



## **ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL**

### **Facultad de Ingeniería en Electricidad y Computación**

**“DISEÑO E IMPLEMENTACIÓN DE UN CONTROL DE PLANTA DE NIVEL GUNT RT450 DEL LABORATORIO DE INSTRUMENTACIÓN INDUSTRIAL UTILIZANDO UN PLC SIMATIC S7 1200 CON AUTOMATIZACIÓN TOTAL INTEGRADA (TIA PORTAL) E IMPLEMENTACIÓN DE UNA INTERFAZ PARA USUARIO DISEÑADA EN LABVIEW CON COMUNICACIÓN MEDIANTE OPC”**

### **INFORME DE PROYECTO DE GRADUACIÓN**

Previo a la obtención del Título de:

**INGENIERO EN ELECTRICIDAD ESPECIALIZACIÓN ELECTRÓNICA Y AUTOMATIZACIÓN INDUSTRIAL**

Presentado por:

**Lenin Jair Barreiro Chévez**

**Jorge Manuel Molina Peñaranda**

Guayaquil – Ecuador

2015

## **AGRADECIMIENTO**

A Dios, El Todopoderoso por brindarme la energía y capacidad necesarias para terminar ésta etapa de mi vida.

A mi familia por el apoyo recibido durante esta etapa, por estar ahí siempre que necesité de una mano amiga.

A mi esposa, Verónica, por estar a mi lado siempre en este camino para alcanzar el título profesional tan ansiado.

A los profesores, pilares fundamentales en este largo proceso de aprendizaje.

A mi compañero Jorge por la gran dedicación empleada en la elaboración de este informe de proyecto.

**Lenin Jair Barreiro Chévez**

A Dios, por darme las señales necesarias de que culminar este camino era posible.

A mi familia en especial a mi esposa, María del Cisne, compañera y motivadora permanente, por su voz de aliento en los momentos difíciles, a mis hijos María Fiorella, Alejandra Estefanía y Jorge Eduardo por su amor. A mi madre Grimanesa por sus oraciones y creer siempre que esto era posible, A mi padre Milton por su apoyo y paciencia. A mis hermanos Fabricio y Claudia por el cariño y estar ahí siempre. A mis amigos y compañeros.

A mis profesores en especial al MSc. Holger Cevallos Ulloa por su constante apoyo y guía. A mi amigo y compañero Lenin por su disponibilidad para trabajar en equipo con eficiencia y responsabilidad.

**Jorge Manuel Molina Peñaranda**

## **DEDICATORIA**

Este trabajo se lo dedico a mis padres que siempre estuvieron apoyándome en todo momento y a mi familia, que son la razón por la cual he logrado llegar a este punto en mi vida.

**Lenin Jair Barreiro Chévez**

Este trabajo se lo dedico a esos seres maravillosos que llenan mi vida de alegría mi esposa e hijos.

**Jorge Manuel Molina Peñaranda**

# TRIBUNAL DE SUSTENTACIÓN

---

**MSc. SARA RÍOS**

SUBDECANA FIEC

---

**MSc. HOLGER CEVALLOS U.**

DIRECTOR DEL PROYECTO DE  
GRADUACIÓN

---

**MSc. CARLOS SALAZAR L.**

MIEMBRO DEL TRIBUNAL

## **DECLARACIÓN EXPRESA**

“La responsabilidad por el contenido de este informe de proyecto de graduación, nos corresponden exclusivamente; y el patrimonio intelectual de la misma, a la ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL”.

(Reglamento de Graduación de la ESPOL)

---

Lenin Jair Barreiro Chévez

---

Jorge Manuel Molina Peñaranda

## RESUMEN

El presente proyecto consiste en la implementación de un control para la Planta de Nivel GUNT RT 450, primero se realiza una descripción detallada de los elementos que se requieren para este proyecto, de hardware y software, haciendo un repaso por las instrucciones de programación en Labview y TIA PORTAL que se utilizarán en el desarrollo del proyecto. Luego se detalla la estrategia de control a seguir, teniendo 2 opciones Manual y Automático, se realiza la programación en TIA PORTAL, siendo cuidadosos en el manejo del tipo de variables a fin de no tener dificultades en la compilación de la aplicación que se carga al PLC implementado para ello un proceso de estandarización de las variables analógicas utilizando las funciones que TIA PORTAL ofrece para este fin.

Se utiliza el bloque PID\_COMPACT, eje central de nuestro control en modo automático, aprovechando su función de autotuning, para calcular los parámetros PID.

Se procede a diseñar el VI en Labview que nos servirá para interactuar con la aplicación de TIA PORTAL cargada en el PLC, se trata de replicar al mínimo detalle el funcionamiento real de la Planta de Nivel a fin de que el usuario utilice esta aplicación en Labview para un manejo remoto del sistema.

Se aprovecha la facilidad que provee NI OPC Servers para establecer comunicación entre una aplicación creada en Labview con cualquier PLC en este caso con SIMATIC S7-1200.

Posteriormente se somete el sistema a diversas pruebas de comunicación y desempeño utilizando para ello parámetros de sistemas de control en el dominio del tiempo, para diversas condiciones de funcionamiento, a fin de verificar que el control diseñado sea confiable y su comunicación estable.

Finalmente se diseñan prácticas para el Laboratorio de Instrumentación Industrial, aplicando los procedimientos desarrollados en el proyecto.



# ÍNDICE GENERAL

AGRADECIMIENTO .....	I
DEDICATORIA .....	III
TRIBUNAL DE SUSTENTACIÓN.....	IV
DECLARACIÓN EXPRESA .....	V
RESUMEN.....	VI
ÍNDICE GENERAL .....	VIII
ABREVIATURAS .....	XII
SIMBOLOGÍA .....	XIII
ÍNDICE DE FIGURAS.....	XV
ÍNDICE DE TABLAS .....	XXV
INTRODUCCIÓN.....	XXVI
CAPÍTULO 1 .....	1
1.1 DESCRIPCIÓN DEL PROBLEMA.....	1
1.2 OBJETIVOS.....	2
1.3 SOLUCIÓN PROPUESTA Y JUSTIFICACIÓN DE LA SOLUCIÓN .....	4
1.4 METODOLOGÍA.....	5
CAPÍTULO 2 .....	8
2.1 DESCRIPCIÓN DE LA PLANTA DE NIVEL GUNT HAMBURG RT450 .....	8
2.2 DESCRIPCIÓN DEL PLC SIMATIC S7 1200 SIEMENS .....	17

2.3 DESCRIPCIÓN DEL SOFTWARE LABVIEW NATIONAL INSTRUMENTS.	23
2.4 DESCRIPCIÓN DEL SOFTWARE TIA PORTAL	28
CAPÍTULO 3	31
3.1 MODO DE FUNCIONAMIENTO MANUAL	34
3.2 MODO DE FUNCIONAMIENTO AUTOMÁTICO	44
CAPÍTULO 4	55
4.1 CREACIÓN, CONFIGURACIÓN Y EDICIÓN DE VARIABLES.	55
4.2 DISEÑO DE VI PARA MONITOREO Y AJUSTE DE PARÁMETROS	65
4.3 EDICIÓN DE NOTAS EXPLICATIVAS SOBRE ELEMENTOS DEL INTERFAZ PARA USUARIO.	68
CAPÍTULO 5	71
5.1 ARQUITECTURA CLIENTE SERVIDOR EN OPC	71
5.2 CREACIÓN DE CANAL DE COMUNICACIÓN.	76
5.3 SELECCIÓN DE TIPO DE COMUNICACIÓN	77
5.4 CONFIGURACIÓN DEL CLIENTE OPC	84
5.5 AJUSTE GENERAL DE PARÁMETROS DE COMUNICACIÓN.	93
CAPÍTULO 6	99
6.1 DISEÑO DE PRUEBAS.	99
6.2 PRUEBAS Y COMPARACIÓN DE RESULTADOS OBTENIDOS	102
6.3 PUESTA EN MARCHA CON LOS AJUSTES ÓPTIMOS DE CONTROL	
	117
CAPÍTULO 7	120
7.1 PRÁCTICA # 1: INTRODUCCION A TIA PORTAL.	121

7.1.1	CREACIÓN DE UN PROYECTO.....	123
7.1.2	MENUS DE AYUDA DE SOFTWARE Y CONFIGURACION DE VISTA 127	
7.1.3	CONFIGURACIÓN DE HARDWARE Y RED.....	130
7.2	PRÁCTICA # 2: PROGRAMACION BASICA DE PLC S7 1200.....	136
7.2.1	MANEJO DE VARIABLES: DIRECCIONAMIENTO Y CONFIGURACION.....	137
7.2.2	BLOQUES: TIPO Y PROGRAMACION .....	143
7.2.3	PROGRAMACIÓN CON CONTACTOS Y BOBINAS .....	147
7.2.4	DEFINICIÓN DE VARIABLES Y TABLAS DE SÍMBOLOS.....	163
7.2.5	DEPURACIÓN DEL PROYECTO Y COMPILACIÓN .....	168
7.3	PRÁCTICA #3 DISEÑO DE INTERFAZ PARA USUARIO EN LABVIEW PARA PLANTA DE NIVEL GUNT RT450 .....	171
7.3.1	VISUALIZACIÓN DE VARIABLES BOOLEANAS Y ANÁLOGAS. ....	172
7.3.2	VISUALIZACIÓN DE ALARMAS.....	181
7.3.3	EDICIÓN DE TEXTOS EXPLICATIVOS SOBRE EL PROCESO .....	183
7.4	PRÁCTICA # 4: CONFIGURACIÓN CLIENTE/SERVIDOR OPC.....	185
7.4.1	CONFIGURACIÓN DE CLIENTE OPC.....	186
7.4.2	CONFIGURACIÓN DE SERVIDOR OPC .....	192
7.5	PRÁCTICA # 5: CONTROL CON INTERFAZ PARA USUARIO DE PLANTA DE NIVEL GUNT RT 450.....	203
7.5.1	CONTROL MANUAL .....	212
7.5.2	CONTROL PID.....	215
	CONCLUSIONES Y RECOMENDACIONES .....	217

ANEXOS.....	220
BIBLIOGRAFÍA.....	224

## ABREVIATURAS

AC	Alternating Current
CPU	Central Processing Unit
DC	Direct Current
DSC	Datalogging and Supervisory Control
HMI	Human Machine Interface
I/O	Input/Output
NI	National Instruments
OLE	Object linking and embedding
PID	Proporcional Integral Derivativo
PLC	Programable Logic Controller
SCADA	Supervisory Control and Data Adquisition
TCP/IP	Transmision Control Protocol Internet Protocol
TIA PORTAL	Totally Integrated Automation Portal
VI	Virtual Instrument

## SIMBOLOGÍA

$\mu\text{F}$	microfaradios
kW	kilovatios
P&ID	Piping instrumentation diagram
mA	miliamperios
cm	centímetros
V	voltios
$\text{m}^3/\text{hora}$	Metros cúbicos por hora
$p_e$	Presión de aire proporcional
$p_s$	Presión de mando
$\text{k}\Omega$	Kilo-ohmios
kHz	Kilo-hertz
Hz	hertz
%AQ	Salida analógica
%AI	Entrada analógica
$K_p$	Ganancia proporcional
$T_I$	Tiempo de acción integral
$T_D$	Tiempo de acción derivativa
a	Coefficiente de retardo derivativo
b	Ponderación de la acción proporcional

$%I$	Entrada digital de PLC
$%Q$	Salida digital de PLC
$y$	Valor de consigna
$x$	Valor de proceso
$t_d$	Tiempo de retardo
$t_r$	Tiempo de levantamiento
$t_p$	Tiempo pico
$M_p$	Sobrepaso máximo

## ÍNDICE DE FIGURAS

Figura 1.1 Elementos que intervienen en el proyecto.....	3
Figura 2.1 Módulo básico RT 450 [1] .....	9
Figura 2.2 Módulo para regulación de nivel RT 450.01 [3] .....	10
Figura 2.3 Diagrama P&ID de la Planta de Nivel GUNT RT 450 [3] .....	12
Figura 2.4 Sensor de nivel RT 450.35 Elobau 212 KK0.5 [7] .....	13
Figura 2.5 Señal de salida para sensor de nivel RT 450.35 Elobau 212 KK0.5 [7] ..	14
Figura 2.6 Válvula RT 450.20 SAMSON con servo accionamiento neumático y regulador de posición i/p SAMSON 3760 .....	15
Figura 2.7 Esquema de funcionamiento del posicionador SAMSON 3760 [8] .....	16
Figura 2.8 CPU 1212 AC/DC/relé y SM1234 ensamblados.....	19
Figura 2.9 Diagrama de conexiones para CPU 1212C AC/DC/relé /relé [4] .....	20
Figura 2.10 Diagrama de conexiones para módulo SM 1234 [4].....	22
Figura 2.11 Interfaz diseñado en Labview (Front panel).....	24
Figura 2.12. Programación en Labview (Block diagram) .....	24
Figura 2.13 Vista de configuración de dispositivos en TIA PORTAL .....	29
Figura 3.1 Selector de fuente de consigna y dial en Labview .....	32
Figura 3.2 Potenciómetro de precisión 10 K $\Omega$ 10 vueltas.....	32
Figura 3.3 Diagrama de flujo selección modo de funcionamiento .....	35
Figura 3.4 Programación en TIA PORTAL de selección modo de operación .....	35
Figura 3.5 Diagrama de Flujo de limitación, normalización y escalado de consigna de potenciómetro .....	37
Figura 3.6 Programación en TIA PORTAL de limitación, normalización y escalado de consigna de potenciómetro .....	38



Figura 3.7 Diagrama de Flujo de selección de Fuente de Consigna y Manejo de la Salida Analógica en Modo Manual.....	39
Figura 3.8 Programación en TIA PORTAL de selección de Fuente de Consigna y Manejo de la Salida Analógica en Modo Manual.....	40
Figura 3.9 Programación en TIA PORTAL de la Salida Analógica en Modo Manual	41
Figura 3.10 Diagrama de Flujo para normalización y escalado de la salida del Bloque PID_COMPACT para Manual y Automático .....	42
Figura 3.11 Programación en TIA PORTAL para normalización y escalado de la salida del Bloque PID_COMPACT para Manual y Automático.....	43
Figura 3.12 Visualización en Labview de %MD26-Escalado Salida Real Válvula ...	43
Figura 3.13 Segmento de la Interfaz de Labview para el ingreso del Setpoint .....	45
Figura 3.14 Programación en TIA PORTAL de la Salida Analógica en Modo Automático.....	45
Figura 3.15 Segmento de la Interfaz de Labview para el ingreso del Setpoint. ....	46
Figura 3.16 Programación en TIA PORTAL para Arranque y Paro de la aplicación desde Planta de Nivel y Labview .....	49
Figura 3.17 Selectores, pulsadores y luces piloto de la Planta de Nivel .....	49
Figura 3.18 Controles para Arranque y Paro desde aplicación en Labview .....	50
Figura 3.19 Diagrama de Flujo de arranque de bomba desde Labview o desde Planta .....	50
Figura 3.20 Programación en TIA PORTAL de arranque de bomba desde Labview o desde Planta.....	51
Figura 3.21 Diagrama de Flujo de condiciones de apagado y alarma para bomba LOWARA .....	53

Figura 4.1 Ruta para seleccionar la ventana de Image Navigator .....	56
Figura 4.22 Imagen sólo de la planta. ....	57
Figura 4.3 Selector para modo manual y automático .....	58
Figura 4.4 Selector para fuente de consigna, dial para apertura de válvula desde Labview e indicador de modo manual para el dial.....	59
Figura 4.5 Indicador para porcentaje de apertura de válvula.....	60
Figura 4.66 Control numérico y barra indicadora para valor de Setpoint. ....	62
Figura 4.7 Run, Stop y Reset para el funcionamiento de la bomba. ....	63
Figura 4.8 Run, Stop, Llave, representación del PLC y Stop del lazo while. ....	64
Figura 4.9 Indicador de Nivel Máximo.....	65
Figura 4.10 Gráfica SETPOINT .....	66
Figura 4.11 Gráfica ERROR.....	67
Figura 4.12 Gráfica APERTURA .....	67
Figura 4.13 Programación para el texto explicativo de válvula Samson .....	69
Figura 4.14 Interfaz de Labview completa.....	70
Figura 5.1 Arquitectura Cliente-Servidor OPC.....	76
Figura 5.2 Asignación de nombre de canal de comunicación.....	77
Figura 5.3 Asignación de tipo de comunicación. ....	78
Figura 5.4 Selección del adaptador de red.....	79
Figura 5.5 Optimización de Escritura .....	80
Figura 5.6 Manejo de números no-normalizados de Punto Flotante .....	81
Figura 5.7 Resumen de configuración del tipo de comunicación para el Canal de comunicación Siemens. ....	82

Figura 5.8 Creación del Dispositivo con el cual nos vamos a comunicar (Fuente de Datos).....	83
Figura 5.9 Selección del Dispositivo con el cual nos vamos a comunicar (Fuente de Datos).....	84
Figura 5.10 Creación de variables (Tags) que trabajan entre Labview y TIA PORTAL .....	85
Figura 5.11 Creación de variable tipo Float.....	86
Figura 5.12 Todas las variables Float creadas. ....	86
Figura 5.13 Todas las variables Float y Boolean creadas. ....	87
Figura 5.14 Creación del I/O Server.....	88
Figura 5.15 Creación del Cliente OPC .....	89
Figura 5.16 Creación de Proyecto.lvproj y librería PlantaNivel.lvlib .....	90
Figura 5.17 Creación de Variables Bound (ligadas) .....	91
Figura 5.18 Ubicación de las variables del NI OPC Server .....	91
Figura 5.19 Finalización del proceso de Variables Bound (ligadas).....	92
Figura 5.20 Todas las variables del Cliente de Datos creadas .....	93
Figura 5.21 Asignación de Identificación de dispositivo (Device ID) .....	94
Figura 5.22 Definición del Scan Mode para el Dispositivo (PLC).....	94
Figura 5.23 Configuración de tiempos máximos de comunicación .....	95
Figura 5.24 Habilitación de opción de Auto-Demotion.....	96
Figura 5.25 Selección del número de puerto TCP/IP .....	96
Figura 5.26 Definición de la ubicación del CPU, rack y slot .....	97
Figura 5.27 Definición del orden para el almacenamiento de bits .....	98
Figura 5.28 Resumen de los parámetros de comunicación del S7-1200.....	98

Figura 6.1 Curva de respuesta al escalón unitario [1] .....	100
Figura 6.2 Interfaz en Labview para un Setpoint : 20 cm .....	103
Figura 6.3 OPC Quick Client para Setpoint : 20 cm .....	103
Figura 6.4 Respuesta del sistema al Setpoint de 20 cm.....	104
Figura 6.5 Interfaz en Labview para un Setpoint : 25 cm .....	105
Figura 6.6 OPC Quick Client para Setpoint : 25 cm .....	106
Figura 6.7 Respuesta del sistema al Setpoint de 25 cm.....	106
Figura 6.8 Interfaz en Labview para un Setpoint : 30 cm .....	107
Figura 6.9 OPC Quick Client para Setpoint : 30 cm .....	108
Figura 6.10 Respuesta del sistema al Setpoint de 30 cm.....	108
Figura 6.11 Interfaz en Labview para un Setpoint : 35 cm.....	109
Figura 6.12 OPC Quick Client para Setpoint : 35 cm .....	110
Figura 6.13 Respuesta del sistema al Setpoint de 35 cm.....	110
Figura 6.14 Interfaz en Labview para un Setpoint : 40 cm .....	111
Figura 6.15 OPC Quick Client para Setpoint : 40 cm .....	112
Figura 6.16 Respuesta del sistema al Setpoint de 40 cm.....	112
Figura 6.17 Interfaz en Labview para un Setpoint : 44 cm .....	114
Figura 6.18 OPC Quick Client para Setpoint : 44 cm .....	114
Figura 6.19 Respuesta del sistema al Setpoint de 44 cm.....	115
Figura 6.20 Puesta en marcha para Setpoint : 25 cm .....	118
Figura 6.21 Valores PID con los que se trabajó en el proyecto. ....	119
Figura 7.1 Comunicación entre dispositivos SIEMENS con TIA PORTAL .....	122
Figura 7.2 Vista del Portal.....	123
Figura 7.3 Creando el proyecto.....	124

Figura 7.4 Configuración de dispositivo .....	125
Figura 7.5 Mostrar los dispositivos.....	125
Figura 7.6 Agregar el dispositivo.....	126
Figura 7.7 Vista del proyecto. ....	127
Figura 7.8 Opciones de ayuda.....	128
Figura 7.9 Índice de ayuda ordenada por temas.....	129
Figura 7.10 Opción Buscar .....	129
Figura 7.11 Vista favoritos.....	130
Figura 7.12 Editor de hardware y redes .....	131
Figura 7.13 Configuración de dispositivos.....	133
Figura 7.14 Catálogo de hardware.....	134
Figura 7.15 PLC y Módulo .....	137
Figura 7.16 Número SINT.....	139
Figura 7.17 Número INT .....	140
Figura 7.18 Dato real .....	141
Figura 7.19 Componentes número Lreal.....	142
Figura 7.20 Programación lineal [4] .....	144
Figura 7.2121 Programación estructurada [4].....	146
Figura 7.22 Arbol de proyecto .....	148
Figura 7.23 Bloque de programación Main. ....	148
Figura 7.24 Barra de Favoritos en bloque MAIN .....	149
Figura 7.25 Contactos y Bobinas en MAIN .....	150
Figura 7.26 Entradas digitales Run Bomba y Stop Bomba.....	152
Figura 7.27 Salida Bomba Encendida.....	152

Figura 7.28 Diagrama de flujo encender y apagar bomba.....	154
Figura 7.29 Programación encender y apagar bomba .....	154
Figura 7.30 Diagrama de flujo de Segmento 2, normalizado y escalado de la variable de nivel.....	156
Figura 7.31 Programación de Segmento 2, normalizado y escalado de la variable de nivel.....	157
Figura 7.32 Bloque NORM_X .....	158
Figura 7.33 Bloque SCALE_X.....	159
Figura 7.34 Diagrama de flujo del segmento 3, Alarma del Sistema .....	159
Figura 7.35 Programación del segmento 3, Alarma del Sistema .....	160
Figura 7.36 Diagrama de flujo del segmento 4, Condición Reset de Alarma .....	161
Figura 7.37. Programación del segmento 4, Condición Reset de Alarma.....	162
Figura 7.38 Diagrama de flujo del segmento 5, Salida apertura de la válvula al 100% .....	162
Figura 7.39 Programación del flujo del segmento 5, Salida apertura de la válvula al 100% .....	163
Figura 7.40 Definir variable .....	164
Figura 7.41 Definición de características de variable. ....	165
Figura 7.42 Asignación de direcciones a todas las variables.....	166
Figura 7.43 Acceso a tabla de variables estándar .....	167
Figura 7.44 Tabla de variables estándar .....	167
Figura 7.45 Variables en representación gráfica de PLC y Módulo SM 1234 .....	168
Figura 7.46 Compilación de programa en PLC .....	169
Figura 7.47 Compilación exitosa del programa del PLC.....	169

Figura 7.48 Carga del programa en el PLC.....	170
Figura 7.49 Creación de un proyecto en blanco de Labview.....	172
Figura 7.50 Creación de Lazo While.....	173
Figura 7.51 Tiempo de ejecución cíclica del Lazo While.....	174
Figura 7.52 Creación de las entradas digitales.....	175
Figura 7.53 Entradas digitales creadas.....	176
Figura 7.54 Creación de Salidas Digitales.....	177
Figura 7.55 Salidas Digitales creadas.....	177
Figura 7.56 Ingreso al Image Navigator.....	178
Figura 7.57 Representación de la bomba.....	179
Figura 7.58 Creación de variables analógicas.....	179
Figura 7.59 Variables analógicas creadas.....	180
Figura 7.60 Lógica de arranque y apagado de bomba, alarmas y reseteo.....	181
Figura 7.61 Creación de string para indicar alarma.....	182
Figura 7.62 Atributos del string.....	183
Figura 7.63 Funcionamiento y programación de textos explicativos.....	184
Figura 7.64 Ruta para creación de librería.....	186
Figura 7.65 Ruta para creación de I/O Server.....	187
Figura 7.66 Selección de tipo de servidor.....	187
Figura 7.67 Selección del servidor OPC con el que vamos a trabajar y cambio de la tasa de actualización.....	188
Figura 7.68 Ruta para creación de variables ligadas.....	189
Figura 7.69 Selección de variables ligadas.....	189
Figura 7.70 Block Diagram sin variables ligadas.....	190

Figura 7.71 Block Diagram con variables ligadas.....	191
Figura 7.72 Creación del canal de comunicación.....	193
Figura 7.73 Selección del protocolo de comunicación.....	193
Figura 7.74 Selección del adaptador de red de la computadora con la que se trabaja .....	194
Figura 7.75. Selección del método de optimización .....	194
Figura 7.76. Selección de respuesta cuando un valor analógico no este normalizado .....	195
Figura 7.77 Resumen de la configuración del canal.....	195
Figura 7.78 Escritura del nombre del dispositivo.....	196
Figura 7.79 Selección del modelo del dispositivo.....	196
Figura 7.80 Escritura de la dirección ip del dispositivo dentro de la red .....	197
Figura 7.81 Resumen de la configuración del canal.....	197
Figura 7.82 Parámetros de sincronización .....	198
Figura 7.83 Auto degradación del dispositivo cuando hay una falla de comunicación .....	198
Figura 7.84. Parámetros de comunicación.....	199
Figura 7.85 Parámetros de puerto de comunicación para S7.....	199
Figura 7.86 Parámetros de ordenamiento.....	200
Figura 7.87 Resumen de configuraciones del dispositivo a utilizar.....	200
Figura 7.88 Creación de variable analoga Nivel Escalado .....	201
Figura 7.89 Creación de variable booleana Run Bomba. ....	201
Figura 7.90 Variables creadas en NI OPC Servers .....	202
Figura 7.91 Ruta para creación del Cyclic Interrupt .....	205



Figura 7.92 Ruta para creación del PID_Compact.....	206
Figura 7.93 Ajustes básicos del PID_Compact.....	206
Figura 7.94 Ajustes del valor real del PID_Compact .....	207
Figura 7.95 Dirección del PID_Compact. ....	208
Figura 7.96 Selección del DB o bloque de datos.....	208
Figura 7.97 Normalizado y escalado de salida de PID .....	209
Figura 7.98 Bloque PID_Compact en el Cyclic Interrupt .....	210
Figura 7.99 Segmento 3 del bloque Main editado. ....	211
Figura 7.100 Bloque PID_Compact con salida Alarma de Nivel .....	211
Figura 7.101 Bloque PID_Compact con todas las variables.....	212
Figura 7.102 Puesta en marcha del bloque PID_Compact.....	213
Figura 7.103 Variables creadas en NI OPC Servers .....	214
Figura 7.104 Block Diagram con lógica para funcionamiento manual y automático .....	215

## ÍNDICE DE TABLAS

Tabla 1 Descripción de elementos del módulo básico RT 450 [3].....	9
Tabla 2 Descripción de componentes y función del Módulo para regulación de nivel RT 450.01 [3].....	11
Tabla 3 Elementos del posicionador electro neumático SAMSON 3760.....	17
Tabla 4 Características técnicas del CPU 1212C AC/DC/relé [4] .....	21
Tabla 5 Características técnicas del módulo SM 1234 [4] .....	23
Tabla 6 Parámetros de la fórmula que utiliza el PID_COMPACT [4].....	34
Tabla 7 Resultados de la optimización del PID_COMPACT .....	46
Tabla 8 Significado de las variables de la ecuación en términos de TIA PORTAL ...	47
Tabla 9 Variables que trabajan entre Labview y TIA PORTAL .....	85
Tabla 10 Resultados obtenidos en las pruebas.....	116
Tabla 11 Tipo de bloques de programación.....	147
Tabla 12 Relación entre número de bits y nivel en cm. ....	158
Tabla 13 Tipo de Datos en NI OPC Servers y TIA PORTAL .....	192

## **INTRODUCCIÓN**

Los equipos para automatización como los PLCs, surgieron de la necesidad de simplificar tareas que mediante lógica cableada resultaban en complicados diseños, aparte de la paralización en las actividades industriales que resultaba de la avería en un determinado elemento aislado, por ello y aprovechando el desarrollo en la tecnología de los microprocesadores que evolucionan constantemente reduciendo su tamaño y aumentando la capacidad de manejo de datos, hoy en día contamos con PLCs cada vez más accesibles en precio y que incorporan funciones que antes eran ejecutadas por elementos aislados, adicionalmente la evolución constante en las versiones de Labview, y uno de sus componentes claves NI OPC Servers ha facilitado la comunicación con una gama cada vez más amplia de PLCs.

Por las razones anteriormente expuestas, hemos considerado conveniente desarrollar el presente proyecto de graduación aprovechando las facilidades prestadas en el Laboratorio de Instrumentación Industrial para introducir la tecnología Siemens con su entorno de programación y desarrollo TIA PORTAL, vinculándolo con Labview que es un software ampliamente utilizado en la actualidad, a fin de desarrollar procedimientos, tecnología y difundir la misma mediante la elaboración de prácticas de laboratorio.

# **CAPÍTULO 1**

## **ANTECEDENTES Y JUSTIFICACIÓN**

### **1.1 DESCRIPCIÓN DEL PROBLEMA**

En la automatización de procesos industriales comúnmente se usan PLC'S para el control, y paneles HMI para monitorear señales y ajuste de parámetros. Frecuentemente ambos elementos necesitan comunicarse mediante un protocolo propietario, de la misma marca que suministra los equipos; por lo tanto es indispensable integrar PLC, panel HMI y software de programación-configuración del mismo fabricante e inclusive de la misma versión; esto constituye un problema que se presenta en las industrias, cuando se dispone de equipos que utilizan diferentes

protocolos y drivers de comunicación, y se necesita monitorear, ajustar parámetros del proceso en un HMI; adicionalmente a medida que los fabricantes lanzan al mercado nuevos PLC, pueden dejar de ofrecer soporte a los antiguos equipos, con lo cual en ocasiones dejan de ser compatibles con nuevas versiones de HMI o SCADA. Es importante mencionar que trabajar con un software que usa el concepto de programación modular como TIA PORTAL y sus equipos vinculados como el PLC SIMATIC S71200, contribuye a diversificar nuestros conocimientos y a estar mejor preparados para responder a los desafíos que profesionalmente se presentan.

## **1.2 OBJETIVOS.**

Implementar comunicación entre un PLC Simatic S7 1200 de Siemens y un Interfaz para usuario diseñado en Labview de National Instruments utilizando la arquitectura Cliente/Servidor OPC (OLE for Process Control), a fin de que la comunicación entre la Fuente de Datos(PLC) y el Cliente de Datos(HMI) sea independiente del fabricante de los componentes de este sistema.



Figura 1.1 Elementos que intervienen en el proyecto.

Diseñar una guía de prácticas de laboratorio para la planta de nivel Gunt RT450 que integre conocimientos de Automatización, Instrumentación Industrial y Comunicaciones Industriales, utilizando los recursos de hardware y software de este proyecto.

Monitorear y ajustar los parámetros de control del funcionamiento de la planta de nivel Gunt RT450, mediante un Interfaz para Usuario diseñado en Labview.

Obtener conocimientos en desarrollo de aplicaciones para automatización industrial con el software TIA PORTAL, y en la configuración de las herramientas para comunicaciones industriales que dispone Labview.

### **1.3 SOLUCIÓN PROPUESTA Y JUSTIFICACIÓN DE LA SOLUCIÓN**

#### **SOLUCIÓN PROPUESTA**

Actualmente en el Laboratorio de Instrumentación Industrial la Planta de Nivel GUNT RT450, tiene como una de las opciones de control un PLC Klockner Moeller PS4-201-MM1; aprovechando las características modulares de la planta se integra como elemento de control el PLC SIMATIC S7 1200, el cual al ser programado en el software TIA PORTAL ofrece óptimas posibilidades de aprendizaje debido a la interfaz gráfica que permite un desarrollo de aplicaciones de automatización eficiente, pues TIA PORTAL es útil para PLCs, Paneles HMI, Redes de Comunicación Industrial, adicionalmente mediante el NI OPC SERVER se establece comunicación entre el PLC SIMATIC S7 1200 y una INTERFAZ PARA USUARIO diseñada en LabVIEW, de esta manera los componentes que gestionan el monitoreo, supervisión y ajuste de parámetros son independientes del fabricante.

#### **JUSTIFICACIÓN DE LA SOLUCIÓN**

Seleccionamos esta solución debido a que, en el Laboratorio de Instrumentación Industrial de la ESPOL se cuenta con el software Labview y el módulo NI OPC SERVER de NATIONAL INSTRUMENTS, que permiten establecer comunicación entre una INTERFAZ PARA USUARIO desarrollada en Labview para monitorear señales y ajustar

parámetros de la Planta de Nivel GUNT RT450, y una amplia gama de PLCs de diferentes marcas y modelos.

Elegimos específicamente el PLC SIMATIC S7 1200 debido a que el software de programación TIA PORTAL vinculado a este PLC, ofrece un entorno de programación gráfico e intuitivo, con documentación de ayuda que permite el autoaprendizaje para el desarrollo de aplicaciones, además integra posibilidades de programación para PLCs, pantallas HMI y accionamientos de velocidad variable, de esta manera al combinar los elementos descritos tenemos una herramienta de aprendizaje práctico de gran utilidad, pues combina conocimientos de Instrumentación Industrial, Automatización y Redes Industriales.

#### **1.4 METODOLOGÍA.**

Levantamiento de información sobre los recursos con que se dispone en el Laboratorio de Instrumentación Industrial de la ESPOL, en lo referente a hardware y software relacionados con la ejecución de este proyecto, a fin de identificar y adquirir los elementos adicionales que se necesiten para la implementación



Diseño de la aplicación en TIA PORTAL, para el control de la Planta de Nivel GUNT RT450, en Modo Manual y Automático; conforme se avanza en el desarrollo de la aplicación se debe hacer simulaciones con las Tablas de Observación de TIA PORTAL, a fin de que el resultado final se ajuste al algoritmo requerido.

Diseño de la INTERFAZ PARA USUARIO con el software Labview; se debe visualizar la evolución de las variables análogas y discretas que intervienen en el proceso, estableciendo alarmas y señalizaciones para niveles críticos de operación; el operador de la INTERFAZ PARA USUARIO debe tener acceso vía teclado a modificar setpoints, además la interfaz debe mostrar textos explicativos sobre las variables que intervienen en el proceso, cuando el usuario lo requiera.

Cableado de las señales de control digitales y análogas entre la Planta de Nivel y el PLC, usando cable multipar apantallado.

Configuración de la comunicación entre PLC SIMATIC S7 1200 y la INTERFAZ PARA USUARIO diseñado; usando el OPC se deben crear los tags necesarios para compartir la información entre los 2 elementos,

en este punto se setean parámetros como: Device Driver, Baud rate, Update Rate, Dead Band %.

Se procede a realizar las pruebas del sistema diseñado funcionando en conjunto, operando la INTERFAZ PARA USUARIO en sus modos manual y automático, para proceder a documentar el comportamiento de las variables de la Planta de Nivel, se evalúan los resultados de las mismas, y se depura el diseño hasta que los parámetros sean los óptimos.

Con los resultados obtenidos y tomando en cuenta los procedimientos utilizados, se procede a redactar la guía de prácticas de laboratorio para la planta de nivel, documentando la misma con detalles explicativos y gráficos de los pasos necesarios para lograr el óptimo funcionamiento de la planta.

## **CAPÍTULO 2**

### **ELEMENTOS DEL SISTEMA**

#### **2.1 DESCRIPCIÓN DE LA PLANTA DE NIVEL GUNT HAMBURG RT450**

Este elemento consiste en un sistema didáctico modular que permite, implementar experimentos que nos guían al conocimiento y experimentación de la automatización, instrumentación y comunicaciones industriales con todos los aspectos relacionados, como son regulación de procesos por medio de distinto tipo de controladores, manejo de sensores y actuadores, para magnitudes físicas, como: nivel, caudal, temperatura y presión en Figura 2.1 apreciamos un esquema del módulo básico RT 450, y en la Tabla 1 tenemos la descripción de los elementos del módulo básico.

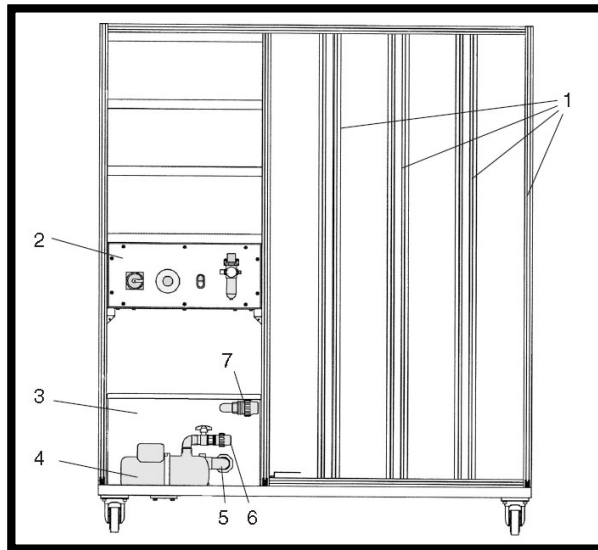


Figura 2.1 Módulo básico RT 450 [1]

1	Rieles de perfil
2	Armario de distribución: Interruptor principal, Interruptor de parada de emergencia, Interruptor I/O de la bomba, Regulador de aire comprimido con manómetro (0..6bares) y acople rápido para experimentos
3	Depósito de agua
4	Bomba <b>marca:</b> LOWARA, <b>Modelo:</b> 2HMS36/A, <b>Caudal:</b> 30-85 litros/minuto, <b>Altura de elevación:</b> 13,2-5,7 metros, <b>Potencia suministrada al líquido por la bomba:</b> 0.3kW, acoplada a un <b>Motor monofásico modelo</b> SM63HM/1036, <b>Voltaje</b> 110-115Vac, <b>I<sub>nominal</sub></b> : 4.25 A, <b>Potencia entregada a la bomba:</b> 0,43 kW, equipada para el arranque con un <b>capacitor</b> 40 $\mu$ F/450V
5	Salida de la bomba
6	Entrada de la bomba
7	Retorno de agua del sistema

Tabla 1 Descripción de elementos del módulo básico RT 450 [3]

Durante las prácticas de este proyecto trabajaremos con la planta equipada con los accesorios necesarios para el control de nivel, por ello la mencionaremos como Planta de Nivel GUNT- RT 450, uno de sus

elementos más importantes es el Módulo para regulación de nivel RT 450.01 que tenemos en la Figura 2.2. Sus componentes y funciones están descritos en Tabla 2

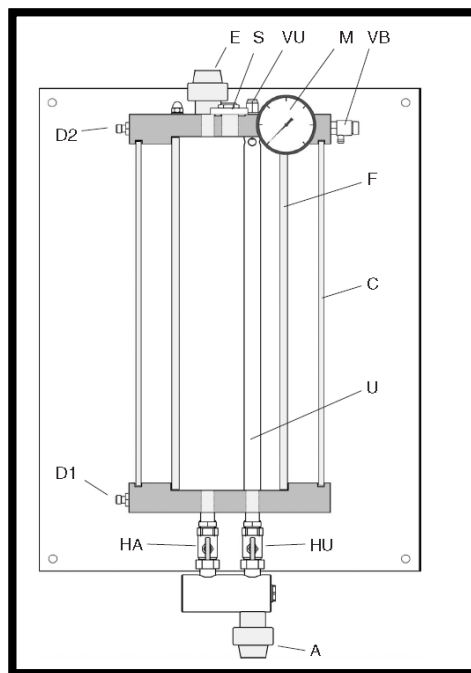


Figura 2.2 Módulo para regulación de nivel RT 450.01 [3]

Letra	Nombre	Función
E	Entrada de agua	Por aquí se bombea agua al depósito
S	Abertura para montaje de sensor de nivel	Aquí va enroscado el sensor de nivel capacitivo RT 450.35
M	Manómetro para la presión en el depósito	Indica la presión interna en el depósito
F	Depósito de nivel de llenado	Para el establecimiento del nivel, está fabricado en vidrio Durán
C	Envoltura protectora	Protección para el depósito
U	Tubo de rebosadero	Impide que el depósito se llene en exceso, siempre y cuando la llave de rebosadero este abierta
A	Salida de Agua	Salida de agua del depósito
HA	Llave de salida de agua	Llave para salida de agua del depósito
HU	Llave de rebosadero	Llave que permite que el agua fluya por el rebosadero cuando el nivel sube del máximo
D1/D2	Conexiones de presión	Medición de nivel a través de la presión hidrostática por encima del fondo del depósito
VB	Válvula de ventilación	Para introducir aire al depósito durante los experimentos
VU	Válvula de sobrepresión	Se abre cuando la presión interna del depósito supera los 2 bares

Tabla 2 Descripción de componentes y función del Módulo para regulación de nivel RT 450.01 [3]

Es necesario referirnos al diagrama P&ID de la Planta de Nivel que utilizaremos en el presente proyecto. Figura 2.3

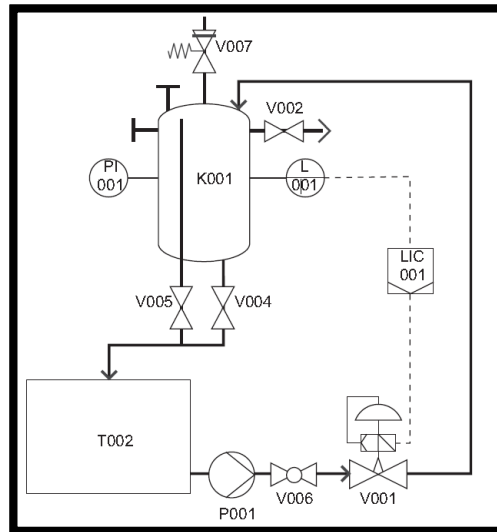


Figura 2.3 Diagrama P&ID de la Planta de Nivel GUNT RT 450 [3]

El agua es impulsada por la bomba P001 Lowara desde el depósito T002 al depósito K001, con la llave de paso V006 abierta, y mediante el accionamiento neumático de la válvula de regulación V001. El nivel de llenado del depósito K001 lo medimos con el sensor de nivel capacitivo L001 Elobau, y se comunica al regulador LIC001 que en el presente proyecto es el control que utiliza el PLC SIMATIC S7-1200 comunicado mediante OPC al Interfaz de Usuario en Labview, este regulador emite la señal que controla a V001, la válvula V002 sirve para la ventilación del depósito, Si se cierran V002 y V005 en el rebozadero se puede trabajar contra un cojín de presión, el manómetro PI001 indica entonces la presión existente en el depósito, Una válvula de seguridad V007 protege el depósito de un exceso de presión mayor a 2 bares. 450 [3]

Otro de los elementos que integran la Planta de Nivel, es el Sensor de Nivel RT 450.35, el cual es de tipo capacitivo, pues la varilla de medición que está sumergida en el agua representa un condensador eléctrico, y dado que las constantes dieléctricas del agua y del aire son muy diferentes, la capacidad del condensador cambia de manera que la podemos medir en función del nivel del agua, está construido con la técnica de 2 conductores y tiene un transformador, se emite una señal estándar de 4-20 mA, proporcional al nivel de llenado [3], el código comercial del sensor utilizado es 212 KK.05 de la marca ELOBAU. Figura 2.4 y se indica el tipo de salida del sensor en Figura 2.5



Figura 2.4 Sensor de nivel RT 450.35 Elobau 212 KK0.5 [7]



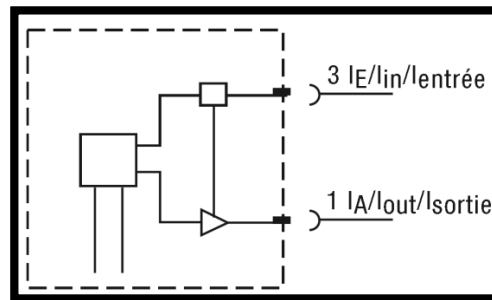


Figura 2.5 Señal de salida para sensor de nivel RT 450.35 Elobau 212 KK0.5 [7]

Finalmente nos referiremos a la válvula de control neumática RT 450.20 SAMSON que es el elemento actuador de este sistema, dicha válvula cuenta con un servo accionamiento neumático y con regulador de posición electro neumático i/p SAMSON 3760 [3]. El factor de flujo, de esta válvula es KV1. Debemos recordar que el factor de flujo nos indica el caudal de agua en  $\text{m}^3/\text{hora}$  a temperatura normal que pasa a través de una válvula con una caída de presión de 1 bar. Figura 2.6

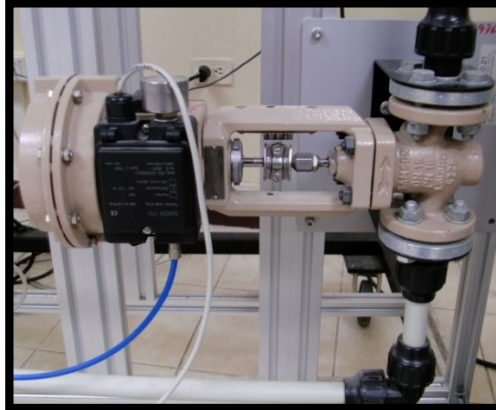


Figura 2.6 Válvula RT 450.20 SAMSON con servo accionamiento neumático y regulador de posición i/p SAMSON 3760

El posicionador electro neumático i/p SAMSON 3760, lleva incorporado un módulo convertidor Figura 2.7, el módulo i/p recibe una corriente de 4-20 mA y allí lo transforma en una presión de aire proporcional  $p_e$  de 0.2 a 1 bares, esta señal de presión  $p_e$ , produce una fuerza sobre la membrana de medición que se compara con la fuerza del resorte de medición, el movimiento de la membrana de medición se transmite por medio de la palanca al obturador doble del piloto, dando así la presión de mando  $p_{st}$  correspondiente, de esta manera el vástago del obturador de la válvula se sitúa en la posición correspondiente a la magnitud de la señal de corriente 4-20 mA que ingresa al módulo convertidor estos detalles de funcionamiento los podemos apreciar en la Figura 2.8 cuyos elementos se indican en la Tabla 3. [8]



Figura 2.7 Módulo convertidor i/p SAMSON

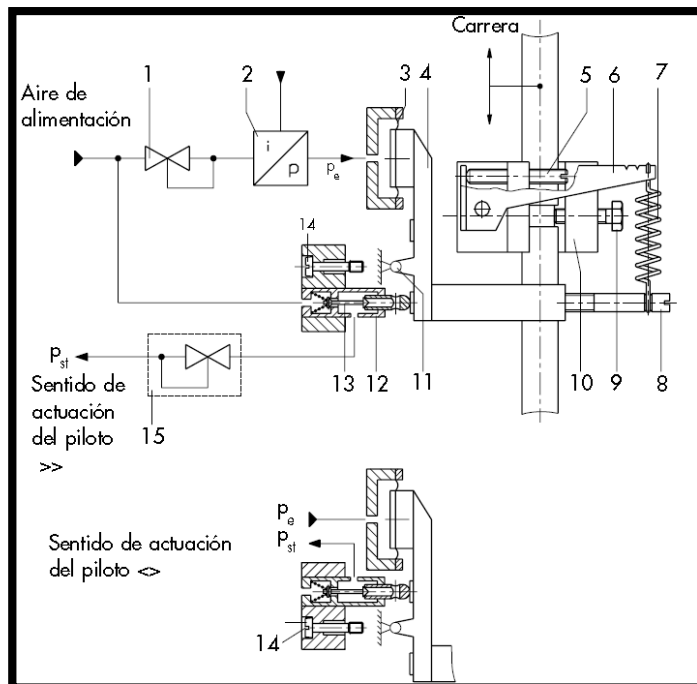


Figura 2.7 Esquema de funcionamiento del posicionador SAMSON 3760 [8]

<b>Número</b>	<b>Elemento</b>
1	Regulador de presión
2	Convertidor i/p
3	Membrana de medición
4	Palanca de la membrana
5	Tornillo de ajuste (cero)
6	Palanca
7	Resorte de medición
8	Tornillo de ajuste (span)
9	Tornillo de bloqueo
10	Estribo de bloqueo
11	Eje de giro
12	Piloto
13	Obturador doble
14	Tornillo
15	Limitador de presión

Tabla 3 Elementos del posicionador electro neumático SAMSON 3760

La característica de modularidad que tiene la planta permite al estudiante implementar nuevas tecnologías de control, como en el caso del presente proyecto mediante un PLC SIMATIC S7-1200 integrado a un Interfaz diseñado en Labview, de la misma manera integramos a la planta, un conjunto de luces piloto, pulsantes y selectores que nos servirán para activar las diferentes fases del control del presente proyecto.

## **2.2 DESCRIPCIÓN DEL PLC SIMATIC S7 1200 SIEMENS**

El modelo S7-1200 engloba diferentes controladores lógicos programables que pueden usarse para distintas tareas, dependiendo del número y tipo de entradas, además tiene varios accesorios como módulos de comunicación, de entradas y salidas tanto analógicas como

digitales, pantallas HMI, tarjetas de memoria que permiten flexibilidad en el diseño de acuerdo a la complejidad de la aplicación que se le dé al equipo.

Cada CPU, se compone de un microprocesador, una fuente de alimentación, circuitos de entrada y salida, para nuestro proyecto hemos seleccionado el CPU 1212C AC/DC/relé y el Módulo de entradas y Salidas analógicas SM1234. En la Figura 2.8 apreciamos los 2 componentes de nuestro proyecto ensamblados, adicionalmente hemos diseñado un circuito transductor que tiene como entrada un potenciómetro  $10k\Omega$  10 vueltas, y como salida una señal de 0-10 Vdc, pues requerimos generar una señal con el potenciómetro, y el módulo SM1234 tiene entrada de 0-10Vdc.

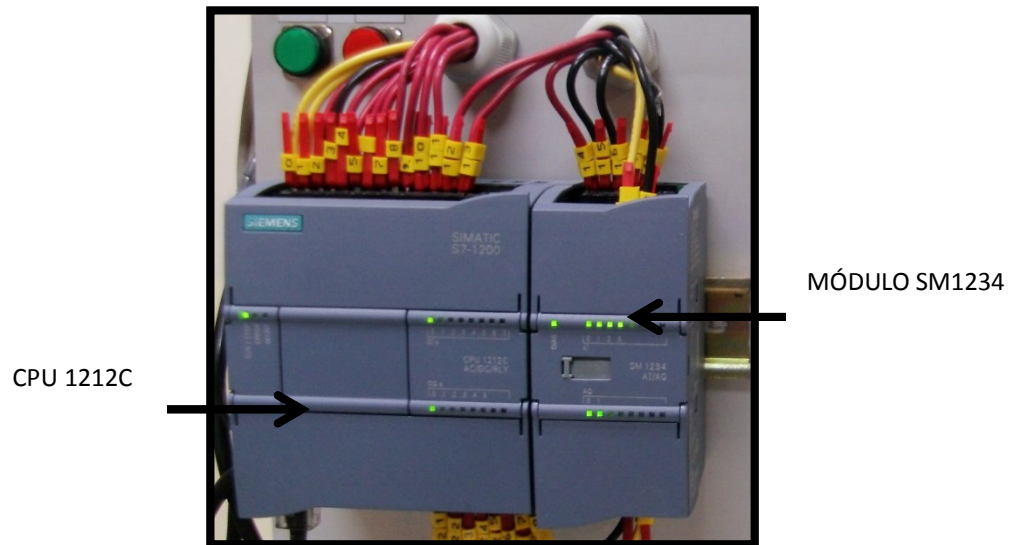


Figura 2.8 CPU 1212 AC/DC/relé y SM1234 ensamblados

Las características técnicas del CPU 1212C AC/DC/relé las referimos en la Tabla 4. y el diagrama de conexiones típico del CPU 1212C en la Figura 2.9, en esta figura apreciamos que la alimentación del equipo puede estar en el rango de 120-240Vac, además posee una fuente de 24Vdc, las entradas digitales se cablean, uniendo en un punto común un extremo de las mismas y cableando este punto común al positivo (+) de una fuente de voltaje de 24 Vdc, el negativo de esta fuente de voltaje se conecta a la bornera del PLC rotulada como 1M, el otro extremo de las entradas digitales va a su correspondiente bornera rotuladas como .0, .1, .2, .3, .4, .5, .6, y .7 . En lo referente a las entradas analógicas, el negativo de las mismas se conecta entre sí en un punto común ese punto común se conecta a la bornera rotulada como 2M y el positivo a cada

entrada analógica a su correspondiente bornera rotulada como 0, 1. En lo referente a las salidas tipo relé, indicamos que se las puede agrupar en 2 niveles de voltaje diferentes, por ejemplo podemos tener la necesidad de conectar un grupo de contactores con bobina a 220Vac y un grupo de luces piloto a 24Vdc, tenemos entonces que un extremo de las bobinas o luces piloto se conecta entre sí formando un punto común, este punto se cablea al negativo si es fuente DC o a una línea que denominaremos L2 si es una fuente AC, el positivo de la fuente DC o la otra línea de la fuente AC que denominaremos L1, se cablea al terminal rotulado 1L para un grupo de cargas de un nivel de voltaje, y el común del otro grupo de cargas a 2L, el otro extremo de las bobinas o luces piloto o válvulas o la carga que fuere se cablea a las borneras numeradas de las salidas relé.

