control SISO

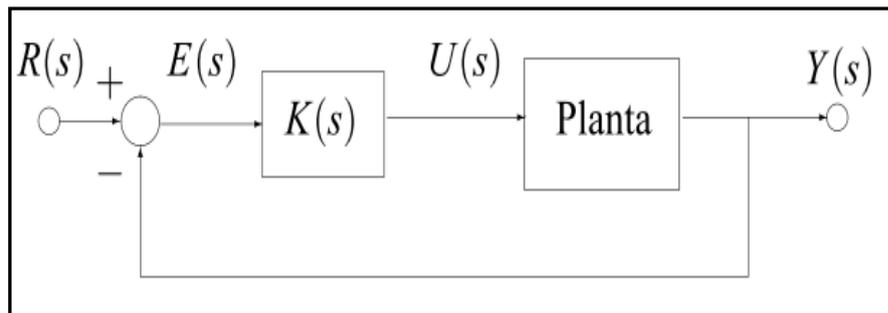


Gráfico68. Estructura de un control PID.

Fuente: http://es.wikipedia.org/wiki/Proporcional_integral_derivativo

²³ VILANOVA, Ramon, Procesos con PID, España, 2004
http://www.automatas.org/hardware/teoria_pid.htm

Los miembros de la familia de controladores PID, incluyen tres acciones: proporcional (P), integral (I) y derivativa (D). Estos controladores son los denominados P, I, PI, PD y PID.

4.2.1. Acción de control proporcional P.

Con esta acción tenemos una salida del controlador que es proporcional al error, es decir: $u(t) = K_p \cdot e(t)$, que descripta desde su función transferencia queda:

$$C_p(s) = K_p$$

En donde K_p es una ganancia proporcional ajustable. Un controlador proporcional puede controlar cualquier planta estable, pero posee desempeño limitado y error en régimen permanente (off-set).

4.2.2. Acción de control integral I.

Esta acción da una salida del controlador que es proporcional al error acumulado, lo que implica que es un modo de controlar lento.

$$U(t) = K_I \int_0^t e(\tau) d\tau \quad C_i(s) = \frac{K_i}{s}$$

La señal de control $u(t)$ tiene un valor diferente de cero cuando la señal de error $e(t)$ es cero. Por lo que se concluye que dada una referencia constante, o perturbaciones, el error en régimen permanente es cero.

4.2.3. Acción de control proporcional-integral PI.

Se define mediante

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int_0^t e(\tau) d\tau$$

Donde T_i se denomina tiempo integral y es quien ajusta la acción integral. La función de transferencia resulta:

$$C_{PI}(S) = K_p \left(1 + \frac{1}{T_i S} \right)$$

“Con un control proporcional, es necesario que exista error para tener una acción de control distinta de cero. Con acción integral, un error pequeño positivo siempre nos dará una acción de control creciente, y si fuera negativa la señal de control será decreciente. Este razonamiento sencillo nos muestra que el error en régimen permanente será siempre cero.

Muchos controladores industriales tienen solo acción PI. Se puede demostrar que un control PI es adecuado para todos los procesos donde la dinámica es esencialmente de primer orden. Lo que puede demostrarse en forma sencilla, por ejemplo, mediante un ensayo al escalón.”²⁴

²⁴ CASTILLO, J.- PEREZ, J.S. Control de temperatura de un horno industrial de secado utilizando un PID analógico. México. 1999. Pag 62.

4.2.4. Acción de control proporcional-derivativa PD.

Se define mediante:

$$u(t) = K_p e(t) + K_p T_d \frac{de(t)}{dt}$$

En donde T_d es una constante denominada tiempo derivativo. Esta acción tiene carácter de previsión, lo que hace más rápida la acción de control, aunque tiene la desventaja importante que amplifica las señales de ruido y puede provocar saturación en el actuador. La acción de control derivativa nunca se utiliza por sí sola, debido a que solo es eficaz durante períodos transitorios. La función de transferencia de un controlador PD resulta:

$$C_{PD}(S) = K_p + SK_p T_d$$

Cuando una acción de control derivativa se agrega a un controlador proporcional, permite obtener un controlador de alta sensibilidad, es decir que responde a la velocidad del cambio del error y produce una corrección significativa antes de que la magnitud del error se vuelva demasiado grande. Aunque el control derivativo no afecta en forma directa al error de estado estacionario, añade amortiguamiento al sistema y, por tanto, permite un valor más grande que la ganancia K , lo cual provoca una mejora en la precisión en estado estable.

4.2.5. Acción de control proporcional-integral-derivativa PID.

Esta acción combinada reúne las ventajas de cada una de las tres acciones de control individuales. La ecuación de un controlador con esta acción combinada se obtiene mediante:

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int_0^t e(\tau) d\tau + K_p T_d \frac{de(t)}{dt}$$

Y su función transferencia resulta:

$$C_{PID}(s) = K_P \left(1 + \frac{1}{T_i s} + T_d s \right)$$

4.3. Métodos clásicos de ajuste de Ziegler y Nichols.

“Los métodos de ajuste de las ganancias de un controlador PID, son el Método de Oscilación o Método de Respuesta en Frecuencia y el Método Basado en la Curva Reacción o Método de Respuesta al Escalón. El primero se basa en un lazo de control solo con ganancia proporcional y de acuerdo a la ganancia utilizada para que el sistema empiece a oscilar y al período de esas oscilaciones, podemos establecer las ganancias del controlador PID. El otro método se resume en ensayar al sistema a lazo abierto con un escalón unitario, se calculan algunos parámetros, como la máxima pendiente de la curva y el retardo, y con ellos establecemos las ganancias del controlador PID. Estos métodos fueron propuestos por Ziegler y Nichols (Z-N) en 1942, quienes se basaron en la práctica para desarrollarlos.”²⁵

4.4. Método de Oscilación.

Este procedimiento es válido solo para plantas estables a lazo abierto y se lleva a cabo siguiendo los siguientes pasos:

²⁵ FOMAGOSA, Jaime, Sistemas SCADA, USA, 2004.
<http://www.eng.newcastle.edu.au/~jhb519/teaching/caut1/Apuntes/PID.pdf>, año 2001

- Utilizando solo control proporcional, comenzando con un valor de ganancia pequeño, incrementar la ganancia hasta que el lazo comience a oscilar. Notar que se requieren oscilaciones lineales y que estas deben ser observadas en la salida del controlador.
- Registrar la ganancia crítica del controlador $K_p = K_c$ y el período de oscilación de la salida del controlador, P_c . (En el diagrama de Nyquist, corresponde a que $K_c G(j\omega)$ cruza el punto $(-1, 0)$ cuando $K_p = K_c$).
- Ajustar los parámetros del controlador según la Tabla 3:

	K_p	T_i	T_d
P	0,50 K_c		
PI	0,45 K_c	$P_c/1.2$	
PID	0,60 K_c	0.5 P_c	$P_c/8$

Tabla 3. Parámetros de ajuste (método de oscilación).

Fuente: CORRIPIO, Armando, Tuning of industrial control systems, USA. 1990. Pag. 79.

Dicha tabla fue obtenida por Ziegler y Nichols quienes buscaban una respuesta al escalón de bajo amortiguamiento para plantas que puedan describirse satisfactoriamente por un modelo de la forma:

$$G_o(s) = \frac{K_o e^{-s\tau_o}}{v_o s + 1} \quad \text{donde } v_o > 0$$

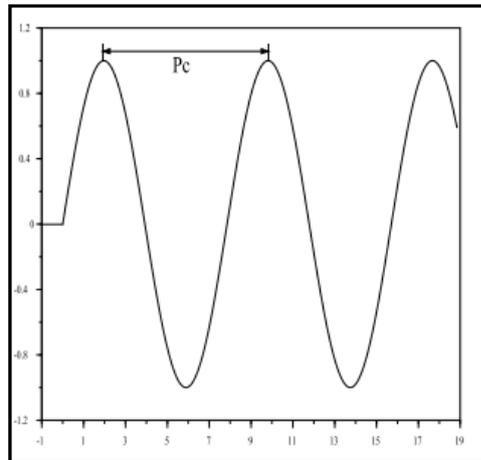


Gráfico69. Respuesta de la planta con ganancia crítica.

Fuente: BANUELOS, M.A.- MARTINEZ, W, Análisis y diseño de un controlador PID analógico. México. [s.a.], Pag 52.

4.5. Método Basado en la Curva Reacción.

Muchas plantas, pueden ser descritas satisfactoriamente por el modelo:

$$G_o(s) = \frac{K_o e^{-s\tau_o}}{v_o s + 1} \quad \text{donde } v_o > 0$$

Una versión cuantitativa lineal de este modelo puede ser obtenida mediante un experimento a lazo abierto, utilizando el siguiente procedimiento:

- Con la planta a lazo abierto, llevar a la planta a un punto de operación normal. Digamos que la salida de la planta se estabiliza en $y(t) = y_0$ para una entrada constante $u(t) = u_0$.
- En el instante inicial t_0 , aplicar un cambio en la entrada escalón, desde u_0 a u_∞ (Esto debería ser en un rango de 10 al 20% de rango completo).