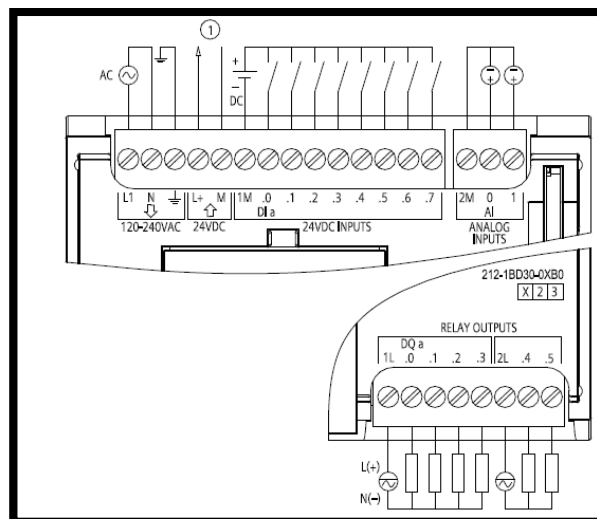


Figura 2.9 Diagrama de conexiones para CPU 1212C AC/DC/relé /relé [4]

<b>CARACTERÍSTICAS</b>	<b>CPU 1212C AC/DC relé</b>
Dimensiones físicas (mm) AxAxP	90x100x75
Memoria de usuario	
Memoria de trabajo	25 KB
Memoria de Carga	1 MB
Memoria Remanente	2 KB
E/S locales	
Digitales	8 entradas/4 salidas
Analógicas	2 entradas
Tamaño de la memoria imagen del proceso	1024 bytes para entradas (I) y 1024 bytes para salidas (Q)
Área de marcas (M)	4096 bytes
Ampliación con módulos de señales	2
Signal Board	1
Módulos de Comunicación	3 (ampliación para el lado izquierdo)
Contadores rápidos	4
Fase simple	3 a 100 kHz 1 a 30 kHz
Fase cuadratura	3 a 80 kHz 1 a 20 kHz
Salidas de impulsos	2
Memory card	Opcional
Tiempo de respaldo del reloj de tiempo real	10 días
PROFINET	1 puerto de comunicación Ethernet
Velocidad de ejecución de funciones matemáticas con números reales	18 $\mu$ s/instrucción
Velocidad de ejecución booleana	0,1 $\mu$ s/instrucción
Intensidad disponible en 24Vdc	300 mA máx. (alimentación de sensores)
Consumo de corriente de entradas digitales	4mA/entrada utilizada
Fuente de Alimentación 85 a 264 Vac	
Frecuencia de línea	47-63 Hz
Voltaje de entradas analógicas	0-10 Vdc
Salidas Digitales	5-30 Vdc ó 5 a 250 Vac

Tabla 4 Características técnicas del CPU 1212C AC/DC/relé [4]



Las características técnicas del módulo de entradas y salidas analógicas SM 1234 las detallamos en la Tabla 5, y el diagrama de conexiones en la Figura 2.10, la forma de conexión de las entradas y salidas analógicas depende si se las configura como entradas analógicas de voltaje o de corriente, si son de voltaje la conexión se realiza para cada entrada a su punto correspondiente 0+,0- , si son de corriente cada entrada analógica se conecta por separado a su respectiva pareja de borneras por ejemplo a 0+, 0-, lo mismo para las salidas analógicas

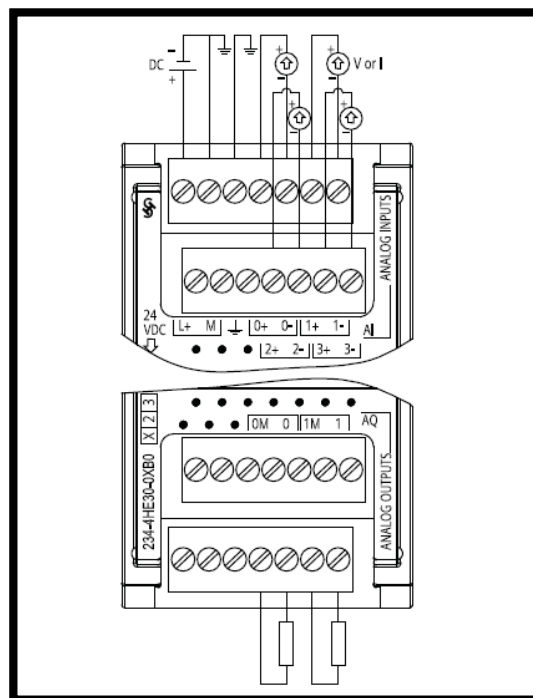


Figura 2.10 Diagrama de conexiones para módulo SM 1234 [4]

<b>CARACTERÍSTICA</b>	<b>SM 1234</b>
Dimensiones físicas (mm) AxAxP	45x100x75
Peso	180 gramos
Disipación de potencia	2,0 W
Entradas Analógicas	4
Tipo	Tensión o intensidad (diferencial), seleccionable en grupos de 2
Rango	$\pm 10$ V ó 0 a 20 mA
Rango total de palabra de datos	-27.648 a 27.648
Resolución	12 bits + bit de signo
Tiempo de conversión analógica digital	625 $\mu$ s (rechazo de 400 Hz)
Diagnóstico de rebase por exceso/defecto	SI
Diagnóstico de cortocircuito a tierra(solo en modo de tensión)	Si en las salidas
Diagnóstico de Rotura de Hilo (solo en modo de intensidad)	Si en las salidas
Salidas Analógicas	2
Tipo	tensión o intensidad
Rango	$\pm 10$ V ó 0 a 20 mA

Tabla 5 Características técnicas del módulo SM 1234 [4]

### 2.3 DESCRIPCIÓN DEL SOFTWARE LABVIEW NATIONAL INSTRUMENTS.

Labview constituye un sistema de programación gráfica para aplicaciones que involucran adquisición, control, análisis y presentación de datos. Es un entorno de programación destinado al desarrollo de aplicaciones, similar a los sistemas de desarrollo comerciales que utilizan C o BASIC pero se diferencia de dichos programas porque emplea la programación gráfica o lenguaje G para crear programas basados en diagramas de bloques mientras que los anteriores se basan en líneas de texto para

crear el código fuente del programa. El uso de Labview presenta muchas ventajas:

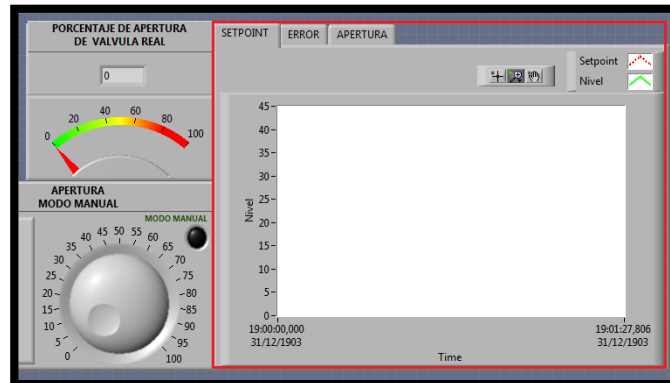


Figura 2.11 Interfaz diseñado en Labview (Front panel)

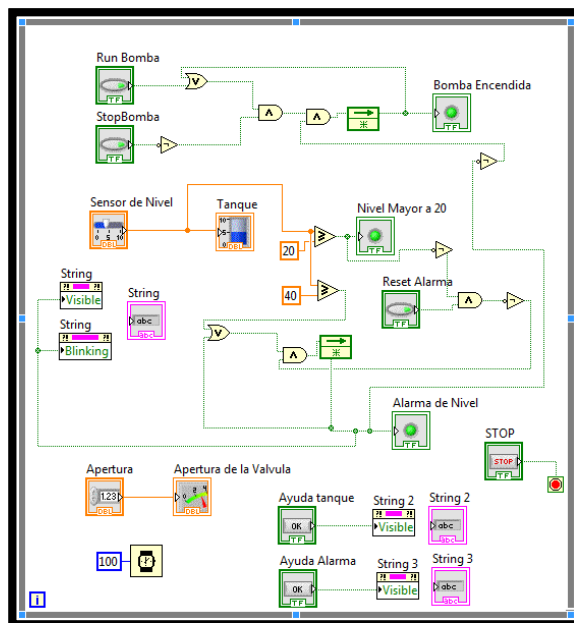


Figura 2.12. Programación en Labview (Block diagram)

- Es muy flexible ya que permite actualizaciones de hardware y software.
- Se pueden crear desde soluciones sencillas hasta llegar a alcanzar un nivel de complejidad alto.
- En un único sistema de desarrollo se integran las funciones de adquisición, análisis y presentación de datos.
- Se pueden incorporar aplicaciones escritas en otros lenguajes.

Para el empleo de Labview no se requiere gran experiencia en programación, ya que se emplean iconos que son familiares a científicos e ingenieros. Emplea un lenguaje más intuitivo en comparación al resto de lenguajes de programación convencionales. En él hay extensas librerías de funciones y subrutinas, también incluye librerías específicas para la adquisición de datos, control de instrumentación, comunicación serie, análisis presentación y guardado de datos, además proporciona potentes herramientas que facilitan la depuración de los programa.

**COMO TRABAJA LABVIEW:** Los programas desarrollados mediante Labview se denominan Instrumentos Virtuales (VIs) ya que su apariencia

y funcionamiento imitan los de un instrumento real. Son equivalentes a las funciones creadas con los lenguajes de programación convencionales. Los VIs tienen una parte interactiva con el usuario (Front Panel) y otra parte de código fuente (Block Diagram). También existen las paletas que permiten hacer modificaciones en dichos bloques. Procedemos a realizar una breve descripción:

**Panel Frontal:** Se trata de la Interfaz gráfica del VI con el usuario, en ella realizamos la representación visual de los elementos que queremos controlar. En él podemos colocar selectores, pulsadores, gráficas para visualizar el comportamiento de variables y muchas otras opciones para manipulación de variables.

**Block Diagram:** En él se realiza la programación del VI, es decir es donde se realiza la implementación del programa del VI para controlar o realizar cualquier procesamiento de las entradas y salidas que se crearon en el panel frontal, en él se incluyen funciones y estructuras integradas en las librerías. En el lenguaje G las funciones y las estructuras son nodos elementales. Los controles e indicadores que se colocaron previamente en el Panel Frontal, aparecen en el diagrama de bloques mediante los terminales. El diagrama de bloques se construye conectando los distintos objetos entre sí, como si se estuviera cableando un circuito. Los cables

unen terminales de entrada y salida con los objetos correspondientes, y por ellos fluyen los datos. Labview posee funciones aritméticas, comparaciones, conversiones, funciones de entrada/salida, de análisis y muchas otras aplicaciones de gran utilidad que hacen fácil la implementación de soluciones.

**Paletas:** Existen 3 tipos

**Paleta de herramientas (Tools Palette):** La cual se puede utilizar tanto en el Front Panel como en el Block Diagram y nos proporciona las herramientas que se requieren para crear y modificarlos.

**Paleta de controles (Controls Palette):** Se utiliza únicamente en el Front Panel y contiene todas las herramientas para crear la interfaz gráfica del VI.

**Paleta de funciones (Functions Palette):** Se emplea únicamente en el diseño del diagrama de bloques y contiene funciones aritméticas, de entrada/salida de señales, entrada/salida de datos a fichero, adquisición de señales, temporización de la ejecución del programa, etc.

## **2.4 DESCRIPCIÓN DEL SOFTWARE TIA PORTAL**

TIA Portal es la herramienta que nos permite sacar todo el potencial de Totally Integrated Automation. Este software optimiza todos los procedimientos de planificación para el manejo de procesos industriales. Esta herramienta de ingeniería unifica los programas SIMATIC STEP 7, SIMATIC WinCC y SINAMICS StartDrive permitiéndonos el manejo de PLCs, HMIs, arrancadores y variadores de velocidad respectivamente. TIA Portal nos brinda una mayor facilidad en la etapa de diseño, puesta en marcha, operación, mantenimiento y hasta la actualización de soluciones de automatización. TIA Portal integra por primera vez los sistemas de ingeniería para las diversas tareas de automatización. Mediante un framework común, una interfaz unitaria y una configuración similar de tareas de automatización equiparables, se garantiza un tiempo mínimo de familiarización y un alto grado de reconocimiento. [6]

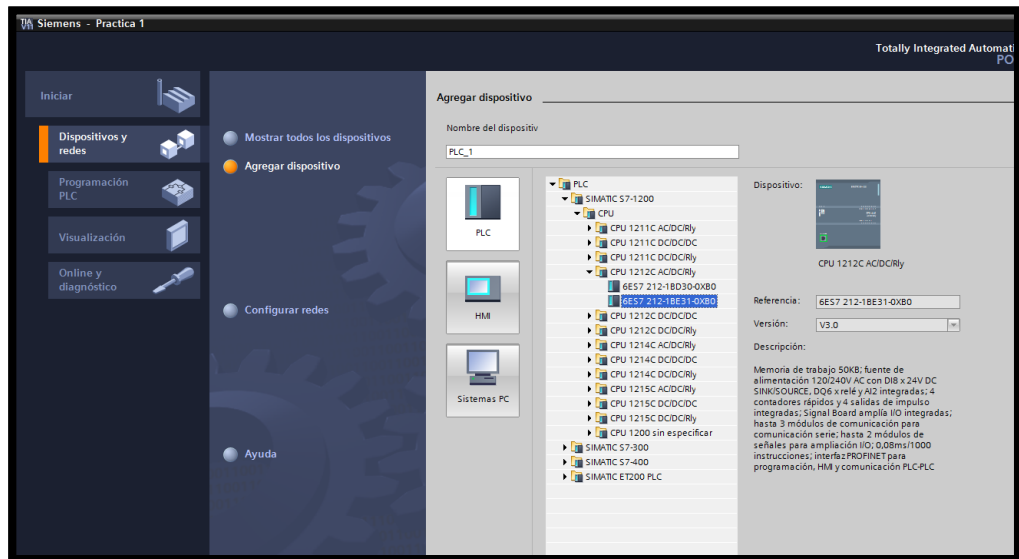


Figura 2.13 Vista de configuración de dispositivos en TIA PORTAL

Esta herramienta nos permite realizar una detección muy rápida de fallos de equipos y sistemas dentro de nuestros procesos ya que para esto no es necesario hacer configuraciones muy exhaustivas ni complejas.

Este software nos permite trabajar con una programación tipo ladder la cual es muy conocida por nosotros y es la de mayor aceptación en lo que a soluciones para procesos industriales se refiere.

Las versiones Basic permiten únicamente programar y configurar los PLCs S7-1200 con sus HMIs compatibles, la versión Professional permite hacerlo con los equipos de la gama S7-300 y S7-400, además posee



simulador, lo que se convierte en un gran factor diferenciador, ya que permite evaluar la programación realizada sin necesidad de conectarse al PLC real.

En resumen con el software TIA Portal tenemos una herramienta que en las manos adecuadas abre un amplio margen para la creación de soluciones en procesos industriales de cualquier nivel de complejidad.

## **CAPÍTULO 3**

### **DISEÑO DE APLICACIÓN PARA CONTROL DE PLANTA DE NIVEL EN TIA PORTAL.**

Tenemos 2 modos de funcionamiento Manual y Automático.

En el modo de funcionamiento manual, vamos a controlar el nivel del líquido en la planta de nivel en lazo abierto, manipulando la apertura o cierre de la válvula con servo accionamiento neumático y regulador de posición i/p, de 2 formas independientes entre sí que se eligen mediante un selector dispuesto en la interfaz de Labview, figura 3.1 el cual nos permite elegir la fuente de consigna para la salida analógica %AQ0, que es la que envía la señal de

corriente 4-20 mA para el regulador de posición i/p. Estas 2 opciones son:

- a) Dial en la interfaz de Labview con escala de 0 a 100, figura 3.1
- b) Potenciómetro de precisión 10 K $\Omega$  10 vueltas, con accesorio para conteo de vueltas y recorrido, figura 3.2



Figura 3.1 Selector de fuente de consigna y dial en Labview



Figura 3.2 Potenciómetro de precisión 10 K $\Omega$  10 vueltas

En el modo de funcionamiento automático, el control de nivel de líquido en lazo cerrado se realiza mediante el objeto tecnológico PID\_COMPACT del PLC SIMATIC S7-1200, el cual utiliza un algoritmo que calcula la ganancia proporcional  $K_p$ , Tiempo de acción integral  $T_I$ , Tiempo de acción derivativa  $T_D$ . El PID\_COMPACT se ejecuta cíclicamente comparando el Setpoint o consigna  $w$  ajustado en la interfaz de Labview con el valor de la entrada, que corresponde al sensor de nivel capacitivo de código comercial ELOBAU 212 KKA 05 el cual en [3] se lo designa como RT 450.35; de esta manera el objeto tecnológico indicado, ajusta cíclicamente el Valor de Salida  $y$ . Este proceso se denomina optimización fina (autotuning) se lo realiza antes de poner el regulador PID en servicio [4], Una vez realizado el proceso anterior con los parámetros PID optimizados se puede poner el regulador PID en marcha. Adicionalmente indicamos en la figura Ecuación (3.1) [4] la fórmula que utiliza el PID\_COMPACT para calcular el valor de salida  $y$ ; en la Tabla 6 se detallan los parámetros que maneja esta fórmula.

$$y = KP * \left[ (b * w - x) + \left( \frac{(w - x)}{TI * S} \right) + \left( \frac{TD * S}{a * TD * S + 1} \right) * (c * w - x) \right]$$

(3.1) Fórmula que utiliza el PID\_Compact para calcular el valor de salida  $y$  [4].

$y$	Valor de Salida	$x$	Valor de proceso
$w$	Consigna	$s$	Operador laplaciano
$K_p$	Ganancia proporcional (acción P)	$a$	Coefficiente de retardo derivativo (acción D)
$T_I$	Tiempo de acción integral (acción I)	$b$	Ponderación de la acción proporcional (acción P)
$T_D$	Tiempo de acción derivativa (acción D)	$c$	Ponderación de acción derivativa (acción D)

Tabla 6 Parámetros de la fórmula que utiliza el PID\_COMPACT [4]

### 3.1 MODO DE FUNCIONAMIENTO MANUAL

Para elegir el modo de funcionamiento manual, tenemos la opción de hacerlo desde la Interfaz de Labview mediante el selector que activa la marca %M0.5 o, poniendo en MAN el selector rotulado MODO dispuesto en el panel de la Planta de Nivel y por ende la entrada digital %I0.4, tal como se aprecia en el diagrama de flujo de la figura 3.3 y la programación en TIA PORTAL de la figura 3.4, adicionalmente se activa la salida %Q0.1 que energiza la luz piloto color azul de la Planta de Nivel rotulada como: Modo Manual ON.

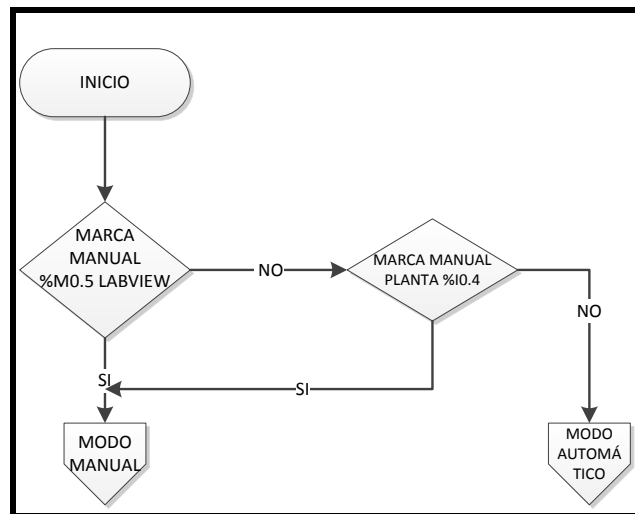


Figura 3.3 Diagrama de flujo selección modo de funcionamiento

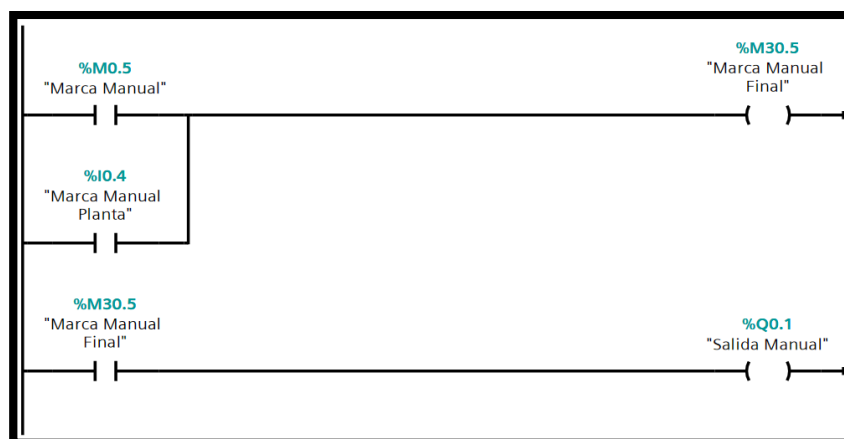


Figura 3.4 Programación en TIA PORTAL de selección modo de operación

Una vez que hemos seleccionado el modo de funcionamiento MANUAL para lo que la variable boolean %M30.5 debe tener el valor TRUE; ya podemos elegir la Fuente de Consigna de la señal analógica, mediante un selector en la Interfaz de Labview que maneja la variable

%M30.3-Fuente de Consigna o con el selector de 2 posiciones ubicado en la Planta de Nivel rotulado Consigna Planta que está conectado a la entrada %I0.5, el resultado de esta selección se almacena en %M30.4-Fuente de Consigna Final, basta con poner en TRUE una de las 2 opciones para que la salida %Q0.3 energice la luz piloto color azul en la Planta de Nivel, rotulada como Consigna Potenciómetro, pues ya teníamos seleccionado el Modo Manual y por lo tanto en TRUE la %M30.5-Marca Manual Final.

Para definir cuál de las 2 Fuentes de Consigna va a actuar, el potenciómetro 10 K $\Omega$  o el dial de la Interfaz de Labview;, utilizamos la función Seleccionar SEL de TIA PORTAL que trabaja así: si el valor de %M30.4 es FALSE , la variable %MD6-Apertura Manual que toma valores según posición del dial ubicado en la Interfaz de Labview, es asignada a %MD42-Valor Manual, por otro lado si el valor de %M30.4 es TRUE, la variable %MD38-Escalado Consigna es asignada a %MD42, esta variable %MD38 es el resultado de la limitación, normalización y escalado de la variable %IW100 que toma valores de la entrada analógica AI2,este proceso conforme a los diagramas de flujo de las Figura 3.5 y 3.7 y la programación en TIA PORTAL en Figuras 3.6 y 3.8.

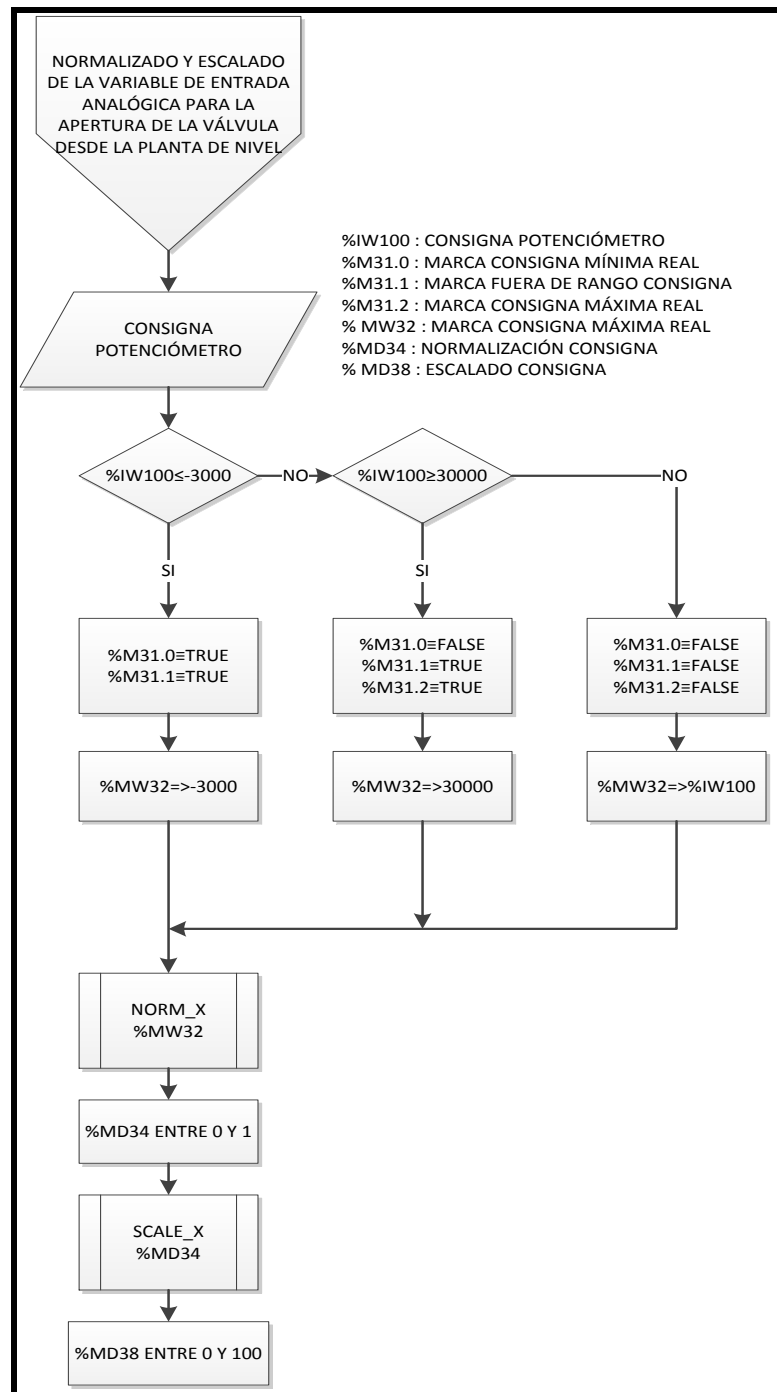


Figura 3.5 Diagrama de Flujo de limitación, normalización y escalado de consigna de potenciómetro



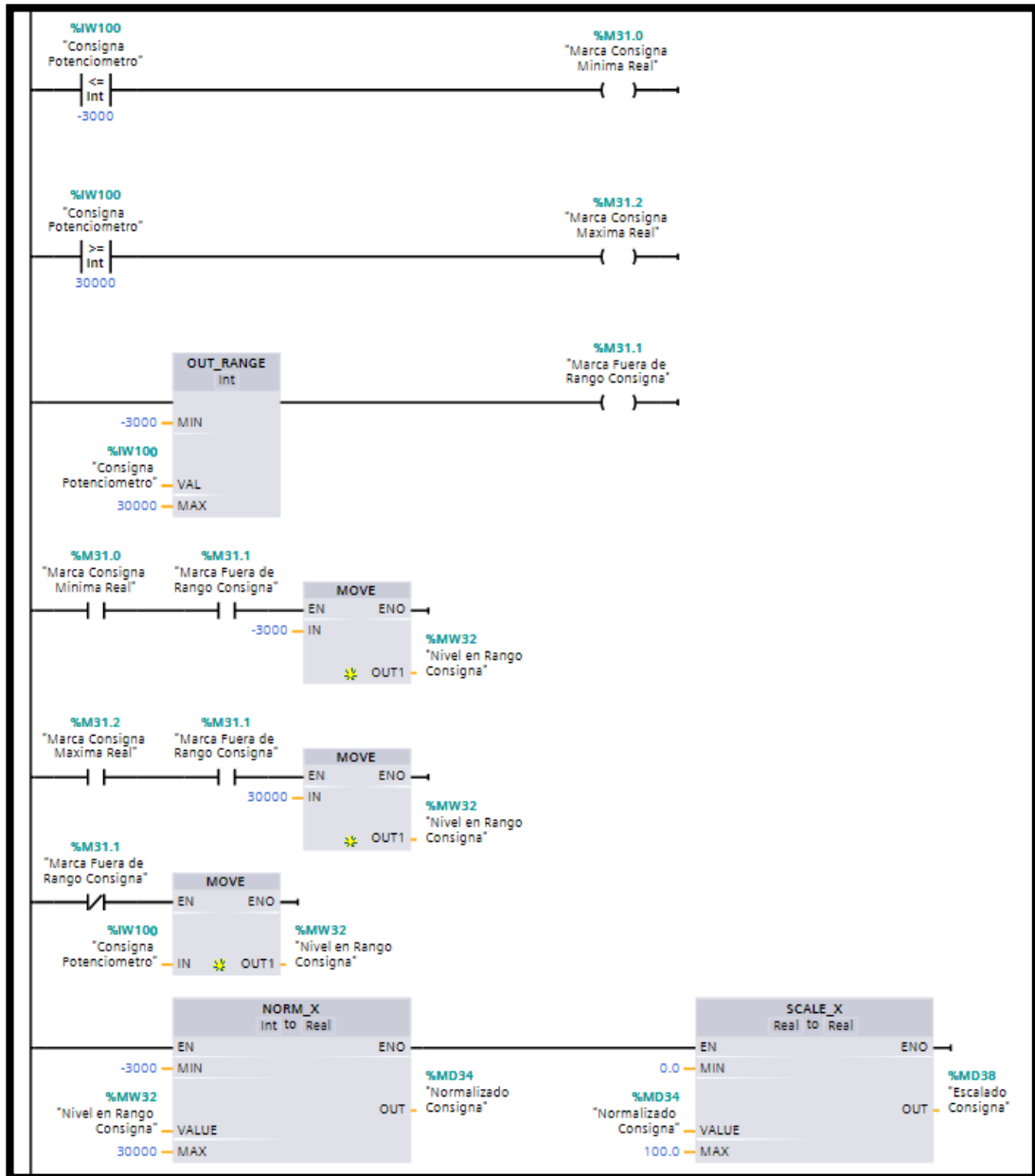


Figura 3.6 Programación en TIA PORTAL de limitación, normalización y escalado de consigna de potenciómetro

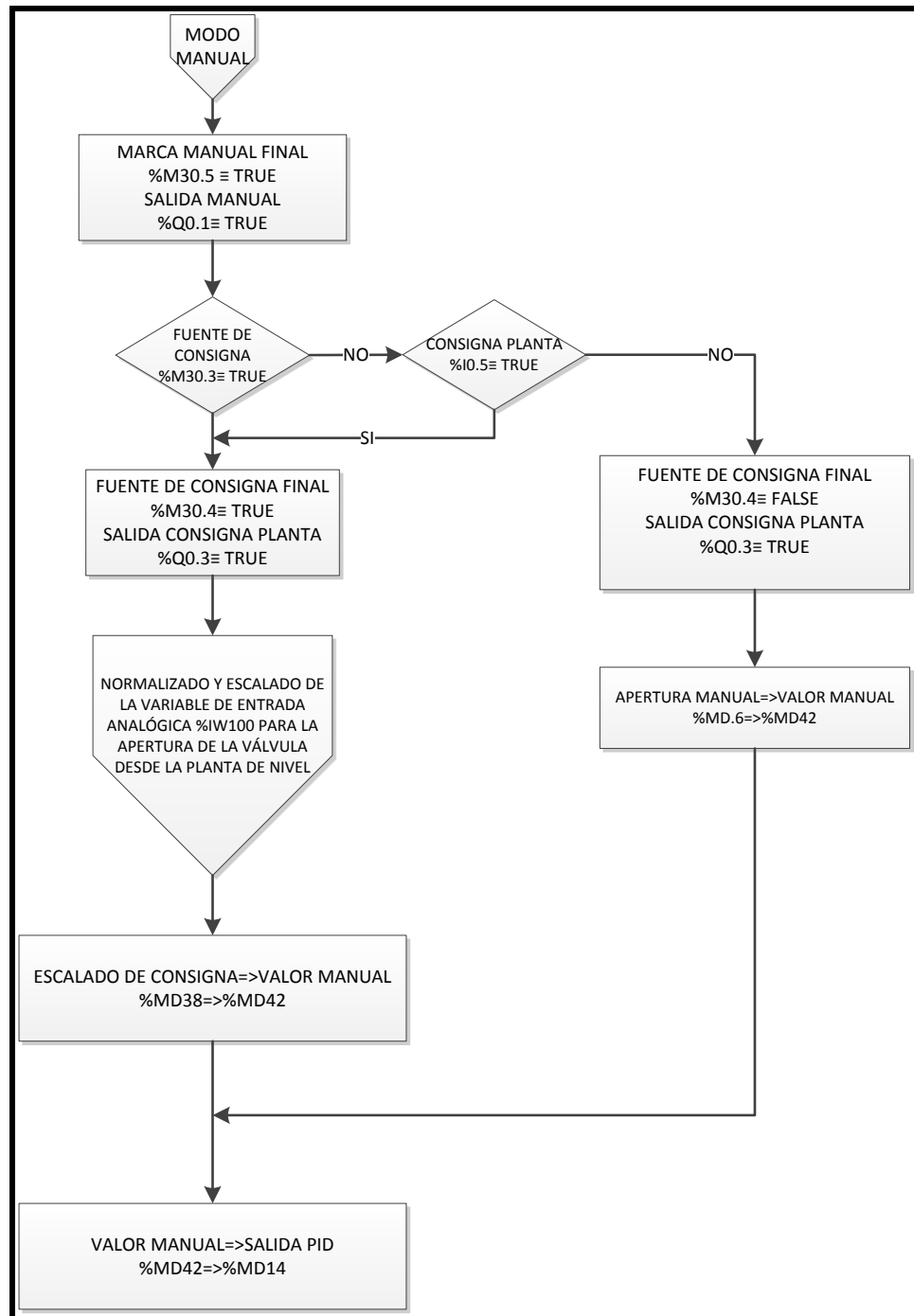


Figura 3.7 Diagrama de Flujo de selección de Fuente de Consigna y Manejo de la Salida Analógica en Modo Manual

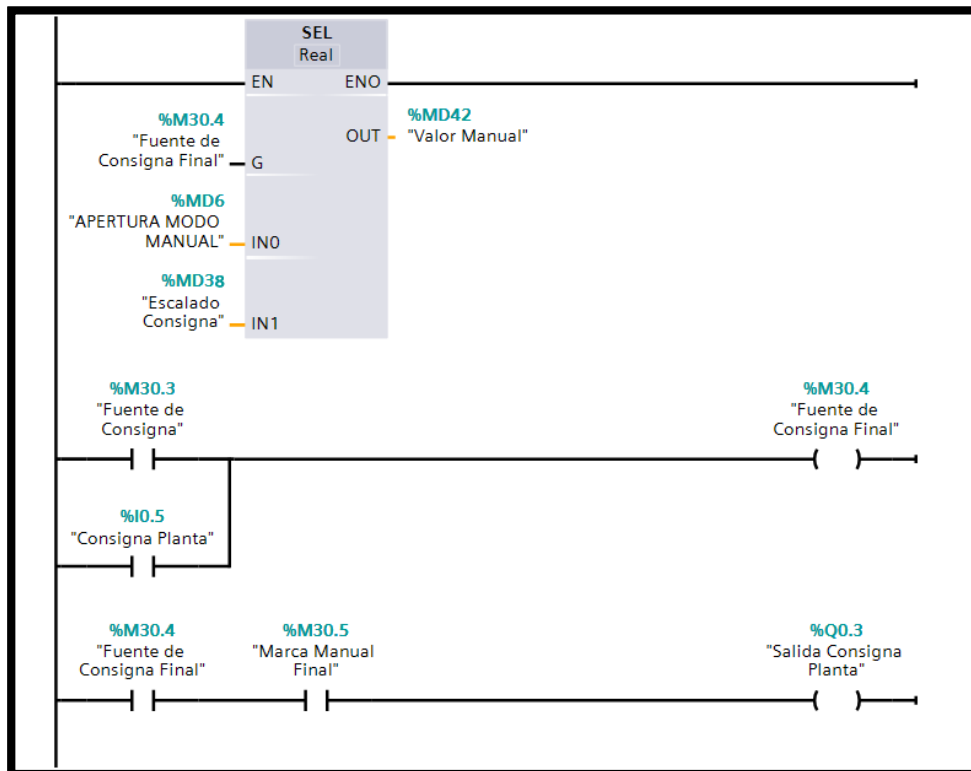


Figura 3.8 Programación en TIA PORTAL de selección de Fuente de Consigna y Manejo de la Salida Analógica en Modo Manual

Finalmente los valores analógicos que tiene %MD42 son asignados a la entrada ManualValue de la instrucción PID\_COMPACT, los valores de %MD42 se asignan a la salida del PID\_COMPACT %MD14, siempre y cuando la variable booleana %M30.5 este en TRUE pues de esta última depende el parámetro ManualEnable que activa o desactiva el modo de operación manual del PID\_Compact, esto se aprecia en la programación en TIA PORTAL de la Figura 3.9, a su vez %MD14 es la salida del PID\_Compact la cual después del proceso de normalizado y escalado

descrito en las Figuras 3.10 y 3.11, está direccionada a la variable analógica %QW96-Válvula la cual provee valores a la salida %AQ0 del módulo SM1234, que es la que físicamente se cablea al posicionador I/P electro neumático de la Válvula de Paso referida como RT450.20 en el manual de la Planta de Nivel GUNT RT450, adicionalmente en las Figuras 3.10 y 3.11 se aprecia un proceso en el que el %MD14-Salida PID se direcciona a %MD26-Escalado Salida Real de Válvula para poder visualizar en la Interfaz de Labview el porcentaje de apertura de la Válvula RT450.20, esto lo podemos en la Figura 3.12

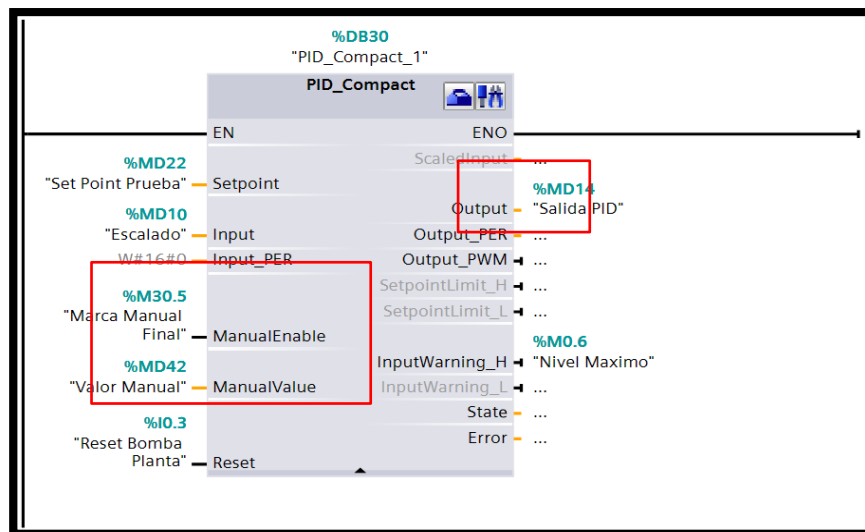


Figura 3.9 Programación en TIA PORTAL de la Salida Analógica en Modo Manual

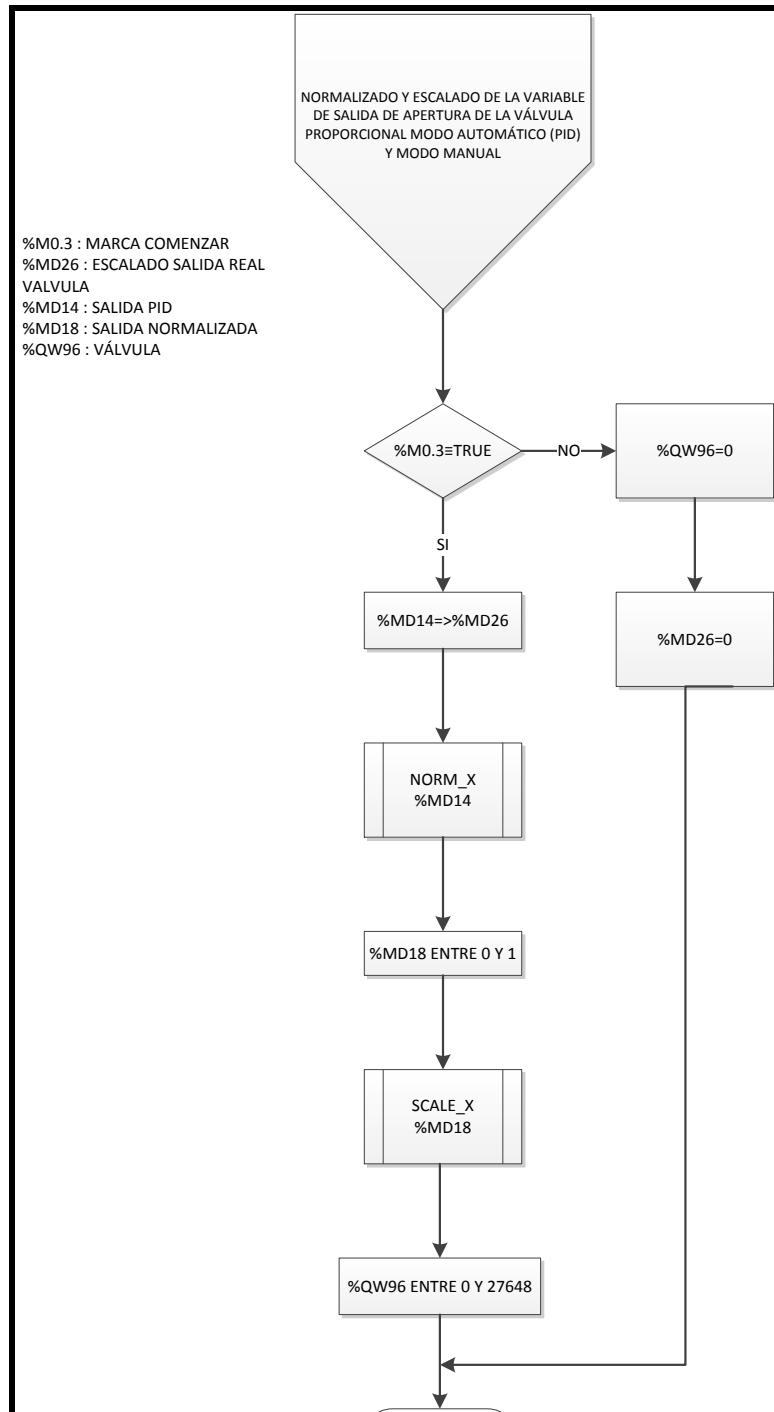


Figura 3.10 Diagrama de Flujo para normalización y escalado de la salida del Bloque PID\_COMPACT para Manual y Automático

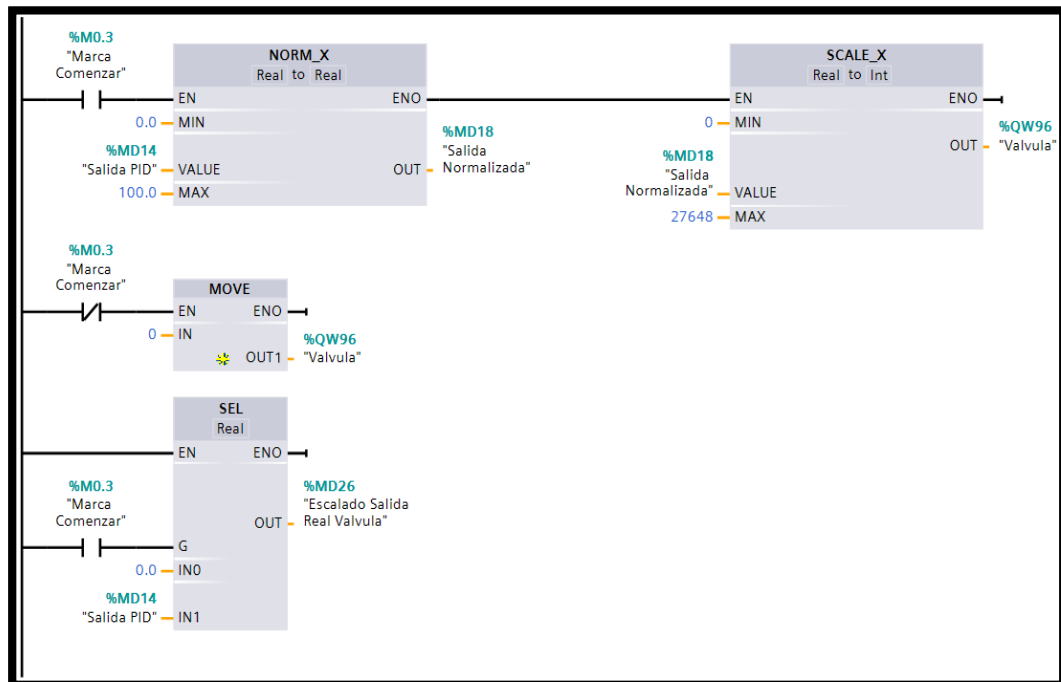


Figura 3.11 Programación en TIA PORTAL para normalización y escalado de la salida del Bloque PID\_COMPACT para Manual y Automático



Figura 3.12 Visualización en Labview de %MD26-Escalado Salida Real Válvula

### 3.2 MODO DE FUNCIONAMIENTO AUTOMÁTICO

Para elegir el modo de funcionamiento automático, lo hacemos de acuerdo al proceso descrito en las Figuras 3.3 y 3.4, para lo cual, tanto %M0.5-Marca Manual cómo %I0.4-Marca Manual Planta, deben tener el valor de FALSE, por lo tanto %M30.5-Marca Manual Final también estará en FALSE, de esta manera el parámetro ManualEnable del PID\_Compact estará desactivado el cual es su estado predeterminado, de esta manera la variable analógica %MD22-Setpoint Prueba, cuyos valores son ingresados por el usuario desde la Interfaz de Labview, Figura 3.13, es la encargada de definir el parámetro Setpoint del PID\_Compact, y el algoritmo indicado en la Ecuación (3.1) es el encargado de comparar los valores %MD10-Escalado del sensor de nivel capacitivo de código comercial ELOBAU 212 KKA 05 con el %MD22-Setpoint y en base a los valores  $K_p$ ,  $T_I$ ,  $T_D$ ,  $a$ ,  $b$ ,  $c$  descritos en la Tabla 2, calcular la variable %MD14-Salida PID, que en la fórmula es indicada como  $y$ . Podemos apreciar los parámetros que intervienen en el funcionamiento automático en la Figura 3.14.

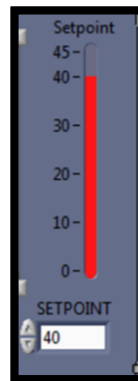


Figura 3.13 Segmento de la Interfaz de Labview para el ingreso del Setpoint

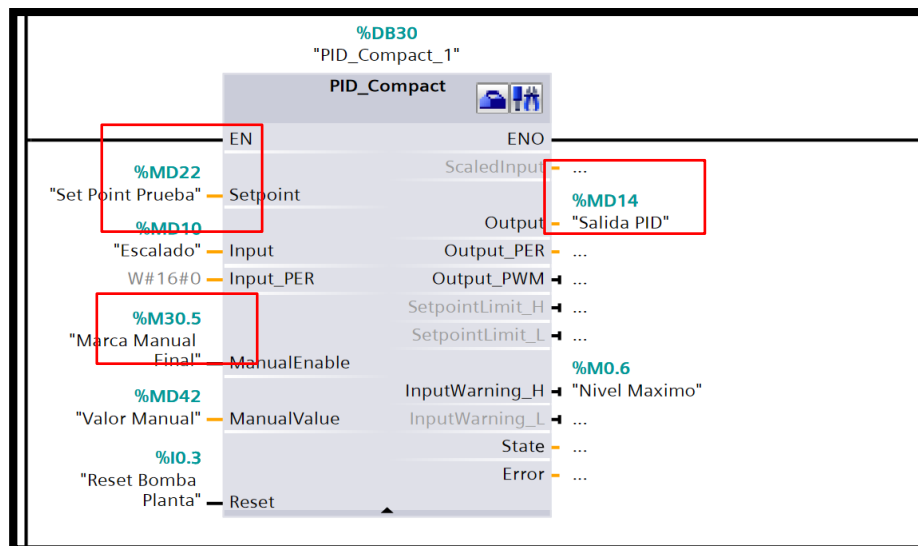


Figura 3.14 Programación en TIA PORTAL de la Salida Analógica en Modo Automático

Es necesario indicar que los parámetros PID:  $K_p$ ,  $T_I$ ,  $T_D$ ,  $a$ ,  $b$ ,  $c$ , para el presente proyecto, se obtuvieron utilizando la función de Optimización del Objeto Tecnológico PID\_Compact localizado en el submenú Ajustes Avanzados de dicho recurso, Figura 3.15



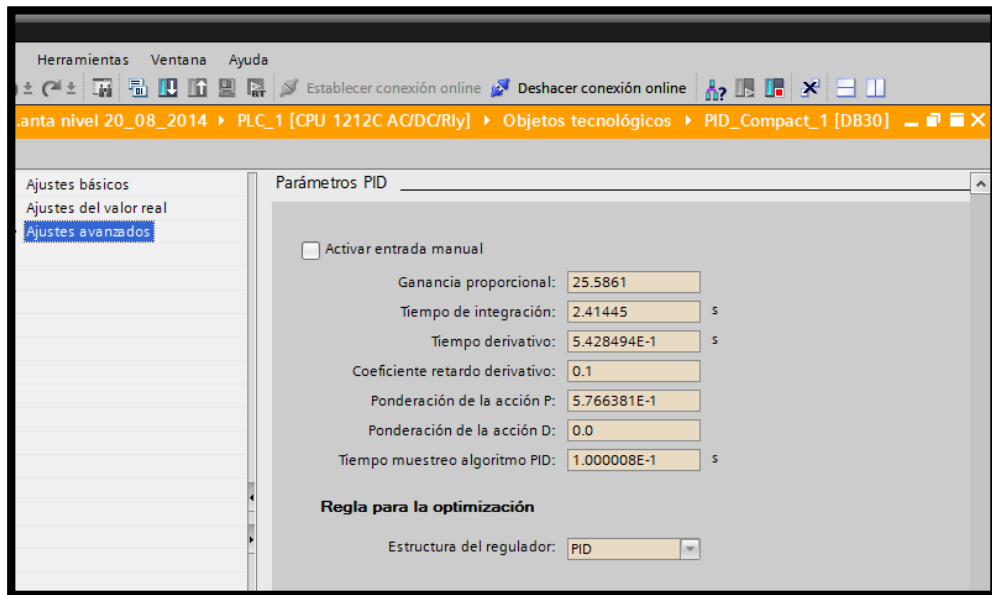


Figura 3.15 Segmento de la Interfaz de Labview para el ingreso del Setpoint.

Indicamos los resultados obtenidos en la optimización del PID\_COMPACT en la siguiente Tabla 7

Parámetro	Simbología	Valores
Ganancia Proporcional	$K_p$	25,861
Tiempo de Integración	$T_I$	2,41445 segundos
Tiempo derivativo	$T_D$	$5,42849 \times 10^{-1}$ segundos
Coeficiente de Retardo Derivativo	$a$	0,1
Ponderación de la acción P	$b$	$5,766381 \times 10^{-1}$
Ponderación de la acción D	$c$	0.0

Tabla 7 Resultados de la optimización del PID\_COMPACT

En base a los valores de la tabla 7 podemos escribir la ecuación en el dominio de la frecuencia, Ecuación (3.2) que describe el comportamiento del controlador PID\_COMPACT en modo automático para el presente proyecto y adicionalmente la Tabla 8 con el significado de las variables de la ecuación en términos de TIA PORTAL.

$$y = 25.861 * \left[ ((5.766391 * 10^{-1} * w) - x) + \left( \frac{(w - x)}{2.41445 * s} \right) + \left( \frac{(5.42849 * 10^{-1} * s)}{(0.542849 * 10^{-1} * s) + 1} \right) * (-x) \right]$$

(3.2) Descripción matemática del controlador PID\_Compact para este proyecto

Variable en la ecuación	Función que cumple	Variable en PID_COMPACT
y	Valor de Salida	%MD14-Salida PID
w	Consigna	%MD22-Setpoint Prueba
x	Valor de Proceso	%MD10-Escalado

Tabla 8 Significado de las variables de la ecuación en términos de TIA PORTAL

Es necesario explicar además las partes de la aplicación que se usan tanto en Modo Manual como en Modo Automático.

Para inicializar la aplicación, podemos hacerlo desde la Interfaz de Labview con la variable booleana %M0.0-Marca Run en TRUE o desde el panel de la planta de nivel accionando el pulsador verde rotulado como MARCHA que está conectado a la entrada digital del PLC %I0.6-Run

Lectura Escritura, para que esto sea posible previamente el selector de dos posiciones rotulado COMENZAR debe estar en posición ON y dicho elemento está conectado a la entrada digital %I0.0-Run Plc, al ser este un típico circuito de control con enclavamiento, la función de paro la cumplen: %M0.1-Marca Stop desde Labview y el pulsador rojo rotulado como PARO que está conectado a la entrada digital del PLC %I0.7-Stop Lectura Escritura, la finalidad de este circuito es controlar la activación o desactivación de la variable %M0.3-Marca Comenzar, la cual es sumamente importante para el resto de la aplicación, pues sirve para inicializar procesos tanto en modo automático como en modo manual, en la programación en TIA PORTAL detallada en Figura 3.16, al ponerse en TRUE %M0.3, activa la salida %Q0.0 la cual está conectada físicamente a la luz piloto amarilla rotulada como LECTURA ESCRITURA PLC ON, las luces piloto, pulsadores y selectores indicados en este párrafo se indican en la Figura 3.17. y los controles de la Interfaz de Labview que intervienen en este punto se muestran en la Figura 3.18

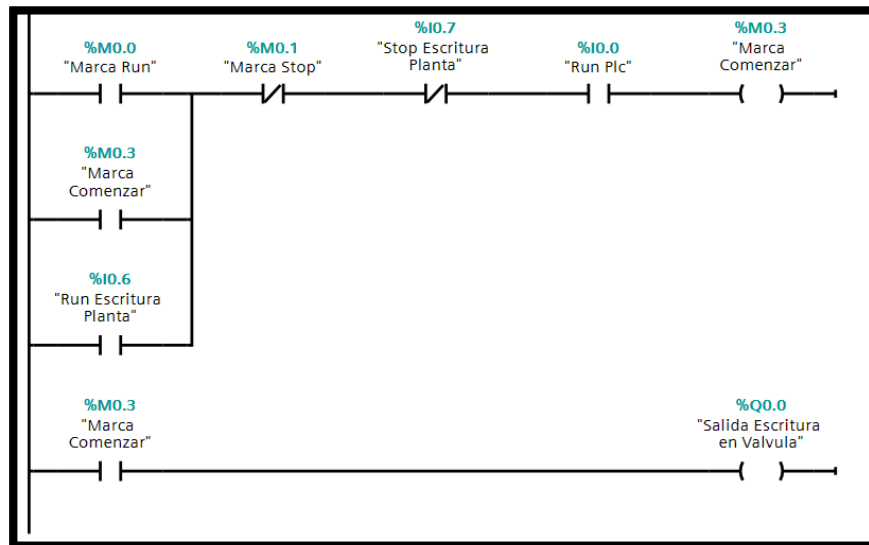


Figura 3.16 Programación en TIA PORTAL para Arranque y Paro de la aplicación desde Planta de Nivel y Labview



Figura 3.17 Selectores, pulsadores y luces piloto de la Planta de Nivel



Figura 3.18 Controles para Arranque y Paro desde aplicación en Labview

Para el arranque de la bomba que impulsa el líquido a través del sistema, tenemos la opción de hacerlo desde la Interfaz en Labview con la variable booleana %M0.2-Run Bomba Labview o desde la planta de nivel con el pulsador verde rotulado como Marcha Bomba Planta de Nivel, que está cableado a la entrada %I0.1 del PLC, con un pulso de cualquiera de las dos señales se energiza la bobina del contactor KM2 conectando la alimentación a la bomba monofásica LOWARA 2HM236/A, esto lo apreciamos en las Figuras 3.19 y 3.20

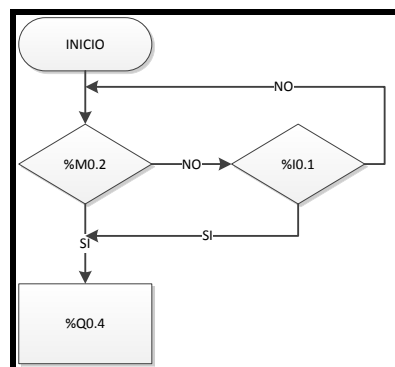


Figura 3.19 Diagrama de Flujo de arranque de bomba desde Labview o desde Planta

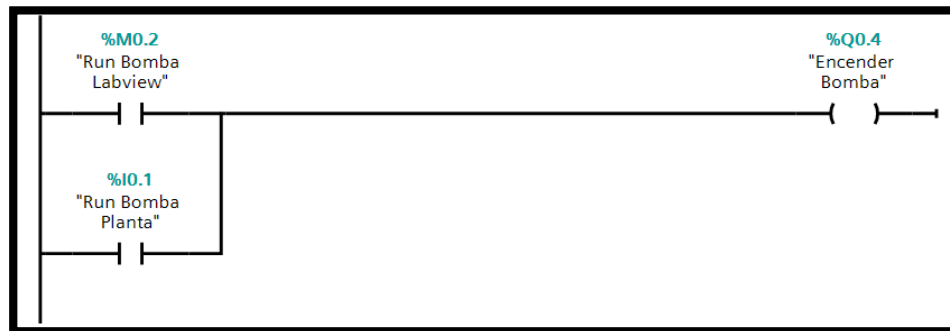


Figura 3.20 Programación en TIA PORTAL de arranque de bomba desde Labview o desde Planta

Para el apagado de la bomba LOWARA hemos previsto que se lo pueda hacer desde la Interfaz de Labview con la variable booleana %M1.2-Stop Bomba Labview, y desde la planta de nivel con el pulsador rojo rotulado STOP BOMBA PLANTA, que está cableado a la entrada del PLC %I0.2-Stop Bomba Planta además la bomba debe apagarse cuando se establece una condición de alarma para cuando el líquido en el tanque rebase cierto nivel máximo, hemos definido este valor en 43,23 centímetros debido a que el tanque de la planta de nivel tiene 45 centímetros y así prevenir un derrame del líquido, el valor de nivel máximo lo hemos configurado en el PID\_COMPACT el cual activa su salida InputWarning mediante %M0.6-Nivel Máximo para el valor indicado, este %M0.6 a su vez activa la variable %M0.7-Marca Nivel Máximo, que cuando está en TRUE, provoca que la salida del PLC %Q0.2-Salida Nivel Máximo energice a la luz piloto roja en la planta

rotulada como ALARMA NIVEL ALTO ON, se prevé además un circuito para resetear la condición de alarma, utilizando un pulsador rojo ubicado en la planta rotulado RESET ALARMA NIVEL ALTO, el cual está cableado a la entrada %I0.3-Reset Bomba Planta o desde la Interfaz en Labview con la variable %M1.0-Reset Bomba Hmi, cualquiera de las 2 opciones permite activar la variable %M1.1-Reset Bomba.

En cualquiera de los 3 casos que se genera la condición para que la bomba LOWARA se apague, ya sea por nivel máximo, apagado desde Interfaz en Labview o apagado desde Planta, la salida del PLC %Q0.2- Apagar Bomba se activa, interrumpiendo el circuito de alimentación al contactor KM2 en la Planta de Nivel. Estas condiciones de apagado, se aprecian en las Figuras 3.21 y 3.22.

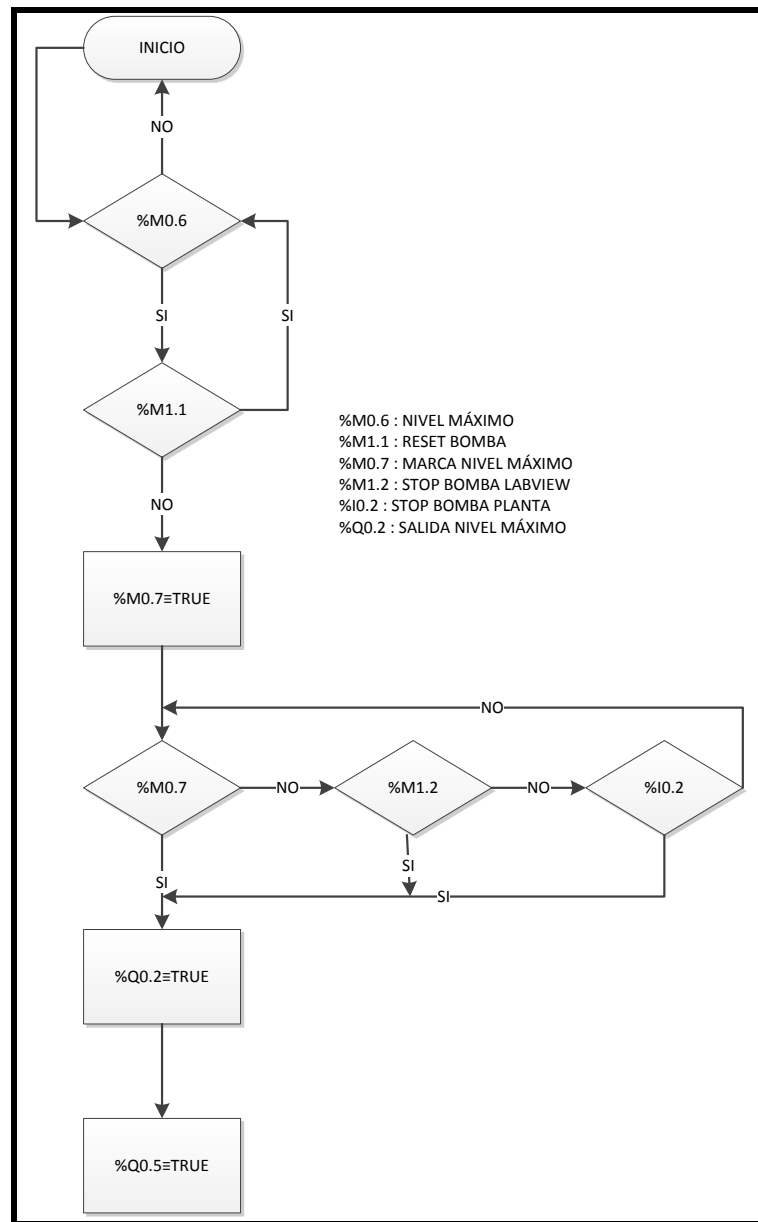


Figura 3.21 Diagrama de Flujo de condiciones de apagado y alarma para bomba

LOWARA



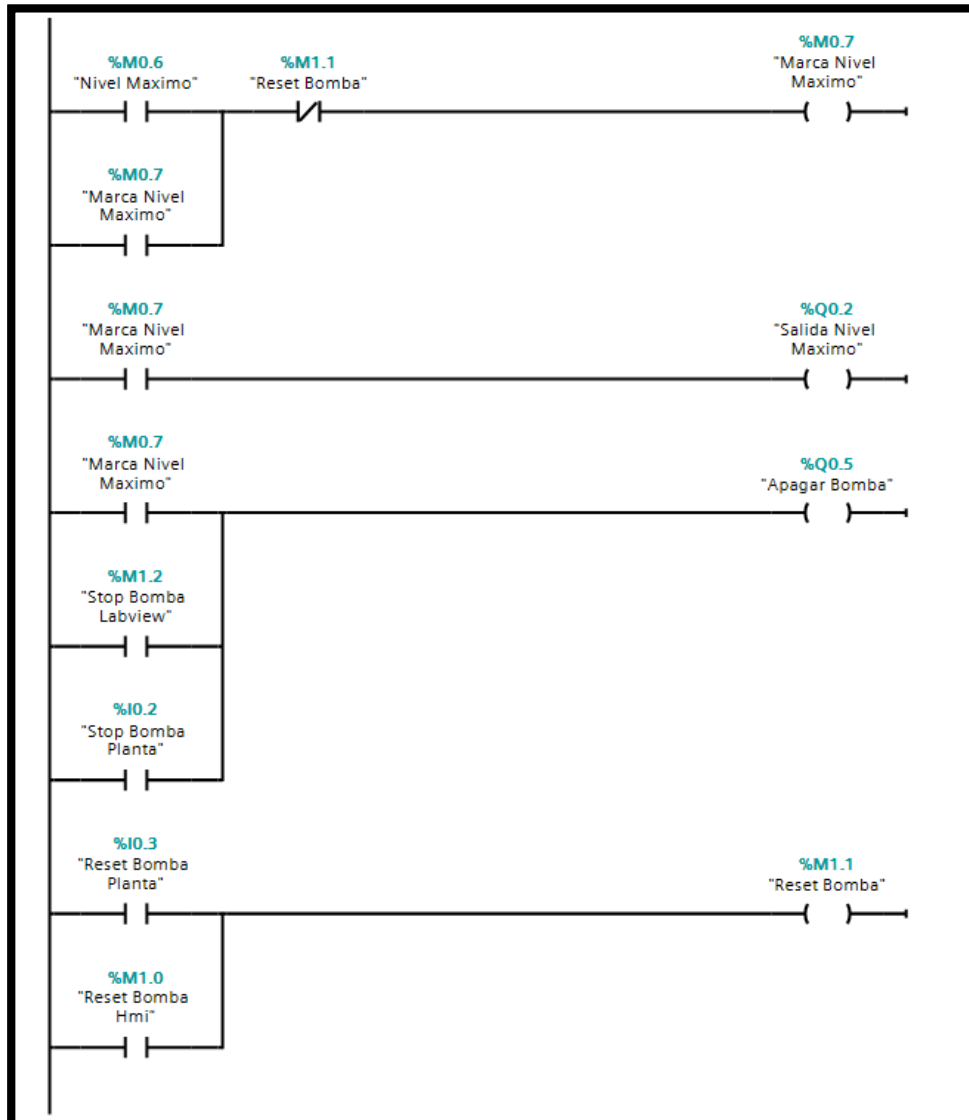


Fig 3.22 Programación en TIA PORTAL de condiciones de apagado y alarma para bomba LOWARA

## **CAPÍTULO 4**

### **DISEÑO DE INTERFAZ PARA USUARIO EN SOFTWARE LABVIEW.**

#### **4.1 CREACIÓN, CONFIGURACIÓN Y EDICIÓN DE VARIABLES.**

Procedemos a crear la Interfaz de Labview para el proyecto, necesitamos realizar una representación gráfica de todas las variables que intervienen en nuestro sistema. Para esto primero abrimos el programa Labview 2013 de National Instruments y seleccionamos Blank VI.

Una vez hecho esto guardamos el VI con el nombre PLANTA DE NIVEL. Comenzamos entonces a crear la Interfaz haciendo prácticamente un

dibujo de cómo queremos que se vea la planta. Siguiendo la ruta Tools-DSC Module-Image Navigator de la Figura 4.1 se abre una ventana donde tenemos gran cantidad de gráficos de elementos de la industria los cuales nos sirven para representar toda la planta.

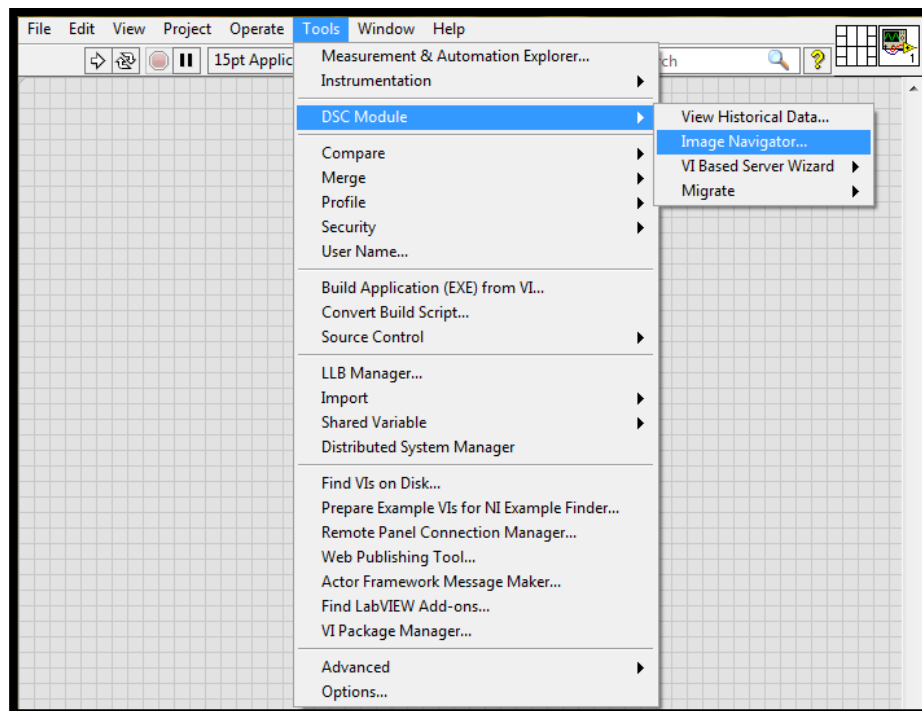


Figura 4.1 Ruta para seleccionar la ventana de Image Navigator

Creamos entonces un dibujo de la planta donde vamos a colocar los elementos que nos sirven para modificar las variables Booleanas y Analógicas. Tenemos en la Figura 4.2 un dibujo que representa a la bomba Lowara, un dibujo del tanque donde va a variar el nivel, un dibujo

del sensor de nivel Elobau, de la válvula de control neumático Samson, del reservorio de agua y de todas las tuberías que en la planta tenemos.

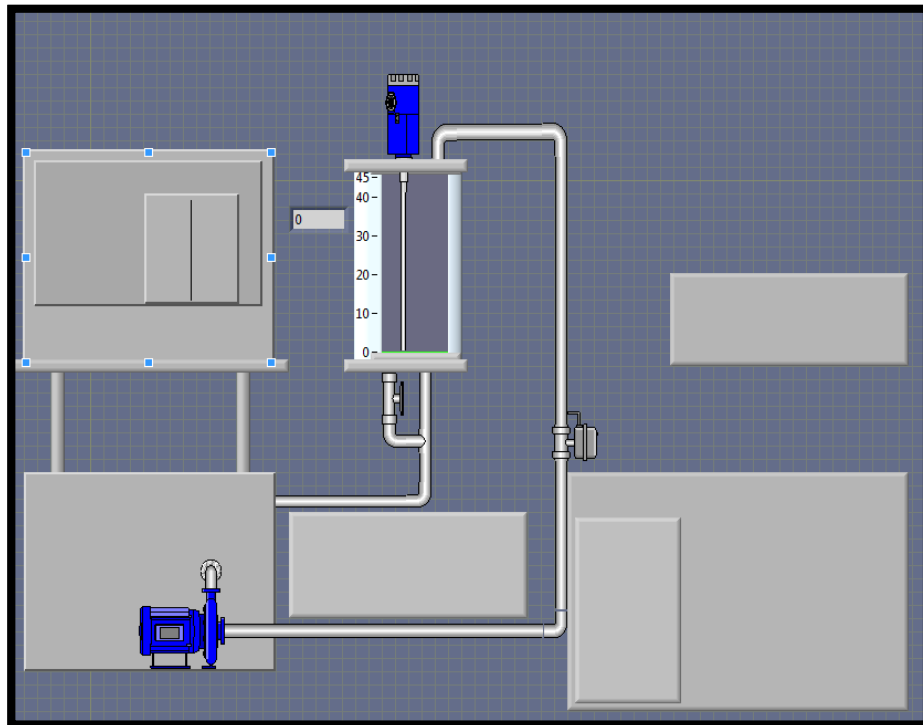


Figura 4.22 Imagen sólo de la planta.

En el tanque creamos un indicador tipo tank que al variar su entrada simula el llenado del mismo.

Como en nuestro programa tenemos dos modos de funcionamiento manual y automático, entonces creamos un selector Figura 4.3 para poder elegir que funcionamiento deseamos. Con el funcionamiento de

dicho selector también activamos 2 indicadores, cuando el selector este en la posición de **Automático** se enciende el indicador de ese lado y mientras que al estar el selector del lado **Manual** se enciende el correspondiente indicador.

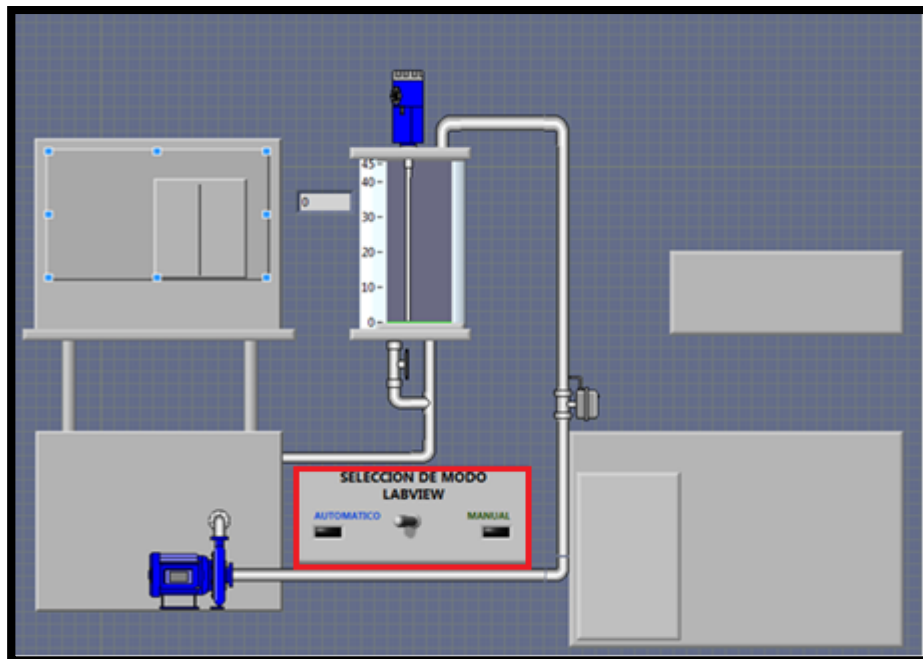


Figura 4.3 Selector para modo manual y automático

Una vez puesto el selector de modo manual y automático colocamos un selector para la fuente de consigna. Este selector nos permite variar la señal para la apertura de la válvula en el modo manual. Si seleccionamos **Entrada Analógica Física** controlamos la apertura de la válvula desde el potenciómetro de precisión que tenemos instalado en la planta de nivel

mientras que al seleccionar Labview controlaremos la apertura en modo manual de la válvula desde la interfaz que estamos diseñando. Para el control desde la interfaz añadimos un dial con variación de 0 a 100 que permitira abrir la válvula de 0 a 100%.

También colocamos un indicador que muestre que se cumplen las condiciones para que el dial maneje la apertura de la válvula, eso se dará cuando todas las condiciones previas se cumplan. El selector, dial y el indicador se los puede observar en la Figura 4.4.

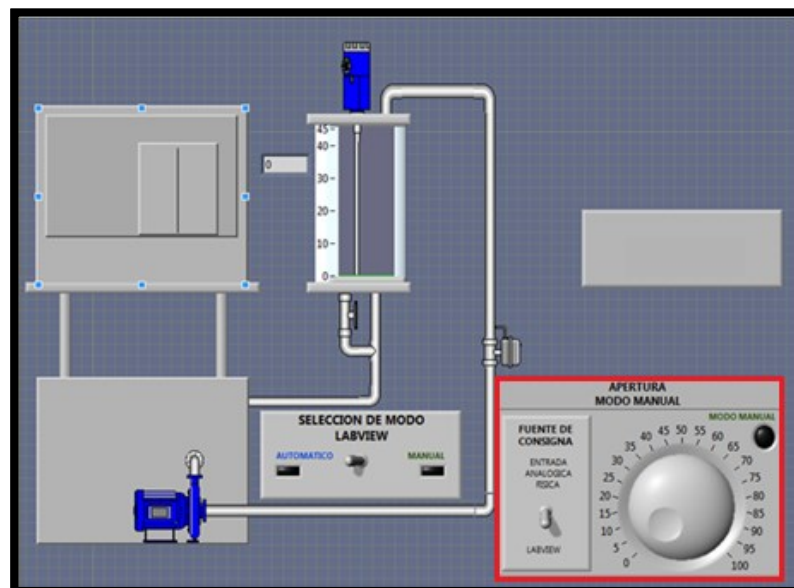


Figura 4.4 Selector para fuente de consigna, dial para apertura de válvula desde Labview e indicador de modo manual para el dial.

Ahora creamos un indicador llamado Porcentaje de Apertura de Válvula Real Figura 4.5. Este muestra en porcentaje el valor de apertura de la válvula. Este valor es la señal que se envía a la válvula para que abra, es decir no es una medición directa de la apertura de la válvula sino la apertura deseada de la misma. Para que esta apertura representada en el indicador sea correcta se tiene que garantizar que la válvula funcione y que la presión de aire en la misma sea la correcta. Se hace esta visualización indirecta ya que esta válvula no tiene una señal que nos indique su apertura sea esta de voltaje o de corriente.

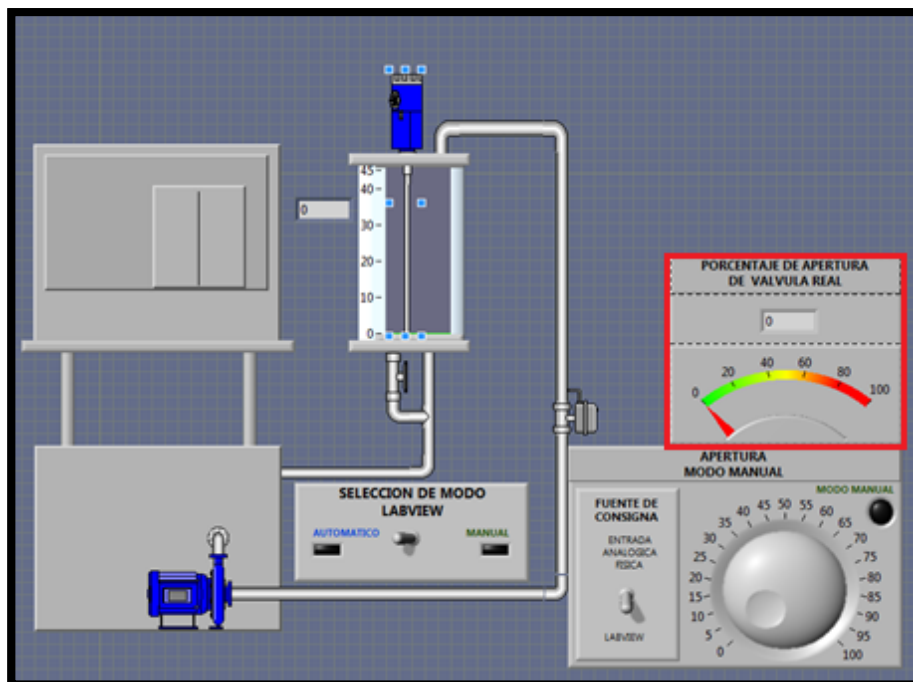


Figura 4.5 Indicador para porcentaje de apertura de válvula.

Necesitamos también enviar una señal de **Setpoint** al PLC. Este valor será el nivel que nosotros deseemos para nuestra planta y será un valor analógico que varíe entre 0 y 45cm que es el máximo nivel del tanque de la planta.

Para enviar este valor colocamos un control numérico, también colocamos una barra indicadora a un costado del tanque para poder comparar el nivel deseado y el nivel actual en el tanque Figura 4.6. Esto es una representación visual que facilita la comprensión del funcionamiento del sistema.

El nivel en el tanque se lo representa en color verde que es el color del líquido en la planta mientras que el valor del Setpoint en rojo para poder diferenciarlos.



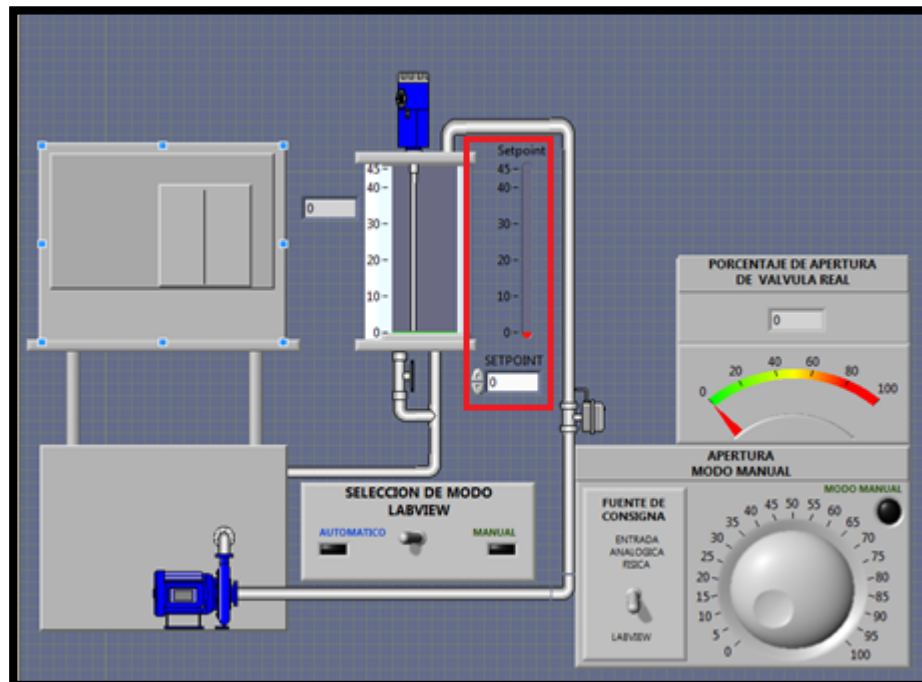


Figura 4.66 Control numérico y barra indicadora para valor de Setpoint.

También queremos controlar el encendido y apagado de la bomba desde la interfaz en Labview y para ello colocamos 3 pulsantes Figura 4.7. Los pulsantes son **Run**, **Stop** y **Reset Bomba**. El pulsante **Run** sirve para el encendido de la bomba, el **Stop** para apagarla en cualquier momento y el pulsante **Reset Bomba** que sirve para resetear la falla que se da cuando el nivel del líquido sobrepase el nivel máximo que en la lógica del PLC esta en 43,23cm, esta falla apaga la bomba. Mientras esa falla este activa no se puede volver a encenderla.

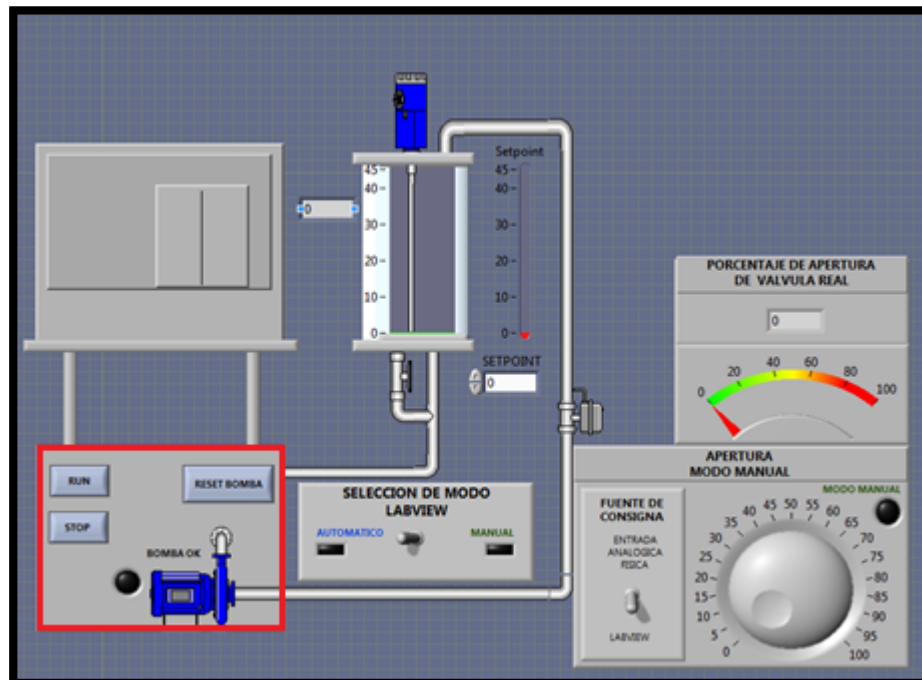


Figura 4.7 Run, Stop y Reset para el funcionamiento de la bomba.

En la lógica que radica en el PLC tenemos dos señales que nos sirven para activar o desactivar la escritura de datos en la válvula, también un bloqueo o llave que en un caso real en una fábrica sería un código de acceso para restringir el manejo del PLC por parte de usuarios indebidos. Para implementar dichas señales en la interfaz de Labview colocamos 2 pulsantes y 3 indicadores Figura 4.8 . Los pulsantes son **Run** y **Stop**. En cambio los indicadores son **On** y **Off** que muestran que esta activa o no la escritura de datos. También un indicador llamado **Llave** que sirve para lo anteriormente indicado.

También colocamos el control llamado **STOP** para detener el lazo while.

Lazo cuyo periodo sera de 100ms.

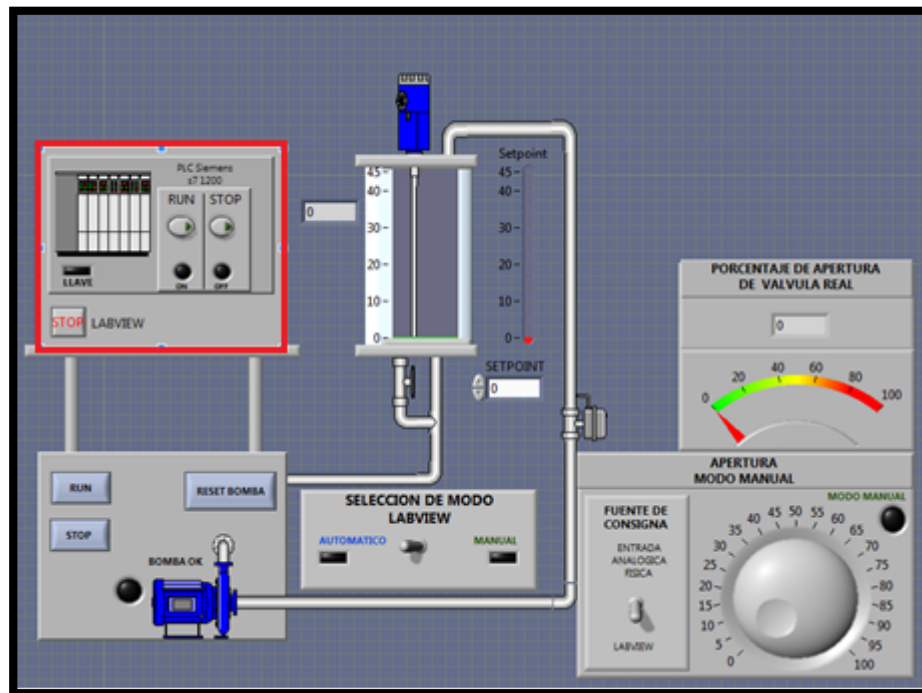


Figura 4.8 Run, Stop, Llave, representación del PLC y Stop del lazo while.

Finalmente El pulsante **Reset Bomba** que mencionamos anteriormente sirve para resetear la alarma de nivel. Esta alarma de nivel la representamos con un indicador llamado **Nivel Maximo** Figura 4.9.

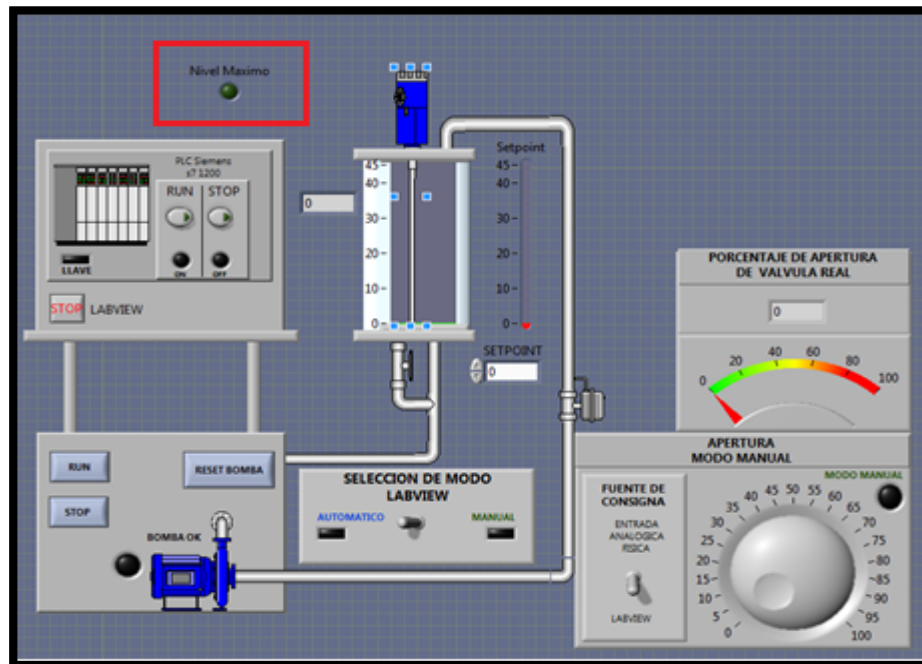


Figura 4.9 Indicador de Nivel Máximo.

## 4.2 DISEÑO DE VI PARA MONITOREO Y AJUSTE DE PARÁMETROS

Teniendo las variables booleanas y analógicas completas vamos a monitorearlas, para hacerlo debemos mostrar su comportamiento gráficamente. Hemos dispuesto 3 gráficas las cuales las ponemos dentro de un Tab Control que es un elemento que permite disponer de varias ventanas dentro de un mismo espacio. Las gráficas son:

**SETPOINT:** En esta gráfica vamos a monitorear el nivel del líquido en el tanque y lo vamos a comparar con el valor que ingresamos en Setpoint. Mandamos las 2 señales a un Waveform Chart que un elemento de

Labview que permite graficar las entradas en función del tiempo. Figura 4.10.

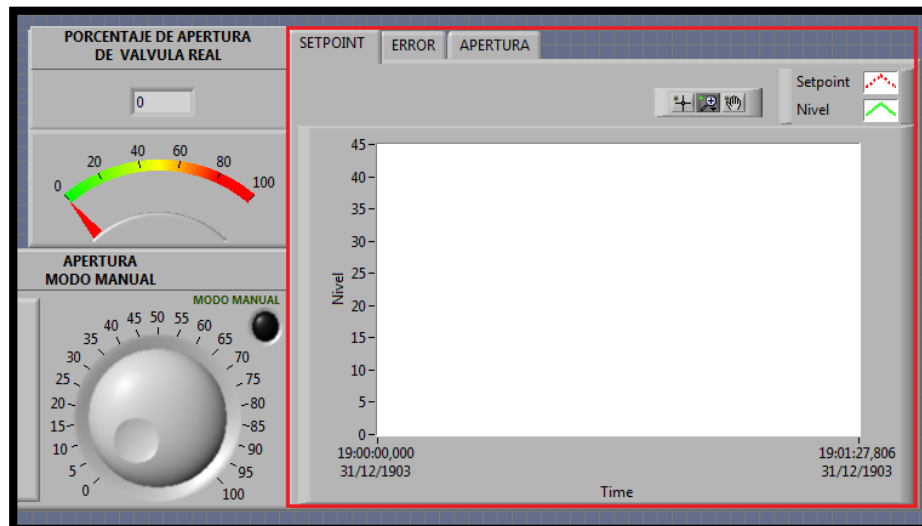


Figura 4.10 Gráfica SETPOINT

**ERROR:** En esta gráfica se compara % Error Actual vs % Error Admitido. % Error Actual es un valor que viene dado de la comparación del valor del Setpoint que damos con la lectura de nivel del líquido en el tanque. % Error Admitido es un valor que ingresamos por medio de un control numérico. Figura 4.11

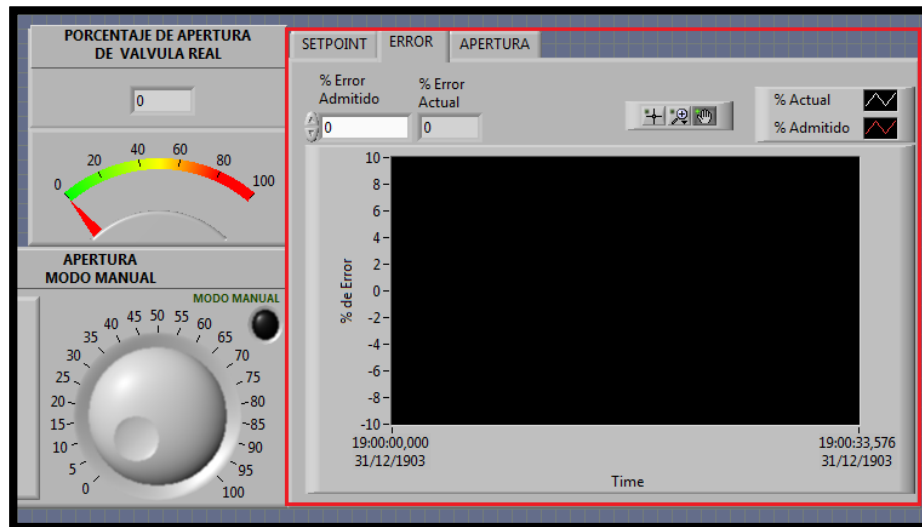


Figura 4.11 Gráfica ERROR

**APERTURA:** En esta gráfica el valor a visualizar es la apertura de la válvula. En esta gráfica solo tendremos un valor para visualizar y no se compara con ningún otro valor. Figura 4.12

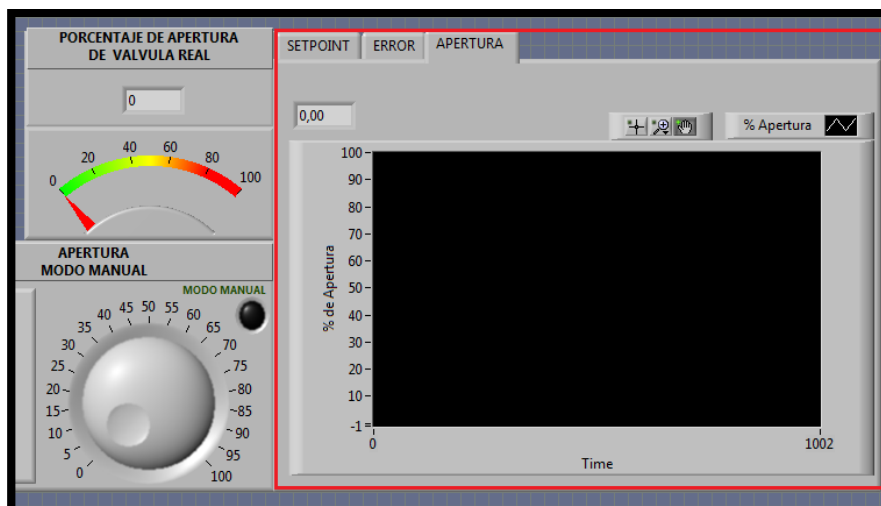


Figura 4.12 Gráfica APERTURA

### **4.3 EDICIÓN DE NOTAS EXPLICATIVAS SOBRE ELEMENTOS DEL INTERFAZ PARA USUARIO.**

Es necesario crear notas explicativas sobre algunos elementos dentro de la interfaz de nuestro programa. Las notas explicativas son muy importantes ya que en ellas podemos indicar cuál es el funcionamiento del sistema, datos de catálogo importantes y valores de referencia que para el caso de un usuario que recién comience a familiarizarse con el sistema le resultarían de mucha ayuda.

Para crear las notas explicativas creamos varios pulsantes cerca de los elementos a los cuales les vamos a poner la ayuda. Para nuestra interfaz estos pulsantes serán de color fucsia que es un color llamativo y se diferencia de los demás elementos.

También creamos los String que son cadenas de caracteres. Modificamos las propiedades de estos String para poder modificar su visibilidad. Esto lo logramos dando clic derecho sobre el ícono de cada String y escogiendo Create-Property Node-Visible.

Esta propiedad la activamos o desactivamos por medio de un control booleano. Este control booleano será cada pulsante que creamos. Como

ejemplo tomemos la válvula Samson con un pulsante a su lado. Cuando se presione el pulsante fucsia a su lado se despliega el texto de ayuda asociado a ese pulsante mostrando para la válvula sus datos. Observamos en la figura 4.13 la programación del texto explicativo de la Samson.

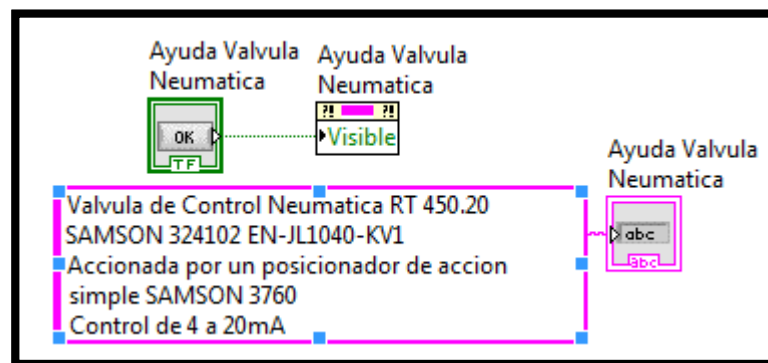


Figura 4.13 Programación para el texto explicativo de válvula Samson

En la figura 4.14 tenemos la imagen de la interfaz de Labview con todo lo que hemos agregado en este capítulo y esta será la base para ingresar las variables que crearemos en el siguiente capítulo.



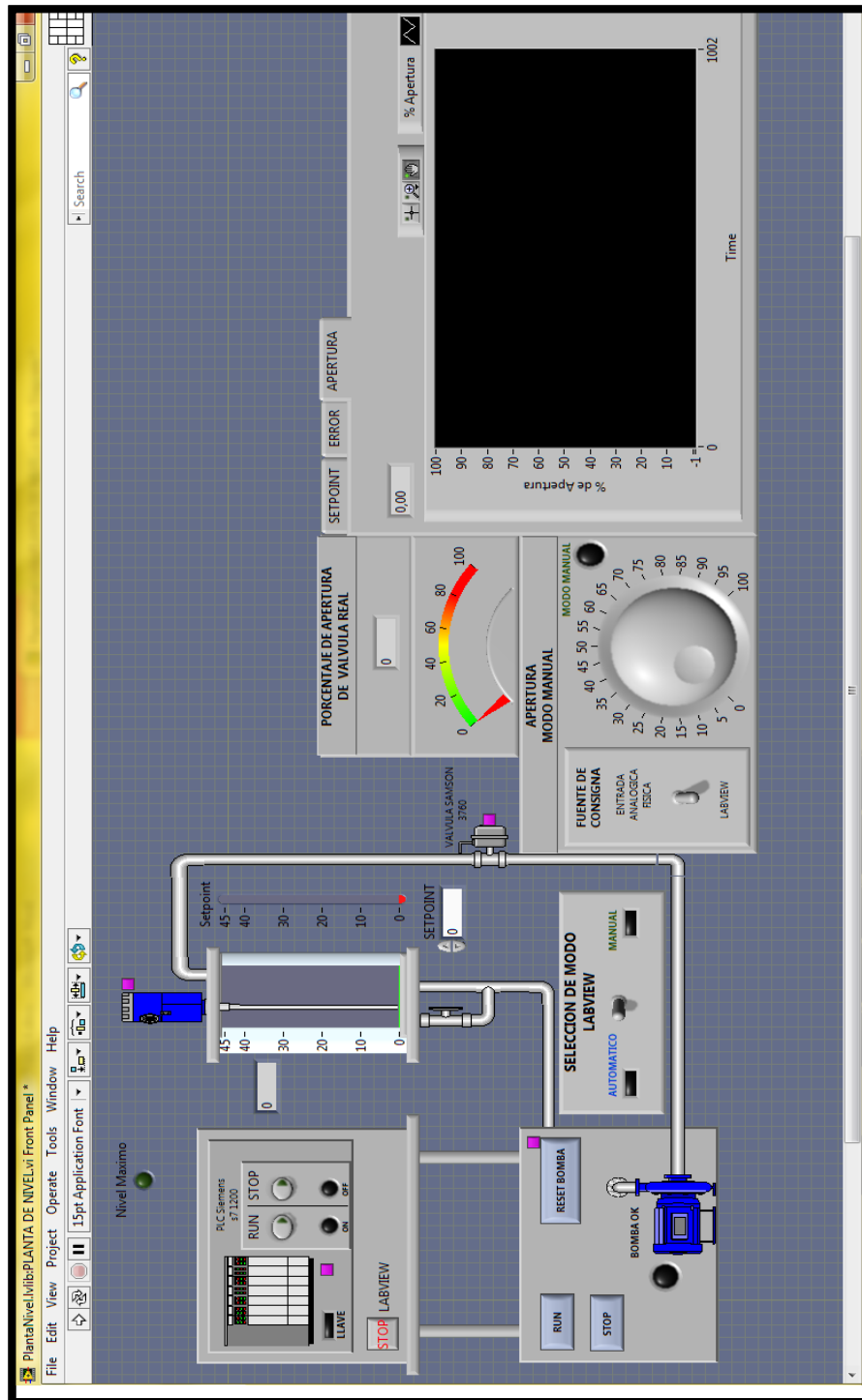


Figura 4.14 Interfaz de Labview completa

## **CAPÍTULO 5**

# **CONFIGURACIÓN DE LA COMUNICACIÓN ENTRE PLC SIMATIC S7 1200 E INTERFAZ PARA USUARIO MEDIANTE OPC.**

### **5.1 ARQUITECTURA CLIENTE SERVIDOR EN OPC**

**OPC:** es el estándar para el intercambio seguro de datos en el espacio de la automatización industrial, garantizando un flujo continuo de información entre dispositivos de diferentes marcas y proveedores, la Fundación OPC es la responsable del desarrollo y mantenimiento de la norma, la cual se nutre de especificaciones diseñadas por proveedores industriales, usuarios finales, estas especificaciones definen la interfaz

entre clientes y servidores, e incluyen el acceso a los datos en tiempo real, monitoreo de alarmas y eventos, acceso a datos históricos [9]. De hecho la comunicación OPC la podemos visualizar como una caja negra que se sitúa entre la Fuente de Datos y el Cliente de Datos que les permite intercambiar información sin saber el uno del otro, a pesar de lo cual los respectivos protocolos nativos siguen siendo necesarios. Figura 5.1.

**Un Servidor OPC:** es básicamente una aplicación de software diseñada para cumplir con una o más especificaciones OPC, al hablar de Servidor no nos referimos al computador donde el software se está ejecutando, en realidad este Servidor se relaciona con el Cliente OPC. Podemos establecer un símil entre un Servidor OPC con un traductor que se relaciona con los protocolos nativos de una Fuente de Datos, la estructura está diseñada para que funcione en forma bidireccional es decir: un Servidor OPC puede leer y escribir en una Fuente de Datos, es necesario dejar en claro que la estructura Servidor OPC/Cliente OPC es tipo maestro-esclavo por lo cual el Servidor OPC solo lea/escriba datos a una Fuente de Datos si el Cliente OPC así lo pide. Los servidores OPC se pueden comunicar con PLCs, RTUs, instrumentos de medición, bases de datos, páginas web en resumen con cualquier Fuente de Datos en la cual se pueda leer o escribir por medios electrónicos. [10] Tal como

indicamos en la Figura 5.1 un Servidor OPC integra los siguientes componentes:

**Módulo de comunicaciones OPC:** Es la parte del Servidor OPC que se comunica directamente con un Cliente OPC. [10]

**Módulo de comunicaciones nativas:** El Servidor OPC para comunicarse con la Fuente de Datos puede hacerlo mediante el protocolo propietario de datos de dicha Fuente, en otros casos mediante un Interfaz de Programación de la Aplicación (API), esto depende de que la experiencia que tenga el diseñador del Servidor OPC para que identifique cuál de las 2 opciones debe seleccionar. [10]

**Módulo de traducción/mapeado:** Este módulo interpreta las instrucciones OPC entrantes desde el Cliente OPC, y las traduce a requerimientos nativos que se envían a la Fuente de Datos y viceversa. [10]

**Cliente OPC** un módulo de software que trabaja como un destino de datos, inicia y controla la comunicación con Servidores OPC en base a los requerimientos recibidos desde la aplicación en la que está embebido

el Cliente OPC en el caso del presente proyecto de graduación la aplicación a la cual nos referimos es la Interfaz diseñado en Labview, El Cliente OPC traduce las peticiones de comunicación que vienen de la Interfaz de Labview en la petición OPC equivalente y la envían al Servidor OPC para que la procese, como esta comunicación es bidireccional [10]; se produce el proceso inverso cuando los datos vuelven del Servidor OPC relacionado con el PLC Simatic S7-1200, el Cliente OPC los traduce al formato nativo de la aplicación que es la Interfaz diseñado en Labview. Tal como se indica en la figura 5.1 podemos entender mejor el funcionamiento detallado anteriormente si lo visualizamos como 3 módulos.

**Módulo de comunicaciones OPC:** Es importante para que el tanto el Cliente OPC como el Servidor OPC se comuniquen y puedan desconectarse sin desestabilizar al Servidor OPC. [10]

**Módulo de comunicaciones con la aplicación:** El Cliente OPC está diseñado para trabajar con la aplicación, en nuestro caso con la Interfaz diseñado en Labview, por lo cual para que los datos pasen de la Interfaz al Servidor OPC pasando por el Cliente OPC se necesita se trabaja con la Interfaz para la programación de la aplicación (API). El Cliente OPC

puede comunicar con la aplicación mediante un protocolo, siempre y cuando este protocolo este soportado por la aplicación. [10].