- Registrar la salida hasta que se estabilice en el nuevo punto de operación. Supongamos que la curva que se obtiene es la que se muestra en la Grafica70. Esta curva se llama curva de reacción del proceso.

Calcular los parámetros del modelo de la siguiente forma:

$$K_o = \frac{Y_\infty - Y_o}{Y_\infty - u_o}; \quad \tau_o = t_1 - t_o; \quad v_o = t_2 - t_1$$

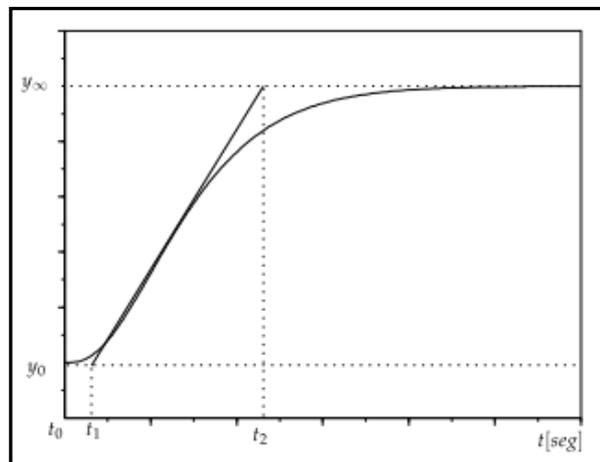


Gráfico70. Respuesta al escalón de la planta.

Fuente: BANUELOS, M.A.- MARTINEZ, W, Análisis y diseño de un controlador PID analógico. México. [s.a.], Pag 54.

El modelo obtenido puede ser utilizado para varios métodos de ajuste de controladores PID. Uno de estos también fue propuesto por Ziegler-Nichols. El objetivo de diseño es alcanzar un amortiguamiento tal que exista una relación de 4:1 para el primer y segundo pico de la respuesta a una referencia escalón. Los parámetros sugeridos por Z-N son los que se muestran en la Tabla4.

TIPO	Kp	Ti	Td
P	T/(K*L)	0	0
PI	0,9*T/(K*L)	0.3/L	0
PID	1,2*T/(K*L)	0,5/L	0,5*L

Tabla 4. Parámetros de ajuste (método curva de reacción).

Fuente: Autores

4.6. Modificaciones de los esquemas de control PID.

En los sistemas de control básicos vistos hasta ahora, si la entrada de referencia es un escalón, debido a la presencia del término derivativo en la acción de control, la variable manipulada $u(t)$ contendrá una función impulso (una delta). En un controlador PID real, en lugar del término derivativo $T_D s$ emplearemos:

$$\frac{T_d s}{\tau_D s + 1}$$

En donde T_D , denominada constante de tiempo derivativa, normalmente es elegida tal que $0.1 \leq T_D \leq 0.2$. Cuanto más pequeña es T_D , mejor es la aproximación entre el término "derivativo filtrado" de la Ecuación y el "derivativo" $T_d s$, es decir son iguales en el límite:

$$\lim_{\tau_d \rightarrow 0} u_{\text{PID}}(t) = K_p e(t) + \frac{K_p}{T_i} \int_{t_0}^t e(\tau) d\tau + K_p T_d \frac{de(t)}{dt}$$

Con la inclusión de un polo evitamos utilizar acciones de control grandes en respuesta a errores de control de alta frecuencia, tales como errores inducidos por cambios de setpoint (referencia) o mediciones de ruido. El argumento clásico por el cual se elige $T_D = 0$ es, además de asegurar un controlador propio, para atenuar ruido de alta frecuencia. Casi todos los controladores industriales PID definen a T_D como una fracción fija de T_d , en lugar de tomarlo como un parámetro independiente de diseño.

4.7. APLICACIÓN AL PROYECTO MONOGRAFICO.

La arquitectura de este sistema es la típica de un servosistema de una entrada y una salida, con el controlador en cascada con la planta

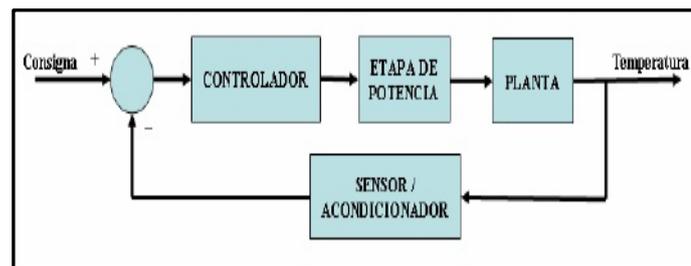


Gráfico71. Estructura del sistema en maqueta.

Fuente: Astrom, Karl - Haggund, Tore, PID controllers: theory, design, and tuning, USA, 1995, Pag. 198.

El invernadero que vamos a controlar está constituido por un sistema térmico y como tal posee una dinámica lenta, siendo esto una ventaja a la hora de realizar prácticas introductorias al control, puesto que se puede ir viendo la evolución del comportamiento del regulador. Una desventaja de utilizar dinámicas lentas es el tiempo empleado en los diferentes pasos para el diseño del controlador.

El sistema térmico para calefacción está constituido por un foco dicróico de 80 Vatios que será controlado por la etapa de potencia que está constituida por un mando de PWM con un optoaislador que activa a un triac (Grafica 72) y el sistema

de refrigeración está constituido por un ventilador el mismo que será activado por un relé accionado por una salida digital del PLC.

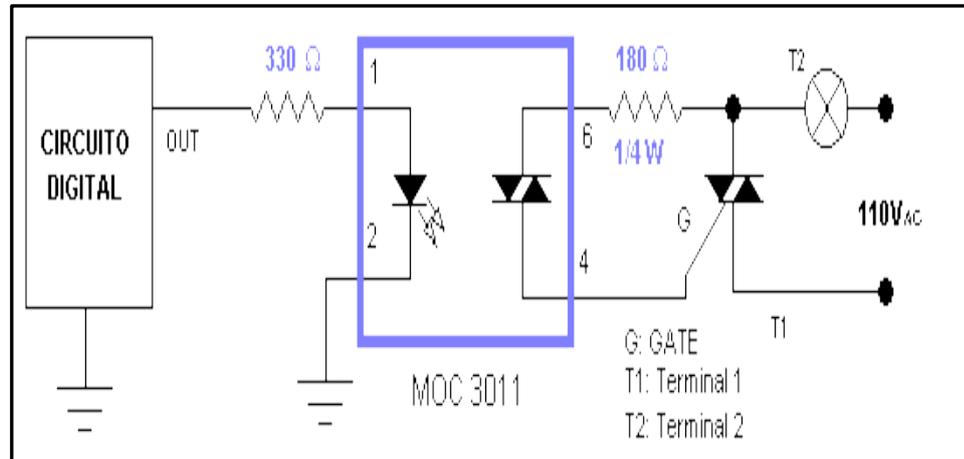


Gráfico72. Etapa de potencia para manejar el dícroico.

Fuente: MARLIN, Thomas, Process control: designing processes and control systems for dynamic performance. Alemania. 1975. Pag. 124.

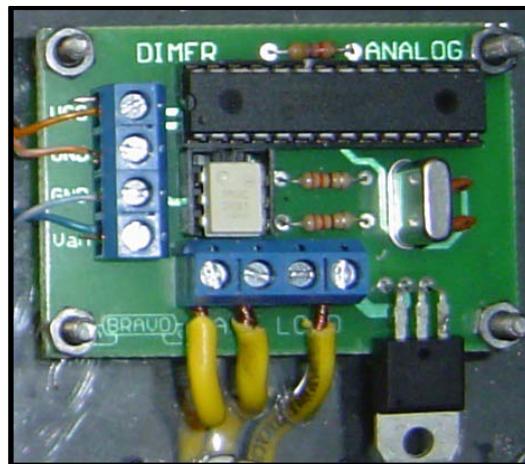


Gráfico73. Placa Física de potencia para activación de calefacción

Fuente: Autores, Laboratorio de Electrónica UDA.

El sensor de temperatura (LM35DZ), mide la magnitud a controlar, con el circuito acondicionador que ya se describió en el capítulo número uno, para conseguir la realimentación deseada.

En el laboratorio se realizó la práctica a lazo abierto del proceso con un pulso de amplitud de 5 voltios obteniendo una curva no muy marcada pero de las cuales se pudo obtener valores aproximados para saber la dinámica de la planta.