**Módulo de traducción/mapeado:** Sirve para traducir de forma bidireccional la información que la Interfaz diseñada en Labview necesita leer o escribir al PLC Simatic S7-1200

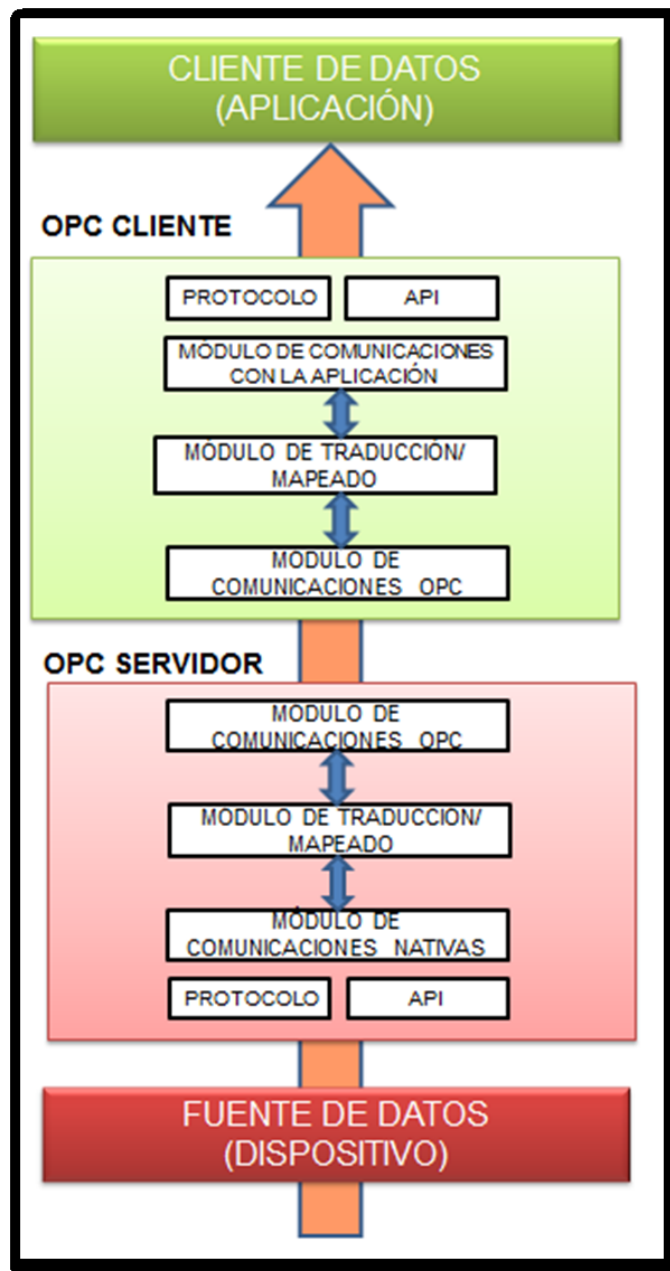


Figura 5.1 Arquitectura Cliente-Servidor OPC

## 5.2 CREACIÓN DE CANAL DE COMUNICACIÓN.

Procedemos a abrir la aplicación NI OPC Servers y hacemos clic en Add a channel, aparece el cuadro de diálogo New Channel-Identification,

damos el nombre del canal en este proyecto, Channel Name: Siemens, clic en Siguiente. Figura 5.1.

Es necesario remarcar que el nombre de canal de comunicación puede tener de 1 hasta 256 caracteres de longitud, los nombres no pueden contener puntos, comillas dobles o comenzar con un sub-guión, tal como se aprecia en Figura 5.2.

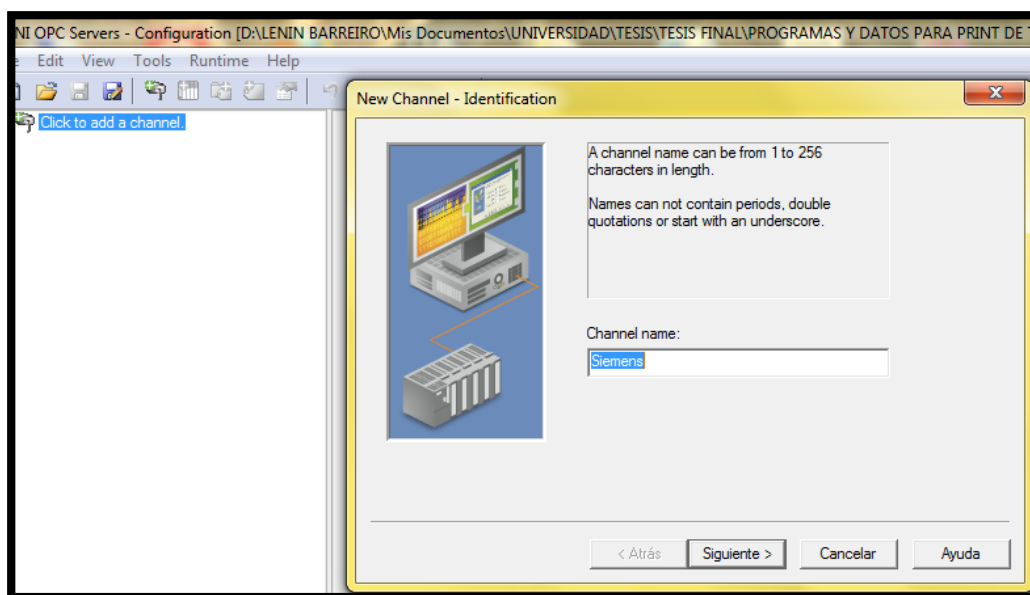


Figura 5.2 Asignación de nombre de canal de comunicación

### 5.3 SELECCIÓN DE TIPO DE COMUNICACIÓN

El Device Driver del dispositivo nos permite indicar el tipo de comunicación que vamos a usar de acuerdo al dispositivo con que



necesitamos comunicar para el canal de comunicación creado, para este proyecto seleccionamos Siemens TCP/IP Ethernet y damos clic en siguiente. Figura 5.3.

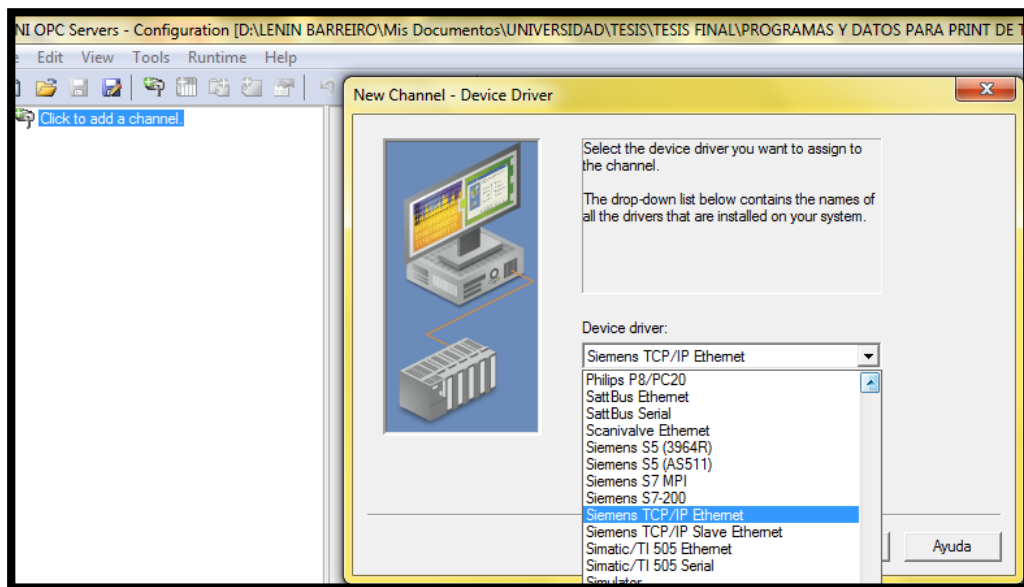


Figura 5.3 Asignación de tipo de comunicación.

Seleccionamos el adaptador de red, para este proyecto utilizamos el que corresponde al computador que se está utilizando: Intel (R) Wifi Link. Figura 5.4, damos clic en siguiente.

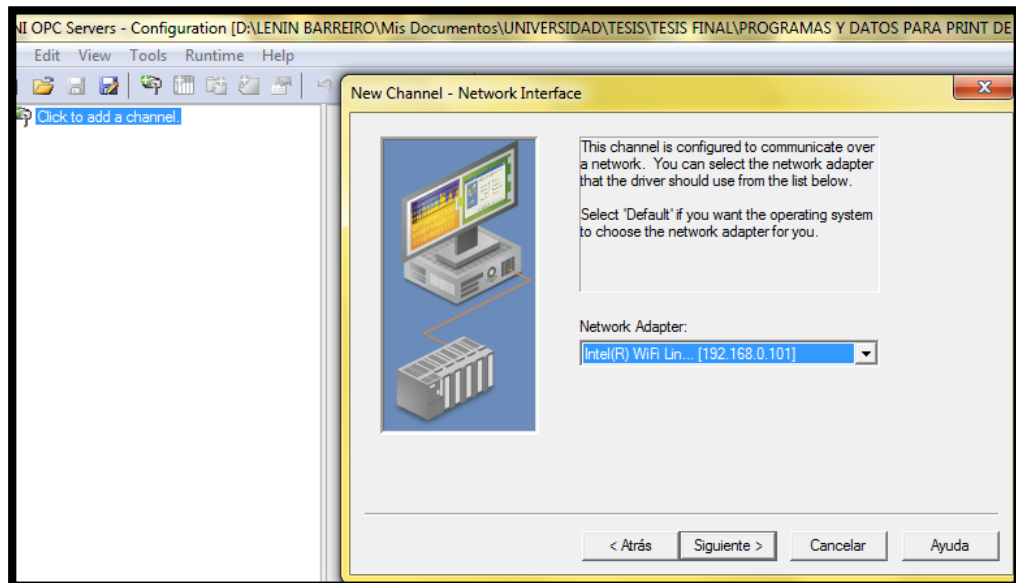


Figura 5.4 Selección del adaptador de red.

Aparece el cuadro de diálogo de Optimización de escritura (Write Optimizations), dicho cuadro nos indica que podemos controlar como el servidor procesa la escritura en el canal creado, se elige el método de optimización y cuantas lecturas por cada escritura. El método de optimización seleccionado es el de escribir solo los últimos valores para todos los tags, y el Duty Cycle lo fijamos en 10 escrituras por cada lectura. Figura 5.5 y damos clic en siguiente.

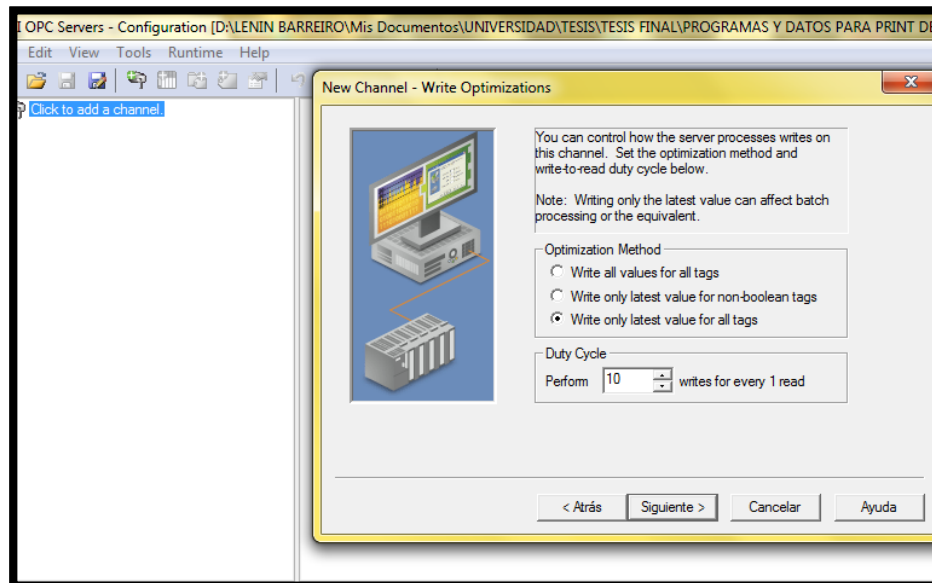


Figura 5.5 Optimización de Escritura

Ahora tenemos la selección del manejo de valores de No normalizados de Punto Flotante, para ello es preciso indicar que los números en Punto Flotante es una notación científica que el CPU usa para modelar a los números reales muy grandes o muy pequeños de forma compacta, por ejemplo tenemos el número  $0.12 \cdot 10^{-5}$  en el cual tenemos un dígito a la izquierda del punto decimal, en el caso anterior el número no está normalizado, pero si lo escribimos como  $1.2 \cdot 10^{-6}$  ya están normalizado, entonces decimos que el número de Punto Flotante está normalizado cuando el número a la izquierda del punto decimal es distinto de cero.[2], en nuestro caso, elegimos reemplazar por ceros todos los valores no-normalizados, pues la otra opción es no modificar y trabajar con valores valores no-normalizados. Figura 5.6

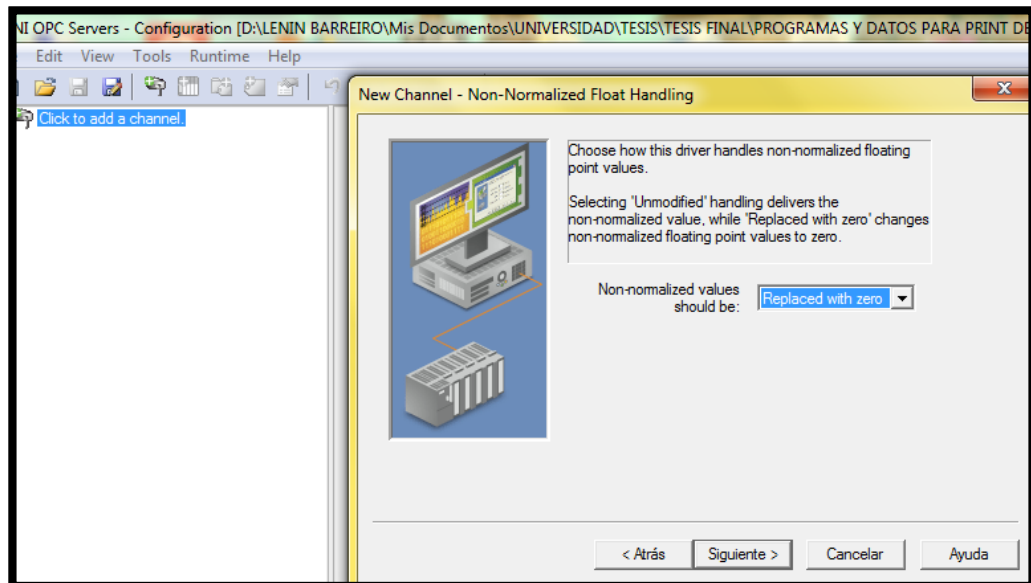


Figura 5.6 Manejo de números no-normalizados de Punto Flotante

Finalmente tenemos un resumen de los parámetros configurados para el tipo de comunicación seleccionada para el canal Siemens Figura 5.7, damos clic en Finalizar.

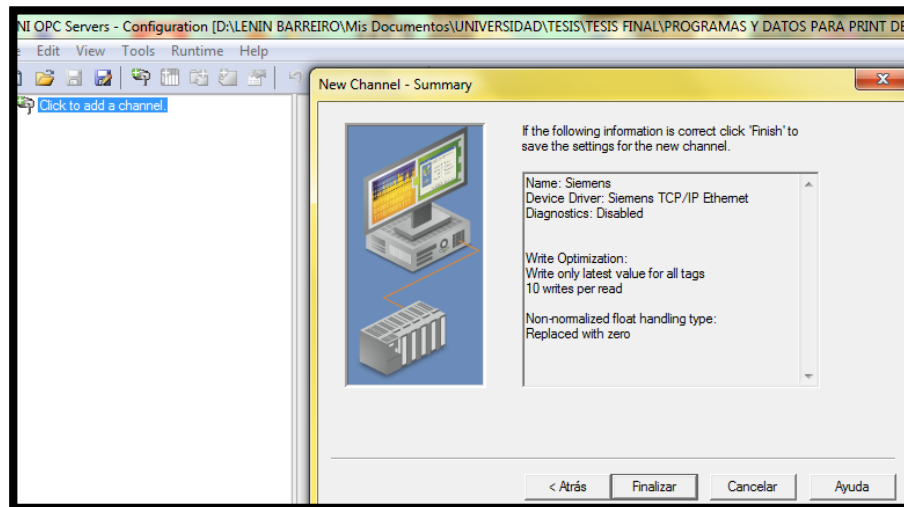


Figura 5.7 Resumen de configuración del tipo de comunicación para el Canal de comunicación Siemens.

Adicionalmente debemos crear el dispositivo (New Device) con el cual nos vamos a enlazar mediante el canal de comunicación para ello damos clic en el canal Siemens y a continuación clic en agregar un dispositivo (Add a device), le asignamos el nombre S7\_1200 Figura 5.8. En la arquitectura Cliente Servidor OPC de la Figura 5.1 el dispositivo creado corresponde a la Fuente de Datos. Clic en Siguiente.

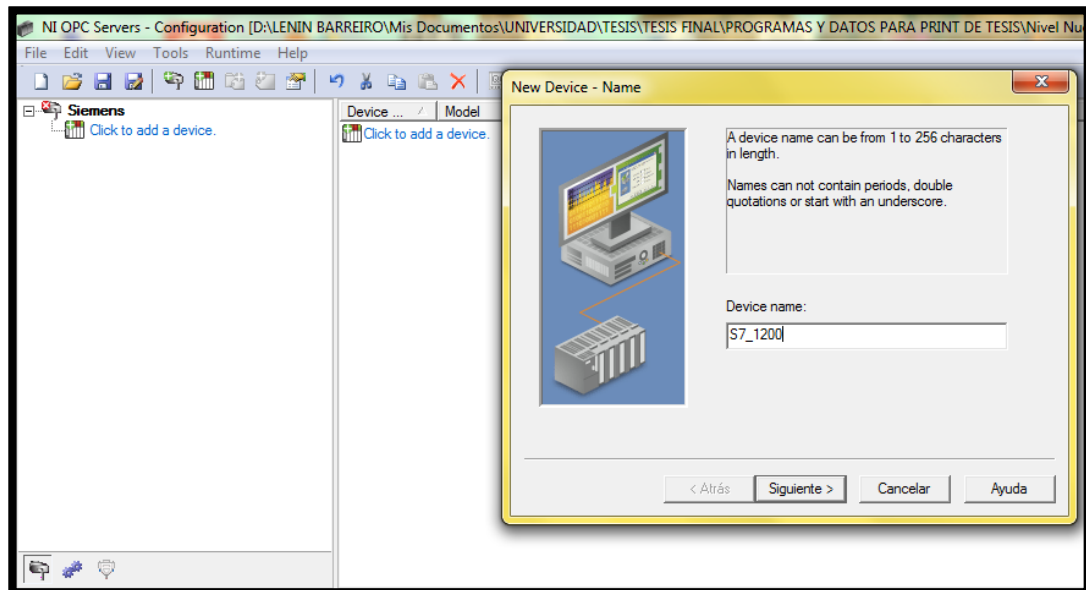


Figura 5.8 Creación del Dispositivo con el cual nos vamos a comunicar (Fuente de Datos)

Procedemos a seleccionar el modelo comercial del dispositivo con el cual vamos a comunicar, el NI OPC Servers cuenta con soporte para la mayoría de los modelos de PLCs o dispositivos que actúan como Fuente de Datos que comercialmente están disponibles, para nuestro caso seleccionamos el S7-1200, pues contamos con el Simatic S7-1200 CPU-1212C. Figura 5.9.

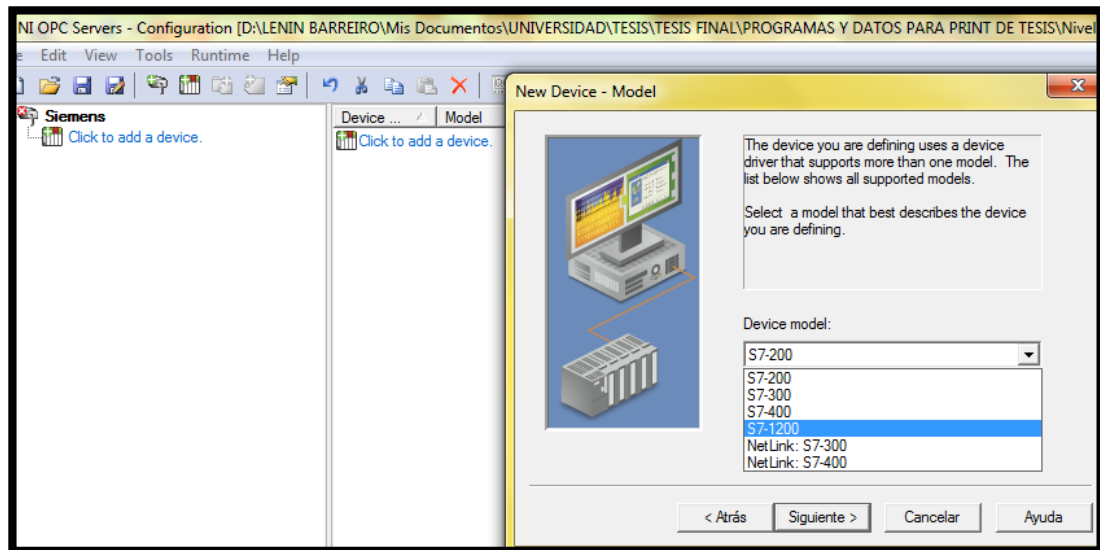


Figura 5.9 Selección del Dispositivo con el cual nos vamos a comunicar (Fuente de Datos)

#### 5.4 CONFIGURACIÓN DEL CLIENTE OPC

Previamente debemos definir en el NI OPC Server, las variables (Tags) que van a trabajar en conjunto entre Labview y TIA PORTAL, para ello creamos el archivo Nivel Nuevo que tiene la configuración del canal y el dispositivo, y procedemos a crear las variables dando clic en Add a static tag. Figura 5.10, adicionalmente en la Tabla 9, indicamos las variables que se van a definir, con su dirección, el tipo de dato en OPC Server y el tipo de dato en TIA PORTAL.

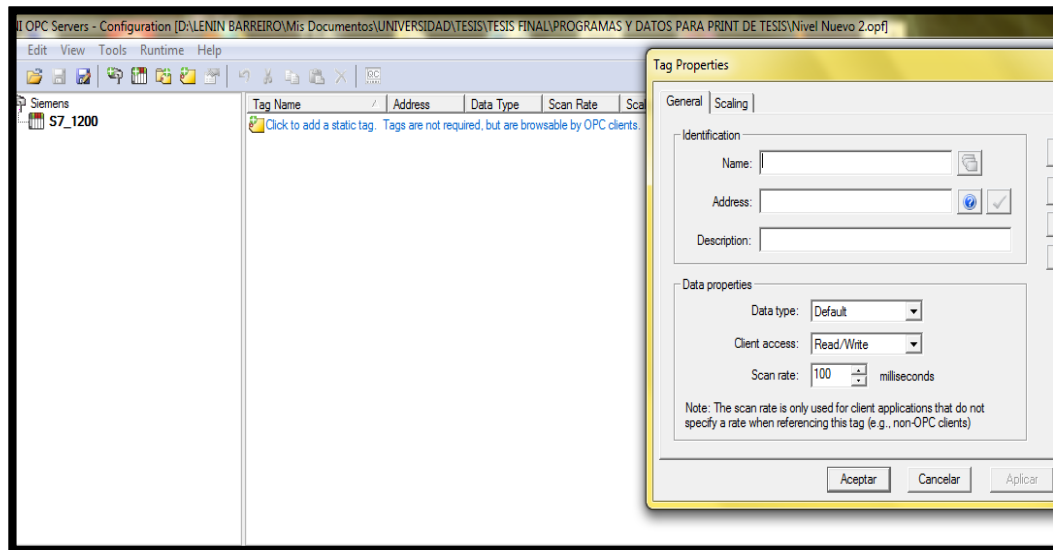


Figura 5.10 Creación de variables (Tags) que trabajan entre Labview y TIA PORTAL

VARIABLE	DIRECCION	TIPO DE DATO EN NI OPC SERVER	TIPO DE DATO EN TIA PORTAL
Apertura Modo Manual	MD6	FLOAT	REAL
Escalado	MD10	FLOAT	REAL
Escalado Salida Real Válvula	MD26	FLOAT	REAL
Fuente de Consigna	M30.3	BOOLEAN	BOOL
Marca Comenzar	M0.3	BOOLEAN	BOOL
Marca Manual	M0.5	BOOLEAN	BOOL
Marca Manual Final	M30.5	BOOLEAN	BOOL
Marca Nivel Máximo	M0.7	BOOLEAN	BOOL
Marca Run	M0.0	BOOLEAN	BOOL
Marca Stop	M0.1	BOOLEAN	BOOL
Reset Bomba Hmi	M1.0	BOOLEAN	BOOL
Run Bomba Labview	M0.2	BOOLEAN	BOOL
Run Plc	I0.0	BOOLEAN	BOOL
Salida Pid	MD14	FLOAT	REAL
Set Point Prueba	MD22	FLOAT	REAL
Stop Bomba Labview	M1.2	BOOLEAN	BOOL

Tabla 9 Variables que trabajan entre Labview y TIA PORTAL



Tenemos 2 tipos de datos en TIA PORTAL y su equivalencia en NI OPC Server, en Figura 5.11 mostramos la creación de una variable (Tag) de tipo Float

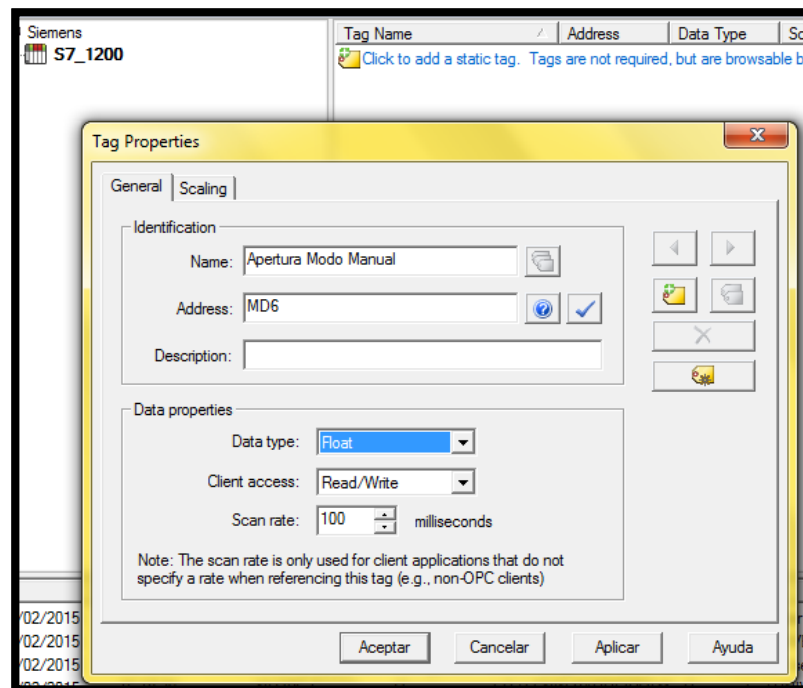


Figura 5.11 Creación de variable tipo Float

Luego procedemos a crear todas las variables tipo Float. Figura 5.12

Tag Name	Address	Data Type	Scan Rate	Scaling
Apertura Modo Manual	MD6	Float	100	None
Escalado	MD10	Float	100	None
Escalado Salida Real Valvula	MD26	Float	100	None
Salida Pid	MD14	Float	100	None
Set Point Prueba	MD22	Float	100	None

Figura 5.12 Todas las variables Float creadas.

Ahora creamos las variables tipo Boolean, en la Figura 5.13 apreciamos todas las variables Float y Boolean.

Tag Name	Address	Data Type	Scan Rate	Scaling
Apertura Modo Manual	MD6	Float	100	None
Escalado	MD10	Float	100	None
Escalado Salida Real Valvula	MD26	Float	100	None
Fuente de Consigna	M30.3	Boolean	100	None
Marca Comenzar	M0.3	Boolean	100	None
Marca Manual	M0.5	Boolean	100	None
Marca Manual Final	M30.5	Boolean	100	None
Marca Nivel Maximo	M0.7	Boolean	100	None
Marca Run	M0.0	Boolean	100	None
Marca Stop	M0.1	Boolean	100	None
Reset Bomba Hmi	M1.0	Boolean	100	None
Run Bomba Labview	M0.2	Boolean	100	None
Run Plc	I0.0	Boolean	100	None
Salida Pid	MD14	Float	100	None
Set Point Prueba	MD22	Float	100	None
Stop Bomba Labview	M1.2	Boolean	100	None

Figura 5.13 Todas las variables Float y Boolean creadas.

Una vez creadas las variables en NI OPC Servers abrimos Labview, y damos clic en Create Project-> Blank Project luego en My Computer clic derecho en New -> I/O Server. Figura 5.14

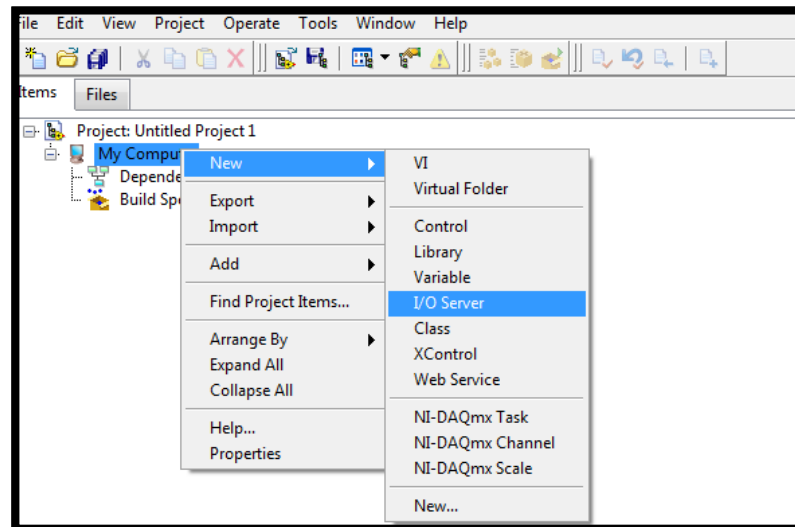


Figura 5.14 Creación del I/O Server.

Luego OPC Client y damos clic en Continue, de esta manera empezamos a configurar el Cliente OPC (OPC Client) en Figura 5.15, el cual según la arquitectura Cliente Servidor de la Figura 5.1 está directamente relacionada con la Aplicación de Labview la cual actúa como Cliente de Datos .

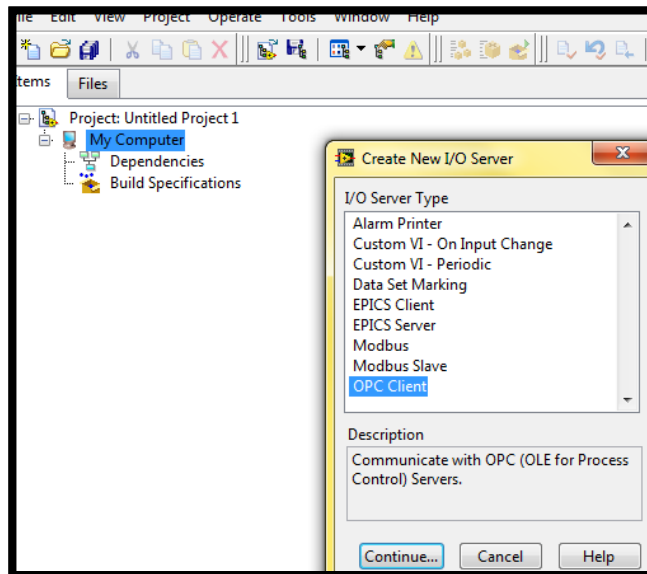


Figura 5.15 Creación del Cliente OPC

Fijamos el Update rate a 100ms, el Deadband en 0% y el Reconnect poll rate en 120 segundos damos clic en Ok. En este punto vamos a precisar el significado de los parámetros que hemos calibrado. [11]

**Update Rate:** Especifica la cantidad de tiempo en milisegundos antes de que el OPC Client recibe valores actualizados [11]

**Deadband:** Indica el porcentaje de cambio en los datos para notificar al Cliente, de dicho cambio, cuando se especifica un porcentaje, se calculará el cambio de datos requerido mediante el rango que da el porcentaje. [11]

Cuando el porcentaje no está habilitado como en nuestro caso con 0%, se notificará todo los cambios de datos.

**Reconnect poll rate:** Especifica cada que tiempo el Cliente OPC intenta reconectarse al Servidor OPC después de perder la conexión, si lo ponemos en 0, el Cliente OPC no volverá a intentar reconectarse.[11]

Se crea una librería nueva, le cambiamos el nombre al proyecto, a la librería y guardamos. Se debe guardar la librería en la misma carpeta dirección que se guarda el proyecto. Figura 5.16

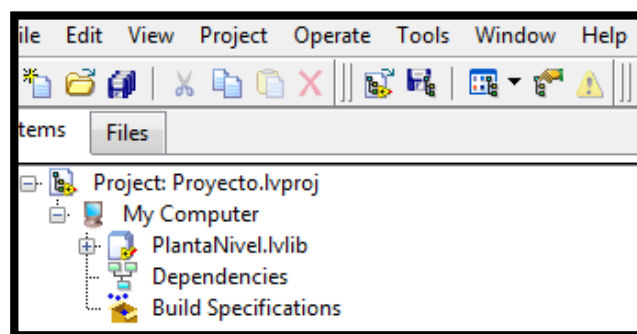


Figura 5.16 Creación de Proyecto.lvproj y librería PlantaNivel.lvlib

Damos clic derecho en la librería que guardamos PlantaNivel.lvlib y escogemos Create Bound Variables (variables ligadas) Figura 5.17.

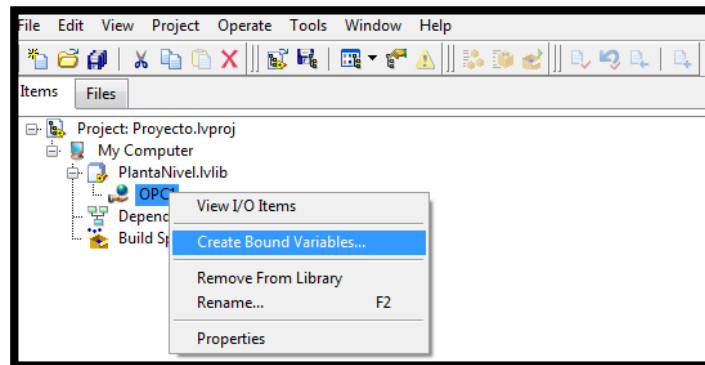


Figura 5.17 Creación de Variables Bound (ligadas)

Seguimos la ruta Project->My Computer->PlantaNivel->OPC1->Siemens->S7\_1200, aquí están las variables del NI OPC Server, Figura 5.18.

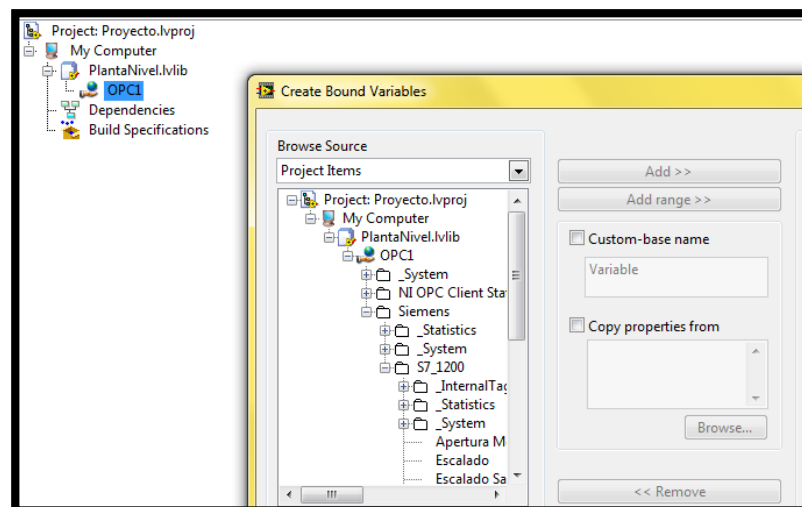


Figura 5.18 Ubicación de las variables del NI OPC Server

Luego seleccionamos todas las variables y damos clic en Add>>, Ok y después en la ventana que aparece en Done. Figura 5.19.

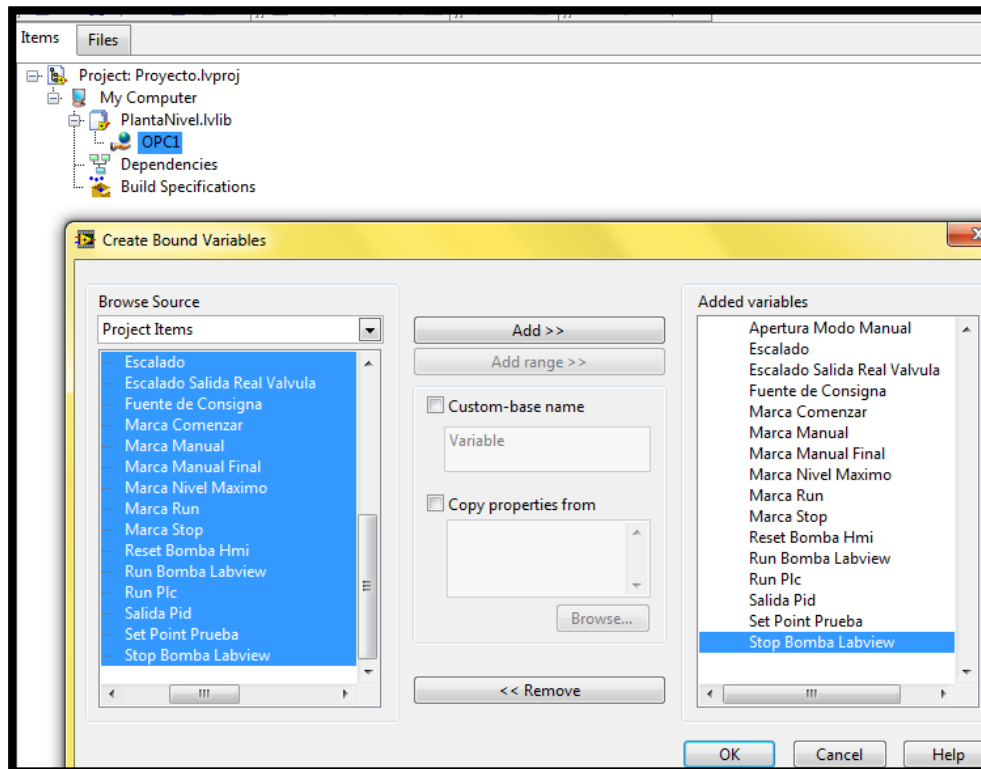


Figura 5.19 Finalización del proceso de Variables Bound (ligadas)

Al final tenemos creadas todas las variables del cliente OPC que vamos a utilizar en nuestro programa Figura 5.20.

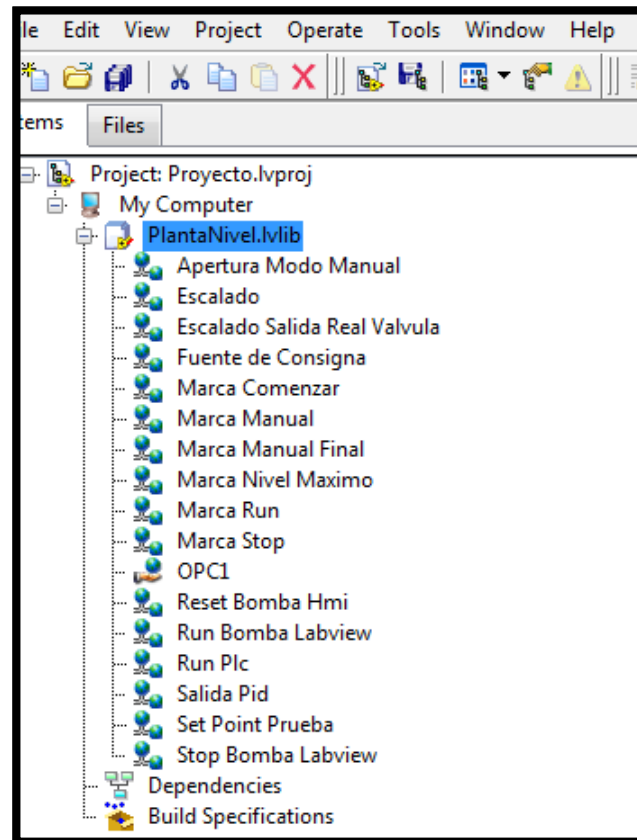


Figura 5.20 Todas las variables del Cliente de Datos creadas

## 5.5 AJUSTE GENERAL DE PARÁMETROS DE COMUNICACIÓN.

Al tener definido con que PLC vamos a trabajar, el siguiente paso es fijar la dirección de dicho dispositivo, en el caso de que nuestro dispositivo forme parte de una red, con la finalidad de comunicarse, con él, se le debe asignar una dirección única conocida como: Device ID: 200.126.14.122 para nuestro caso. Figura 5.21.



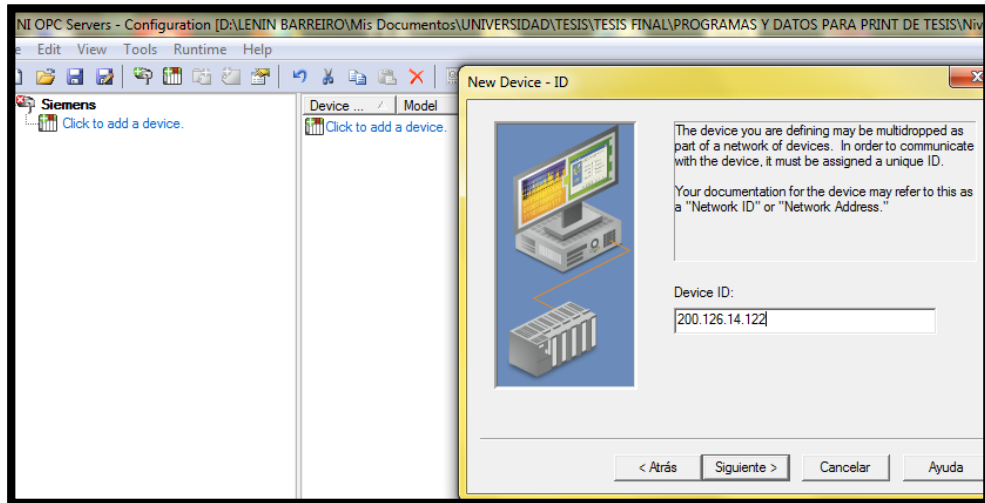


Figura 5.21 Asignación de Identificación de dispositivo (Device ID)

Definimos el Scan Mode, al seleccionar :Respect Client specified scan rate, estamos eligiendo, el valor de scan rate, que se escogió al crear el Cliente OPC, para nuestro caso 100 milisegundos. Figura 5.22.

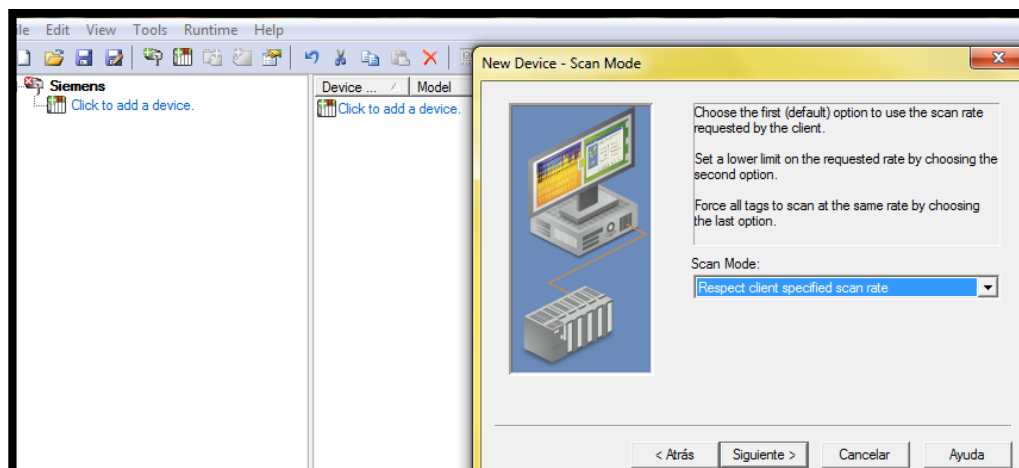


Figura 5.22 Definición del Scan Mode para el Dispositivo (PLC)

Configuramos los parámetros de tiempos máximos de comunicación para nuestro dispositivo (PLC). Figura 5.23.

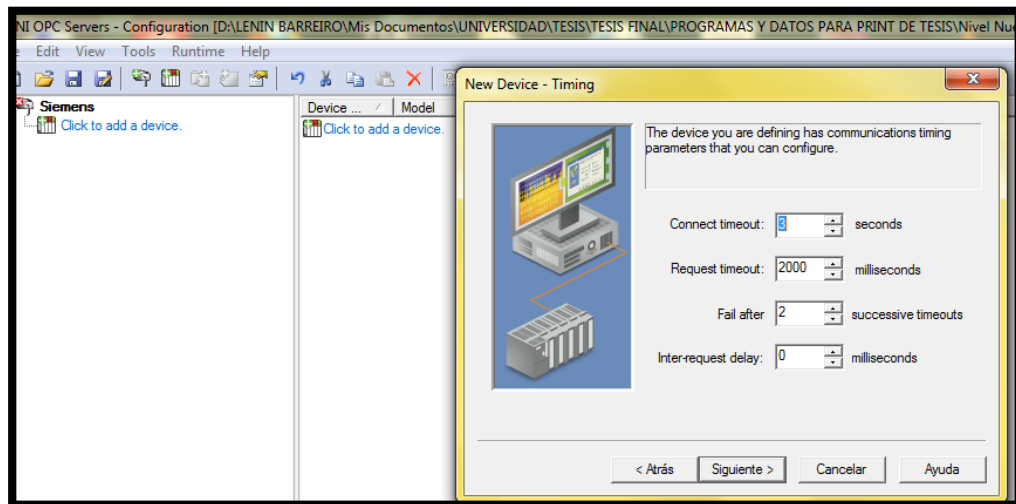


Figura 5.23 Configuración de tiempos máximos de comunicación

Podemos activar la opción Auto-Demotion para que el Servidor pueda intentar reconectar con el dispositivo en caso de una pérdida de comunicación, en nuestro caso no habilitamos esta opción. Pero en caso de que se seleccionare esta opción, durante el tiempo que el Auto-Demotion está siendo aplicado, no se enviará ninguna solicitud de lectura de datos al PLC . Figura 5.24.

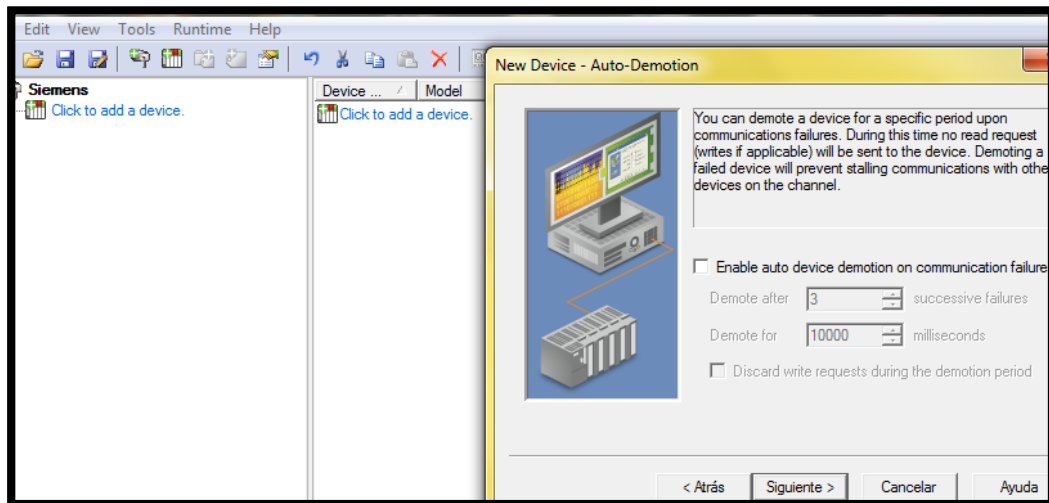


Figura 5.24 Habilitación de opción de Auto-Demotion

Selecciono la opción por defecto que es el TCP/IP Port Number : 102.

Figura 5.25.

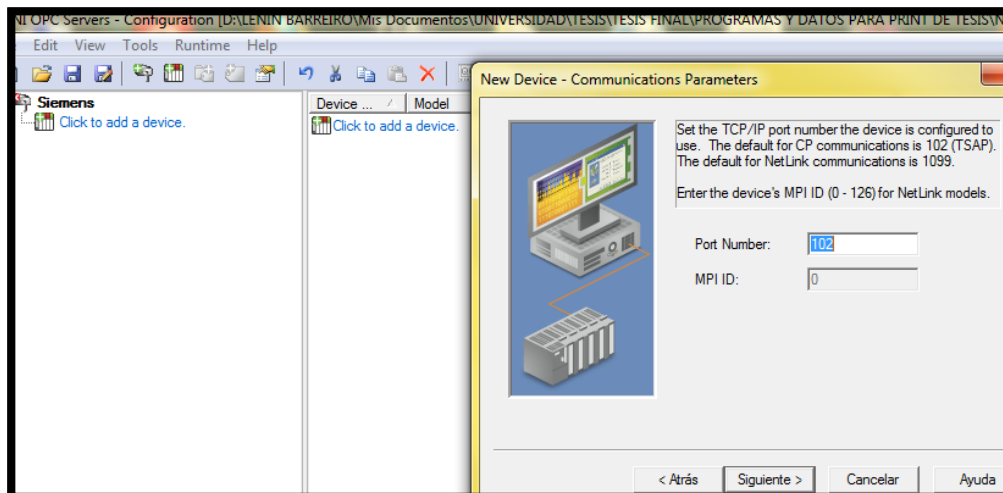


Figura 5.25 Selección del número de puerto TCP/IP

Para definir el S7 Comm. Parameters, tomo en cuenta que, en este proyecto solamente tengo un PLC con un módulo de entradas-salidas analógicas, por lo tanto, el CPU reside en el rack número 0 espacio(Slot) 1. Figura 5.26.

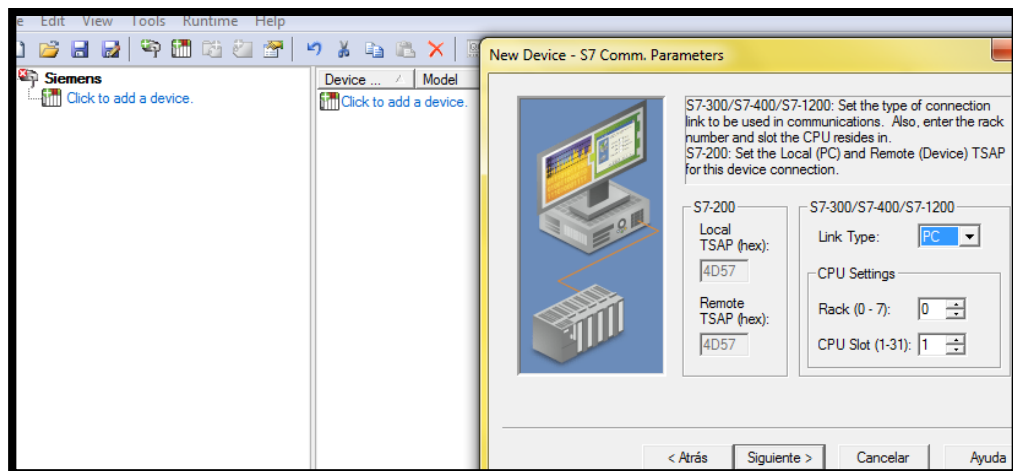


Figura 5.26 Definición de la ubicación del CPU, rack y slot

En al cuadro de diálogo Addressing Options seleccionamos Big Endian, porque este es el orden de almacenamiento de bits en memoria por defecto para los controladores Siemens S7. Figura 5.27.

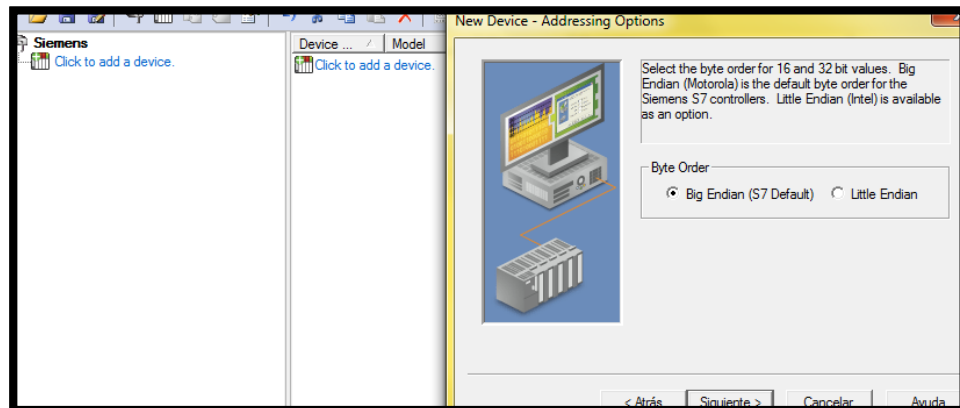


Figura 5.27 Definición del orden para el almacenamiento de bits

Finalmente un resumen de los parámetros del S7-1200 Figura 5.28.

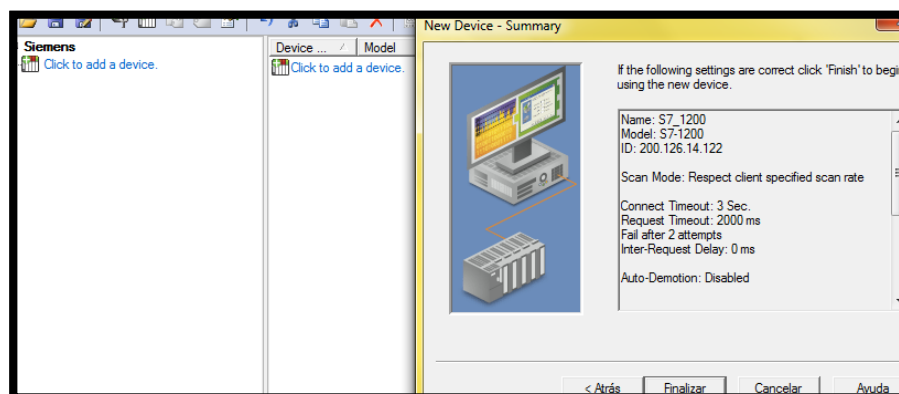


Figura 5.28 Resumen de los parámetros de comunicación del S7-1200

## **CAPÍTULO 6**

### **PRUEBAS, VALIDACIÓN Y PUESTA EN FUNCIONAMIENTO.**

#### **6.1 DISEÑO DE PRUEBAS.**

Para el diseño de pruebas, tendremos 2 enfoques. El primero relacionado con la desempeño del controlador utilizado que es el PID\_COMPACT, el cual lo haremos en términos de cantidades en el dominio del tiempo, para lo cual compararemos los parámetros indicados en la Figura 6.1, para diferentes valores de Setpoint, analizando si existe una variación significativa en los parámetros en el dominio del tiempo que nos permitan, concluir si los ajustes realizados con el Autotuning del PID\_COMPACT, fueron los adecuados tomando en cuenta que la apertura de la válvula HA de salida de agua del depósito fue del 50%.

El segundo aspecto que nos interesa evaluar es si la comunicación establecida mediante el OPC, fue óptima, esto lo podemos verificar mediante la inspección de las variables en el OPC Quick Client, analizando la calidad de la comunicación para las distintas variables, y si el cambio de los valores de Setpoint se ve reflejado en dicha herramienta, además con la correcta visualización de los valores en la Interfaz de Labview.

Es preciso incluir en el presente reporte ciertas definiciones para la respuesta transitoria en el dominio del tiempo, a fin de usar estas definiciones para evaluar las gráficas obtenidas.

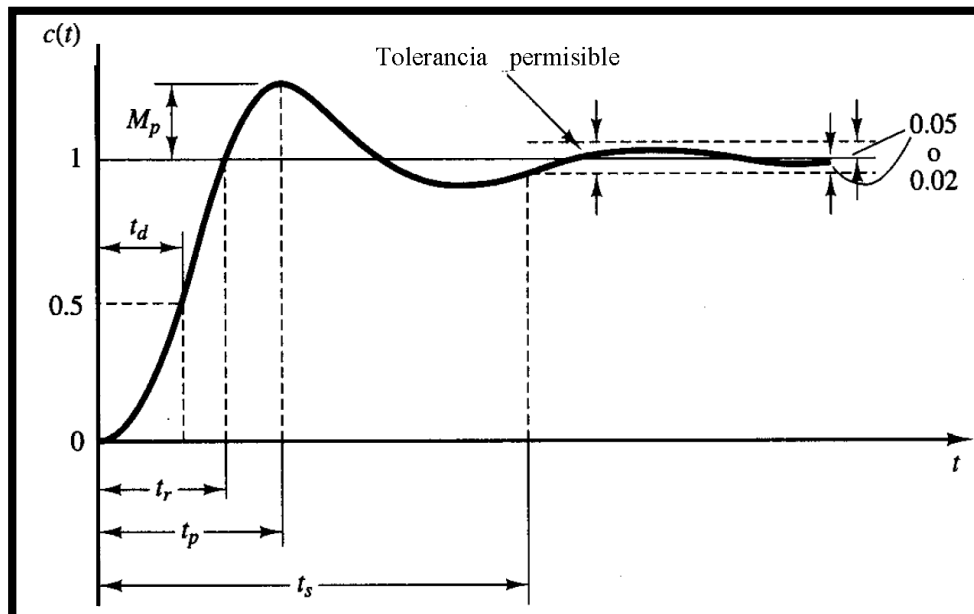


Figura 6.1 Curva de respuesta al escalón unitario [1]

**Tiempo de retardo  $t_d$**  : Es el tiempo requerido para que la respuesta alcance la primera vez la mitad del valor final. [1]

**Tiempo de levantamiento  $t_r$** : Es el tiempo requerido para que la respuesta pase del 10 al 90%, del 5 al 95% o del 0 al 100% de su valor final. Para sistemas subamortiguados de segundo orden, por lo común se usa el tiempo de levantamiento de 0 a 100%. Para sistemas sobreamortiguados, suele usarse el tiempo de levantamiento de 10 a 90%. [1]

**Tiempo pico  $t_p$**  : el tiempo pico es el tiempo requerido para que la respuesta alcance el primer pico del sobrepaso. [1]

**Sobrepaso máximo (porcentaje)  $M_p$** : el sobrepaso máximo es el valor pico máximo de la curva de respuesta, medido a partir de la unidad. Si el valor final en estado estable de la respuesta es diferente de la unidad, es común usar el porcentaje de sobrepaso máximo. [1] Se define mediante la Ecuación 6.1, tomando en cuenta que el  $c(t)$  son los distintos valores que tiene la variable de nivel en nuestro informe.



$$Mp = \left( \frac{c(tp) - c(\infty)}{c(\infty)} \right) * 100\%$$

Ecuación 6.1 Porcentaje de Sobrepasso máximo [1]

**Tiempo de asentamiento  $t_s$ :** es el tiempo que toma  $c(t)$  en alcanzar valores dentro de un rango de tolerancia permisible que puede ser del 2% o el 5%. [1]

## 6.2 PRUEBAS Y COMPARACIÓN DE RESULTADOS OBTENIDOS

**Setpoint de 20 cm:** En la Figura 6.2 observamos que esta seleccionado un valor de setpoint de 20, que corresponde a la variable Setpoint Prueba, el porcentaje de apertura de la válvula está en 5,13774% y el Nivel en 20,0727 cm. En la Figura 6.3 tenemos una vista del OPC Quick Client para este caso observando que Setpoint Prueba está en 20. En la Figura 6.4 tenemos una gráfica amplificada de la respuesta en el dominio del tiempo.

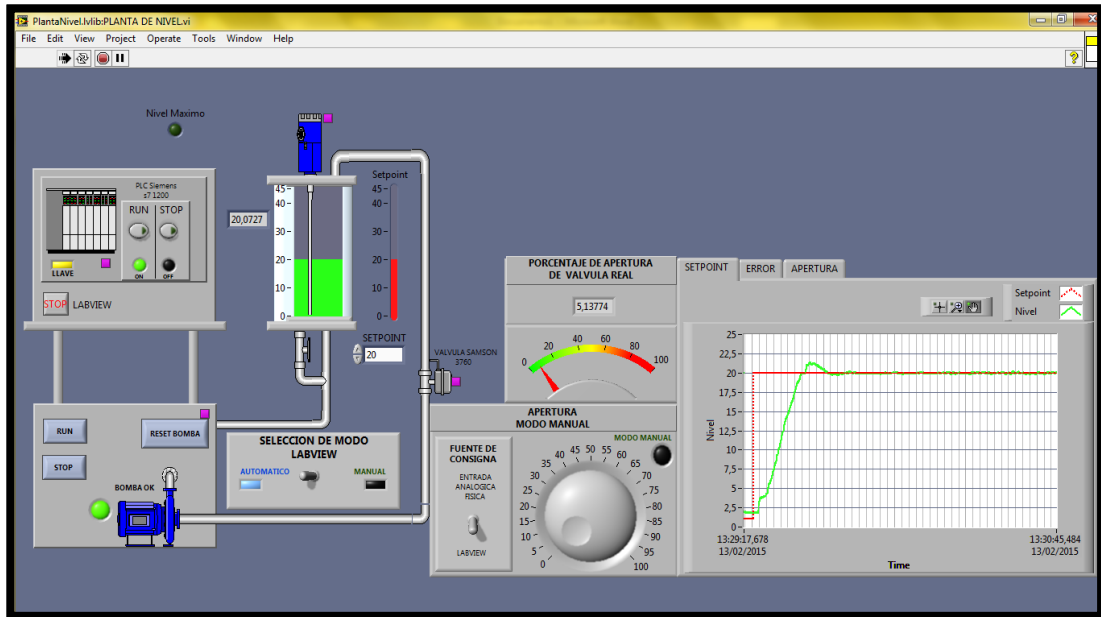


Figura 6.2 Interfaz en Labview para un Setpoint : 20 cm

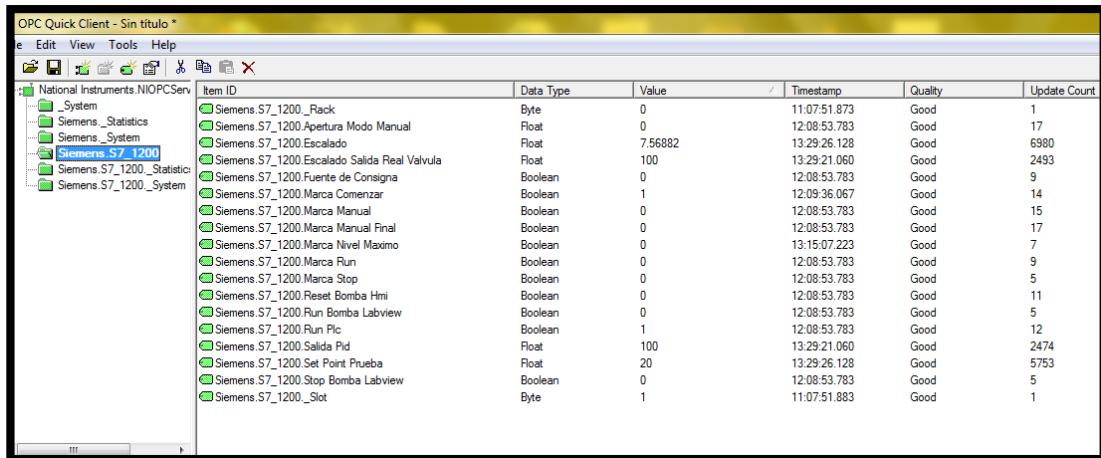


Figura 6.3 OPC Quick Client para Setpoint : 20 cm

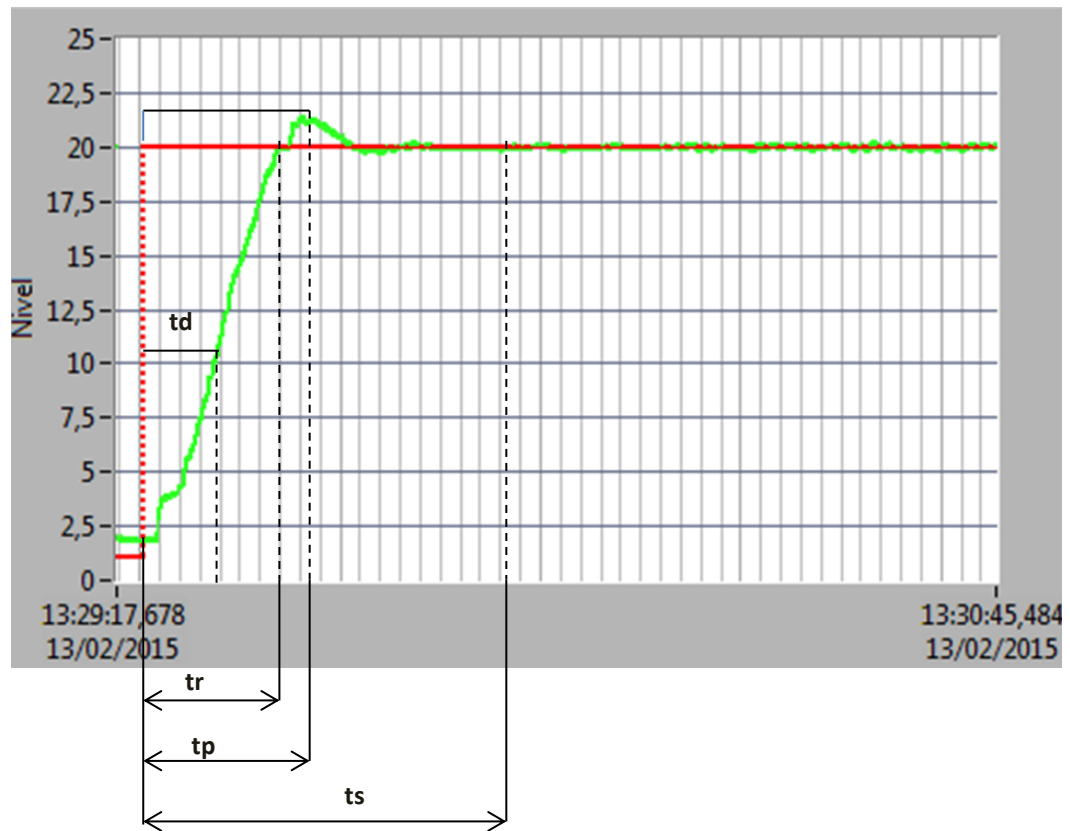


Figura 6.4 Respuesta del sistema al Setpoint de 20 cm.

**Setpoint de 25 cm:** En la Figura 6.5 observamos que está seleccionado un valor de setpoint de 25, que corresponde a la variable Setpoint Prueba, el porcentaje de apertura de la válvula está en 16,0005% y el Nivel en 25,0327 cm. En la Figura 6.6 tenemos una vista del OPC Quick Client para este caso observando que Setpoint Prueba está en 25. En la

Figura 6.7 tenemos una gráfica amplificada de la respuesta en el dominio del tiempo.

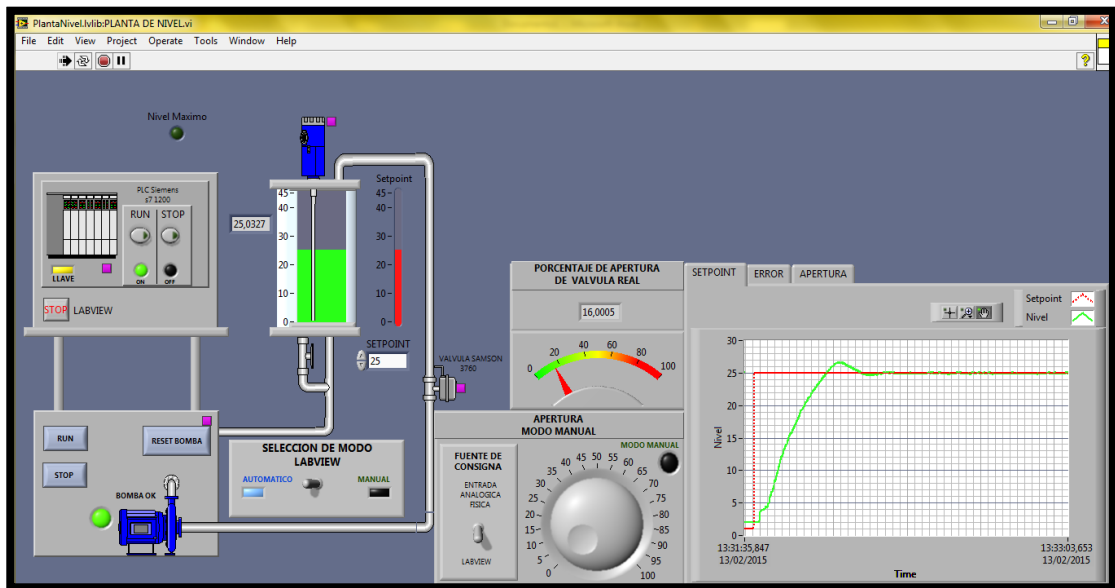


Figura 6.5 Interfaz en Labview para un Setpoint : 25 cm

OPC Quick Client - Sin título \*

File Edit View Tools Help

National Instruments.NIOPCServ

Item ID	Data Type	Value	Timestamp	Quality	Update Count
Siemens.S7_1200.Flack	Byte	0	11:07:51.873	Good	1
Siemens.S7_1200.Apertura Modo Manual	Float	0	12:08:53.783	Good	17
Siemens.S7_1200.Escalado	Float	19.1634	13:31:51.130	Good	7122
Siemens.S7_1200.Escalado Salida Real Valvula	Float	94.8162	13:31:51.130	Good	2593
Siemens.S7_1200.Fuente de Consigna	Boolean	0	12:08:53.783	Good	9
Siemens.S7_1200.Marca Comenzar	Boolean	1	12:09:36.067	Good	14
Siemens.S7_1200.Marca Manual	Boolean	0	12:08:53.783	Good	15
Siemens.S7_1200.Marca Manual Final	Boolean	0	12:08:53.783	Good	17
Siemens.S7_1200.Marca Nivel Maximo	Boolean	0	13:15:07.223	Good	7
Siemens.S7_1200.Marca Run	Boolean	0	12:08:53.783	Good	9
Siemens.S7_1200.Marca Stop	Boolean	0	12:08:53.783	Good	5
Siemens.S7_1200.Reset Bomba Hmi	Boolean	0	12:08:53.783	Good	11
Siemens.S7_1200.Run Bomba Labview	Boolean	0	12:08:53.783	Good	5
Siemens.S7_1200.Run Plc	Boolean	1	12:08:53.783	Good	12
Siemens.S7_1200.Salida Pid	Float	94.8162	13:31:51.130	Good	2574
Siemens.S7_1200.Set Point Prueba	Float	25	13:31:51.130	Good	5885
Siemens.S7_1200.Stop Bomba Labview	Boolean	0	12:08:53.783	Good	5
Siemens.S7_1200.Slot	Byte	1	11:07:51.883	Good	1

Figura 6.6 OPC Quick Client para Setpoint : 25 cm

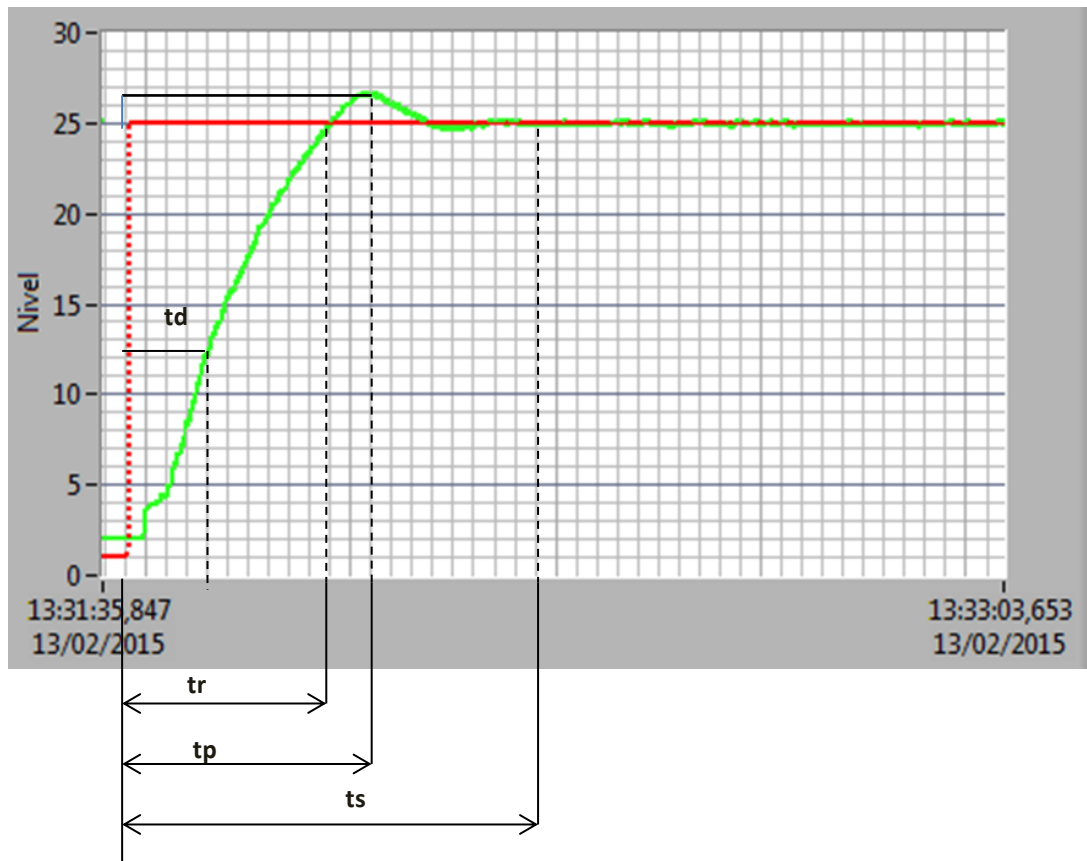


Figura 6.7 Respuesta del sistema al Setpoint de 25 cm.

**Setpoint de 30 cm:** En la Figura 6.8 observamos que esta seleccionado un valor de setpoint de 30, que corresponde a la variable Setpoint Prueba, el porcentaje de apertura de la válvula está en 20,6411% y el Nivel en 29,9881 cm. En la Figura 6.9 tenemos una vista del OPC Quick Client para este caso observando que Setpoint Prueba está en 30. En la Figura 6.10 tenemos una gráfica ampliada de la respuesta en el dominio del tiempo.

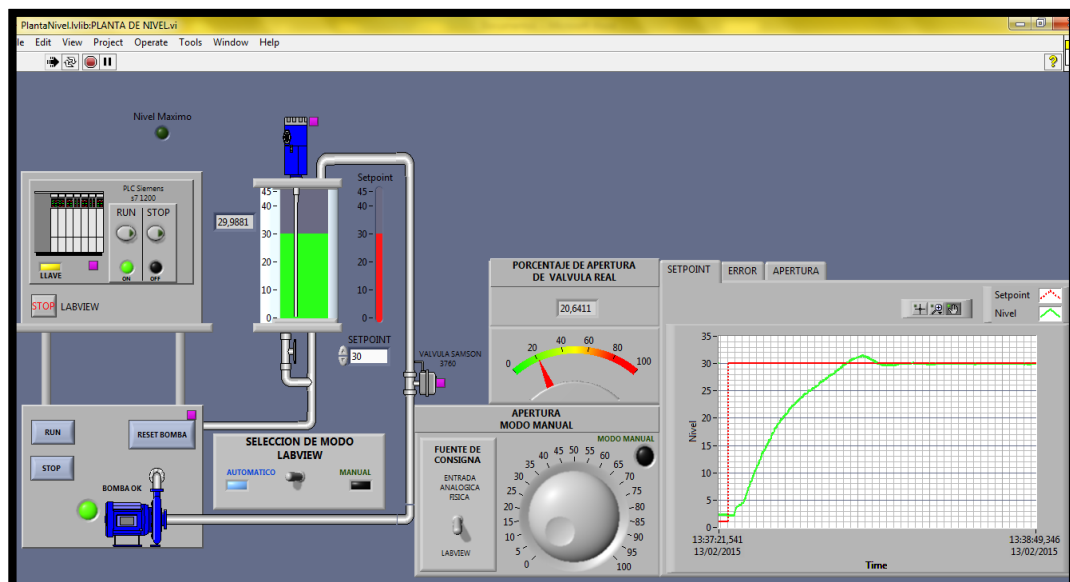


Figura 6.8 Interfaz en Labview para un Setpoint : 30 cm

OPC Quick Client - Sin título \*

Item ID	Data Type	Value	Timestamp	Quality	Update Count
Siemens S7_1200_Rack	Byte	0	11:07:51.873	Good	1
Siemens S7_1200_Apertura Modo Manual	Float	0	12:08:53.783	Good	17
Siemens S7_1200_Escalado	Float	28.2272	13:37:53.255	Good	7476
Siemens S7_1200_Escalado Salida Real Valvula	Float	99.071	13:37:53.255	Good	2797
Siemens S7_1200_Fuente de Consigna	Boolean	0	12:08:53.783	Good	9
Siemens S7_1200_Marca Comenzar	Boolean	1	12:09:36.067	Good	14
Siemens S7_1200_Marca Manual	Boolean	0	12:08:53.783	Good	15
Siemens S7_1200_Marca Manual Final	Boolean	0	12:08:53.783	Good	17
Siemens S7_1200_Marca Nivel Maximo	Boolean	0	13:15:07.223	Good	7
Siemens S7_1200_Marca Run	Boolean	0	12:08:53.783	Good	9
Siemens S7_1200_Marca Stop	Boolean	0	12:08:53.783	Good	5
Siemens S7_1200_Reset Bomba Hmi	Boolean	0	12:08:53.783	Good	11
Siemens S7_1200_Run Bomba Labview	Boolean	0	12:08:53.783	Good	5
Siemens S7_1200_Run Pic	Boolean	1	12:08:53.783	Good	12
Siemens S7_1200_Salida Pid	Float	99.071	13:37:53.255	Good	2774
Siemens S7_1200_Set Point Prueba	Float	30	13:37:53.255	Good	6218
Siemens S7_1200_Stop Bomba Labview	Boolean	0	12:08:53.783	Good	5
Siemens S7_1200_Slot	Byte	1	11:07:51.883	Good	1

Figura 6.9 OPC Quick Client para Setpoint : 30 cm

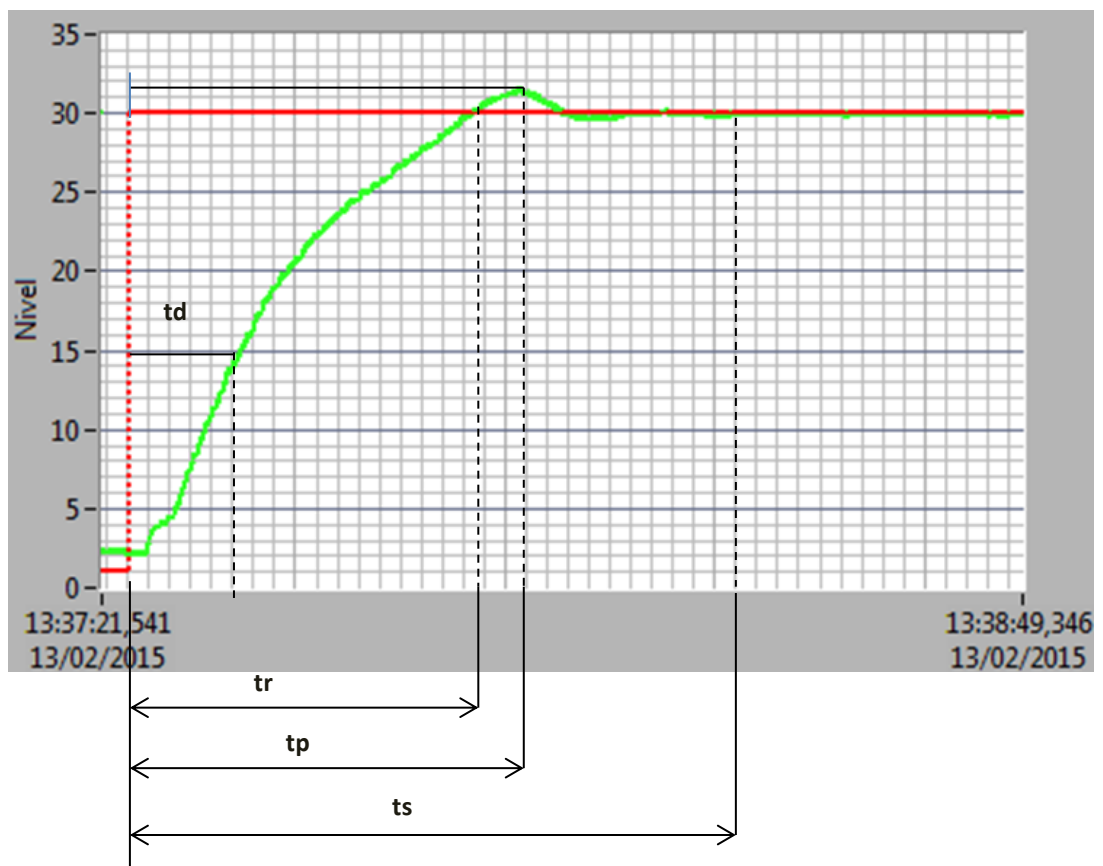


Figura 6.10 Respuesta del sistema al Setpoint de 30 cm

**Setpoint de 35 cm:** En la Figura 6.11 observamos que esta seleccionado un valor de setpoint de 35 que corresponde a la variable Setpoint Prueba, el porcentaje de apertura de la válvula está en 23,2113% y el Nivel en 35,027 cm. En la Figura 6.12 tenemos una vista del OPC Quick Client para este caso observando que Setpoint Prueba está en 35. En la Figura 6.13 tenemos una gráfica ampliada de la respuesta en el dominio del tiempo.

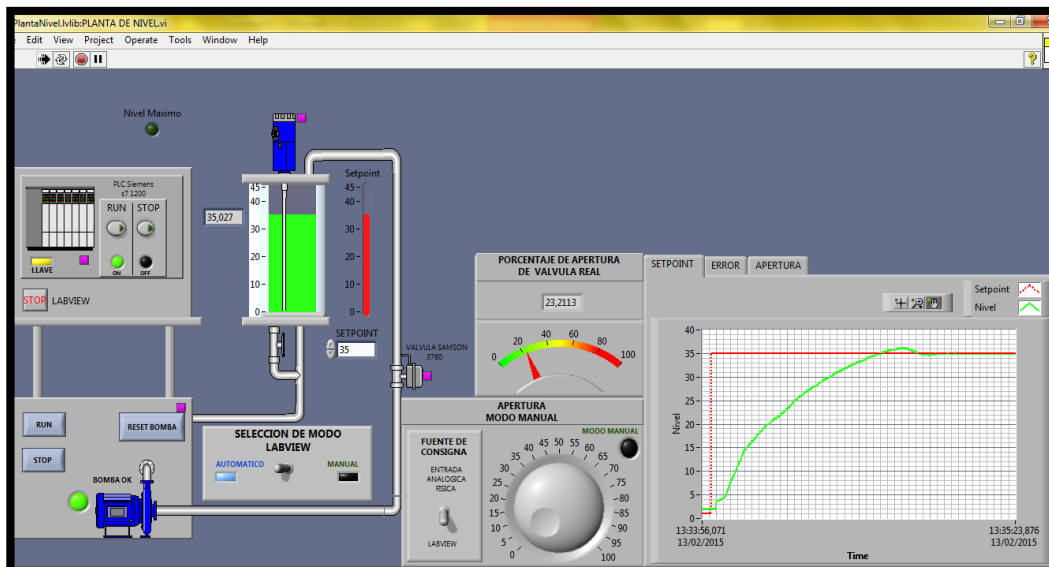


Figura 6.11 Interfaz en Labview para un Setpoint : 35 cm



Item ID	Data Type	Value	Timestamp	Quality	Update Count
Siemens S7_1200_Flask	Byte	0	11:07:51.873	Good	1
Siemens S7_1200_Apertura Modo Manual	Float	0	12:08:53.783	Good	17
Siemens S7_1200_Escalado	Float	34.5587	13:36:10.763	Good	7376
Siemens S7_1200_Escalado Salida Real Valvula	Float	99.6735	13:36:10.763	Good	2752
Siemens S7_1200_Fuente de Consigna	Boolean	0	12:08:53.783	Good	9
Siemens S7_1200_Marca Comenzar	Boolean	1	12:09:36.067	Good	14
Siemens S7_1200_Marca Manual	Boolean	0	12:08:53.783	Good	15
Siemens S7_1200_Marca Manual Final	Boolean	0	12:08:53.783	Good	17
Siemens S7_1200_Marca Nivel Maximo	Boolean	0	13:15:07.223	Good	7
Siemens S7_1200_Marca Run	Boolean	0	12:08:53.783	Good	9
Siemens S7_1200_Marca Stop	Boolean	0	12:08:53.783	Good	5
Siemens S7_1200_Reset Bomba Hmi	Boolean	0	12:08:53.783	Good	11
Siemens S7_1200_Run Bomba Labview	Boolean	0	12:08:53.783	Good	5
Siemens S7_1200_Run Plc	Boolean	1	12:08:53.783	Good	12
Siemens S7_1200_Salida Pid	Float	99.6735	13:36:10.763	Good	2733
Siemens S7_1200_Set Point Prueba	Float	35	13:36:10.763	Good	6120
Siemens S7_1200_Stop Bomba Labview	Boolean	0	12:08:53.783	Good	5
Siemens S7_1200_Slot	Byte	1	11:07:51.883	Good	1

Figura 6.12 OPC Quick Client para Setpoint : 35 cm

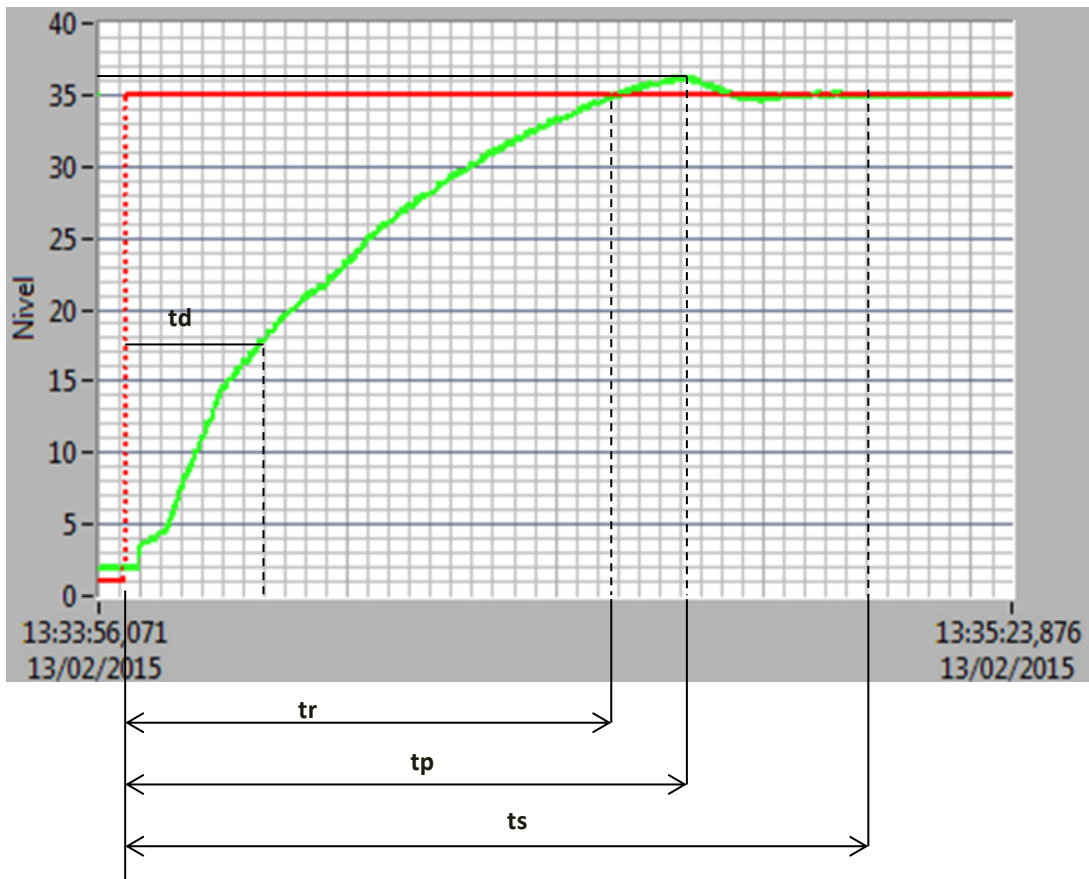


Figura 6.13 Respuesta del sistema al Setpoint de 35 cm

**Setpoint de 40 cm:** En la Figura 6.14 observamos que esta seleccionado un valor de setpoint de 40, que corresponde a la variable Setpoint Prueba, el porcentaje de apertura de la válvula está en 24,3909% y el Nivel en 39,937 cm. En la Figura 6.15 tenemos una vista del OPC Quick Client para este caso observando que Setpoint Prueba está en 40. En la Figura 6.16 tenemos una gráfica ampliada de la respuesta en el dominio del tiempo.

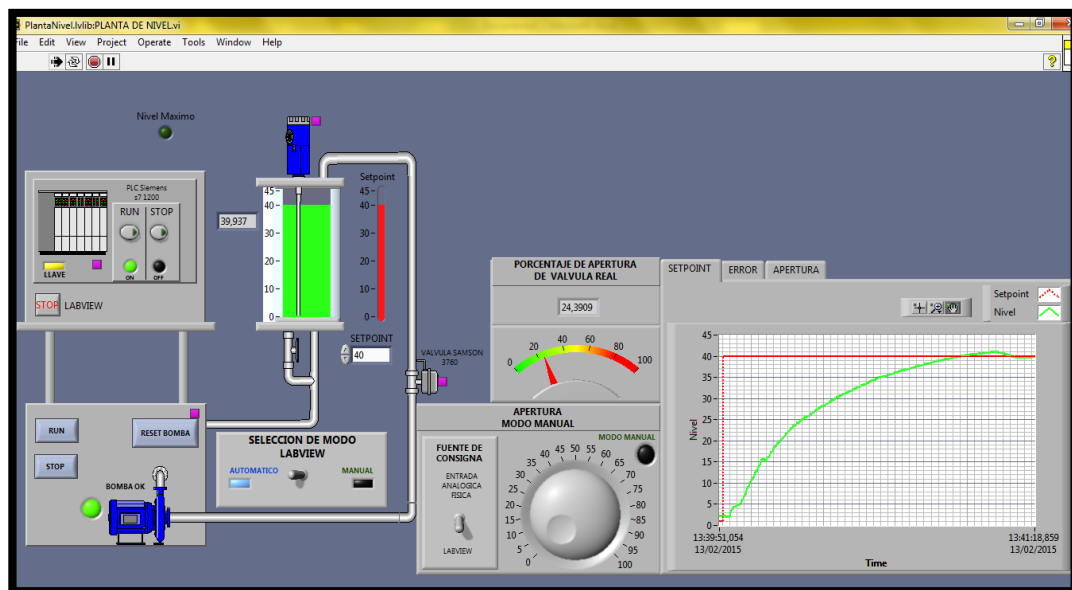


Figura 6.14 Interfaz en Labview para un Setpoint : 40 cm

OPC Quick Client - Sin título \*

File Edit View Tools Help

National Instruments.NIOPCServ

Item ID	Data Type	Value	Timestamp	Quality	Update Count
Siemens.S7_1200_Flack	Byte	0	11:07:51.873	Good	1
Siemens.S7_1200_Apertura Modo Manual	Float	0	12:08:53.783	Good	17
Siemens.S7_1200_Escalado	Float	4.70465	13:39:55.983	Good	7597
Siemens.S7_1200_Escalado Salida Real Valvula	Float	100	13:39:52.941	Good	2872
Siemens.S7_1200_Fuente de Consigna	Boolean	0	12:08:53.783	Good	9
Siemens.S7_1200_Marca Comenzar	Boolean	1	12:09:36.067	Good	14
Siemens.S7_1200_Marca Manual	Boolean	0	12:08:53.783	Good	15
Siemens.S7_1200_Marca Manual Final	Boolean	0	12:08:53.783	Good	17
Siemens.S7_1200_Marca Nivel Maximo	Boolean	0	13:15:07.223	Good	7
Siemens.S7_1200_Marca Run	Boolean	0	12:08:53.783	Good	9
Siemens.S7_1200_Marca Stop	Boolean	0	12:08:53.783	Good	5
Siemens.S7_1200_Reset Bomba Hmi	Boolean	0	12:08:53.783	Good	11
Siemens.S7_1200_Run Bomba Labview	Boolean	0	12:08:53.783	Good	5
Siemens.S7_1200_Run Plc	Boolean	1	12:08:53.783	Good	12
Siemens.S7_1200_Salida Pid	Float	100	13:39:52.941	Good	2849
Siemens.S7_1200_Set Point Prueba	Float	40	13:39:55.983	Good	6335
Siemens.S7_1200_Stop Bomba Labview	Boolean	0	12:08:53.783	Good	5
Siemens.S7_1200_Slot	Byte	1	11:07:51.883	Good	1

Figura 6.15 OPC Quick Client para Setpoint : 40 cm

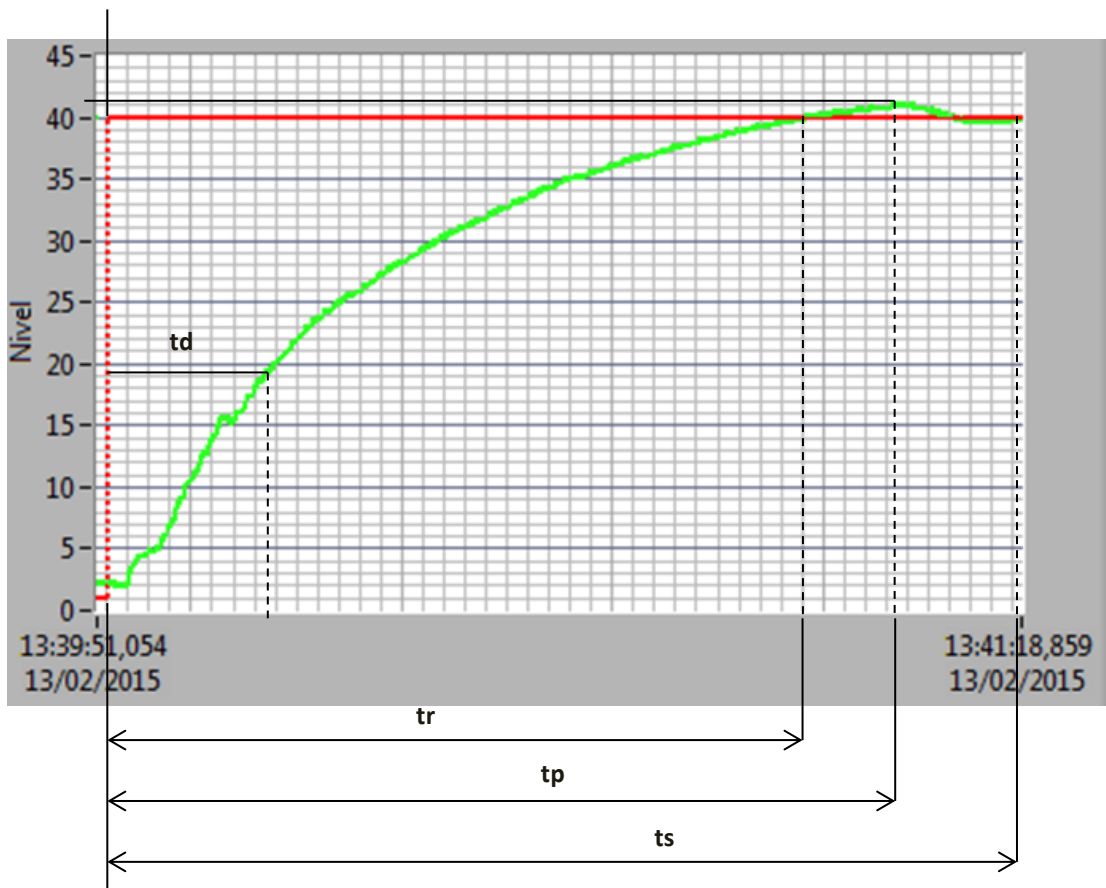


Figura 6.16 Respuesta del sistema al Setpoint de 40 cm

**Setpoint de 44 cm:** En la Figura 6.17 observamos que esta seleccionado un valor de setpoint de 44, que corresponde a la variable Setpoint Prueba, el porcentaje de apertura de la válvula está en 100% y el Nivel en 25,4509 cm. En la Figura 6.18 tenemos una vista del OPC Quick Client para este caso observando que Setpoint Prueba está en 44. En la Figura 6.19 tenemos una gráfica amplificada de la respuesta en el dominio del tiempo, en este valor de Setpoint tenemos que se llega al valor fijado como límite máximo de 43.23 cm por lo tanto, en la Interfaz de Labview vemos que la luz piloto BOMBA OK, está apagada, en la gráfica de Nivel vs Tiempo, apreciamos que la línea verde que nos indica el Nivel actual, cae paulatinamente, hasta llegar a 0, el porcentaje de apertura de válvula está en 100% pero como la Bomba está apagada, no hay flujo de agua.

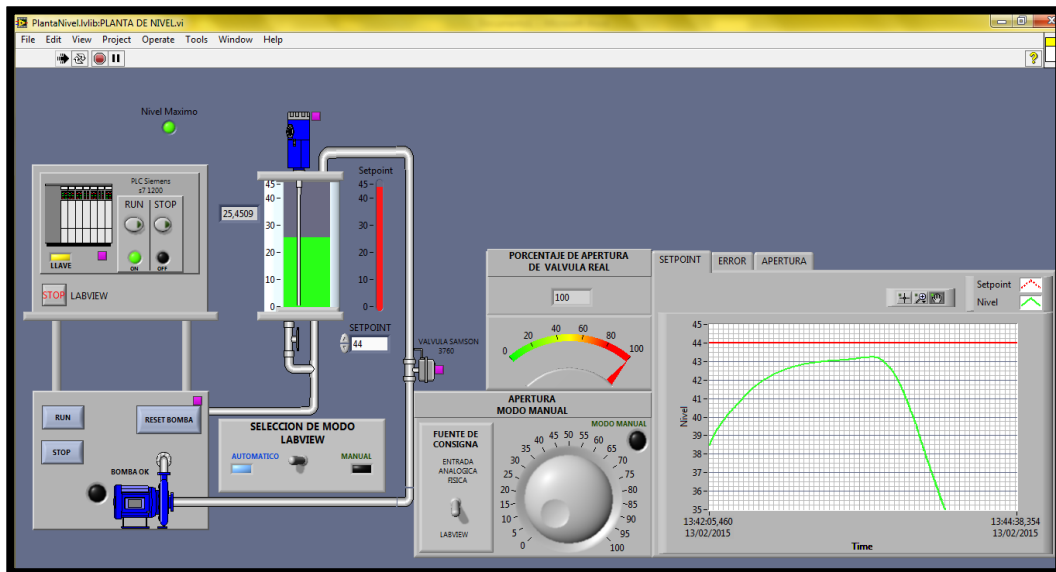


Figura 6.17 Interfaz en Labview para un Setpoint : 44 cm

OPC Quick Client - Sin título \*

Item ID	Data Type	Value	Timestamp	Quality	Update Count
Siemens S7_1200_Rack	Byte	0	11:07:51.873	Good	1
Siemens S7_1200_Apertura Modo Manual	Float	0	12:08:53.783	Good	17
Siemens S7_1200_Escalado	Float	41.9676	13:42.32.139	Good	7751
Siemens S7_1200_Escalado Salida Real Valvula	Float	100	13:41:56.647	Good	2923
Siemens S7_1200_Fuente de Consigna	Boolean	0	12:08:53.783	Good	9
Siemens S7_1200_Marca Comenzar	Boolean	1	12:09:36.067	Good	14
Siemens S7_1200_Marca Manual	Boolean	0	12:08:53.783	Good	15
Siemens S7_1200_Marca Manual Final	Boolean	0	12:08:53.783	Good	17
Siemens S7_1200_Marca Nivel Maximo	Boolean	0	13:15:07.223	Good	7
Siemens S7_1200_Marca Run	Boolean	0	12:08:53.783	Good	9
Siemens S7_1200_Marca Stop	Boolean	0	12:08:53.783	Good	5
Siemens S7_1200_Reset Bomba Hmi	Boolean	0	12:08:53.783	Good	11
Siemens S7_1200_Run Bomba Labview	Boolean	0	12:08:53.783	Good	5
Siemens S7_1200_Run Plc	Boolean	1	12:08:53.783	Good	12
Siemens S7_1200_Salida Pid	Float	100	13:41:56.647	Good	2900
Siemens S7_1200_Set Point Prueba	Float	44	13:42.32.139	Good	6484
Siemens S7_1200_Stop Bomba Labview	Boolean	0	12:08:53.783	Good	5
Siemens S7_1200_Slot	Byte	1	11:07:51.883	Good	1

Figura 6.18 OPC Quick Client para Setpoint : 44 cm

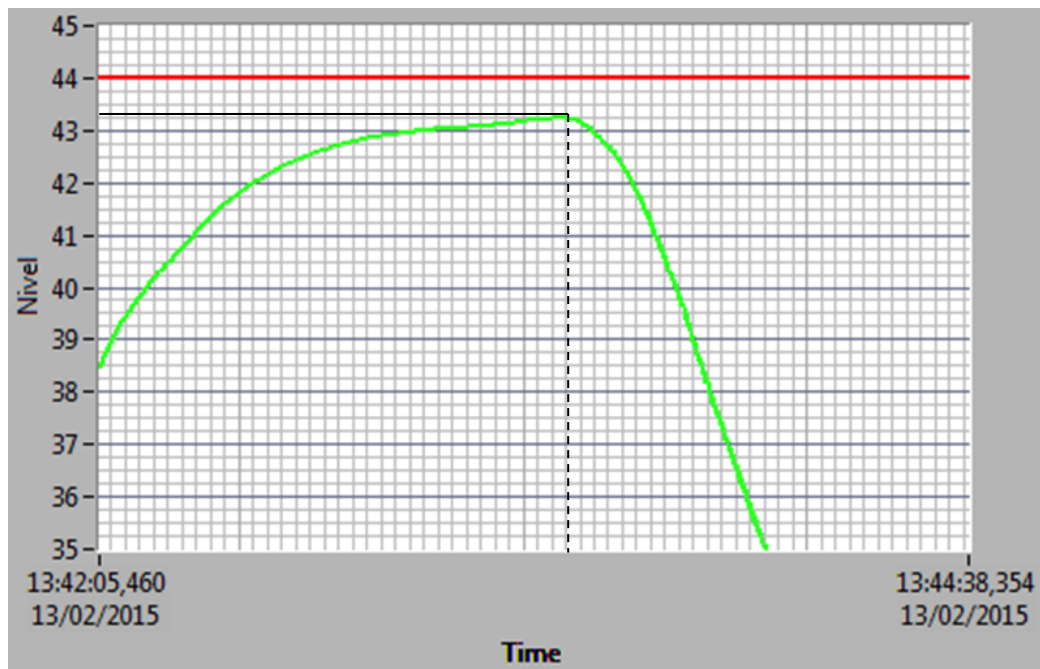


Figura 6.19 Respuesta del sistema al Setpoint de 44 cm

**Comparación de Resultados Obtenidos:** En base a los datos de los gráficos para los distintos valores de Setpoint y los cálculos utilizando la ecuación 6.1 tenemos la siguiente Tabla 10 en base a la cual seleccionaremos el punto de operación óptimo para nuestra Planta de Nivel.9

Parámetro	Valores de Setpoint (cm)				
	20	25	30	35	40
Tiempo de retardo: $t_d$ (seg.)	8	8	10,2	14	15,2
Tiempo de levantamiento: $t_r$ (seg.)	14	20	32,8	47,2	47,2
Tiempo pico: $t_p$ (seg.)	17	20	36	54,8	68
Tiempo de asentamiento $t_s$ (seg.)	36,25	36	38	72	76,7
Valor de Nivel para $t_p$ : $C(t_p)$ cm.	21,66	26,5	31,5	36,3	41,3
Setpoint: $C(\infty)$ cm.	20	25	30	35	40
$C(t_p)$ cm - $C(\infty)$ cm.	1,66	1,5	1,5	1,3	1,3
$M_p$ (%)	8,30%	6,00%	5,00%	3,71%	3,25%

Tabla 10 Resultados obtenidos en las pruebas

En base a los resultados obtenidos y analizando los datos de la Tabla 10, podemos afirmar que en general para todos los puntos de operación el Control de Planta de Nivel utilizado en las pruebas, muestra resultados aceptables desde el punto de vista del Sobrepaso máximo (%), pues este no llega al rango de valores de entre 10% hasta el 60%, si se

aplicaran las reglas de Ziegler-Nichols [1]. En términos de Tiempo de asentamiento observamos que a partir del Setpoint de 35 cm este parámetro se duplica, lo cual nos indica que la zona de trabajo [35cm – 40 cm] es complicada tomando en cuenta el tiempo necesario para alcanzar la estabilidad en el sistema.

### **6.3 PUESTA EN MARCHA CON LOS AJUSTES ÓPTIMOS DE CONTROL**

Considerando todos los parámetros de tiempo, y analizando que 6% es un valor de Sobrepasso máximo más que aceptable, podemos seleccionar como punto de operación para nuestro proceso el Setpoint de 25 cm, procedemos a realizar la correspondiente Puesta en Marcha Figura 6.20



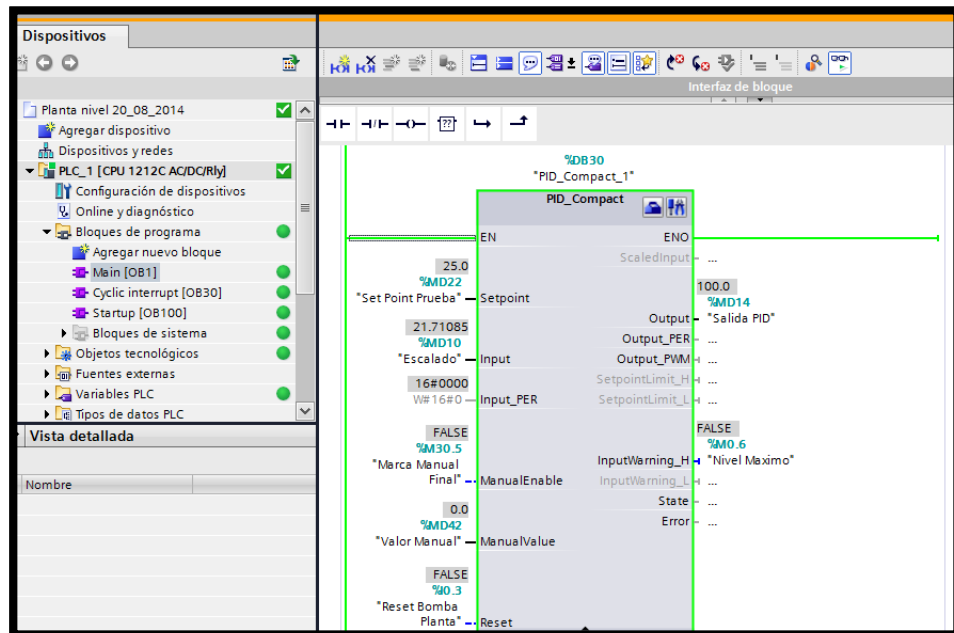


Figura 6.20 Puesta en marcha para Setpoint : 25 cm

Entonces podemos confirmar después de las pruebas realizadas que, los ajustes obtenidos por Autotuning del PID\_Compact son óptimos, pues funcionan adecuadamente para un amplio rango de valores de la Planta de Nivel, dejamos establecido en la Figura 6.21 Cuáles son los valores PID con los que se trabajó en este proyecto.