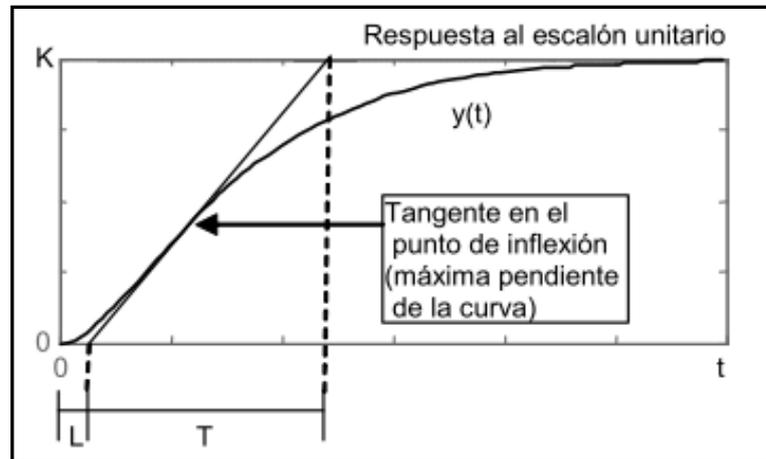


Gráfico74. Etapa de potencia para temperatura.

Fuente: BANUELOS, M.A.- MARTINEZ, W, Análisis y diseño de un controlador PID analógico. México. [s.a.], Pag 65.

Los valores obtenidos son $K=5$, $L=0.05$ segundos ya que el proceso comenzó de inmediato por lo que se puede decir tendiente a cero y $T=3$ minutos (llegando al 66% de la curva). Por lo tanto con estos valores y empleando el método de Ziegler-Nichols obtendremos los siguientes valores.

Realizando los respectivos cálculos obtenemos los siguientes valores.

TIPO	Kp	Ti	Td
P	12	0	0
PI	10,8	6	0
PID	14,4	0,025	10

Tabla 5. Tabla de valores calculados para Kp, Ki y Kd.

Fuente: Autores, Laboratorio de Electrónica UDA.

Los valores de Ti a programar serán $1/K_i$ para PI es 0.16666 y para PID es 40.

Una vez realizados el cálculo respectivo vamos a programarlos en el asistente de configuración PID del software Step7-Micro/Win mediante los siguientes pasos.

PASO 1: Estando en la pantalla principal de Step7-Micro/Win nos dirigimos a la barra de tareas para elegir la opción Herramientas y escogemos el Asistente de Operaciones

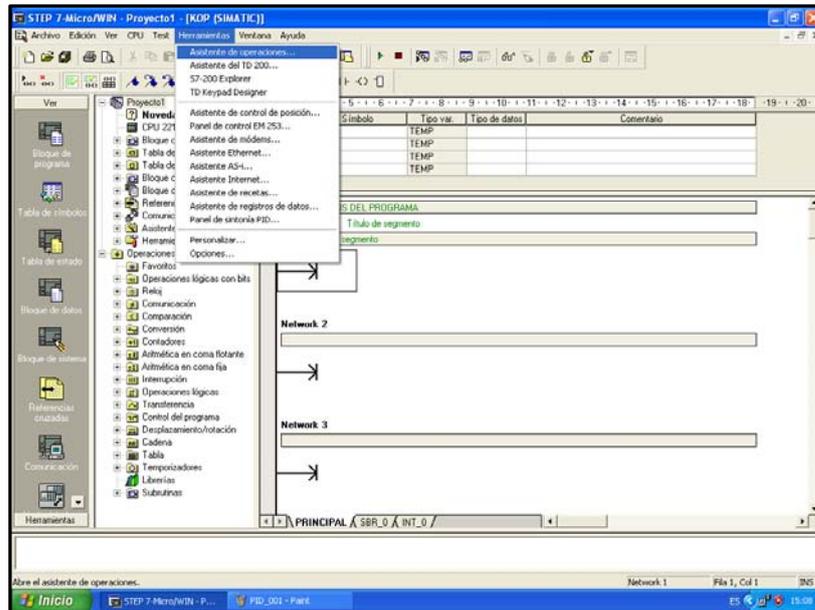


Gráfico75. Asistente de Operaciones para PID.

Fuente: Autores, Programación en el software Step7-Micro/Win.

PASO 2: En el siguiente cuadro tendremos que optar por el asistente que deseamos utilizar que en nuestro caso será PID y pulsamos Siguiente.

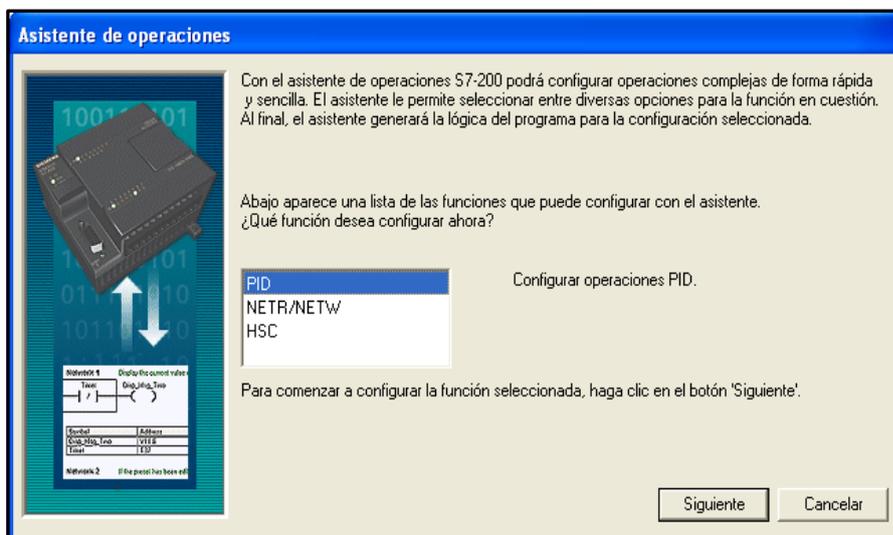


Gráfico76. Selección de Asistente PID.

Fuente: Autores, Programación en el software Step7-Micro/Win.

PASO 5: En esta ventana configuraremos las entradas y salidas con el tipo de escalamiento que puede ser unipolar o bipolar, las salida analógica o digital y los límites de la salida que marca de 0 a 32.000 pero con la variación de voltaje se debe hacer una relación para saber el valor que estamos obteniendo.

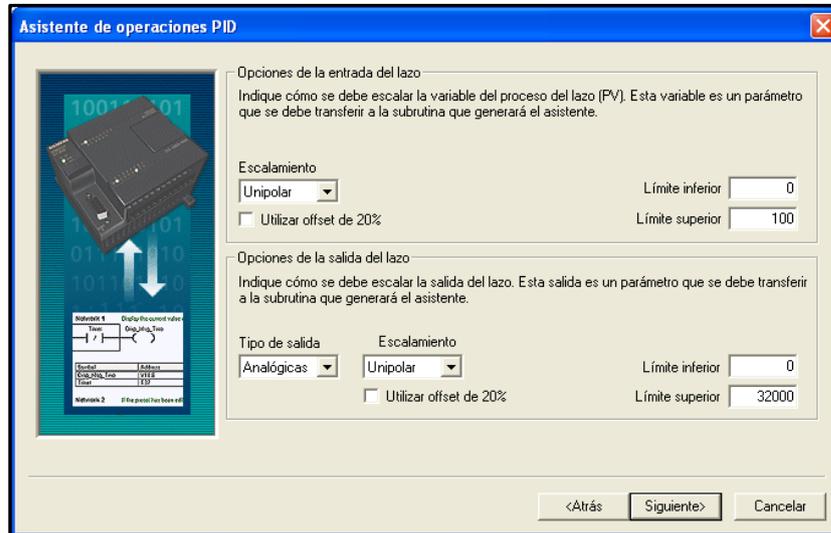


Gráfico79. Configuración de ingresos y salidas en PID.

Fuente: Autores, Programación en el software Step7-Micro/Win.

PASO 6: Ahora tenemos la configuración de las alarmas que puede ser de alta o baja prioridad dependiendo de las necesidades, además del valor que se desea que sea la alarma.

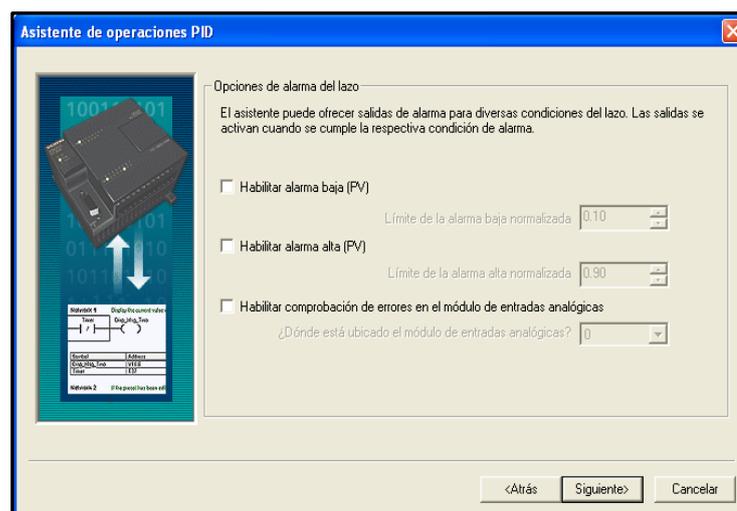


Gráfico80. Programación de alarmas del PID.