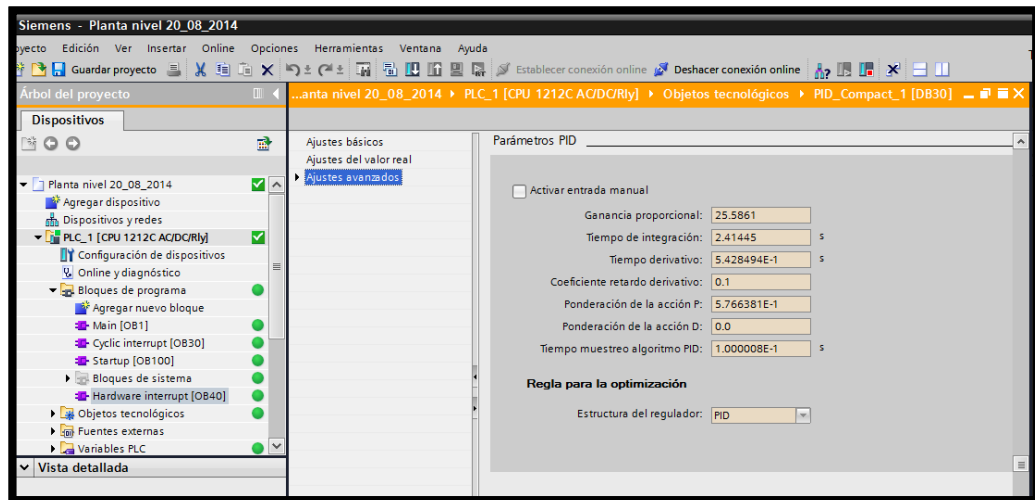


Figura 6.21 Valores PID con los que se trabajó en el proyecto.

## **CAPÍTULO 7**

# **GUÍA DE PRÁCTICAS PARA EL LABORATORIO DE INSTRUMENTACIÓN INDUSTRIAL.**

## 7.1 PRÁCTICA # 1: INTRODUCCION A TIA PORTAL.

### Objetivos:

Iniciar al estudiante en el uso del software TIA Portal para la programación de PLC's de la marca Siemens.

Escoger el PLC y el módulo correcto que se tenga en físico en el programa.

Utilizar la ayuda del software para conocer el funcionamiento del programa, y todo lo que permite realizar en el mismo.

**Introducción:** TIA Portal (Totally Integrated Automation) es un entorno para programación de PLC's y equipos de automatización de la marca Siemens orientado a facilitar la interacción entre el usuarios y los dispositivos, permite integrar distintas aplicaciones de software industrial para procesos de producción en un mismo interfaz, facilitando en gran medida el aprendizaje, la interconexión y la operación, sin una variedad amplia de sistemas de diferentes orígenes. [6]

Es fácil de manejar ya que la interfaz de usuario es muy sencilla, no se pierde mucho tiempo en aprender a utilizar un nuevo software y se puede dedicar más tiempo a la ingeniería. En la siguiente imagen podemos observar el nivel de complejidad al que se puede llegar integrando PLC's, HMI's, variadores de frecuencia, accionamientos y computadores mediante diferentes protocolos de comunicación gracias a las facilidades que nos brinda TIA PORTAL para configurar la comunicación entre dispositivos. [6].

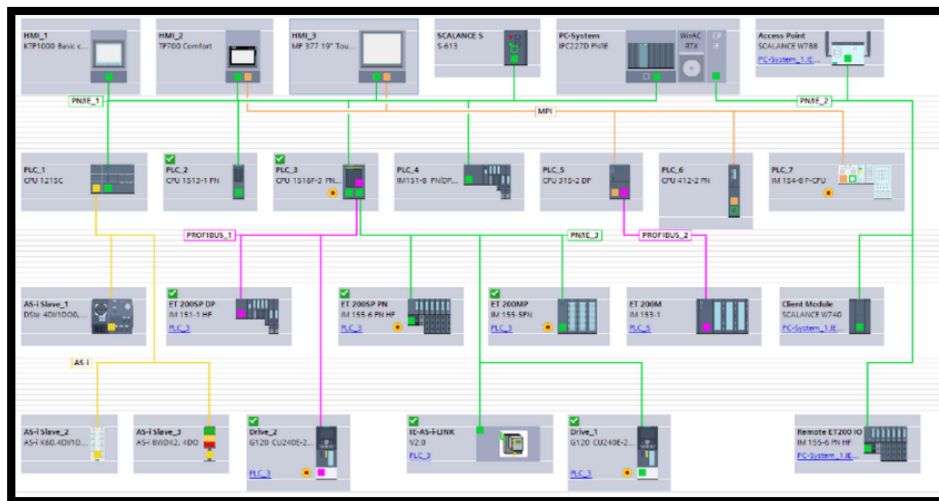


Figura 7.1 Comunicación entre dispositivos SIEMENS con TIA PORTAL

### 7.1.1 CREACIÓN DE UN PROYECTO

Para crear un proyecto en TIA PORTAL damos clic en el acceso directo que se crea en el escritorio al momento de instalar el programa o también en la dirección INICIO/Todos los programas /Siemens Automation/TIA Portal V11.

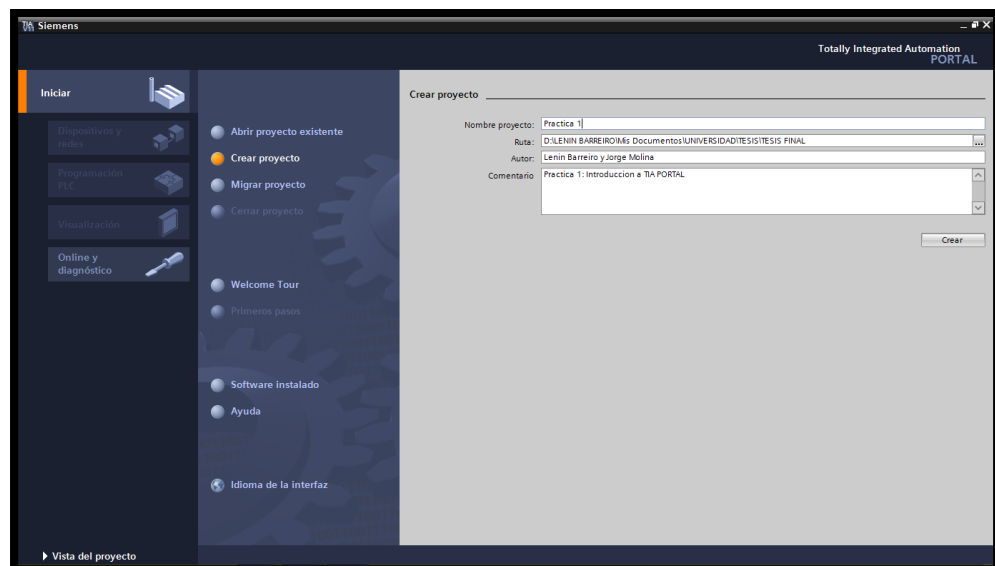


Figura 7.2 Vista del Portal

La primera pantalla que aparece es la vista del portal Figura 7.2 la cual es una vista orientada a las tareas del proyecto.

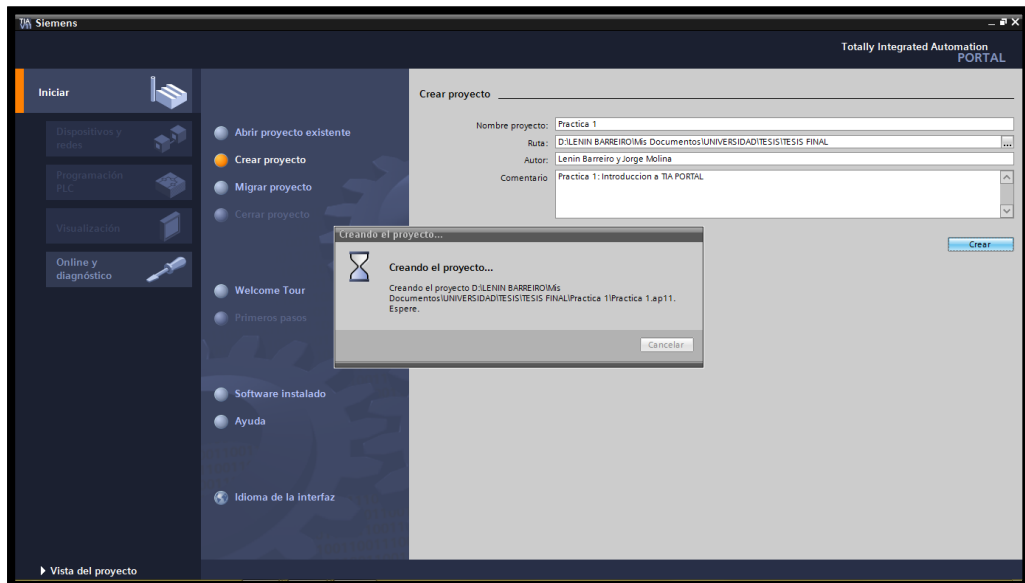


Figura 7.3 Creando el proyecto

Seleccionamos crear proyecto, Figura 7.3 damos un nombre al mismo, seleccionamos la ubicación en la que se va a guardarlo, adicionamos un comentario sobre que trata el proyecto y damos clic en crear.

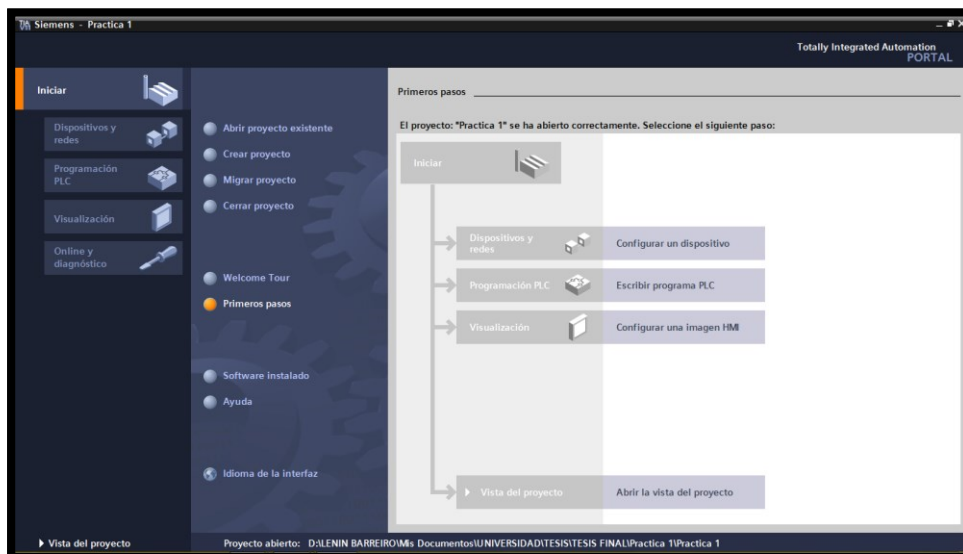


Figura 7.4 Configuración de dispositivo

Una vez creado el Proyecto escogemos Configurar un dispositivo, Figura 7.4

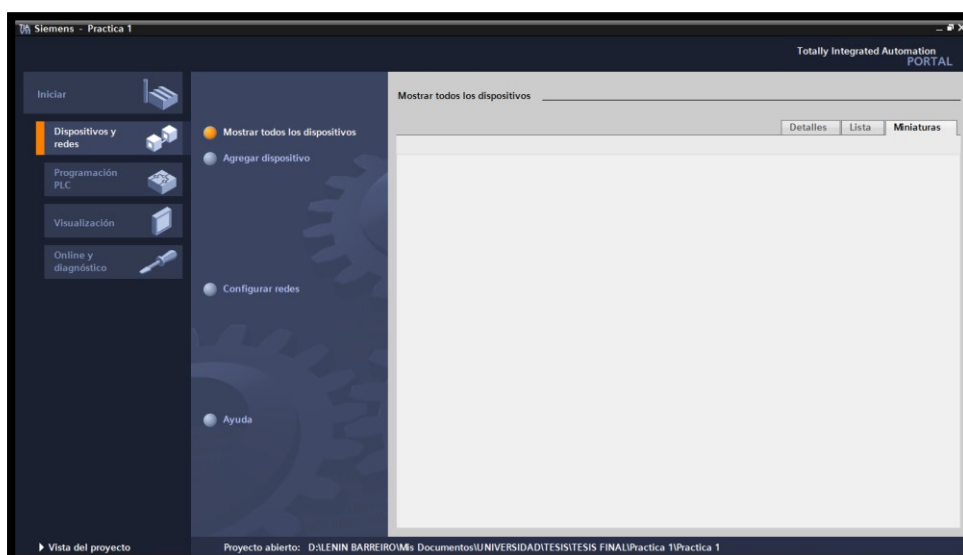


Figura 7.5 Mostrar los dispositivos



En la Figura 7.5 seleccionamos Mostrar Dispositivos como previamente no hemos agregado ninguno, esta vista sale vacía.

Escogemos Agregar dispositivo Figura 7.6

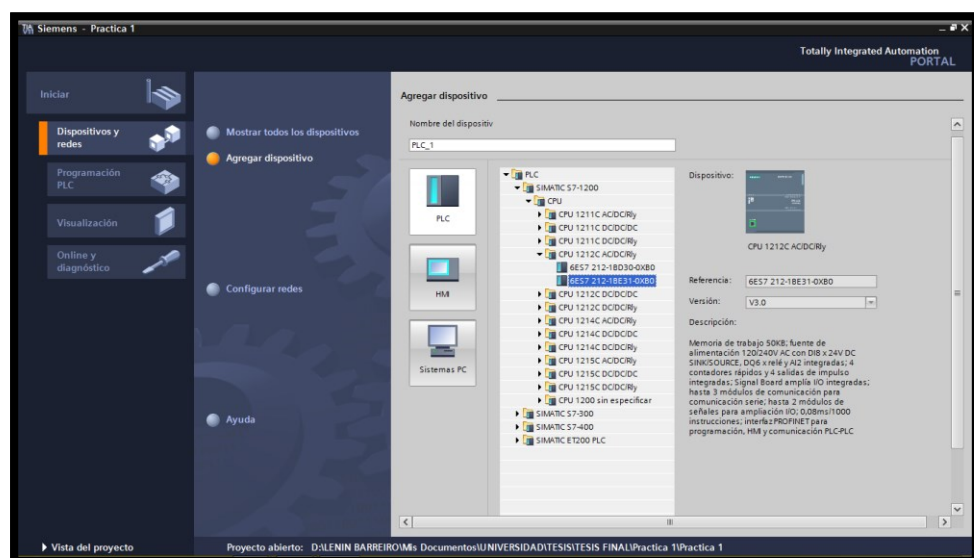


Figura 7.6 Agregar el dispositivo.

Luego escogemos el PLC que tenemos en disponible el cual es un Simatic S7 1200 con un CPU 1212C AC/DC/Rly y damos clic en agregar. Cabe recalcar que la versión del firmware debe coincidir con el que tenemos en el PLC. En caso de no tener la misma versión se deberá actualizarlo bajándolo de la página de soporte de Siemens y cargarlo mediante una sd card.

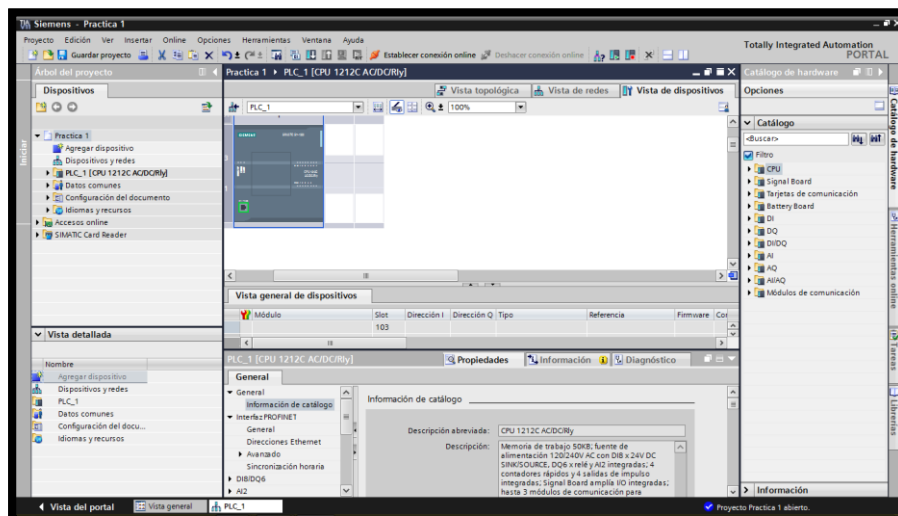


Figura 7.7 Vista del proyecto.

Una vez agregado el dispositivo aparece la pantalla **La vista del proyecto** que es una vista de los componentes del proyecto con sus áreas de trabajo y editores, sirve para programar y editar todos los parámetros del PLC y sus módulos.

### 7.1.2 MENUS DE AYUDA DE SOFTWARE Y CONFIGURACION DE VISTA

Una manera fácil de aprender todo lo relacionado a TIA Portal es acceder al menú de ayuda del programa, el cual presenta una interfaz muy sencilla y didáctica para ser manejado por el usuario. Para acceder al menú elija el comando "Mostrar ayuda" del menú "Ayuda" o pulse <F1> para visualizarlo.

Una vez hecho lo anteriormente mencionado aparecerá la siguiente pantalla



Figura 7.8 Opciones de ayuda.

En esta pantalla se muestran todas las opciones que maneja el menú de ayuda Figura 7.8.

**Contenido:** Muestra ayuda relacionada a TIA PORTAL ordenada por diferentes temas los cuales tienen pestañas para la expansión a subtemas asociados. Figura 7.9.

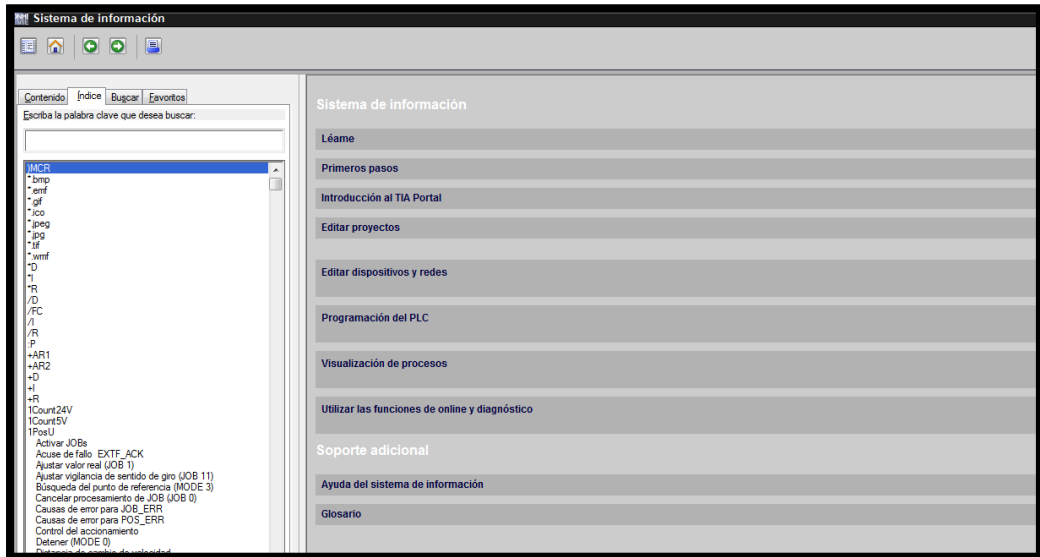


Figura 7.9 Índice de ayuda ordenada por temas

**Índice:** Muestra todos los temas de ayuda pero ordenados alfabéticamente.



Figura 7.10 Opción Buscar

**Buscar:** En esta opción escribimos una palabra clave del tema que necesitamos saber, damos Enter o clic en “Mostrar temas” y aparecen todos los temas relacionados con la palabra clave, Figura 7.10.

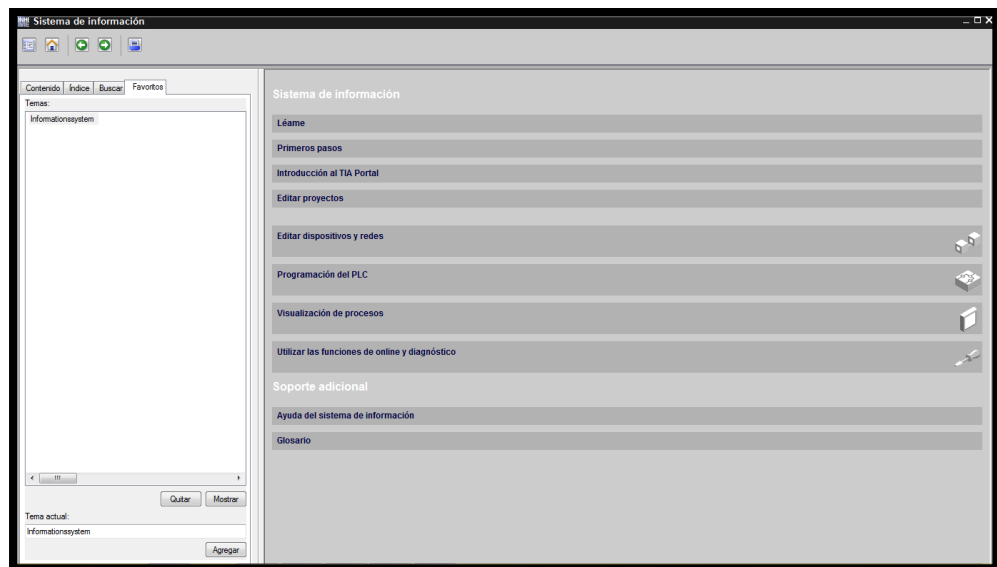


Figura 7.11 Vista favoritos

**Favoritos:** Aquí vamos guardando los temas que a nuestro parecer sean de gran importancia en un futuro. Figura 7.11.

### 7.1.3 CONFIGURACIÓN DE HARDWARE Y RED.

Para crear un sistema de automatización, los diferentes componentes de hardware deben configurarse, parametrizarse y

conectarse entre sí. Las tareas correspondientes se realizan en la vista de dispositivos y redes.

**Función del editor de hardware y redes:** Al hacer doble clic en la entrada "Dispositivos y redes" del árbol del proyecto se abre el editor de hardware y redes. El editor de hardware y redes es el entorno de desarrollo integrado para configurar, parametrizar y conectar dispositivos y módulos en red. Ofrece todo el soporte necesario para realizar el proyecto de automatización

**Estructura del editor de hardware y redes:** El editor de hardware y redes incluye los componentes siguientes. Figura 7.12.

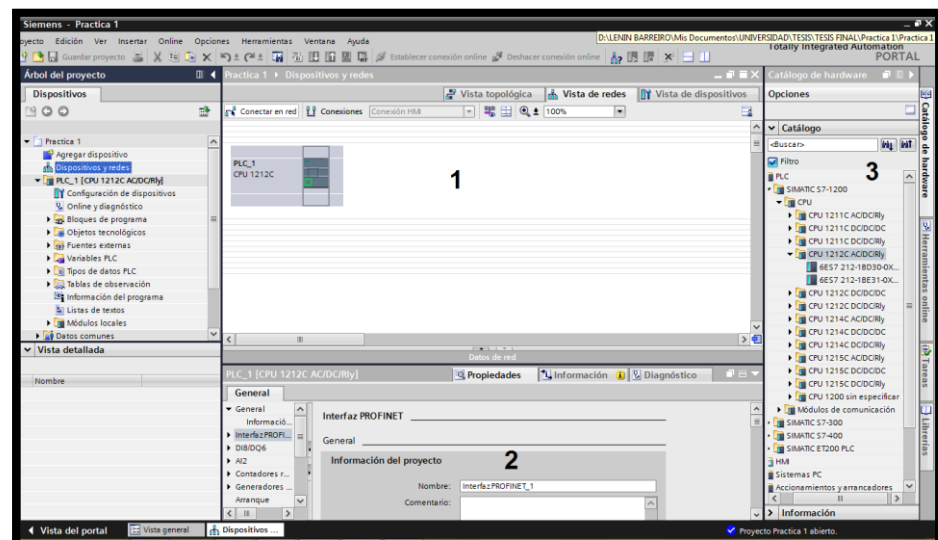


Figura 7.12 Editor de hardware y redes

**Vista de dispositivos, Vista de redes y Vista topológica (1):** El editor de hardware y redes ofrece tres vistas distintas para su proyecto. Es posible cambiar entre estas tres vistas en cualquier momento, en función de si deben crearse y editarse dispositivos y módulos individuales, redes y configuraciones de dispositivos enteras o bien la estructura topológica del proyecto.

**Ventana de inspección (2):** La ventana de inspección ofrece información sobre el objeto actualmente seleccionado. También permite modificar los ajustes del objeto seleccionado.

**Catálogo de hardware (3):** Desde el catálogo de hardware se arrastran los dispositivos y módulos necesarios para el sistema de automatización hasta la vista de dispositivos, redes o topología.

Se pueden utilizar dos o más PLC's si un proyecto lo necesita para lo cual arrastramos el PLC desde el catálogo de hardware y lo conectamos con el que nos viene por default que es el que seleccionamos al momento de crear el proyecto. En nuestro caso solo trabajamos con un PLC y un módulo de entradas y salidas analógicas ya que las que vienen con el PLC no son compatibles o

suficientes para la cantidad y tipo de sensores que vamos a utilizar.

El modulo con el que contamos es SM 1234 AI4/AQ2 que es un módulo de 4 entradas y dos salidas analógicas configurables entre corriente y voltaje. Para agregarlo al PLC damos doble clic en la entrada "Configuración de dispositivos" del árbol del proyecto y aparece la siguiente pantalla. Figura 7.13

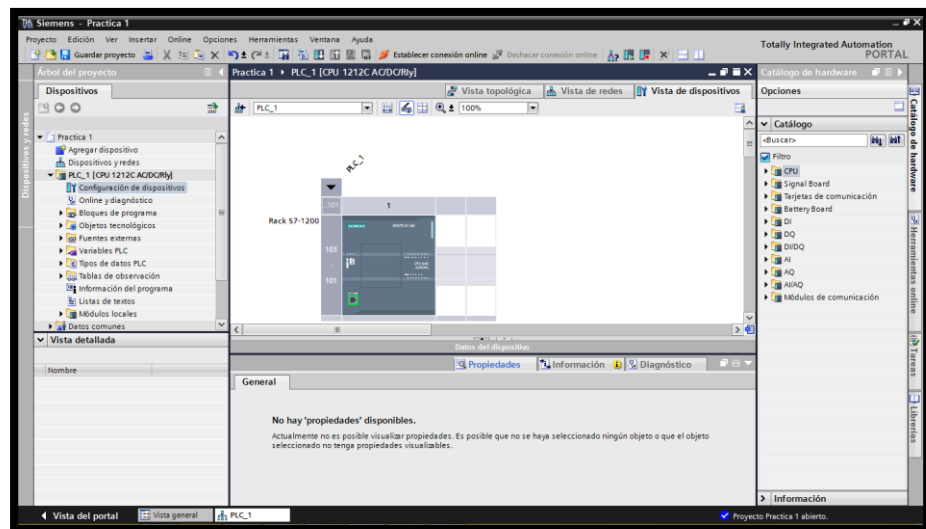


Figura 7.13 Configuración de dispositivos.

Nótese que el catálogo de hardware cambia y ahora aparecen todos los módulos que se pueden agregar a nuestro PLC. Del catálogo de hardware escogemos el módulo de entradas y salidas



analógicas con referencia 6ES7 234-4HE32-0XB0 que es el que nosotros tenemos, lo arrastramos y lo colocamos junto al PLC.

Figura 7.14.

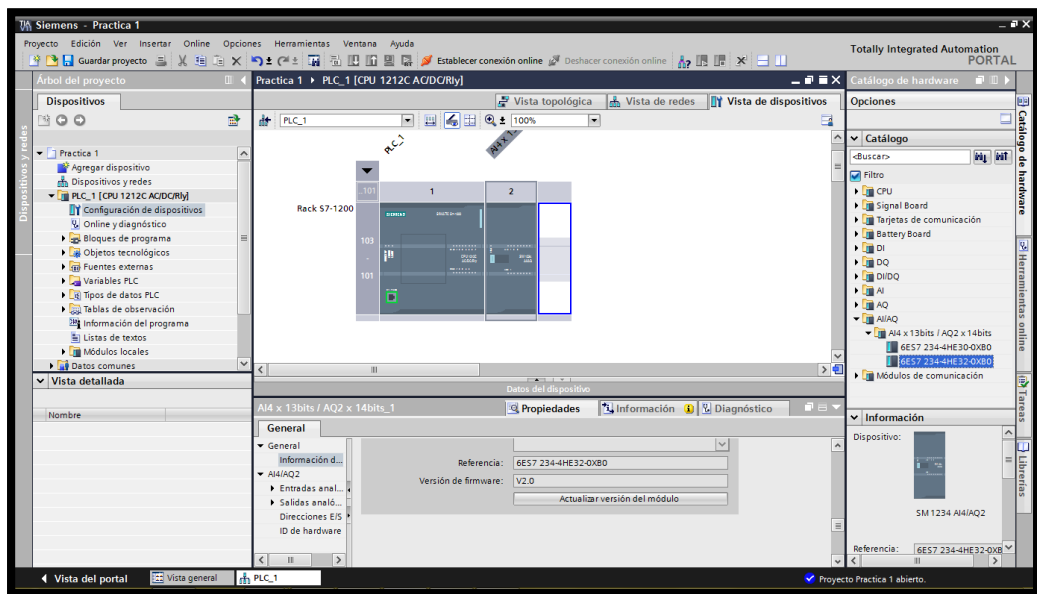


Figura 7.14 Catálogo de hardware

Trabajamos con la versión de firmware 2.0, en caso de no venir cargada esta versión en el módulo descargarla de la página de soporte de Siemens.

Una vez cargado el módulo tenemos completos todo lo que vamos a utilizar en nuestro proyecto. Dando clic sobre el PLC o el módulo aparece en la ventana de diagnóstico información sobre

los mismos, ahí podemos observar información general del dispositivo es decir serie, descripción, voltaje de alimentación y todas las especificaciones del mismo. También se pueden configurar los parámetros de contadores, entradas y salidas sean estas digitales o analógicas. En prácticas posteriores veremos la parametrización de estos valores dependiendo de la aplicación y sensores que tengamos a mano. Para esto guarde el proyecto creado ya que en el mismo se proseguirá con las demás prácticas.

**Tarea:**

- ¿Qué es y para qué sirve TIA Portal?
- Liste 4 PLC's de la Marca Siemens, describa cada uno, sus características y ordénelos de acuerdo a su nivel de recursos.
- ¿Qué es STEP 7 y WinCC? Describa su relación con TIA Portal.

## 7.2 PRÁCTICA # 2: PROGRAMACION BASICA DE PLC S7 1200

### Objetivos:

- Conocer los tipos de variables que maneja el PLC Siemens S7 1200.
- Conocer los diferentes bloques de programación que maneja el programa TIA Portal.
- Hacer un programa sencillo en TIA Portal.
- Aprender a compilar un programa en TIA Portal y cargarlo en el PLC.

**Introducción:** TIA Portal nos brinda un entorno muy sencillo para las tareas de programación, en esta práctica vamos a conocer estas virtudes para poder explotar al máximo todos los recursos que nos brinda el programa.

### 7.2.1 MANEJO DE VARIABLES: DIRECCIONAMIENTO Y CONFIGURACION

Procedemos a abrir el proyecto guardado de la práctica #1, damos clic en configuración de dispositivos y hacemos un zoom al PLC y el módulo. Figura 7.15

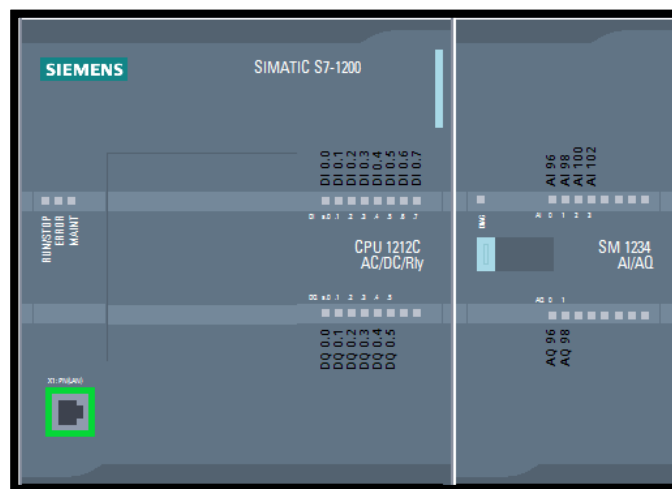


Figura 7.15 PLC y Módulo

Podemos tener una primera impresión de lo que son las variables y las direcciones de memoria de las variables que maneja el PLC reflejadas visualmente como entradas y salidas sean estas digitales o analógicas.

En este caso las variables digitales de entradas ocupan el byte 0 de las direcciones de memoria y cada entrada digital ocupa un bit

del mismo es por esto que van de 0 a 7 ocupando los 8 bit del byte.

Es los mismo para las salidas digitales, por lo contrario las entradas y salidas analógicas ocupan 2 bytes de memoria completos.

**Tipos de datos:** Se nombran los tipos de datos más comunes, estos son los que vamos a utilizar para nuestras prácticas.

**BOOL:** Un operando del tipo de datos BOOL representa un valor de bit y puede contener uno de los siguientes valores:

- TRUE
- FALSE

**BYTE:** Un operando del tipo de datos BYTE es una secuencia de 8 bits.

**WORD:** Un operando del tipo de datos WORD es una secuencia de 16 bits.

**DWORD:** Un operando del tipo de datos DWORD es una secuencia de 32 bits.

**SINT:** Un operando del tipo de datos SINT tiene una longitud de 8 bits y consta de dos componentes: Un signo y un valor numérico en complemento a dos. Los estados lógicos de los bits 0 a 6 representan el valor de posición del número. El estado lógico del bit 7 representa el signo. El signo puede adoptar los valores "0" (positivo) ó "1" (negativo). Un operando del tipo de datos SINT ocupa un byte en la memoria.

La Figura 7.16 muestra el entero +44 como número binario:

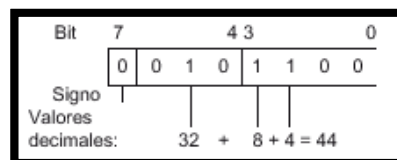


Figura 7.16 Número SINT

**INT:** Un operando del tipo de datos INT tiene una longitud de 16 bits y consta de dos componentes: Un signo y un valor numérico en complemento a dos. Los estados lógicos de los bits 0 a 14 representan el valor de posición del número. El estado lógico del

bit 15 representa el signo. El signo puede adoptar los valores "0" (positivo) ó "1" (negativo). Un operando del tipo de datos INT ocupa dos bytes en la memoria

La Figura 7.17 muestra el entero +44 como número binario:

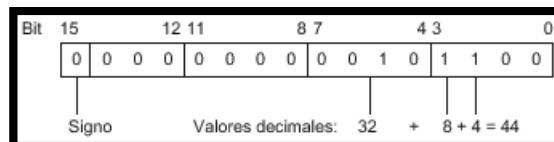


Figura 7.17 Número INT

**DINT:** Un operando del tipo de datos DINT tiene una longitud de 32 bits y consta de dos componentes: Un signo y un valor numérico en complemento a dos. Los estados lógicos de los bits 0 a 30 representan el valor de posición del número. El estado lógico del bit 31 representa el signo. El signo puede adoptar los valores "0" (positivo) ó "1" (negativo). Un operando del tipo de datos DINT ocupa cuatro bytes en la memoria.

**REAL:** Los operados del tipo de datos REAL Figura 7.18 tienen una longitud de 32 bits y se utilizan para representar valores en

coma flotante. Un operando del tipo de datos REAL consta de los tres componentes siguientes:

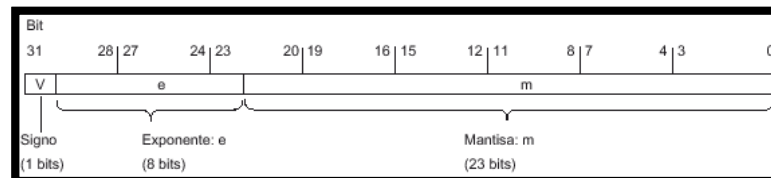


Figura 7.18 Dato real

- Signo: el estado lógico del bit 31 determina el signo. El bit 31 puede adoptar los valores "0" (positivo) ó "1" (negativo).
- Exponentes de 8 bits en base 2: El exponente se incrementa en una constante (base, +127), de modo que presenta un rango de valores de entre 0 y 255.
- Mantisa de 23 bits: se representa sólo la fracción de la mantisa. En los números en coma flotante normalizados, la parte entera de la mantisa siempre es 1, y no se guarda.

**LREAL:** Los operados del tipo de datos LREAL tienen una longitud de 64 bits y se utilizan para representar valores de coma



flotante. Un operando del tipo de datos LREAL consta de los tres componentes siguientes:

- Signo: El estado lógico del bit 63 determina el signo. El bit 63 puede adoptar los valores "0" (positivo) ó "1" (negativo).
- Exponentes de 11 bits en base 2: El exponente se incrementa en una constante (base, +1023), de modo que presenta un rango de valores de entre 0 y 2047.
- Mantisa de 52 bits: Se representa sólo la fracción de la mantisa. En los números en coma flotante normalizados, la parte entera de la mantisa siempre es 1, y no se guarda, Figura 7.19.

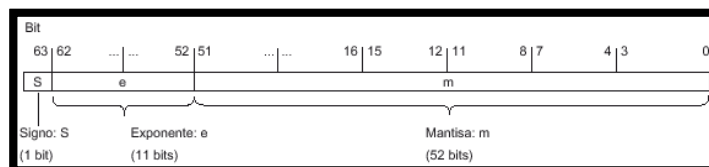


Figura 7.19 Componentes número Lreal

Para conocer los demás tipos de datos que se pueden utilizar para programa en TIA Portal acceda al menú de ayuda y búsqueda en la parte de contenido la siguiente dirección:

Programación del PLC/Cree el programa de usuario/Principios básicos de programación/Tipos de datos

### **7.2.2 BLOQUES: TIPO Y PROGRAMACION**

Programación lineal y estructurada

**Programación lineal:** Las tareas de automatización pequeñas pueden solucionarse escribiendo todo el programa de usuario linealmente en un OB de ciclo. Esto se recomienda sólo para programas sencillos. La Figura 7.20 muestra esquemáticamente un programa lineal: El OB de ciclo "Main1" contiene el programa de usuario completo.

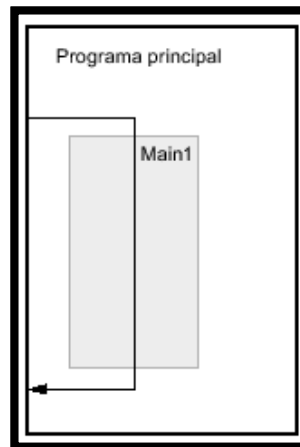


Figura 7.20 Programación lineal [4]

**Programación estructurada:** Para simplificar la creación y mantenimiento de tareas complejas de automatización, es conveniente subdividirlas en tareas parciales más pequeñas que correspondan a las diversas funciones tecnológicas del proceso de automatización o que puedan utilizarse varias veces. En el programa de usuario, estas tareas parciales se representan mediante bloques. Cada bloque es una sección independiente del programa de usuario. La estructuración del programa ofrece las ventajas siguientes:

- Los programas grandes pueden programarse de forma clara.

- Es posible estandarizar partes de programas y utilizarlas varias veces con diferentes parámetros.
- La organización del programa se simplifica.
- El programa se puede modificar más fácilmente.
- La comprobación del programa se simplifica, puesto que puede realizarse por secciones.
- La puesta en marcha se simplifica.

La figura siguiente muestra un esquema de un programa estructurado: El OB de ciclo "Main1" llama subprogramas consecutivamente que, a su vez, ejecutan tareas parciales definidas.

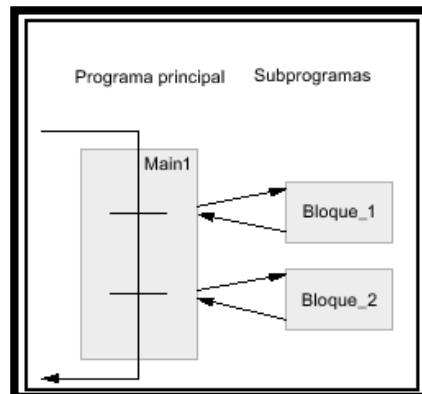


Figura 7.2121 Programación estructurada [4]

**Tipos de Bloques:** Para realizar las tareas en un sistema de automatización se dispone de distintos tipos de bloques. La Tabla 11 siguiente muestra los tipos de bloques disponibles.

TIPO DE BLOQUE	DESCRIPCIÓN BREVE
Bloques de Organización (OB)	Determinan la estructura del programa de usuario
Funciones (FC)	Contienen rutinas de programa para tareas que se repiten con frecuencia. Carecen de “memoria”
Bloques de Función (FB)	Son bloques lógicos que depositan sus valores de forma permanente en bloques de datos de instancia, de modo que siguen disponibles tras procesar el bloque
Bloques de Datos de Instancia	Cuando se llama a un bloque de función, se le asignan los bloques de datos de instancia, en los que se almacenan los datos del programa
Boques de Datos Globales	Los bloques de datos globales son áreas de datos para almacenar datos que pueden ser utilizados por cualquier bloque

Tabla 11 Tipo de bloques de programación

### 7.2.3 PROGRAMACIÓN CON CONTACTOS Y BOBINAS

Para iniciar esta sección abrimos el proyecto creado en la práctica anterior cuyo nombre es Práctica 1, lo guardamos como Práctica 2 y seguimos la siguiente ruta en el árbol de proyecto, Figura 7.22.

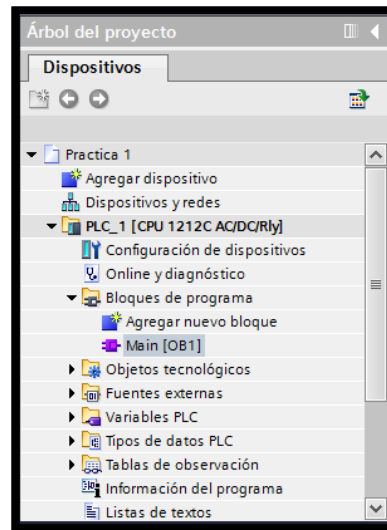


Figura 7.22 Árbol de proyecto

Damos doble clic en Main [OB1] y aparece la siguiente pantalla

Figura 7.23:

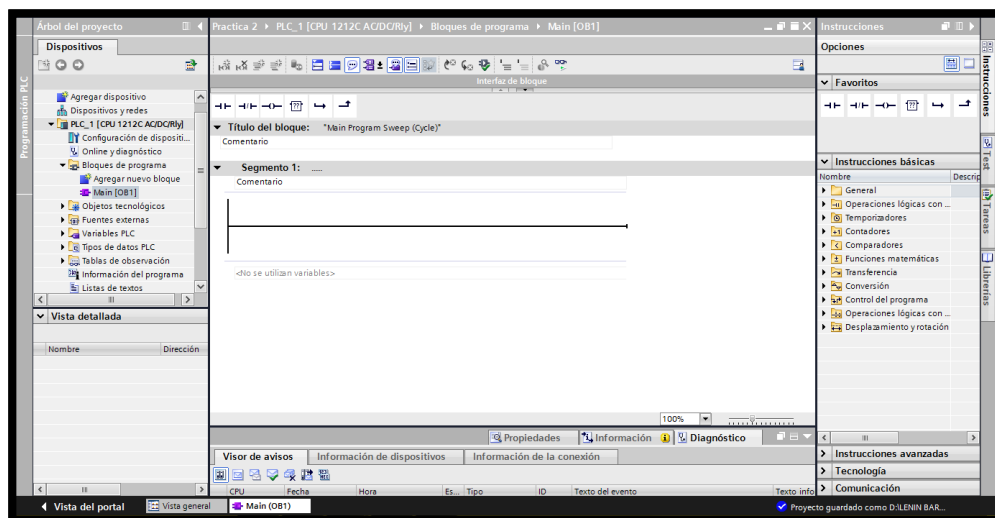


Figura 7.23 Bloque de programación Main.

Los contactos y las bobinas son los elementos más básicos de programación en lenguaje tipo ladder, al ser elementos tan usados, se encuentran sus accesos directos en la barra de favoritos. Figura 7.24

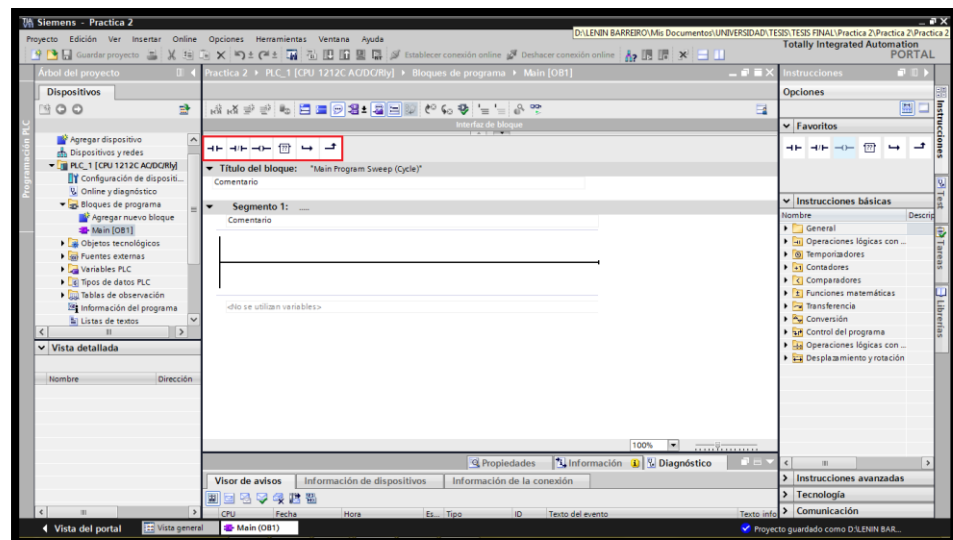


Figura 7.24 Barra de Favoritos en bloque MAIN

Para añadirlos basta únicamente con arrastrarlos al segmento donde se quieren ubicar. En la programación ladder los contactos se ubican a la izquierda (para evaluar) y las bobinas a la derecha (para modificar). Figura 7.25



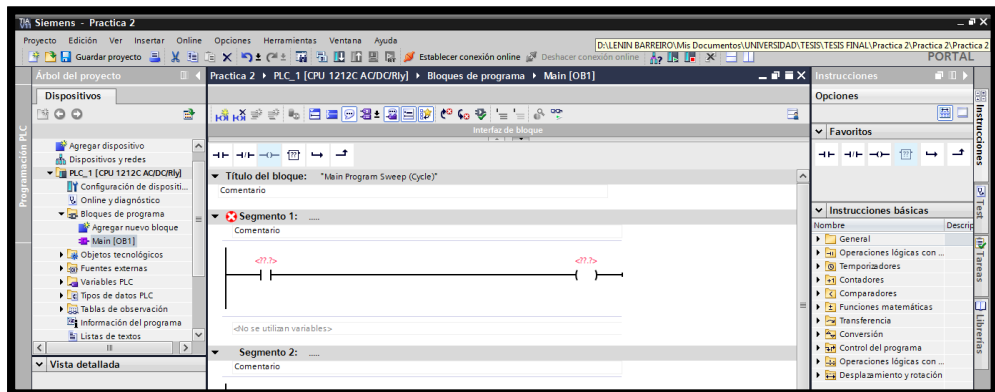


Figura 7.25 Contactos y Bobinas en MAIN

Adicional del lado derecho en Instrucciones Básicas tenemos todos los elementos necesarios para crear cualquier tipo de programa.

**Descripción del Programa a crear:** Para nuestra práctica vamos a crear un programa que controle el encendido y apagado de la bomba en la planta de nivel GUNT RT450 del laboratorio de Instrumentación Industrial de la FIEC y que indique mediante tres salidas lo siguiente.

- 1 salida que indique que la bomba este encendida
- 1 salida que indique que el nivel alcanzo los 20 cm

- 1 salida que indique que el nivel alcanzo los 40 cm

Al momento de pasar los 40 cm se debe apagar la bomba y no se debe poder encender a menos que el nivel descienda a menos de 20 cm.

Vamos a tener el valor de apertura de la válvula SAMSON al 100% constante, lo único que vamos a variar es el encendido y apagado de la bomba.

**Elaboración del Programa:** Primero añadimos dos entradas digitales 1 para el encendido y otra para el apagado de la bomba, las llamaremos Run Bomba y Stop Bomba. Figura 7.26.

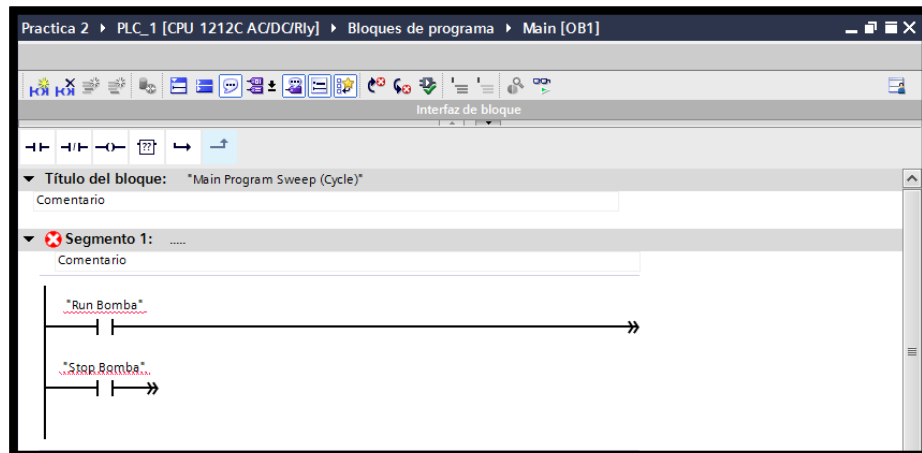


Figura 7.26 Entradas digitales Run Bomba y Stop Bomba

Luego añadimos la salida digital que va a mostrar que la bomba está encendida, Figura 7.27.

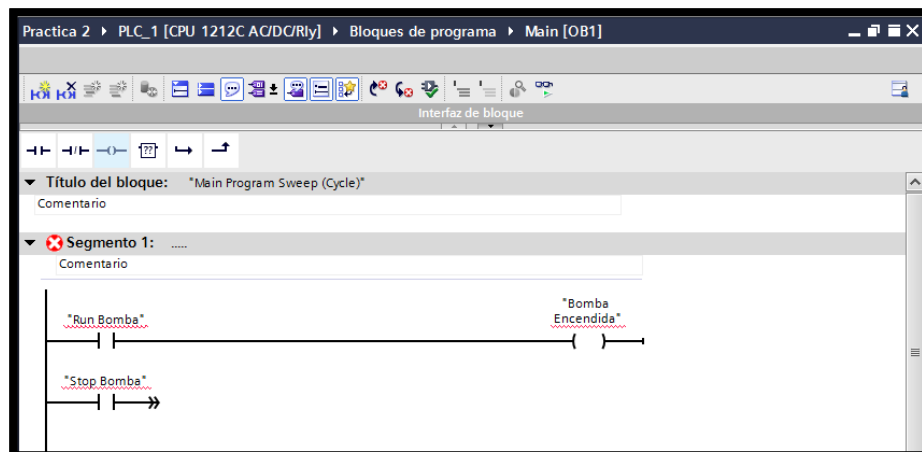


Figura 7.27 Salida Bomba Encendida

De aquí en adelante comenzamos a agregar marcas dependiendo de la lógica de nuestra programación.

En Título de Bloque escribimos una descripción general de los que nuestro programa va a hacer y en Segmento una descripción de los que las líneas de código dentro de este hacen.

Es recomendable separar por segmentos los distintos procesos que se van a realizar dentro del programa.

Luego de realizar la programación tenemos 5 segmentos. Podemos describir el funcionamiento de cada segmento por medio de diagramas de flujo.

**Segmento 1:**

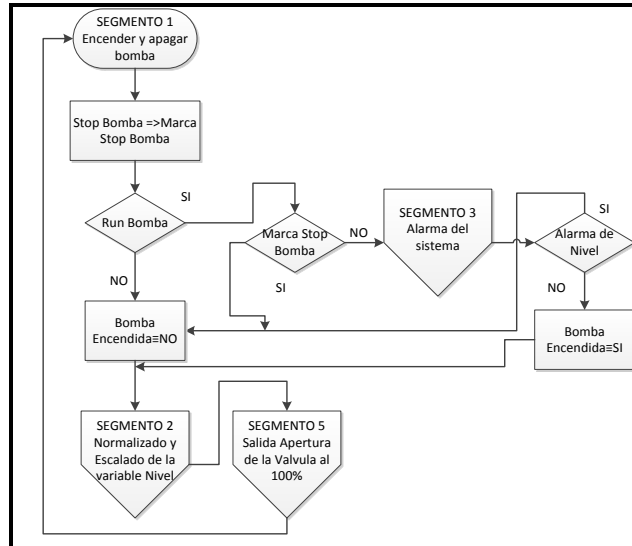


Figura 7.28 Diagrama de flujo encender y apagar bomba

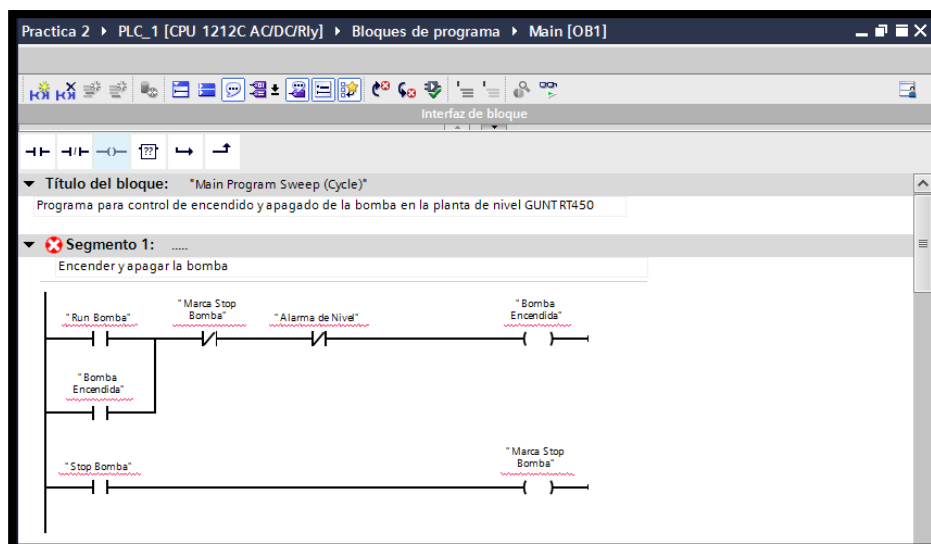


Figura 7.29 Programación encender y apagar bomba

Con la programación dentro de este segmento de código podemos encender la bomba mediante una salida digital llamada Bomba Encendida. Para esto la entrada digital Run Bomba debe de ser verdadera y las marcas digitales Marca Stop Bomba y Alarma de Nivel deben de ser falsas.

Para conocer el valor de la marca digital Alarma de Nivel debemos de acceder al segmento 3.

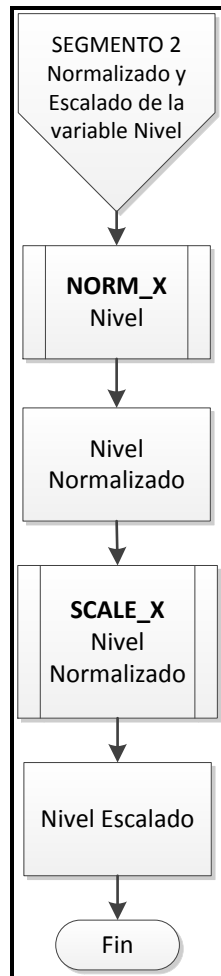
**Segmento 2:**

Figura 7.30 Diagrama de flujo de Segmento 2, normalizado y escalado de la variable de nivel

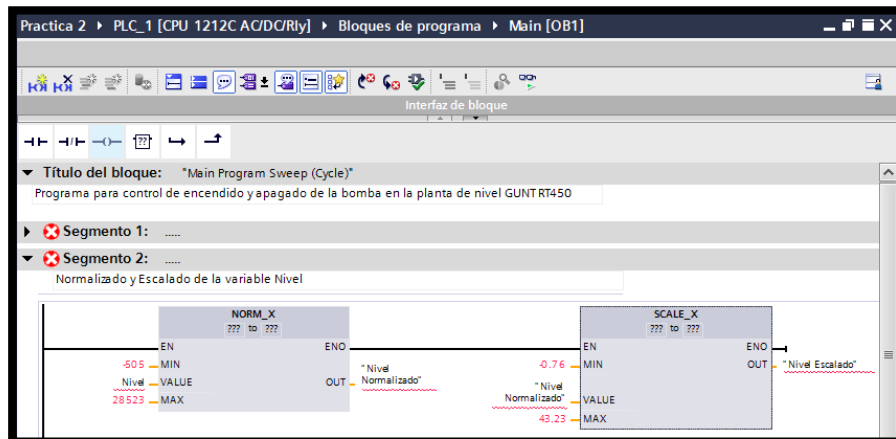


Figura 7.31 Programación de Segmento 2, normalizado y escalado de la variable de nivel

En el segmento 2, Figuras 7.30 y 7.31 se realiza el normalizado y escalado de la variable de entrada analógica que nos muestra el nivel de líquido en el tanque de la planta GUNT RT450. Esa variable primero la hacemos pasar por un bloque de TIA Portal llamado NORM\_X el cual permite representar la entrada que es en bits en un valor que va desde 0 a 1. Los valores puestos en los límites mínimo y máximo del bloque fueron escogidos de las pruebas realizadas para el proyecto y son la fiel representación del rango de nivel en bits con los que vamos a trabajar. Estos son los valores, Tabla 12.



NUMERO DE BITS ENTRADA ANALÓGICA	NIVEL EN CM
-505	-0,76
28523	43,23

Tabla 12 Relación entre número de bits y nivel en cm.

Para el bloque NORM\_X utilizamos los números de bits, Figura 7.32.

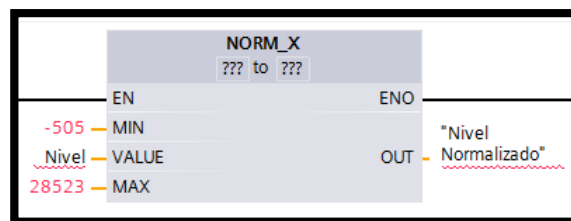


Figura 7.32 Bloque NORM\_X

Después de lo que se realiza el normalizado de la variable *Nivel* tenemos como resultado la variable *Nivel Normalizado*, esta variable la pasamos ahora por el bloque SCALE\_X. Este bloque nos permite convertir los valores que van de 0 a 1 a una representación lineal de los mismos que va entre el nivel mínimo y el nivel máximo.

Para el bloque SCALE\_X utilizamos el nivel en cm, Figura 7.33.

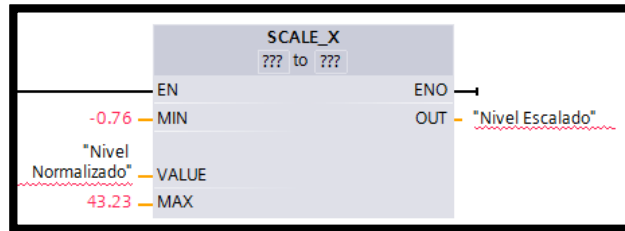


Figura 7.33 Bloque SCALE\_X

Finalmente tenemos como resultado la variable *Nivel Escalado* que es una representación real del nivel del tanque en cm.

### Segmento 3:

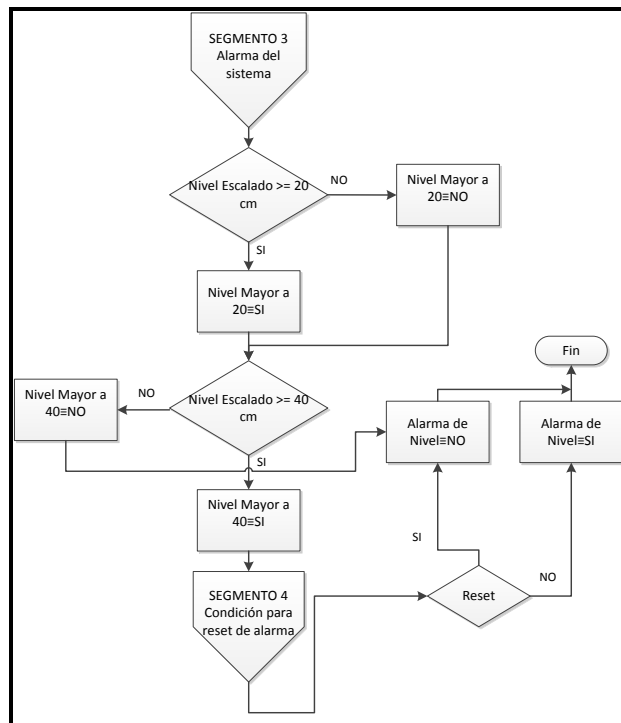


Figura 7.34 Diagrama de flujo del segmento 3, Alarma del Sistema

En el segmento 3 trabajamos con la variable Nivel Escalado y utilizamos en dos comparadores. En el primero se vuelve verdadera la salida Nivel mayor a 20 cuando el líquido sobrepasa los 20cm, en el segundo comparador se vuelve verdadera la Salida Mayor a 40 al pasar el nivel del líquido los 40cm. Figuras 7.34 y 7.35.

Para activar la variable Alarma de Nivel es necesario que la variable Nivel Mayor a 40 sea verdadera y que la variable Reset sea falsa, para conocer el valor de la variable Reset tenemos ver su lógica en el segmento 4.

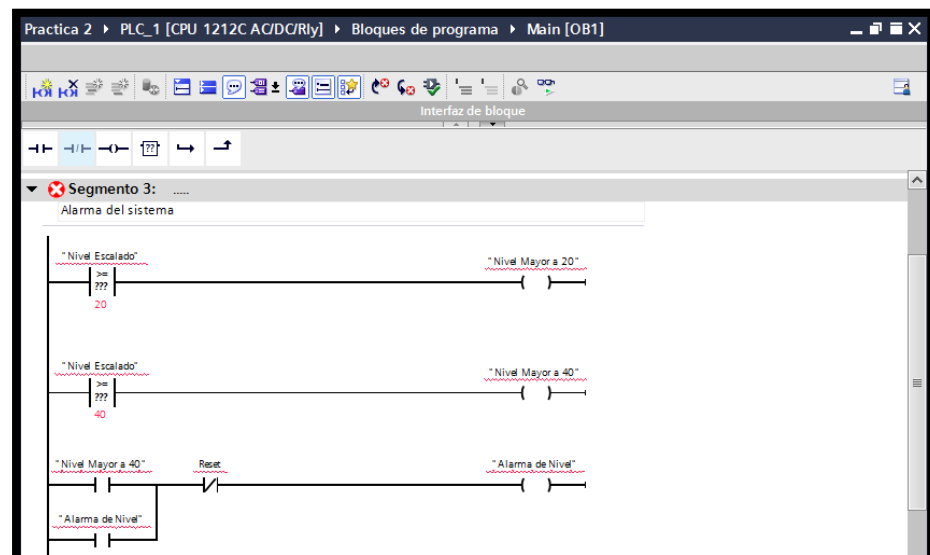


Figura 7.35 Programación del segmento 3, Alarma del Sistema

#### Segmento 4:

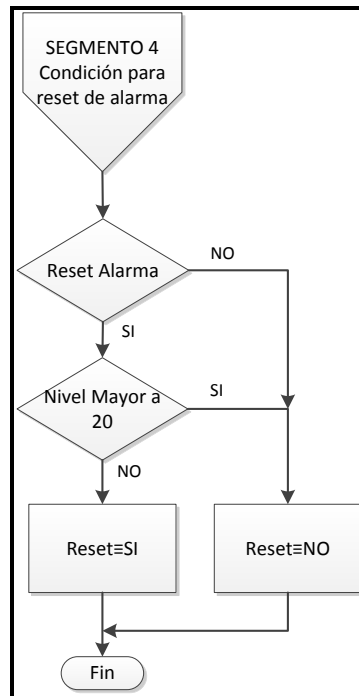


Figura 7.36 Diagrama de flujo del segmento 4, Condición Reset de Alarma

En el segmento 4 se vuelve verdadera la variable Reset siempre y cuando sea verdadera la variable Reset Alarma y la variable Nivel Mayor a 20. En este segmento de programación está el bloqueo para que no se pueda resetear la alarma a menos que el nivel del líquido baje de 20cm, Figuras 7.36, 7.37.

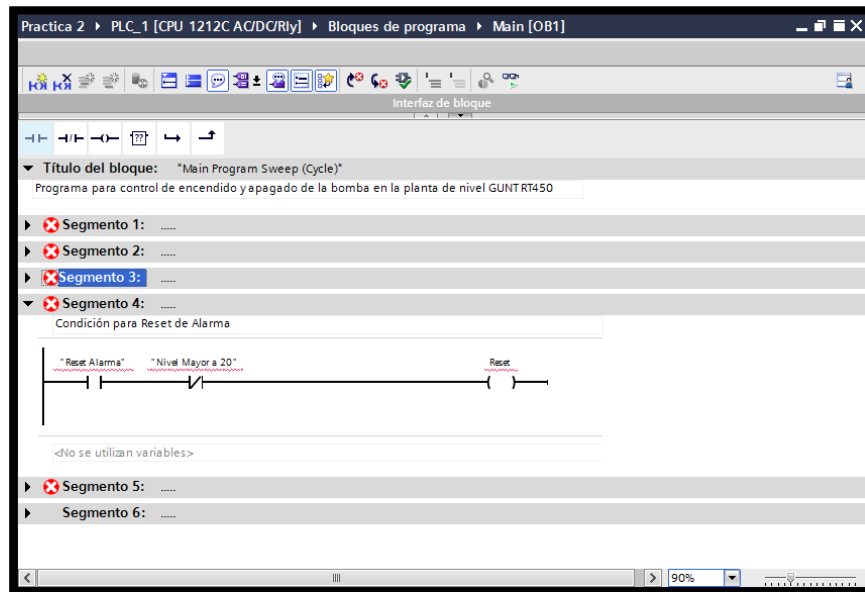


Figura 7.37. Programación del segmento 4, Condición Reset de Alarma

## Segmento 5

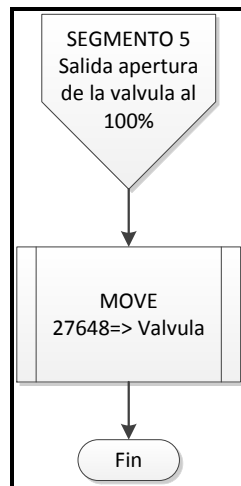


Figura 7.38 Diagrama de flujo del segmento 5, Salida apertura de la válvula al 100%

El segmento 5 se carga el valor 27648 en la variable *Válvula*, esto hace que la válvula se abra al 100%. Este valor es fijo para nuestro programa.

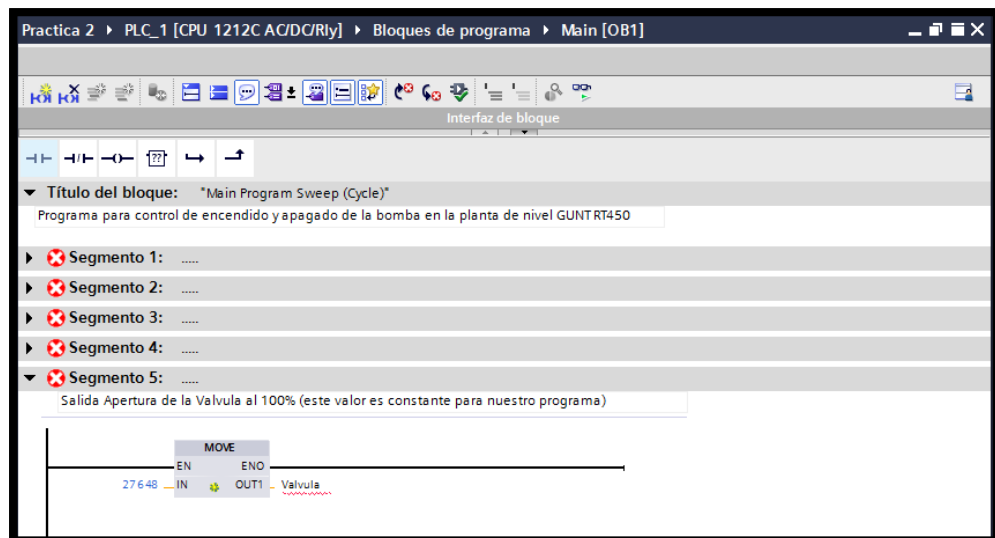


Figura 7.39 Programación del flujo del segmento 5, Salida apertura de la válvula al 100%

#### 7.2.4 DEFINICIÓN DE VARIABLES Y TABLAS DE SÍMBOLOS

**Definición de variables:** Hemos ya realizado toda la programación, tenemos todas las variables pero no las tenemos definidas. No es necesario conocer de antemano la disponibilidad de entradas y salidas, sino que se deben escribir los símbolos deseados para ellas.

Las variables aparecen subrayadas en rojo, TIA PORTAL identifica como error, debido a que son símbolos que no tienen dirección definida aún.

A continuación se muestra una manera simple de cómo asignar las direcciones de entradas y salidas.

Damos clic derecho sobre una variable y seleccionamos Definir variable, Figura 7.40.

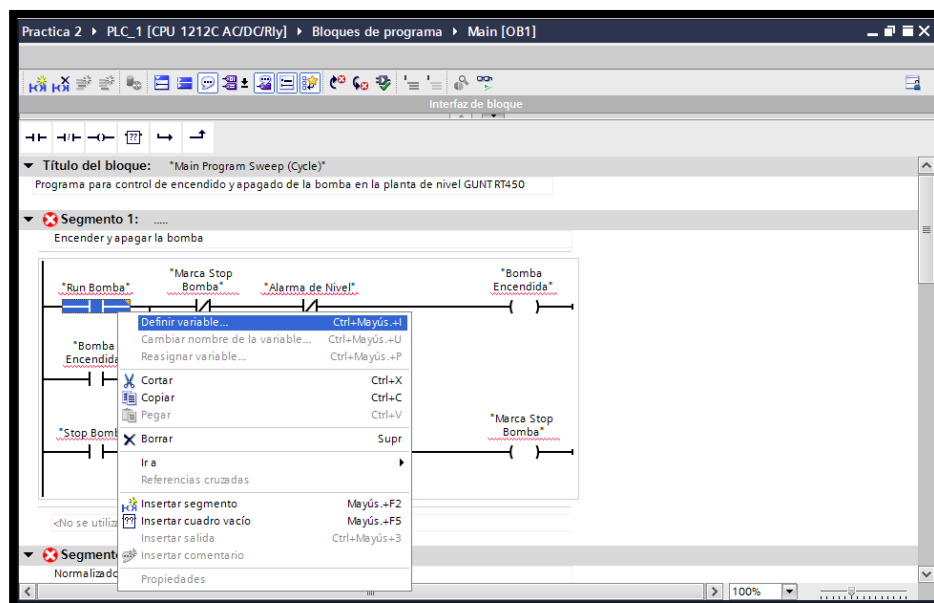


Figura 7.40 Definir variable

Aparece una ventana en la cual podemos escoger si la variable es una marca, una entrada o una salida Figura 7.41. Además aquí se elige en que tabla de variables se va a guardar. Hacemos esto para todas las variables.

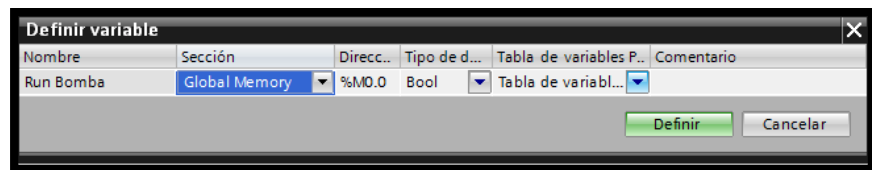


Figura 7.41 Definición de características de variable.

Después de haber asignado las direcciones a todas las variables estas ya no aparecen subrayadas con rojo, tomamos como ejemplo el segmento 1, Figura 7.42.



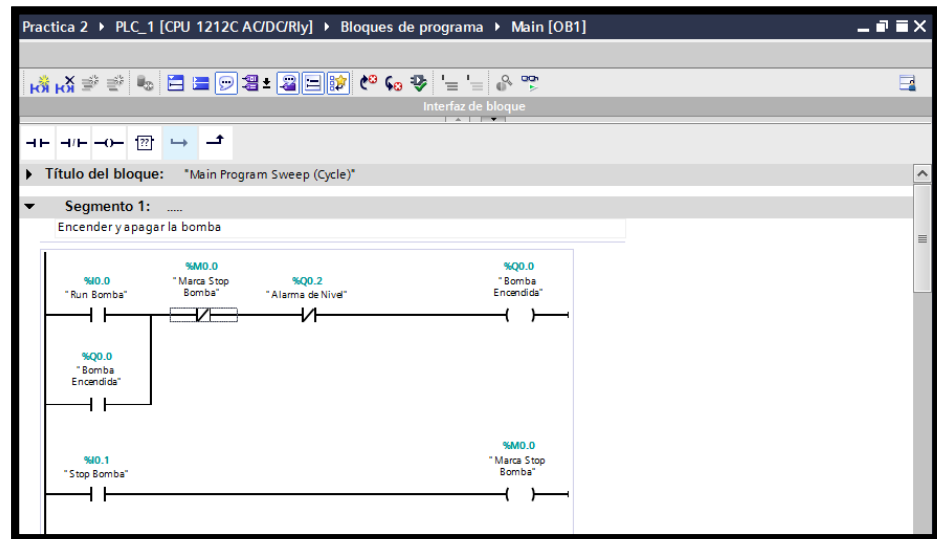


Figura 7.42 Asignación de direcciones a todas las variables

Además en la parte debajo de cada segmento aparece un resumen de todas las variables utilizadas en el mismo.

**Tabla de símbolos:** Como ya direccionamos las variables estas aparecen ya en la tabla de símbolos. Para acceder a esta seguimos la siguiente dirección, Figura 7.43.

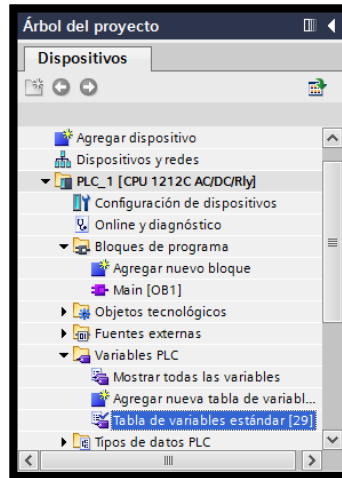


Figura 7.43 Acceso a tabla de variables estándar

Dando clic en Tabla de variables estándar aparece la misma, Figura 7.44.

	Nombre	Tipo de datos	Dirección	Rema...	Visible...	Accesi...	Comentario
1	Run Bomba	Bool	%I0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
2	Merca Stop Bomba	Bool	%M0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
3	Stop Bomba	Bool	%I0.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
4	Bomba Encendida	Bool	%Q0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
5	Nivel	Int	%IW96	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
6	Nivel Mayor a 20	Bool	%Q0.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
7	Nivel Mayor a 40	Bool	%M0.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
8	Alarma de Nivel	Bool	%Q0.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
9	Reset Alarma	Bool	%I0.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
10	Reset	Bool	%M0.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
11	Valvula	Int	%QW96	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
12	Nivel Normalizado	Real	%MD0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
13	Nivel Escalado	Real	%MD4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
14	<Agregar>			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

Figura 7.44 Tabla de variables estándar

Además podemos observar que en Configuración de dispositivos aparecen ya las variables que utilizamos en la representación física del PLC, Figura 7.45.

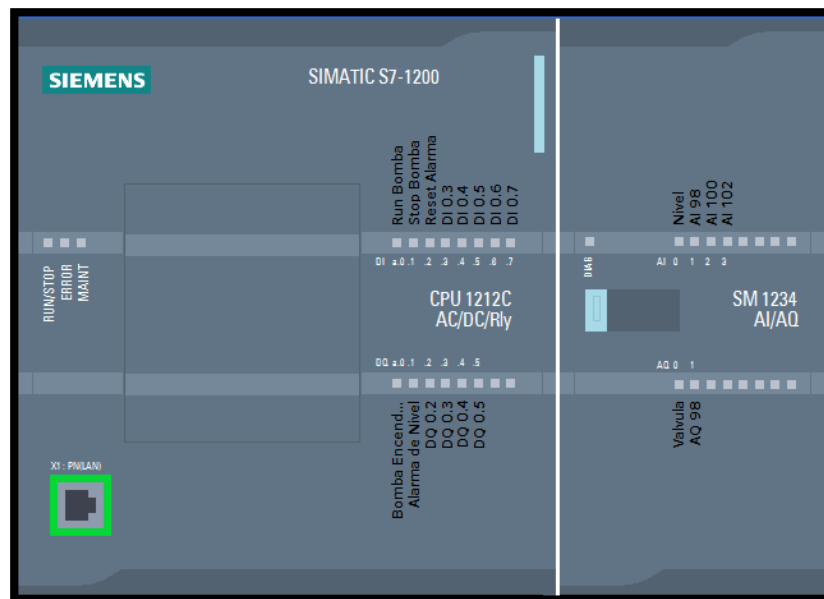


Figura 7.45 Variables en representación gráfica de PLC y Módulo SM 1234

### 7.2.5 DEPURACIÓN DEL PROYECTO Y COMPILACIÓN

Una vez lista toda la programación tenemos que compilar la misma para verificar errores o inconsistencias.

Una vez terminada la programación o si queremos evaluar su coherencia, podemos hacer clic en el botón “Compilar”, Figura 7.46.

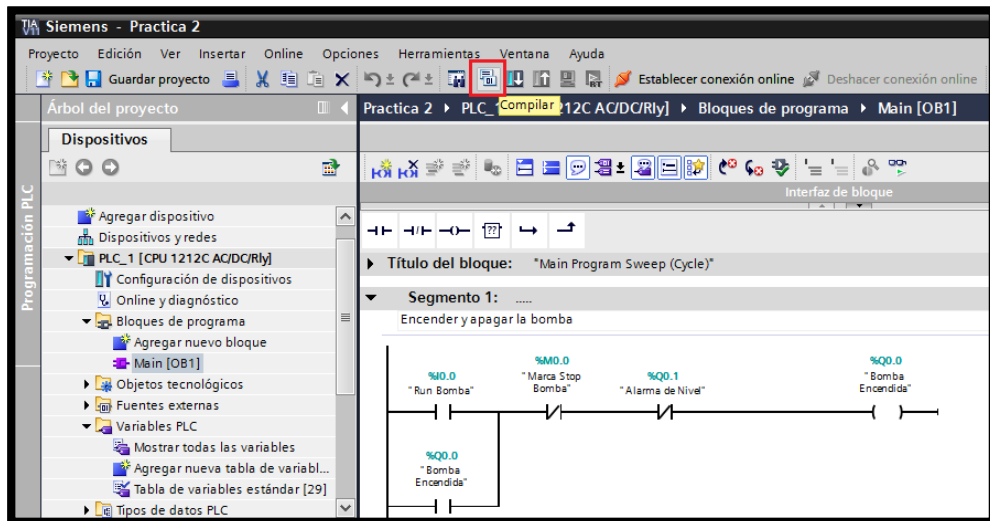


Figura 7.46 Compilación de programa en PLC

Una vez compilado y si no hay errores aparecen lo siguiente que muestra que la compilación fue exitosa, Figura 7.47.

General		Referencias cruzadas	Compilar	Sintaxis
Compilación finalizada (errores: 0; advertencias: 0)				
Ruta	Descripción	Fallos	Advertenc...	Hora
PLC_1		0	0	12:55:10
Bloques de programa		0	0	12:55:10
Main (OB1)	Bloque compilado correctamente.	0	0	12:55:10
	Compilación finalizada (errores: 0; advertencias: 0)	0	0	12:55:11

Figura 7.47 Compilación exitosa del programa del PLC

Por ultimo después de compilado procedemos a cargar el proyecto al PLC. Seleccionamos el PLC en el árbol del proyecto y damos clic en Cargar, Figura 7.48.

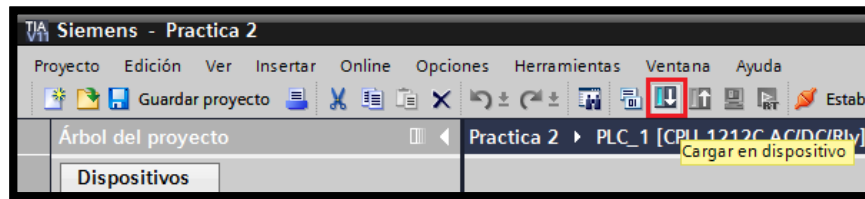


Figura 7.48 Carga del programa en el PLC.

### TAREA:

- Describa el funcionamiento de los bloques NORM\_X y SCALE\_X para esto utilice la ayuda de TIA Portal.
- Edite el programa para que la apertura de la válvula sea del 50%.
- Adicione al programa una salida digital que indique cuando el nivel este en un rango de 30 a 35cm.

### 7.3 PRÁCTICA #3 DISEÑO DE INTERFAZ PARA USUARIO EN LABVIEW PARA PLANTA DE NIVEL GUNT RT450

#### Objetivos:

- Familiarizar al estudiante con el entorno de programación de Lobbies.
- Aprender a activar una alarma cuando el valor de una variable llegue o sobrepase un límite establecido.
- Uso de comentarios como medio didáctico de información dentro de un VI.

**Introducción:** Labview (acrónimo de Laboratory Virtual Instrumentation Engineering Workbench) es un entorno de desarrollo mediante el cual con un lenguaje de programación visual gráfico se pueden diseñar diferentes sistemas. Este software es recomendado para sistemas de pruebas, control y diseño simulado o real y embebido, pues acelera la productividad. En él se utiliza el lenguaje llamado G que significa lenguaje gráfico.

Este programa fue creado por National Instruments (1976) y está disponible para las plataformas Windows, UNIX, MAC y GNU/Linux. La versión 2013 es la última completa y la 2014 está ya disponible en demo y se la puede descargar de la página de National Instruments [12].

### 7.3.1 VISUALIZACIÓN DE VARIABLES BOOLEANAS Y ANÁLOGAS.

**Variables Booleanas:** Primero creamos un proyecto en blanco en Labview con nombre Practica 3, un VI con el mismo nombre y los guardamos. Figura 7.49.

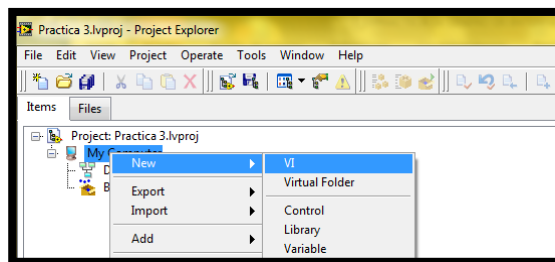


Figura 7.49 Creación de un proyecto en blanco de Labview

Una vez hecho esto abrimos el archivo Práctica 3.vi y comenzamos a trabajar. Teniendo en cuenta el programa elaborado en TIA Portal de la práctica 2 vamos a crear un VI con el cual podamos mostrar gráficamente dicho funcionamiento. Nuestro programa va a tener un funcionamiento cíclico por lo cual

creamos un While Loop o lazo while en el Block Diagram siguiendo la ruta que se muestra en la imagen, Figura 7.50.

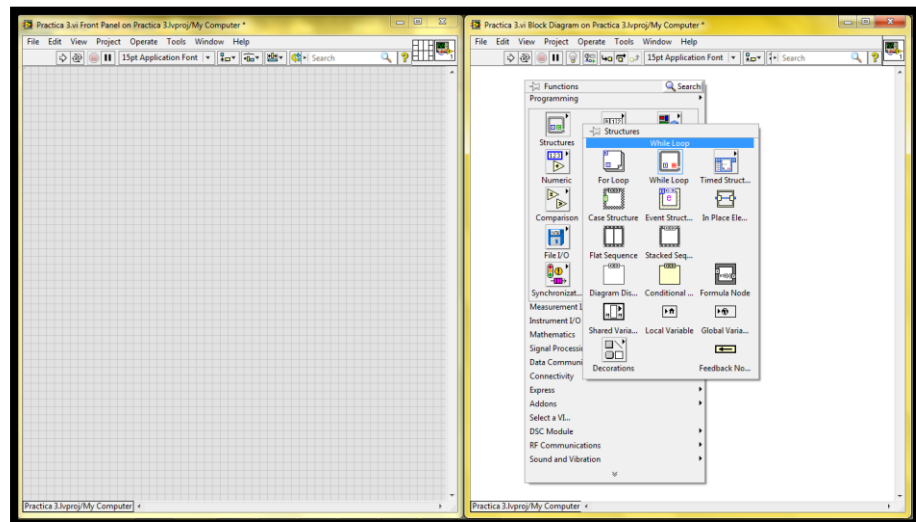


Figura 7.50 Creación de Lazo While

Una vez creado el While Loop creamos un control para el Stop y colocamos un temporizador el cual hará que todo el proceso dentro del While Loop se repita cada cierto tiempo, este tiempo para nuestro caso será de 100ms. Este temporizador los encontramos en la ruta que se muestra en la Figura 7.51.



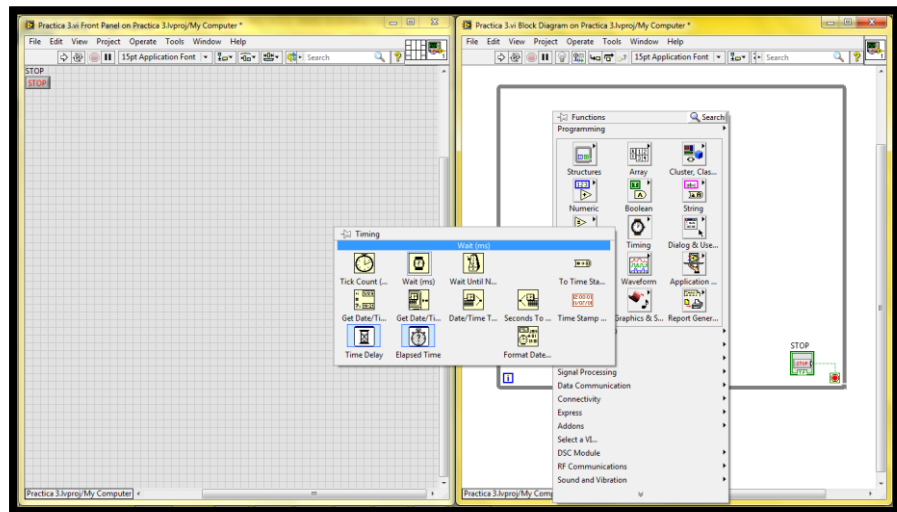


Figura 7.51 Tiempo de ejecución cíclica del Lazo While

Ahora creamos 3 pulsantes que van a representar las entradas digitales de nuestro programa, estas son: Run Bomba, Stop Bomba, Reset Alarma

Esto lo hacemos en el Front Panel siguiendo la ruta que se muestra en la Figura 7.52.

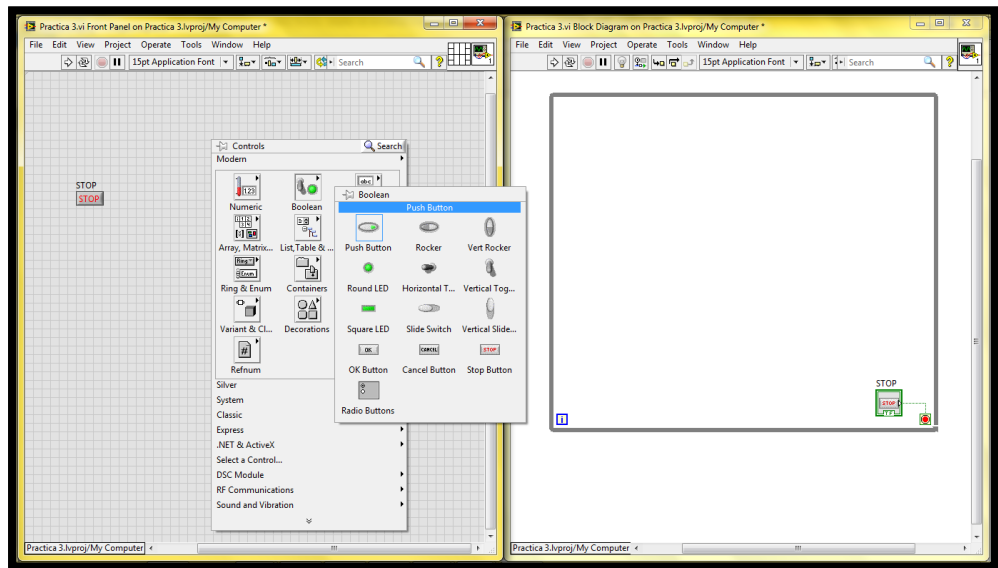


Figura 7.52 Creación de las entradas digitales.

Observamos que tenemos lo que tenemos en el Front Panel y en Block Diagram, en el primero vamos a observar la parte estética del programa mientras que en el Block Diagram la programación como se ve en la Figura 7.53.

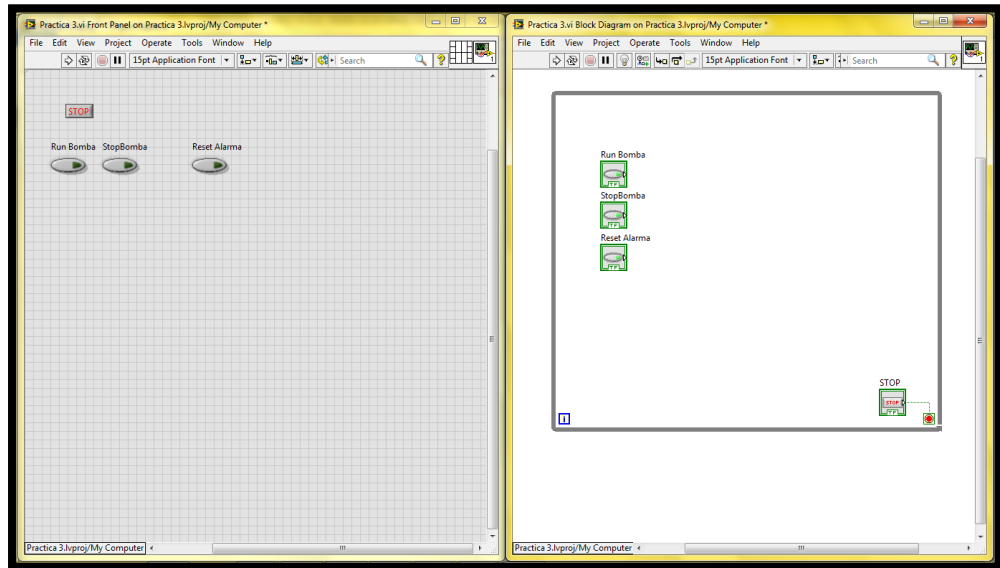


Figura 7.53 Entradas digitales creadas

Ahora creamos indicadores que van a representar las salidas digitales de nuestro programa, estas son: Bomba Encendida, Nivel Mayor a 20, Alarma de Nivel

Esto lo hacemos en el Front Panel siguiendo la ruta que se muestra en la Figura 7.54.

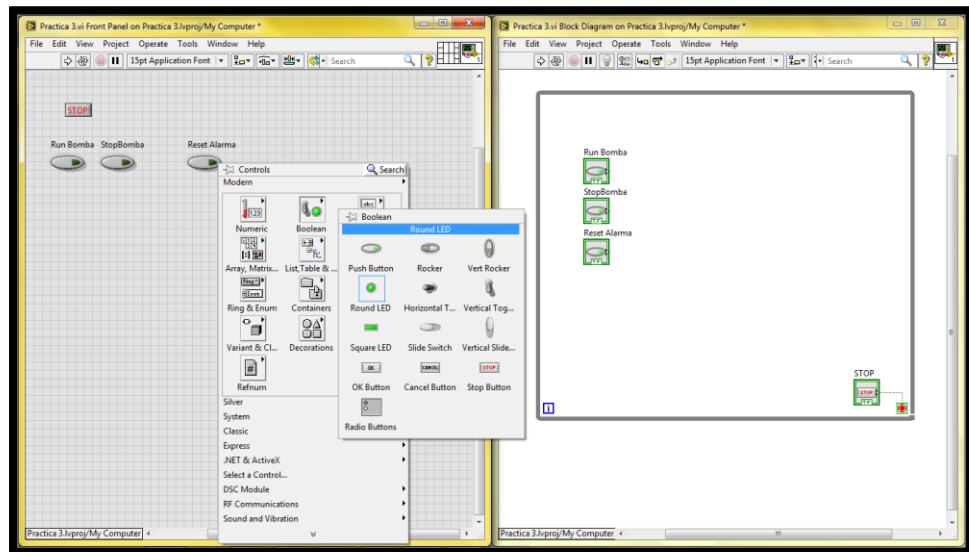


Figura 7.54 Creación de Salidas Digitales

Quedando como se muestra en la Figura 7.55

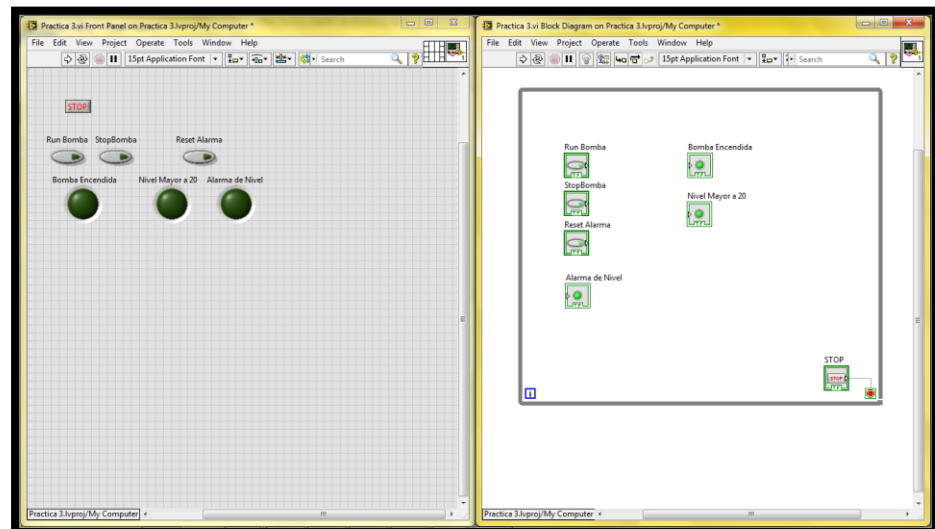


Figura 7.55 Salidas Digitales creadas.

También procedemos a agregar una imagen de una bomba para darle más estética al programa, para hacerlo seguimos la ruta de la Figura 7.56

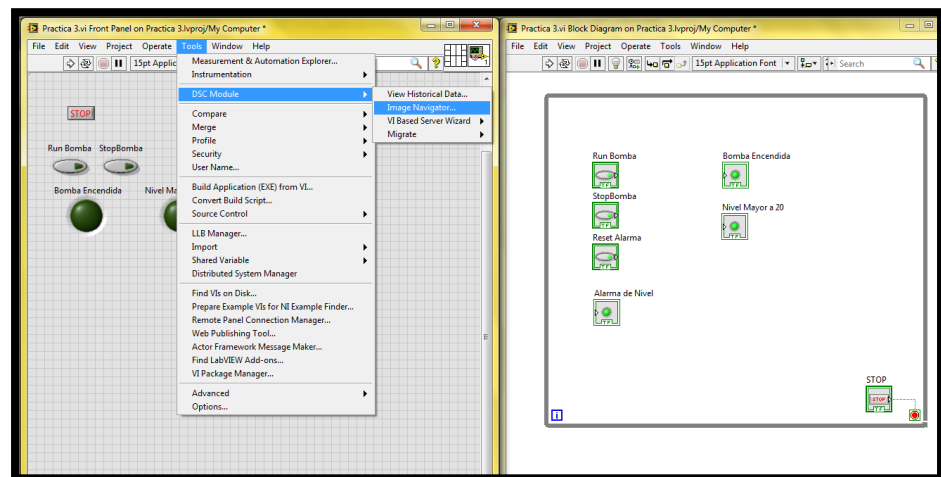


Figura 7.56 Ingreso al Image Navigator

En el DSC Module tenemos un sin fin de imágenes de elementos de la industria los cuales nos pueden servir para representar casi cualquier proceso, Figura 7.57



Figura 7.57 Representación de la bomba

**Variables Analógicas:** Ahora creamos 1 slider que representará el sensor de nivel que va a variar entre 0 y 45cm, un indicador tipo tanque para observar ese nivel, así mismo un indicador que muestre la apertura de la válvula que va de 0 a 100% y cuyo valor será ingresado por el usuario. Estos elementos los encontramos en la ruta descrita en la Figura 7.58.

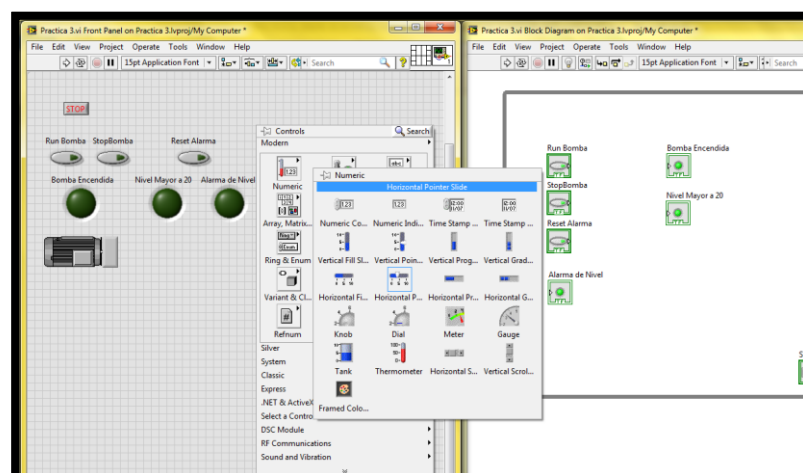


Figura 7.58 Creación de variables analógicas.

Al final tenemos las variables analógicas creadas en la Figura 7.59

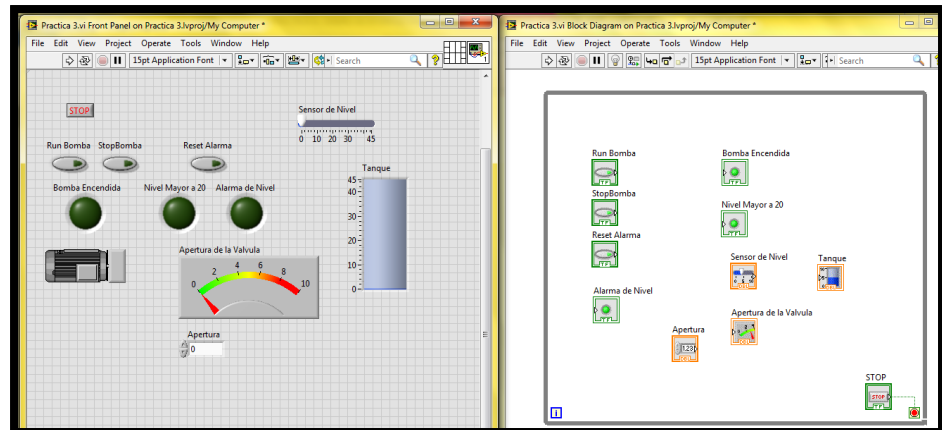


Figura 7.59 Variables analógicas creadas.

Una vez teniendo todas las variables procedemos con la lógica de programación, esta es muy sencilla ya que por el momento solo queremos representar como serían si las enviáramos desde el PLC. En prácticas posteriores haremos eso. La lógica es la misma con la que elaboramos el programa en la practica 2.

Se activa la señal de alarma cuando el nivel del tanque pasa los 40cm. Esta señal apaga la bomba. Se puede resetear esa señal de alarma por medio del botón Reset Alarma y hacerla falsa pero para esto el nivel del tanque tiene que estar por debajo de los 20cm. Una vez cumplidas podemos encender nuevamente la

bomba con Run Bomba. También la podemos apagar pulsando Stop Bomba, toda esta lógica la podemos observar en la Figura 7.60.

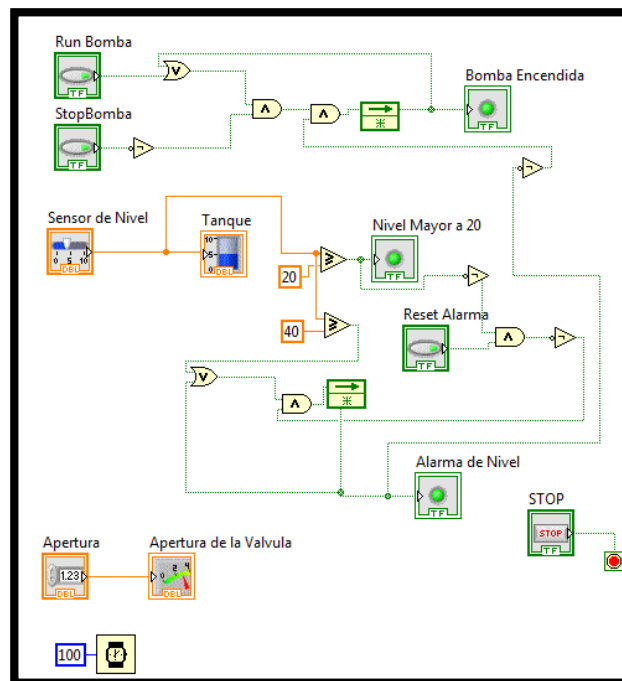


Figura 7.60 Lógica de arranque y apagado de bomba, alarmas y reseteo

### 7.3.2 VISUALIZACIÓN DE ALARMAS

Para visualizar una alarma mostraremos un texto que aparezca cuando el indicador Alarma de Nivel se encienda, para hacer esto tenemos que cambiar las propiedades del texto para que mediante un control que en este caso es la alarma aparezca y desaparezca



de manera intermitente. Tenemos que hacer lo siguiente, primero creamos un String o texto de la siguiente manera en Figura 7.61.

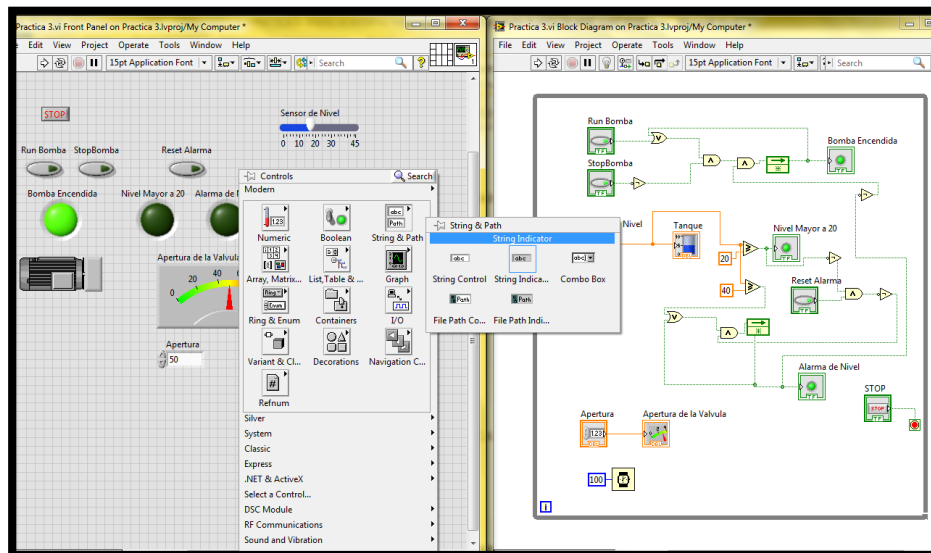


Figura 7.61 Creación de string para indicar alarma

Una vez creado el texto o string en el Block Diagram damos clic sobre su icono y escogemos Create- Property Node-Visible para que solo aparezca cuando nosotros la activemos y después clic derecho y la ruta Create-Property Node-Blinking para que parpadee. Después de haberlo hecho aparecen los siguientes iconos, Figura 7.62.

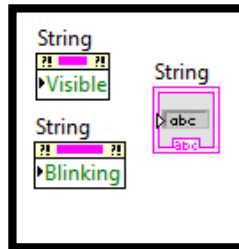


Figura 7.62 Atributos del string.

Estos iconos están asociados al texto y ahora lo único que tenemos que hacer es ponerles como entrada la variable con la cual queremos activar estas propiedades que en nuestro caso es Alarma de Nivel.

### 7.3.3 EDICIÓN DE TEXTOS EXPLICATIVOS SOBRE EL PROCESO

La edición de textos explicativos es bastante similar a lo que hicimos para la acción en estado de alarma donde mostramos un texto que aparece solo cuando esa condición se cumple. En este caso la variable que va a activar la propiedad de visibilidad del texto de ayuda va a ser un pulsante. Creamos entonces un pulsante que al presionarlo despliegue el texto indicando para qué sirve el tanque y otro para el indicador de alarma. En la Figura 7.63 se indica su funcionamiento y programación del texto de ayuda para el indicador de nivel del Tanque.