Fuente: Autores, Programación en el software Step7-Micro/Win.

PASO 7: Ahora se deberá proponer una dirección de la memoria en donde el PID efectuara los cálculos y programara el proceso pero con el cuidado que se sobre monte con algún otro segmento de programa ya que este tipo de sobre montaje no produce error en el código, si se desea el asistente puede optar por la opción Proponer Dirección.

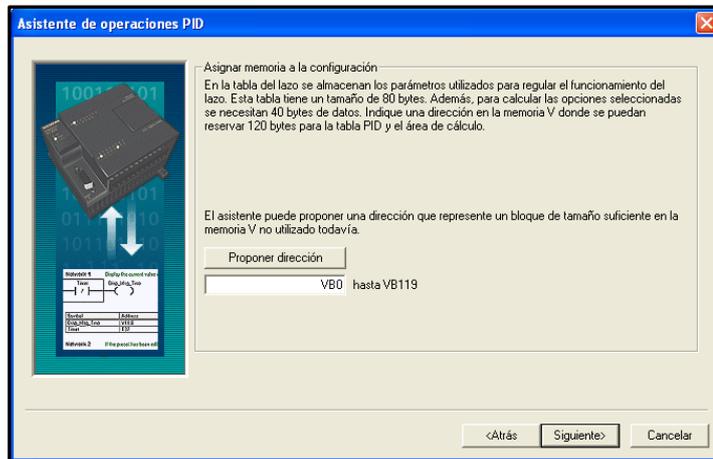


Gráfico81. Dirección de memoria para el PID.

Fuente: Autores, Programación en el software Step7-Micro/Win.

PASO 8: Seguidamente damos los nombres a las subrutinas de inicialización de interrupción que van a ser llamadas en el programa principal y pulsamos Siguiente.

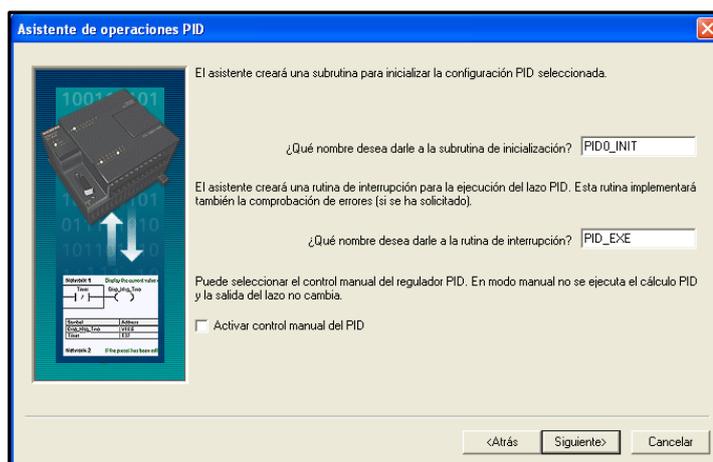


Gráfico82. Nombres a las subrutinas de PID.

Fuente: Autores, Programación en el software Step7-Micro/Win.

PASO 9: Finalizamos el Asistente con la aceptación de los parámetros antes establecidos pulsando Finalizar.

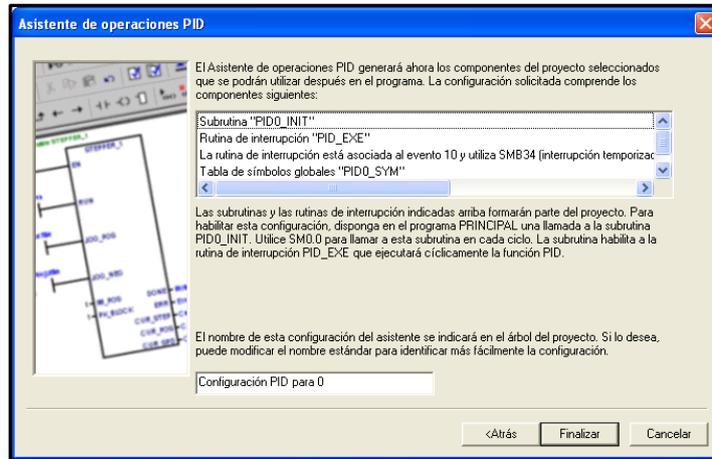


Gráfico83. Finalización del Asistente de Operaciones PID.

Fuente: Autores, Programación en el software Step7-Micro/Win.

PASO 10: Estando nuevamente en la pantalla principal de Step7-Micro/Win obtendremos las subrutinas con el nombre anteriormente configurados, por lo que podemos tenerlos en el programa principal e ingresar las variables de ingreso como de salida.

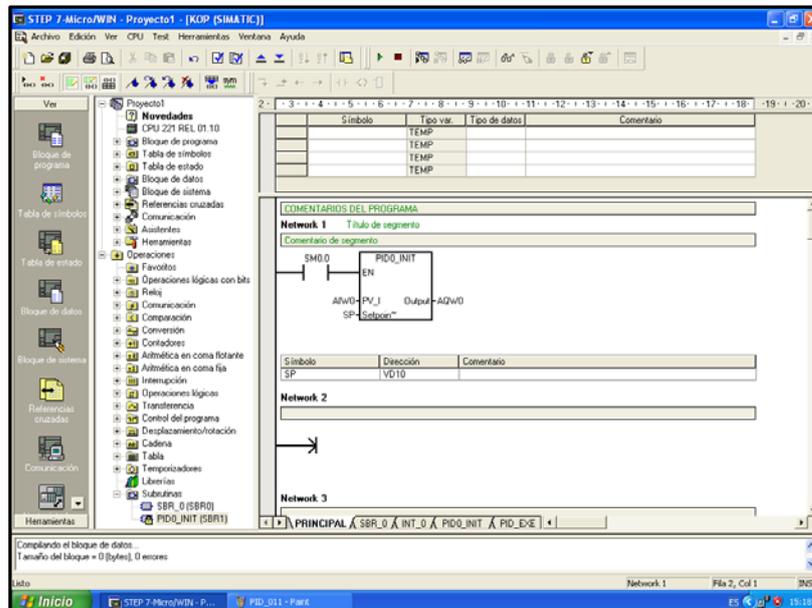


Gráfico84. Bloque de PID en programa principal.

Fuente: Autores, Programación en el software Step7-Micro/Win.

PASO 11: Para tener una visualización del proceso y sus respectivas curvas podemos ingresar en la Barra de Tareas en la opción de Herramientas luego ingresar en Panel de sintonía PID.

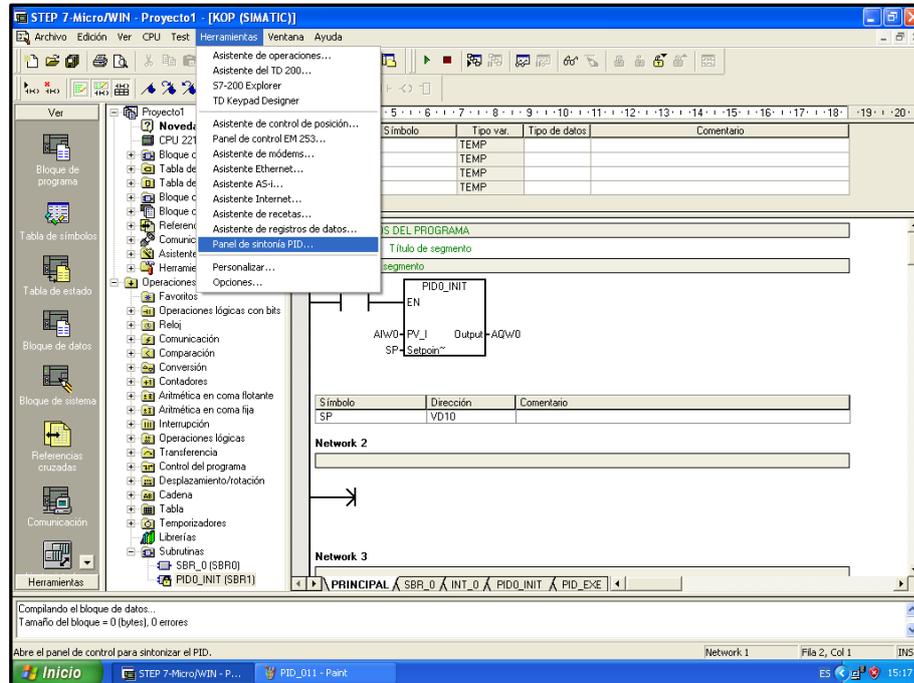


Gráfico85. Panel de sintonía PID.

Fuente: Autores, Programación en el software Step7-Micro/Win.

Una vez que se ha configurado el Asistente debemos direccionar el ingreso de la variable a controlar que en nuestro caso será la Temperatura interna del invernadero, esta variable será de tipo análogo así que para obtener estos valores debemos conectar al Auomata Programable un modulo de entrada analógica.

El modulo empleado en este trabajo monográfico será el EM235 que debe ser configurado la zona operativa de la tensión de entrada (Voltaje) o intensidad de entrada (Corriente), existen 6 microinterruptores DIP en la parte inferior del módulo EM235. Los potenciómetros "OFFSET" y "GAIN" sirven para calibrar los módulos de entradas analógicas.



Gráfico86. Configuración del modulo EM235.

Fuente: Autores, Programación en el software Step7-Micro/Win.

La programación del bloque PID a si como para la comunicación del S7-200 como Servidor está disponible en el ANEXO3.

CAPITULO 5

INSTALACION Y PRUEBAS DEL SISTEMA DE CONTROL.

El principal objetivo de este proyecto es el desarrollo de un sistema SCADA permita comprobar los conocimientos teóricos de modelado y control además de la comunicación entre dos PLC de la serie S7-200 y con el sistema de visualización del proceso a través del software Labview.



Gráfico87. Maqueta de proceso.

Fuente: Autores, Foto en laboratorio de control automático UDA.

Todo esto se desarrollo sobre una maqueta de invernadero que se divide en dos fases:

En la primera fase se desarrollara un sistema digital de llenado de tanque con niveles máximo y mínimo a través de un plc S7-200, en la maqueta este sistema está representado por dos recipientes pequeños. El tanque inferior representa la

alimentación de líquido que podría ser el sistema de agua potable en un invernadero real, con una bomba eléctrica que será la que suministre el líquido hasta que en el tanque el líquido este en el nivel máximo.

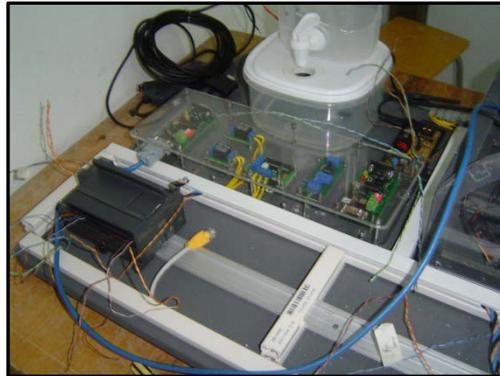


Gráfico88. Llenado de tanque en maqueta.

Fuente: Autores, Foto en laboratorio de control automático UDA.

La segunda fase consta de un invernadero adaptado con un el sensor de temperatura (LM35) que será conectado al modulo analógico EM235 para obtener los valores reales de temperatura, además de un diodo que será el que servirá para el sistema de calefacción y se adapto un ventilador en la parte posterior para tener un sistema de refrigeración pudiendo actuar sobre estos para mantener la temperatura al nivel deseado.

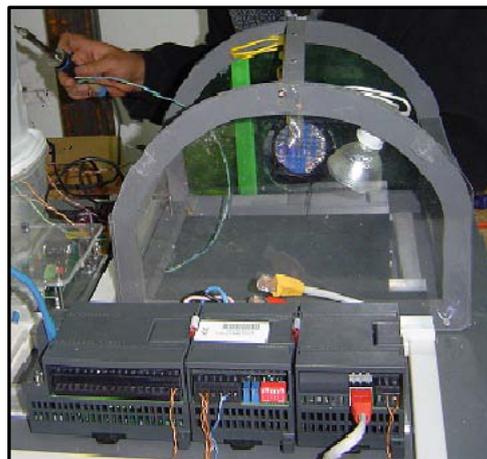


Gráfico89. Maqueta de Invernadero.

Fuente: Autores, Foto en laboratorio de control automático UDA.

Para la comunicación entre los dos PLC se incorporo un swich por lo que cada plc obtiene una dirección IP, también se acoplo dos tarjetas que convierten el protocolo RS232 a RS485 que servirán para obtener, procesar y visualizar los datos en el software Labview.



Gráfico90. Placas de comunicación y de actuadores.

Fuente: Autores, Foto en laboratorio de control automático UDA.

A la maqueta original se le han añadido los siguientes actuadores a lo largo de este proyecto ventilación forzada mediante un ventilador de corriente alterna y otro dicroico para simular la perturbación en el interior del invernadero mediante un mando manual de un swich.



Gráfico91. Dicroico que simula perturbación.

Fuente: Autores, Foto en laboratorio de control automático UDA.

5.1. Calibración del sensor de temperatura en diferentes ambientes.

El LM35 es un sensor de temperatura con una precisión calibrada de 1°C variando su salida 10 mV por cada grado centígrado, obteniendo así una tensión muy baja con relación a la tensión que necesita el plc en su entrada analógica para tomar los correctivos en el proceso por lo que se uso un amplificador operacional no inversor esto aumenta la ganancia sin invertir la señal.

Como ya se describió en el capítulo una ganancia es de 10 calculando con la siguiente formula: $V_o = V_i(1 + (R_2/R_1))$, despejando de aquí y sabiendo que la ganancia es 10 quedaría que $9R_1 = R_2$, es decir, que si R_1 vale 1k R_2 debe valer 9k. No hay ese valor exacto así que la opción fue poner un potenciómetro el mismo que ayudara que la calibración sea exacta.

Al tener un voltaje de salida del sensor que sea perceptible al modulo analógico del PLC este puede ser ubicado dentro de cualquier ambiente siempre y cuando este dentro del rango que soporte el sensor.

5.2. Ubicación del sensor de temperatura y del tanque en la maqueta.

Las ubicaciones de los sensores así como de los actuadores deben de ser estratégicas a fin de no provocar confusiones ni interferencias al momento de dar mantenimiento en una planta.

En esta maqueta el sensor de temperatura será ubicado en el centro del invernadero a fin de tener una temperatura ambiente real dentro del invernadero y con ello tratar en lo posible de disminuir el error en la medición.



Gráfico92. Sensor de temperatura en el interior del invernadero.

Fuente: Autores, Foto en laboratorio de control automático UDA.

La disposición del Tanque fue de ubicar en la parte izquierda del tablero a fin de evitar el cableado de los sensores y actuadores por la misma canaleta que contiene el cableado del invernadero y con esto asegurar el fácil reconocimiento de cada uno de los procesos dentro de la planta total.

5.3. Configuración del software.

Una vez realizado la configuración de cada uno de los software ya descritos en los capítulos dos, tres y cuatro respectivamente, podemos verificar y manipular los datos de la planta en un computador personal mediante el software Labview de la siguiente manera.

LLENADO DE TANQUE

En la pantalla del proceso se visualizara el estado de la bomba, cuando está encendida o apagada mediante un grafico de un led, cabe recalcar que esta variable no puede ser modificada ya que esta variable es programada de solo lectura.

TEMPERATURA DEL INVERNADERO

La visualización de la pantalla se complementa con un botón de control en donde se programara la temperatura a la que se desea tener, además se mostrara las graficas de la temperatura ambiente, la temperatura deseada para saber que tan distante esta la una de la otra, así como la respectiva salida del sistema calefactor.

Si existe alguna perturbación como sería un día caluroso o algún elemento externo que influya sobre la temperatura programada se encenderá en la pantalla un led indicando cuando se activo el sistema de refrigeración que para nuestro proyecto será un ventilador.

Las constantes controladoras del lazo PID también pueden ser modificadas desde la pantalla ya que son programadas como variables tanto de lectura así como de escritura lo que hace que sea un sistema dinámico.

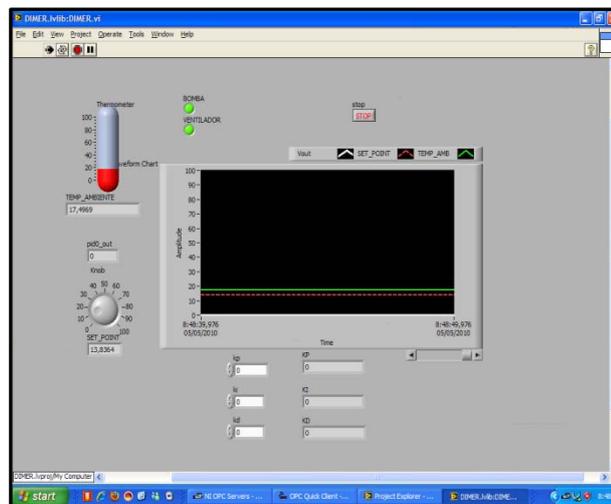


Gráfico93. Pantalla de programa principal en Labview.

Fuente: Autores, Foto en laboratorio de control automático UDA.

5.4. Verificación del correcto funcionamiento del sistema

Una vez realizado todas las acciones antes mencionadas, las prácticas se realizaron con el fin de establecer las mejores constantes que establezca el sistema de una forma rápida y con un error casi nulo.

Partiendo desde los cálculos de la Tabla 5 los valores que se proponen para el control PID son $K_p=14.4$, $T_i=1/0,025=40$, $T_d=10$ para luego ir variando hasta tener un sistema óptimo para la planta.

En un principio los parámetros de configuración del lazo PID no se alteraron para poder obtener la curva característica del sistema puro.

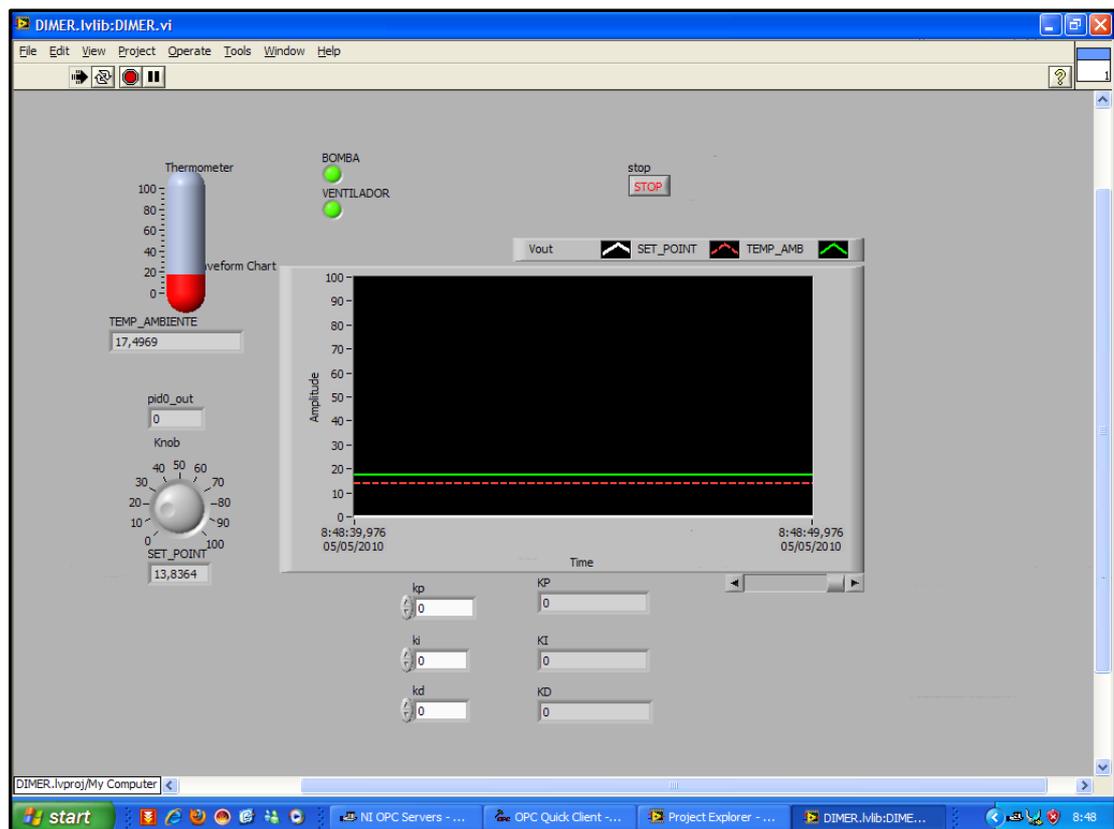


Gráfico94. Forma de onda del proceso sin parámetros.

Fuente: Autores, Foto en laboratorio de control automático UDA.

En un objetivo de tener un proceso rápido y por ende una forma de onda rápida se altero la constante K_p hasta el valor de 18, k_i a un valor de 0,1 y anulando el valor de K_d , obteniendo así un sistema oscilatorio y con error sin llegar a la referencia marcada como setpoint.

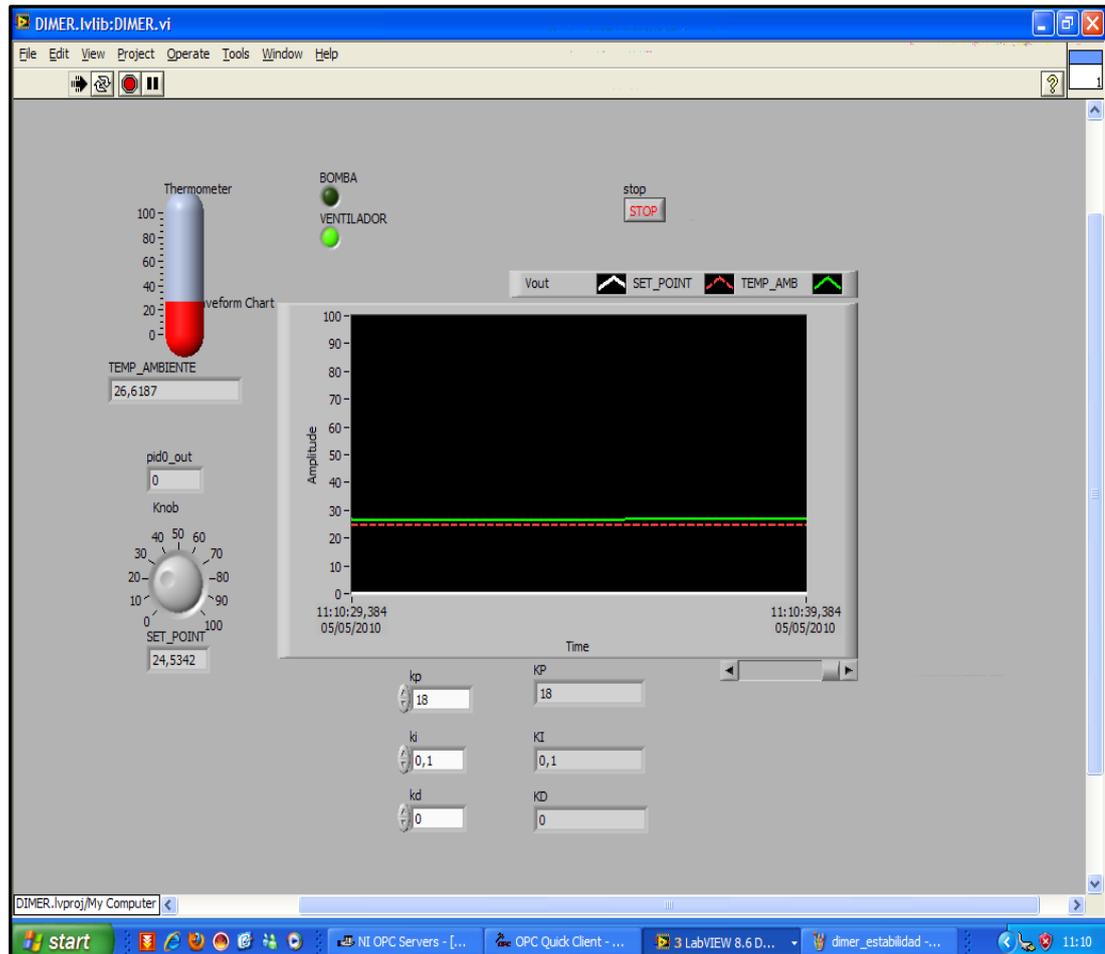


Gráfico95. Forma de onda con error.

Fuente: Autores, Foto en laboratorio de control automático UDA.

Seguidamente se altero la constante K_d hasta el valor de 3 y anulando el valor de K_i , obteniendo como resultado un sistema con error sin llegar a la referencia marcada como set point además de la activación y desactivación frecuente del actuador de la calefacción teniendo alta probabilidad de quemar el actuador.

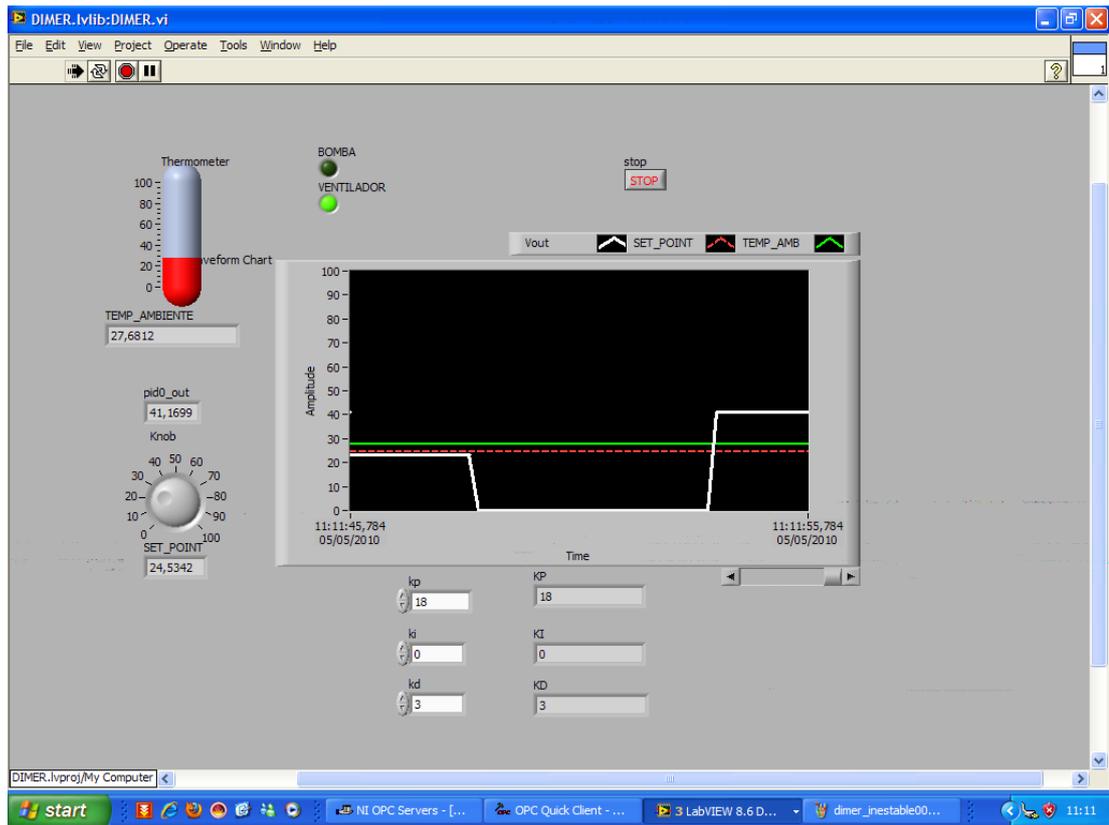


Gráfico96. Forma de onda con error y actuador oscilante.

Fuente: Autores, Foto en laboratorio de control automático UDA.

Luego de modificar todos los parámetros se obtuvo las siguientes observaciones indicadas en la Tabla7:

	Kp ALTO	Ti BAJO	Td ALTO
ESTABILIDAD	Oscilante	No muy oscilante	Estable
VELOCIDAD	Veloz	Veloz	Veloz
ERROR	Disminuye	Tiene error hasta cierto punto	Disminuye

Tabla 6. Tabla de Reacción del Sistema.

Fuente: Autores.

Con el control PI se pudo tener un sistema estable y llegando a la señal de referencia por lo que podría decir que el error es cero además que se noto que el funcionamiento del actuador de la calefacción era más estable.

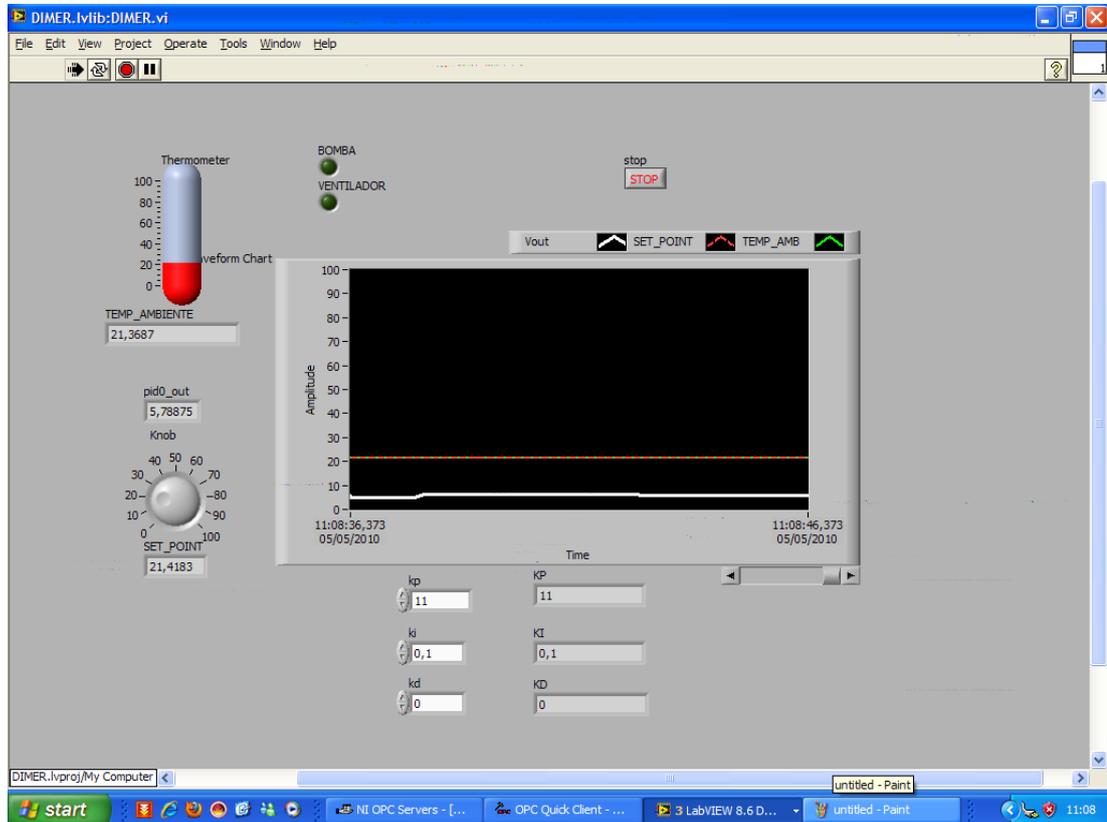


Gráfico97. Forma de onda del sistema estable y sin error.

Fuente: Autores, Foto en laboratorio de control automático UDA.

De esta forma hemos verificado el correcto funcionamiento del sistema ya que los resultados obtenidos concuerdan con los estudiados.

CONCLUSIONES.

Se ha demostrado la implantación de un sistema SCADA con el control de temperatura y de llenado de tanque. El controlador PID analógico propuesto permite observar de manera inmediata la señal producida por cada uno de los parámetros del controlador, lo que complementa las explicaciones teóricas. La etapa de potencia se asemeja a las implementadas en controladores comerciales, lo cual es una aportación adicional del sistema. El proceso de caracterización del sistema permite comparar el modelo del sistema con los resultados experimentales. El sistema cumple el objetivo de acercar al proceso a la referencia deseada mediante la definición del problema, modelado, sintonización, implementación y comprobación experimental.

Con Sistemas de esta categoría, las ganancias de las industrias pueden aumentar notablemente. A pesar de su alto costo, ayuda a reducir tiempos de proceso con lo cual aumenta la productividad. Por otra parte, el retorno de la inversión es a corto plazo.

RECOMENDACIONES

Los sistemas SCADA. Hoy en día son más frecuentes en los diferentes procesos por lo que se sugiere que este trabajo monográfico sea una base de referencia para tener claro el sistema de comunicación y visualización dentro del proceso.

Para mayor exactitud se obtener los valores generados por el proceso en una memoria dentro del plc, ya que se podría generar una base de datos dentro del ordenador pero esta no sería tan segura si este ordenador en algún momento se llegara a colapsar, dejando sin información del proceso o el error ocurrido.

Dentro de este trabajo monográfico en su parte práctica hemos descrito la posibilidad de conectar varios módulos de extensión al PLC como por ejemplo modulo analógico EM235 y el modulo de comunicación, por lo que se sugiere colocar primero el modulo de comunicación y luego el modulo analógico, ya que de hacerlo en forma contraria se producirá un error de lectura de módulos en el PLC.

BIBLIOGRAFIA

Referencias Bibliográficas.

- [1] ASTROM, Karl - HAGGLUND, Tore, PID controllers: theory, design, and tuning, Second edition, USA, Instrument Society of America. 1995.
- [2] BANUELOS, M.A - MARTINEZ, W - PEREZ, J.L, Análisis y diseño de un controlador PID analógico, Primera Edición, UNAM-México, Editorial Centro de Instrumentos, [s.a.].
- [3] CORRIPIO, Armando, Tuning of industrial control systems, USA, Instrument Society of America, 1990.
- [4] HERRERA, Luis, Lecciones Sobre Sistemas SCADA – HMI. Argentina, Presentaciones del curso de POSTGRADO – Especialización en control automático y automatización, 2008.
- [5] MARLIN, Thomas, Process control: designing processes and control systems for dynamic performance.USA, First edition, Mc Graw-Hill. 1975.
- [6] MC MILLAN, Gregory, Tuning and control loop performance: a practitioners' guide, USA, Third edition. Instrument Society of America. 1994.
- [7] NATIONAL INSTRUMENT, LabVIEW User Manual 7, USA, 2003.
- [8] RAYO, G - CASTILLO, J - PEREZ, J, Control de temperatura de un horno industrial de secado utilizando un PID analógico, Artículo, México, 1999.
- [9] ROOTS, William, Fundamentals of temperature control, First edition, Australia, Academic Press. 1969.
- [10] SMITH, Carlos - CORRIPIO, Armando, Principles and practice of automatic process control, Second edition. USA, John Wiley & Sons. 1997.

Referencias Electrónicas.

[1] FORCE. OPC Overview. <http://www.kisters.de/spanish/>. 1996. [consulta 7 de diciembre de 2009]

[2] NATIONAL INSTRUMENTS. Scada construido con Labview [en línea] <http://www.ni.com>. 2003. [consulta 18 de abril de 2009].

[3] SIEMENS. Manual de Instrucciones Simatic s7-200. <http://www.siemens.com>. 1995. [consulta 13 de julio de 2009]

[4] SIEMENS. El controlador PID. <http://www.od.siemens.de>. 1998. [consulta 18 de junio de 2009]

[5] SIEMENS. Manual de modulo EM235 s7200 analógico. <http://www.librostecnicos.com.ar>. 2001. [consulta 20 de junio de 2009]

[6] SIEMENS. Tutorial de Ethernet en Simatic S7-200. http://www.ambientum.com/catalegs/100000999/00181_p3a.htm. 2002. [consulta 3 de febrero de 2010]

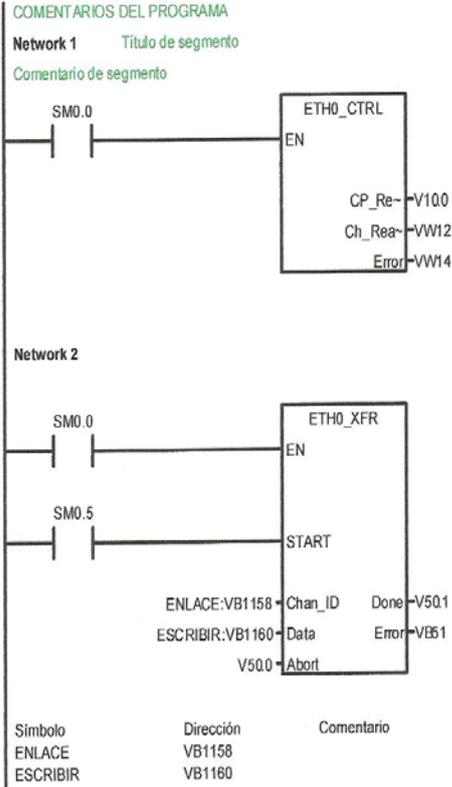
ANEXO1

Programa de llenado de tanque con la configuración de un cliente.

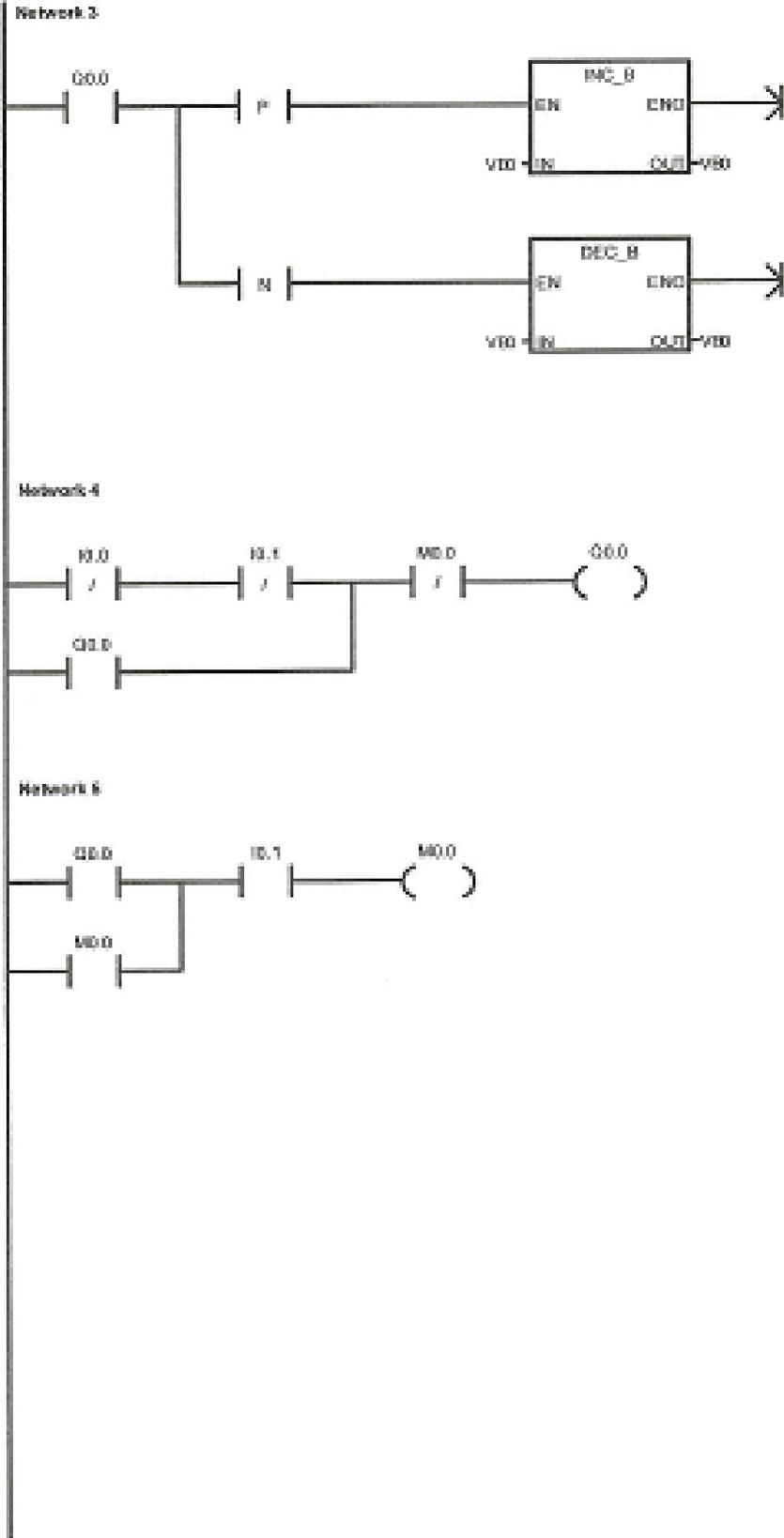
CLIENTE / PRINCIPAL (OB1)

Bloque PRINCIPAL
 Autor:
 Fecha de creación: 13.04.2010 10:39:40
 Fecha de modificación: 30.04.2010 21:32:47

Símbolo	Tipo var.	Tipo de datos	Comentario
	TEMP		



CLIENTE / PRINCIPAL (GB1)



ANEXO2

Variables de comunicación entre el PLC S7-200 y Labview.

OPC Quick Client - Sin título *

File Edit View Tools Help

Item ID	Data Type	Value	Timestamp	Quality	Update Count
SERVERIDOR.57-200.BOMEA	Byte	0	18:10:09:206	Good	1
SERVERIDOR.57-200.ID	Float	0	18:10:09:136	Good	1
SERVERIDOR.57-200.KI	Float	0.1	18:10:09:136	Good	1
SERVERIDOR.57-200.KP	Float	11	18:10:09:136	Good	1
SERVERIDOR.57-200.pd_out	Float	0	18:10:09:136	Good	1
SERVERIDOR.57-200.SET_POHIT	Float	0.2045	18:10:09:437	Good	1
SERVERIDOR.57-200.TEMP_AMBIENTE	Float	0.210625	18:10:38:098	Good	65
SERVERIDOR.57-200.VALOR_SET_POHIT	Float	0.2045	18:10:09:136	Good	1
SERVERIDOR.57-200.VENTILADOR	Byte	1	18:10:09:206	Good	1

Date	Time	Event
07/05/2010	18:10:09	Added group 'SERVI...
07/05/2010	18:10:09	Added group 'SERVI...
07/05/2010	18:10:09	Added 11 items to q...
07/05/2010	18:10:09	Added group 'SERVI...
07/05/2010	18:10:09	Added 16 items to g...
07/05/2010	18:10:09	Added 5 items to q...

Ready

Item Count: 54

ANEXO3

Programación PID y S7-200 como Servidor.

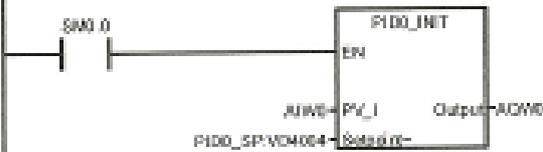
SENYDOP / PRINCIPAL (381)

Diagrama: PRINCIPAL
 Autor:
 Fecha de creación: 13.04.2010 10:07:17
 Fecha de modificación: 01.05.2010 15:44:52

Simbolo	Tipo var.	Tipo de datos	Comentario
	TEMP		

COMENTARIOS DEL PROGRAMA

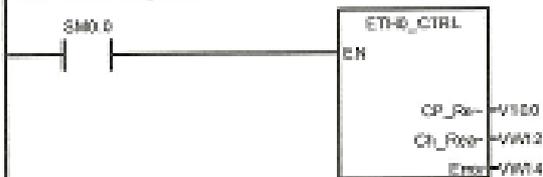
Network 1



Simbolo	Dirección	Comentario
PID0_SP	VD4004	Consigna de proceso normalizada

Network 2

Título de segmento
Comentario de segmento



Network 3



SERVIDOR / PRINCIPAL (SBI)