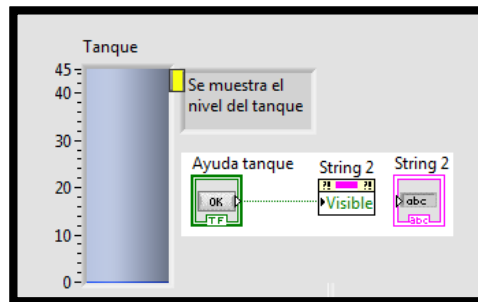


Figura 7.63 Funcionamiento y programación de textos explicativos

### TAREA:

- Realice un diagrama de flujo que muestre el funcionamiento del programa.
- Realice prints de pantalla del funcionamiento del programa.
- Modifique el programa y muestre en una gráfica en el Labview la variación de la señal de nivel en función del tiempo.
- Explique cómo cambiar el funcionamiento de los pulsantes Run Bomba y Stop Bomba para que funcionen como selectores.

- Agregue mensajes de ayuda a los demás pulsantes, al indicador de bomba encendida y al de apertura de válvula.

#### **7.4 PRÁCTICA # 4: CONFIGURACIÓN CLIENTE/SERVIDOR OPC**

##### **Objetivos:**

- Familiarizar al estudiante con el manejo del programa NI OPC Servers.
- Aprender a configurar el Servidor OPC en NI OPC Servers.
- Aprender a configurar el Cliente OPC en Labview.

**Introducción:** Este software proporciona una manera sencilla de comunicarse entre dispositivos de diferentes fabricantes sin la necesidad de utilizar el protocolo propietario de los mismos. El servidor proporciona una amplia gama de controladores de dispositivos enchufables y componentes que se adaptan a la mayoría de las necesidades de comunicación.

**Desarrollo:** Para esta práctica vamos a comunicar el PLC S71200 de Siemens con el software Labview de National Instrument mediante el NI OPC Servers. Vamos a utilizar los dos programas ya creados. El programa en TIA Portal practica 2 y programa de Labview practica 3.

#### 7.4.1 CONFIGURACIÓN DE CLIENTE OPC

En la configuración del cliente trabajaremos ya en el programa Labview. Abrimos el proyecto practica 3 y los guardamos como practica 4 dentro de la misma carpeta que guardamos el archivo del programa NI OPC Servers. Luego damos clic derecho sobre My computer y seguimos la siguiente dirección en la Figura 7.64.

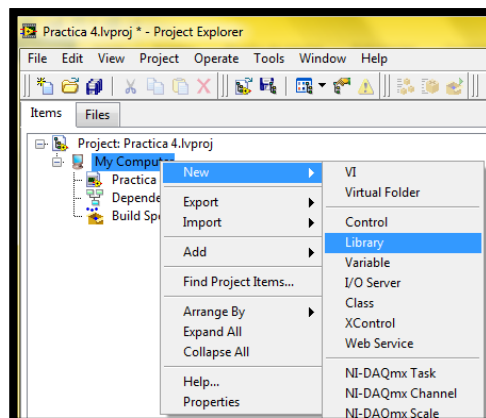


Figura 7.64 Ruta para creación de librería

Procedemos a crear el I/O server como se muestra en las Figuras 7.65, 7.66, 7.67.

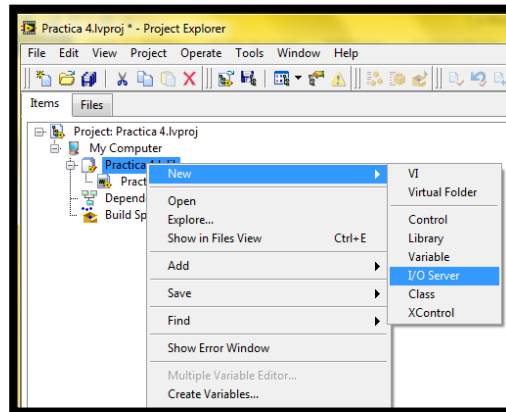


Figura 7.65 Ruta para creación de I/O Server

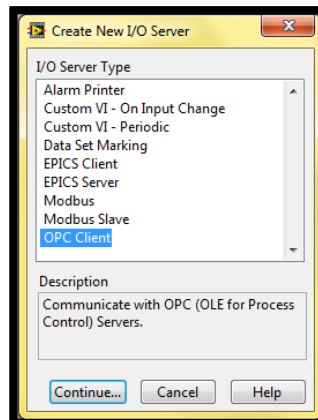


Figura 7.66 Selección de tipo de servidor

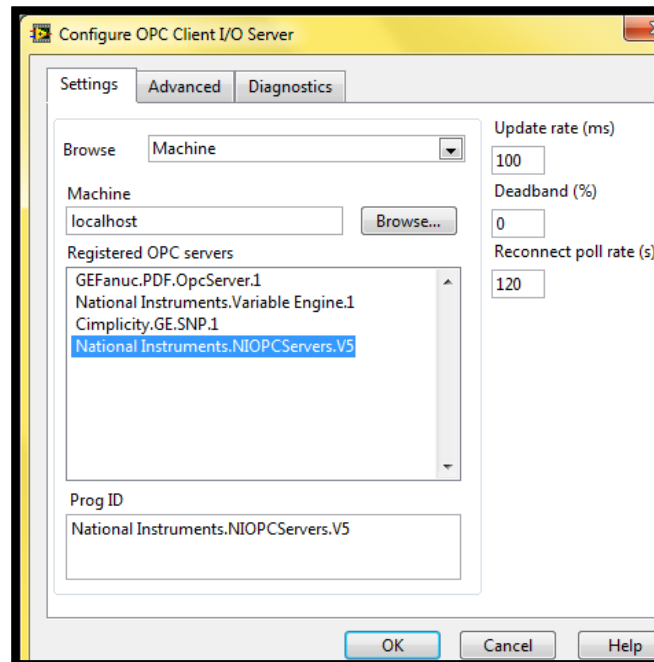


Figura 7.67 Selección del servidor OPC con el que vamos a trabajar y cambio de la tasa de actualización

Cambiamos el valor de la tasa de actualización de valores equiparando al valor de actualización de las variables que creamos en la configuración del servidor OPC. Luego de crear el I/O server creamos las variables ligadas o bound variables con la ruta de las Figuras 7.68, 7.69.

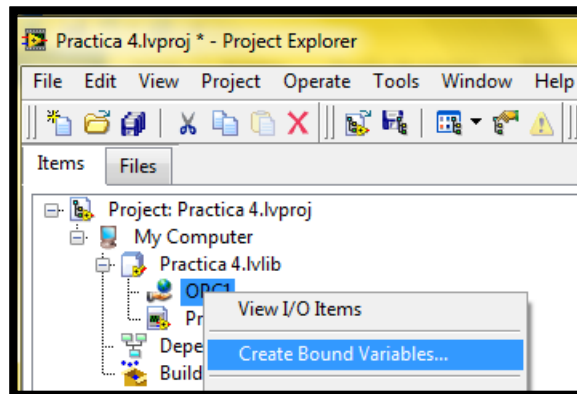


Figura 7.68 Ruta para creación de variables ligadas.

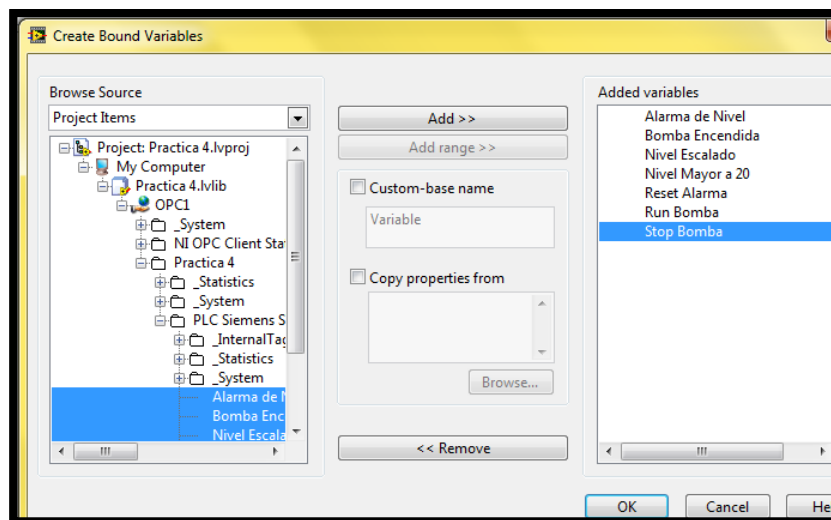


Figura 7.69 Selección de variables ligadas

Ahora que ya tenemos las variables ligadas creadas, abrimos el archivo practica 4.vi que es el de la practica 3 pero cambiado el nombre. Una vez abierto arrastramos las variables al Block Diagram del vi y colocamos las variables donde queremos



compartirlas. Borrarnos toda la lógica de programación de la interfaz de Labview y dejamos que la lógica que radica en el programa de TIA Portal controle todo el sistema. En la Figura 7.70 observamos el Block Diagram antes del cambio.

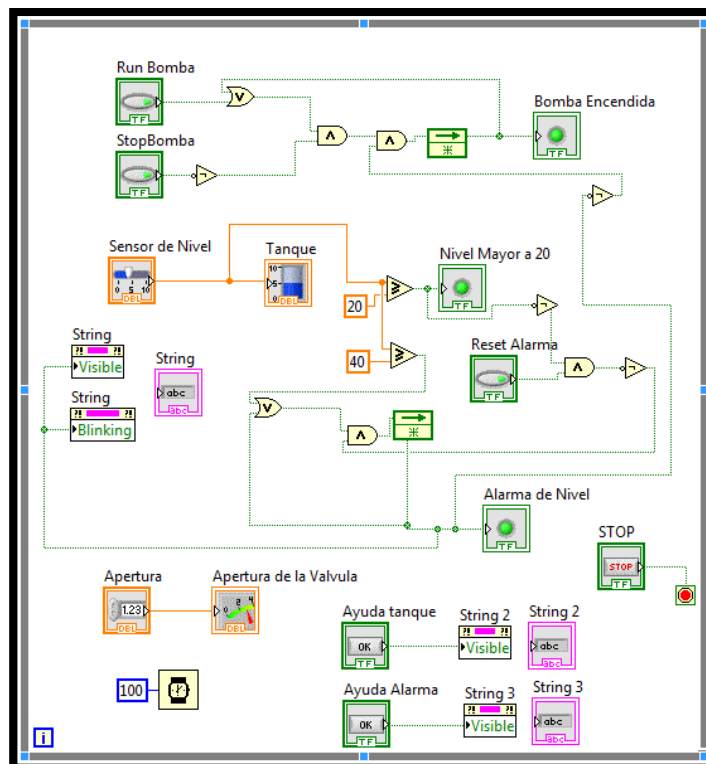


Figura 7.70 Block Diagram sin variables ligadas

Vamos a borrar toda la lógica de programación y a colocar las variables compartidas. El resultado final lo observamos en la Figura 7.71.

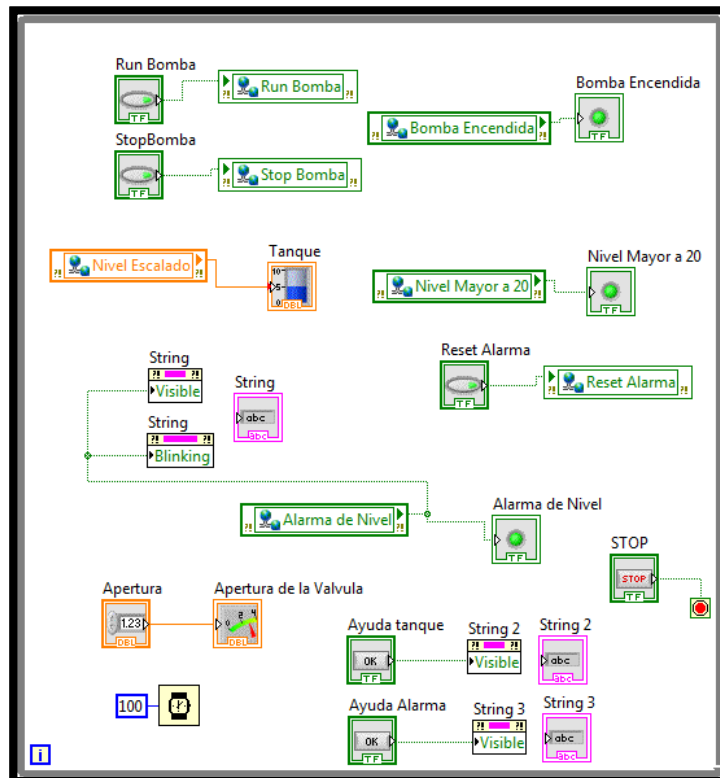


Figura 7.71 Block Diagram con variables ligadas

El único cambio que se hizo en el Front Panel es que ya no tenemos el slider para controlar el nivel del tanque ya que ese valor ahora viene del sensor que está conectado al PLC.

### 7.4.2 CONFIGURACIÓN DE SERVIDOR OPC

Para configurar el servidor OPC primero definimos las variables que vamos a compartir desde TIA Portal, Tabla 13.

VARIABLE	DIRECCION	TIPO DE DATO EN NI OPC SERVERS	TIPO DE DATO EN TIA PORTAL
Nivel Escalado	MD4	FLOAT	REAL
Run Bomba	I0.0	BOOLEAN	BOOL
Stop Bomba	I0.1	BOOLEAN	BOOL
Reset Alarma	I0.2	BOOLEAN	BOOL
Bomba Encendida	Q0.0	BOOLEAN	BOOL
Nivel Mayor a 20	Q0.1	BOOLEAN	BOOL
Alarma de Nivel	Q0.2	BOOLEAN	BOOL

Tabla 13 Tipo de Datos en NI OPC Servers y TIA PORTAL

Primero abrimos el programa NI OPC Servers y guardamos el archivo con el nombre Practica 4, también hay que guardar una copia del programa de TIA Portal y del proyecto de Labview en la misma carpeta. Todos los archivos para conveniencia tienen que estar en la misma carpeta.

Luego en NI OPC Servers damos clic en **Clic to add a channel** y comenzamos a crear el canal de comunicación. Seguimos los pasos descritos en las Figuras: 7.72, 7.73, 7.74, 7.75, 7.76, 7.77

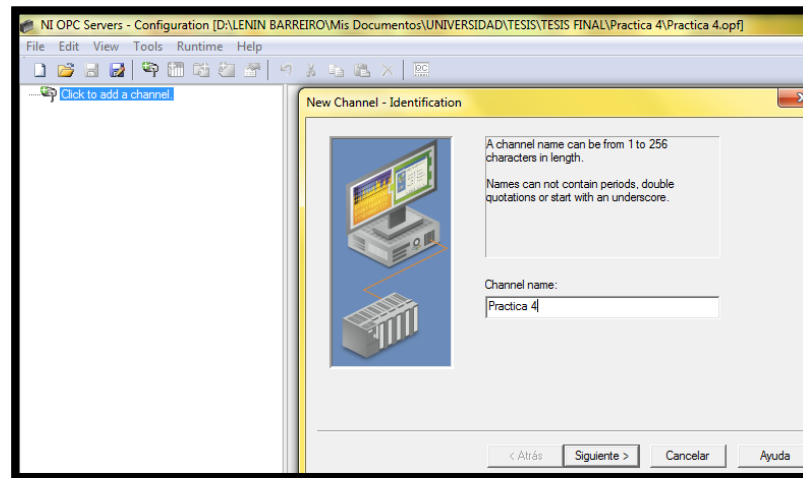


Figura 7.72 Creación del canal de comunicación

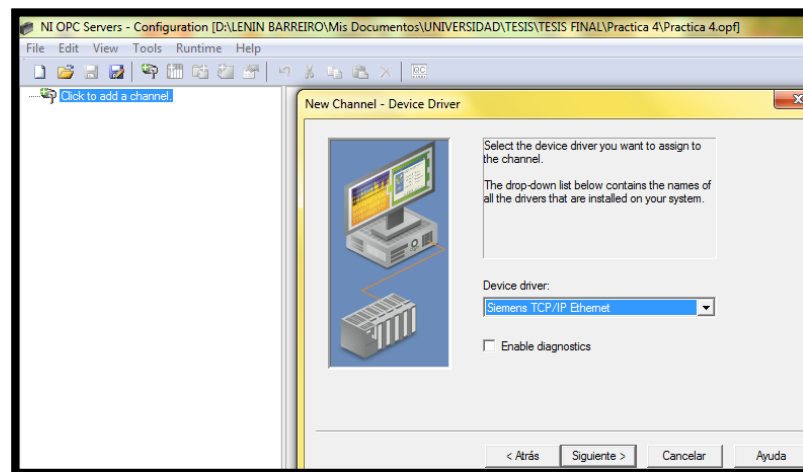


Figura 7.73 Selección del protocolo de comunicación

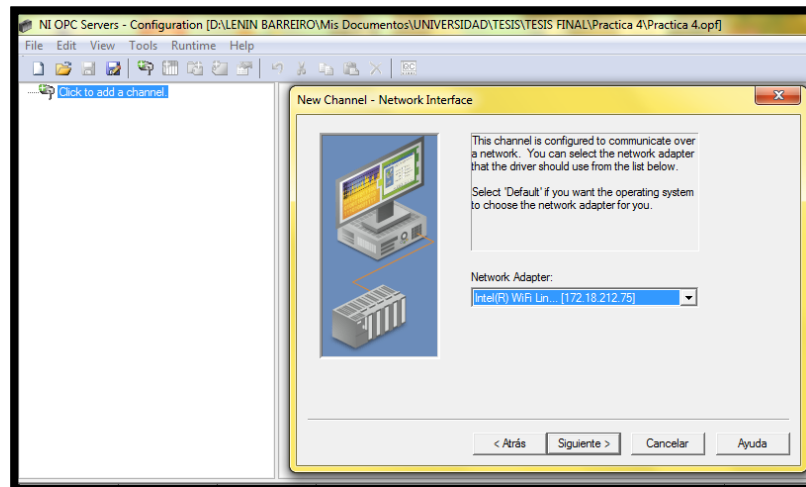


Figura 7.74 Selección del adaptador de red de la computadora con la que se trabaja

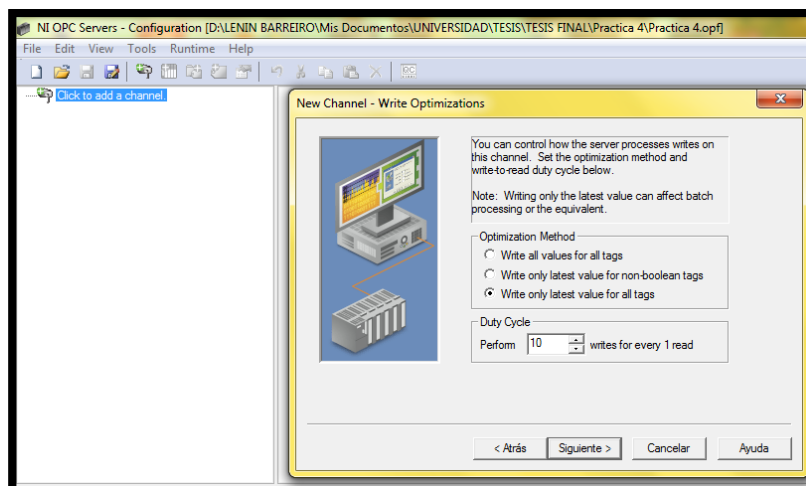


Figura 7.75. Selección del método de optimización

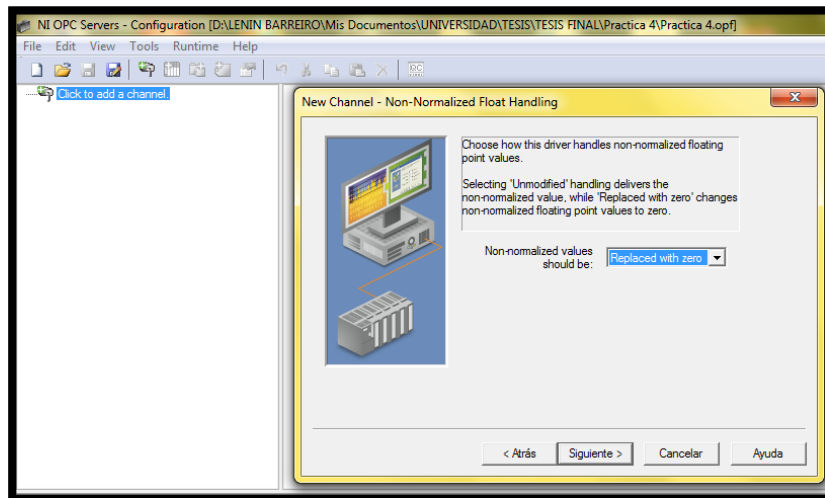


Figura 7.76. Selección de respuesta cuando un valor analógico no este normalizado

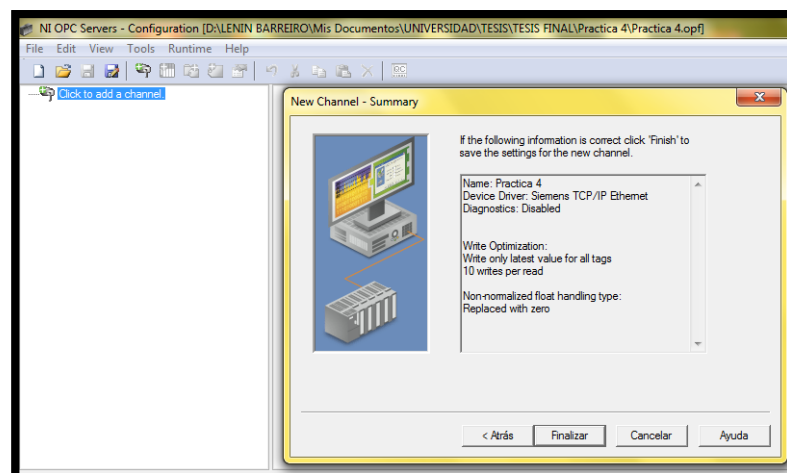


Figura 7.77 Resumen de la configuración del canal

Una vez creado el canal procedemos a seleccionar los parámetros del equipo con el que vamos a trabajar. Seguimos las especificaciones de las Figuras: 7.78, 7.79, 7.80.

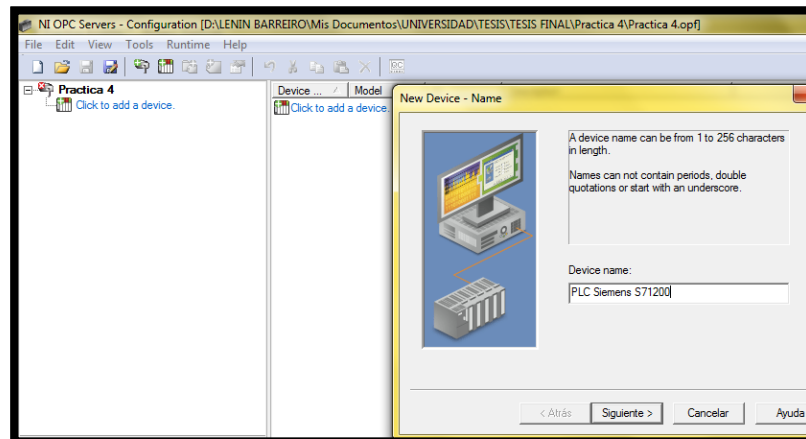


Figura 7.78 Escritura del nombre del dispositivo

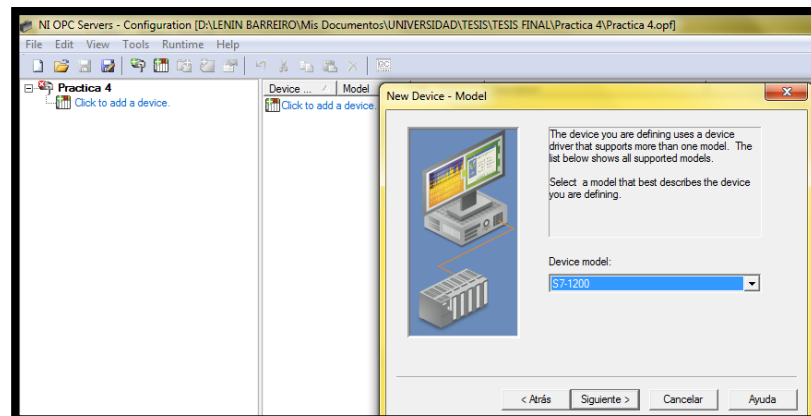


Figura 7.79 Selección del modelo del dispositivo

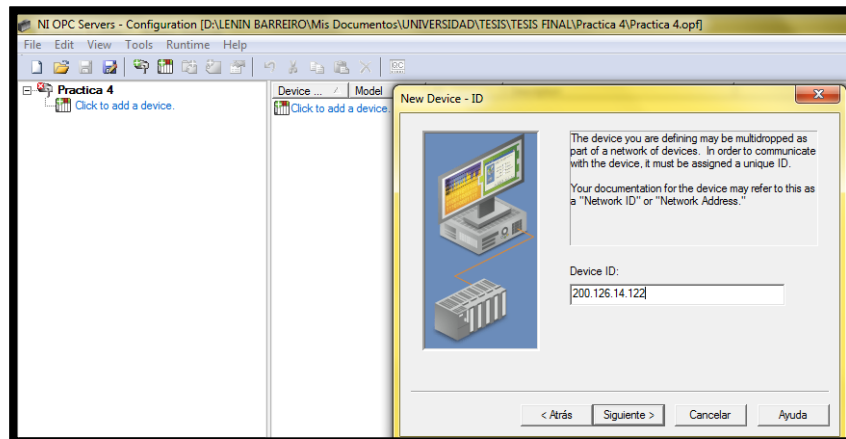


Figura 7.80 Escritura de la dirección ip del dispositivo dentro de la red

Los siguiente parámetros dejarlos por defecto ya que para nuestra aplicación no debemos cambiarlos. Figuras: 7.81, 7.82, 7.83, 7.84, 7.85, 7.86, 7.87.

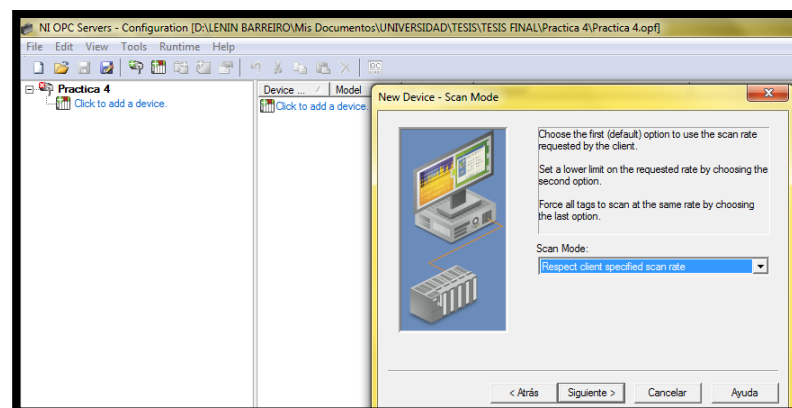


Figura 7.81 Resumen de la configuración del canal



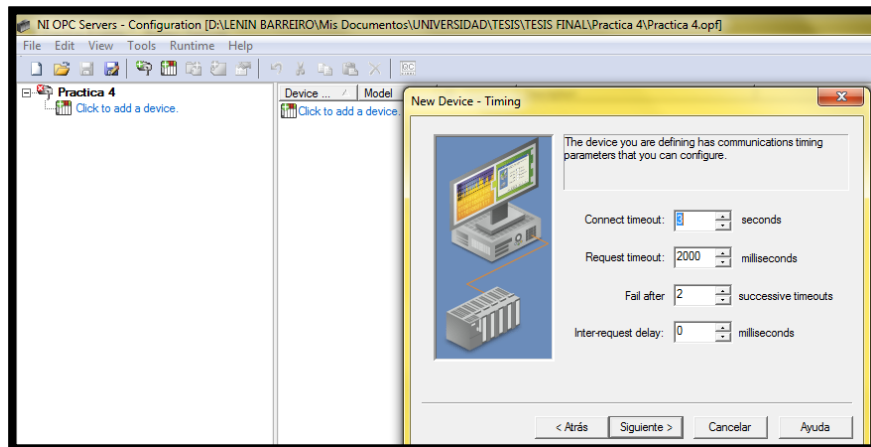


Figura 7.82 Parámetros de sincronización

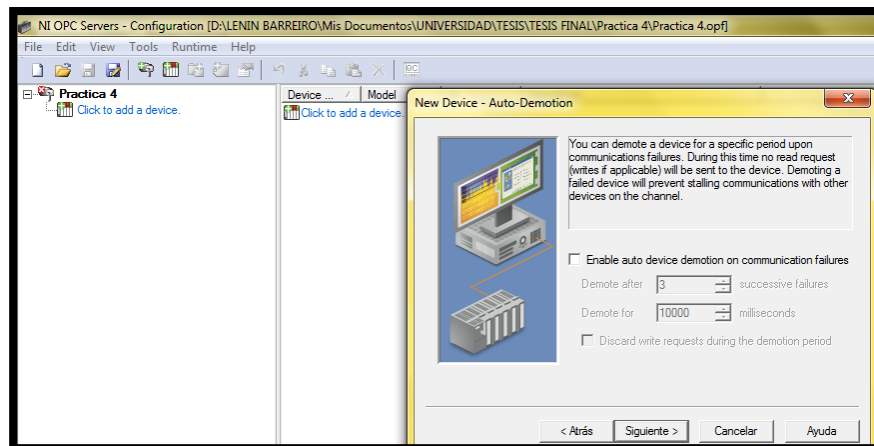


Figura 7.83 Auto degradación del dispositivo cuando hay una falla de comunicación

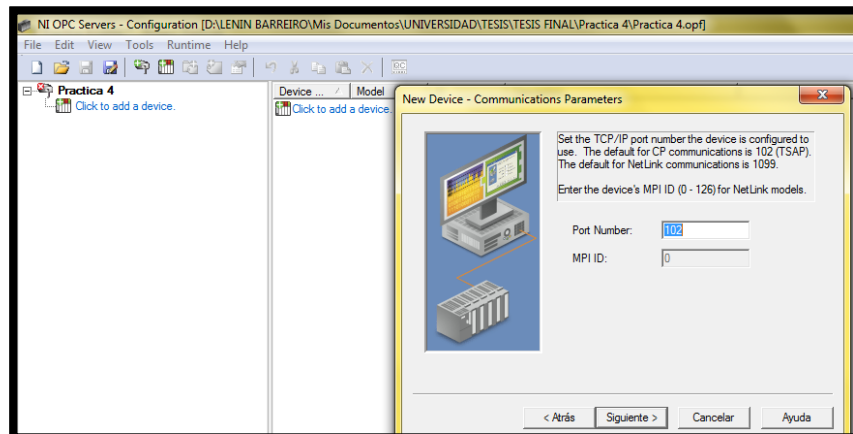


Figura 7.84. Parámetros de comunicación

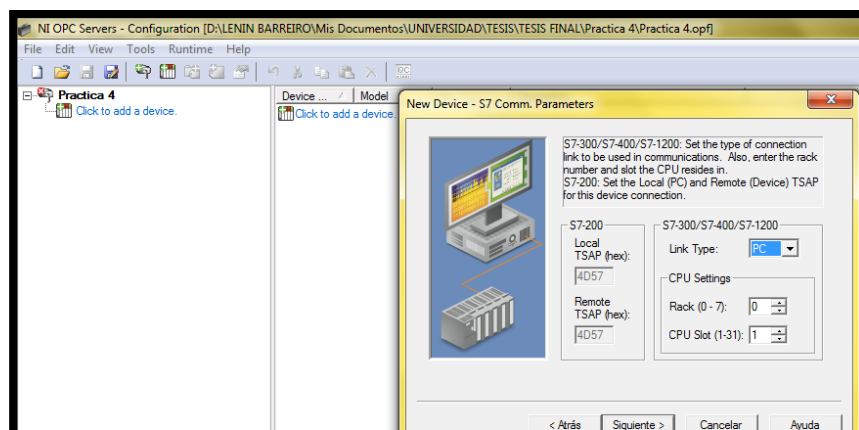


Figura 7.85 Parámetros de puerto de comunicación para S7

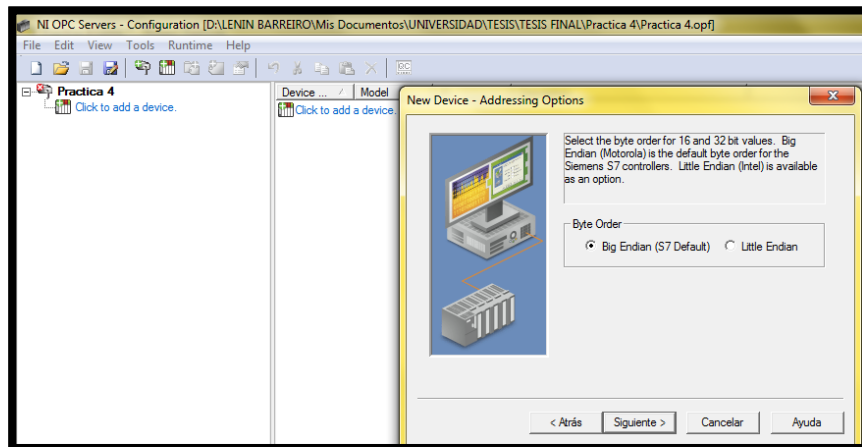


Figura 7.86 Parámetros de ordenamiento

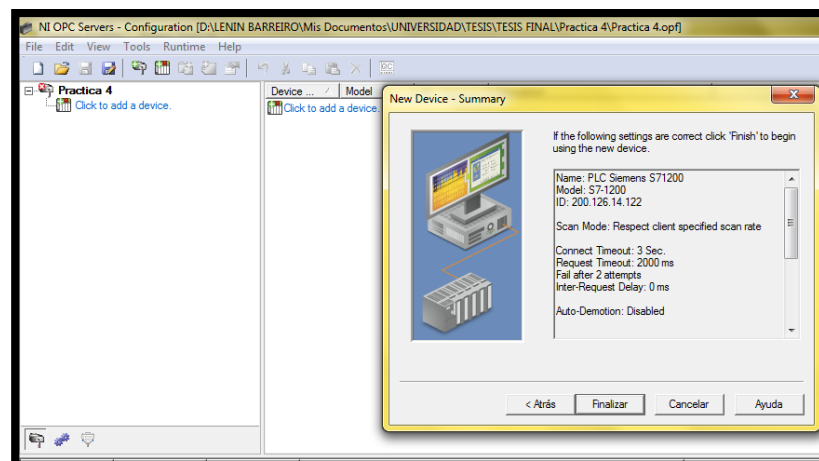


Figura 7.87 Resumen de configuraciones del dispositivo a utilizar

Ya creado el dispositivo seguimos con el último paso en que es la creación de las variables a compartir entre TIA Portal y Labview.

Primero creamos la variable análoga Nivel Escalado figura 7.88

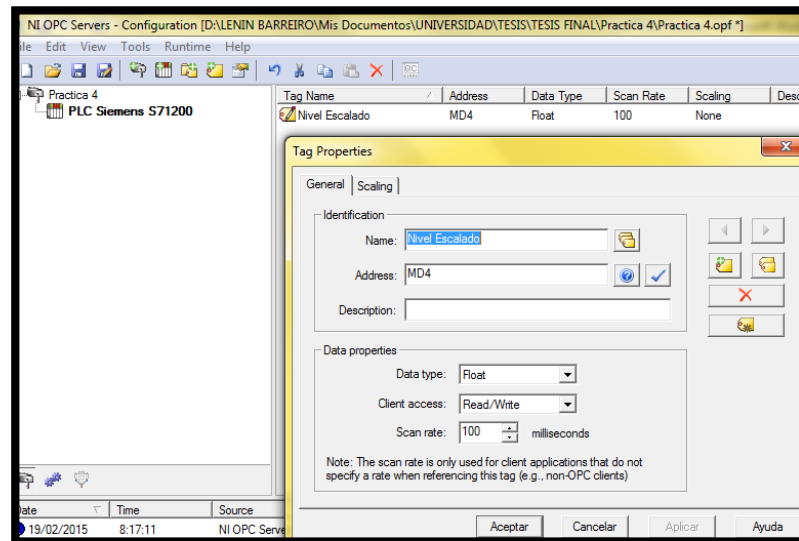


Figura 7.88 Creación de variable analógica Nivel Escalado

Luego de crear la variable analógica procedemos a crear las variables booleanas, comenzamos con Run Bomba. Figura 7.89

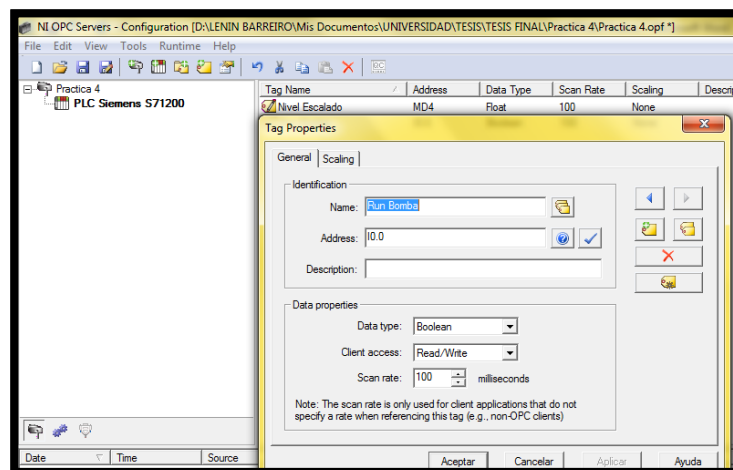
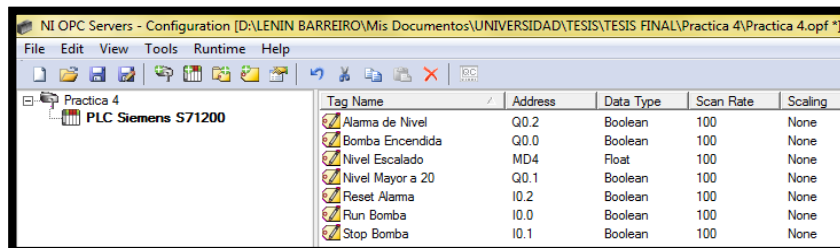


Figura 7.89 Creación de variable booleana Run Bomba.

Finalmente tenemos creadas todas las variables a compartir entre los dos programas. Figura 7.90.



Tag Name	Address	Data Type	Scan Rate	Scaling
Alarma de Nivel	Q0.2	Boolean	100	None
Bomba Encendida	Q0.0	Boolean	100	None
Nivel Escalado	MD4	Float	100	None
Nivel Mayor a 20	Q0.1	Boolean	100	None
Reset Alarma	I0.2	Boolean	100	None
Run Bomba	I0.0	Boolean	100	None
Stop Bomba	I0.1	Boolean	100	None

Figura 7.90 Variables creadas en NI OPC Servers

### TAREA:

- Realice prints del funcionamiento del programa.
- Añada a la interfaz de Labview un Waveform Chart que muestre la variación del nivel en función del tiempo.
- Muestre una variable booleana en la interfaz de Labview que indique cuando el nivel está entre 25 cm y 30 cm. Esta lógica de esta variable tiene que ser programada en TIA Portal y compartida mediante el NI OPC Servers.

## 7.5 PRÁCTICA # 5: CONTROL CON INTERFAZ PARA USUARIO DE PLANTA DE NIVEL GUNT RT 450

### Objetivos:

- Familiarizarse con el uso del objeto tecnológico PID\_Compact de TIA Portal.
- Manejo del objeto tecnológico PID\_Compact en modo manual.
- Manejo del objeto tecnológico PID\_Compact en modo automático.

**Introducción:** El objeto tecnológico PID\_Compact ofrece un regulador PID continuo con optimización integrada. Es posible elegir entre el modo manual y el automático. El regulador PID registra de forma continuada el valor real dentro de un lazo de regulación y lo compara con la consigna deseada. A partir del error de regulación resultante, la instrucción PID\_Compact calcula un valor de salida, con el que el valor real se iguala con la consigna con la máxima rapidez y estabilidad posibles. En los reguladores PID, el valor de salida se compone de tres acciones que destacamos a continuación [4]:

**Acción P:** La acción P del valor de salida aumenta proporcionalmente al error de regulación [4].

**Acción I:** La acción I del valor de salida aumenta hasta que se compensa el error de regulación [4].

**Acción D:** La acción D aumenta con una velocidad de variación creciente del error de regulación. El valor real se iguala lo más rápidamente posible con la consigna. Si la velocidad de variación del error de regulación vuelve a reducirse, también lo hace la acción D [4].

La instrucción PID\_Compact calcula los parámetros P, I y D para su sistema regulado de forma autónoma durante la optimización inicial. Los parámetros pueden optimizarse aún más a través de una optimización fina. No es necesario determinar los parámetros manualmente.

**Desarrollo:** Para esta práctica utilizaremos los programas creados en la práctica 4. Abrimos el programa practica 4 de TIA Portal. En el árbol del proyecto seguimos la ruta de la Figura 7.91 y creamos un bloque de organización Cyclic Interrupt. Este bloque se ejecuta con un periodo que depende del tiempo de ciclo. Dentro de este bloque vamos a crear el

PID\_Compact ya que queremos que se ejecute la toma de datos de manera cíclica y que a su vez cada cierto tiempo se actualicen los valores de entrada y salida del bloque.

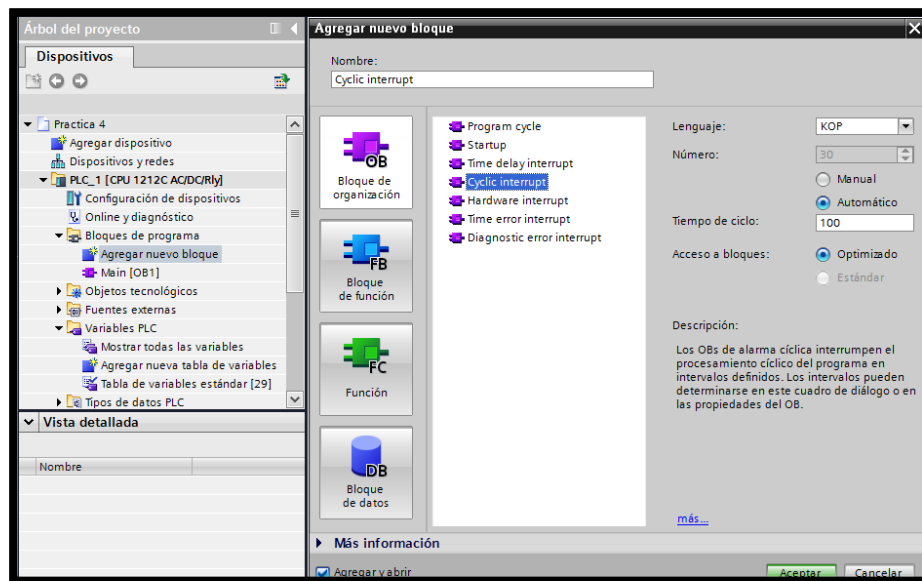


Figura 7.91 Ruta para creación del Cyclic Interrupt

Luego en el árbol del proyecto seguimos la ruta de la Figura 7.92 y creamos un objeto, el PID\_Compact.



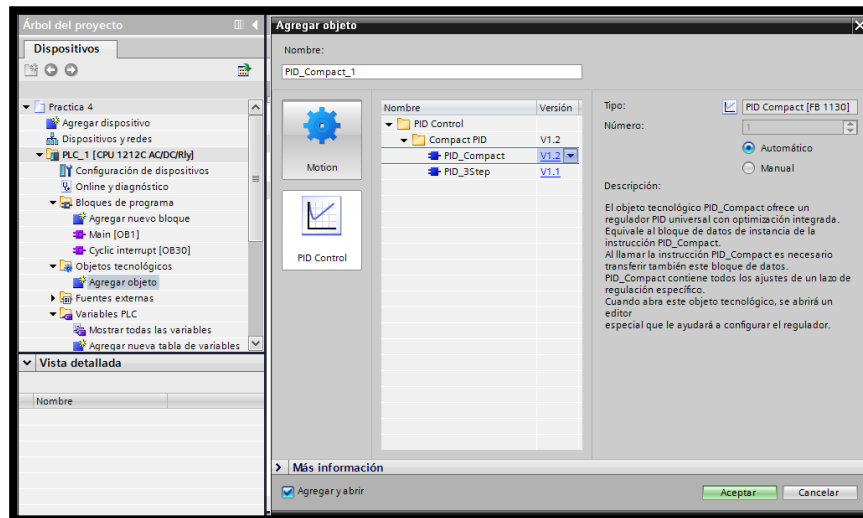


Figura 7.92 Ruta para creación del PID\_Compact.

Una vez creado el objeto tecnológico se abre una ventana donde vamos a configurar todos los ajustes del bloque, Figuras 7.93 y 7.94.

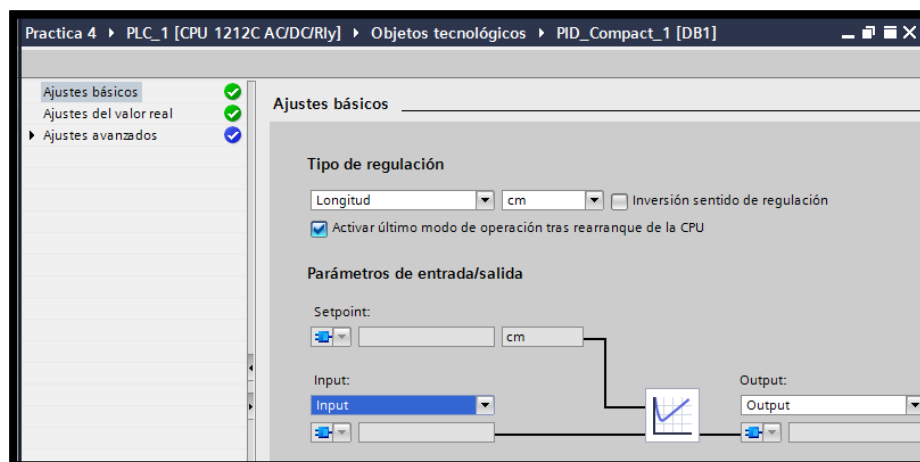


Figura 7.93 Ajustes básicos del PID\_Compact

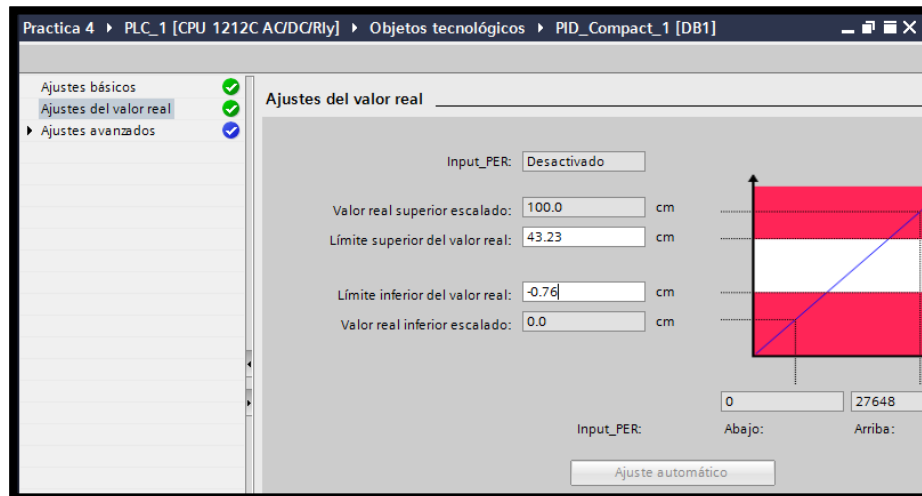


Figura 7.94 Ajustes del valor real del PID\_Compact

Los valores que colocamos en los ajustes del valor real son los que se encontraron haciendo las pruebas en la planta y coinciden con los valores en el bloque de escalado de la variable análoga de nivel del tanque.

Ya con los parámetros puestos abrimos el Cyclic Interrupt y del lado de las instrucciones Figura 7.95 arrastramos el bloque PID\_Compact hacia el segmento 1. En la ventana que aparece al soltar el bloque seleccionamos el DB que creamos anteriormente donde tenemos los parámetros que configuramos para el PID\_Compact. Esto lo observamos en la figura 7.96.

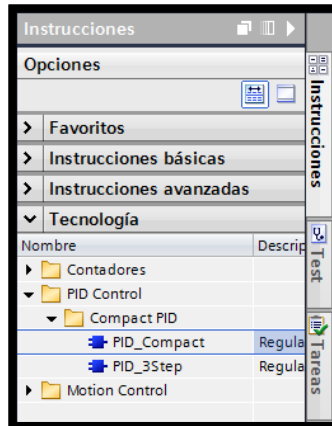


Figura 7.95 Dirección del PID\_Compact.

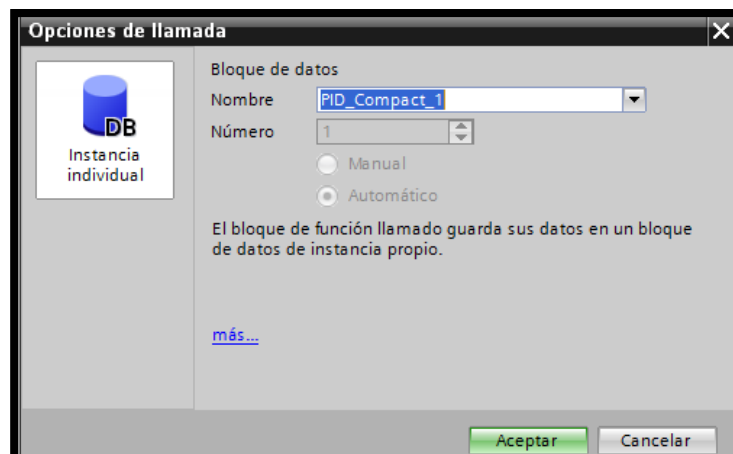


Figura 7.96 Selección del DB o bloque de datos

Tenemos que hacer varias modificaciones en nuestros programas para trabajar con los 2 modos de funcionamiento que tiene el PID\_Compact para controlar una variable, manual y automático. Vamos a trabajar en los dos. La salida análoga del Bloque PID\_Compact que controlará la válvula para nuestro caso y como la configuramos dará un valor de 0 a

100. Ese valor no podemos enviarlo directamente a la válvula, para ello tenemos que pasar dicha variable por los procesos de TIA Portal NORM\_X y SCALE\_X en ese orden.

Procedemos a crear las variables necesarias y editar el segmento 5 de programación donde enviábamos una apertura constante a la válvula.

Figura 7.97.

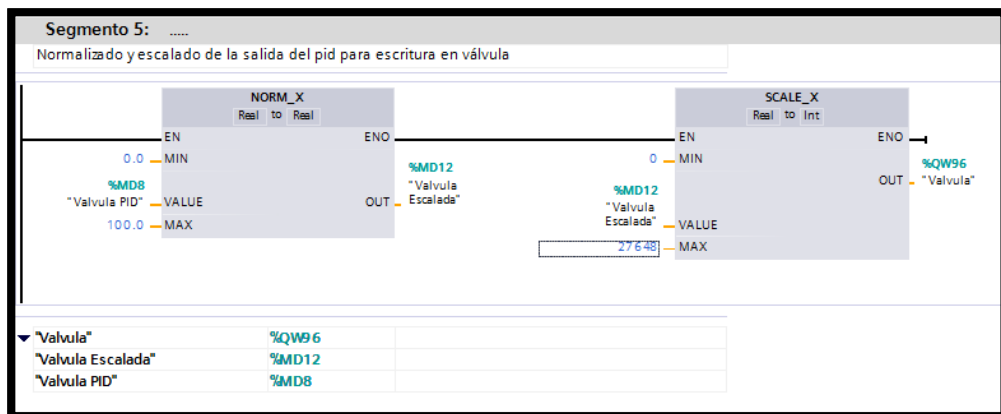


Figura 7.97 Normalizado y escalado de salida de PID

En el bloque PID\_Compact del Cyclic Interrupt también tenemos una salida boolean llamada InputWarnig\_H que es verdadera cuando se sobrepasa el límite máximo para la variable de entrada que en nuestro caso es el nivel. Eso nos servirá para activar la alarma de nivel por medio

de esa salida. Ya no con la lógica de comparadores que teníamos. Figura 7.98.

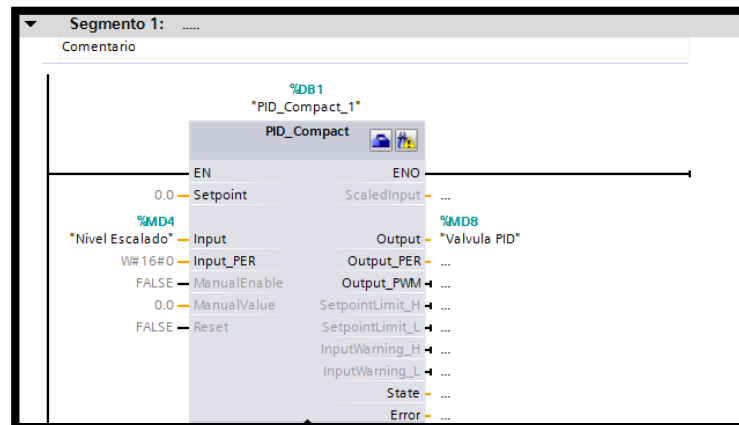


Figura 7.98 Bloque PID\_Compact en el Cyclic Interrupt

Editamos el segmento 3 donde está la lógica de la alarma de nivel quedando de la siguiente manera. Figura 7.99.

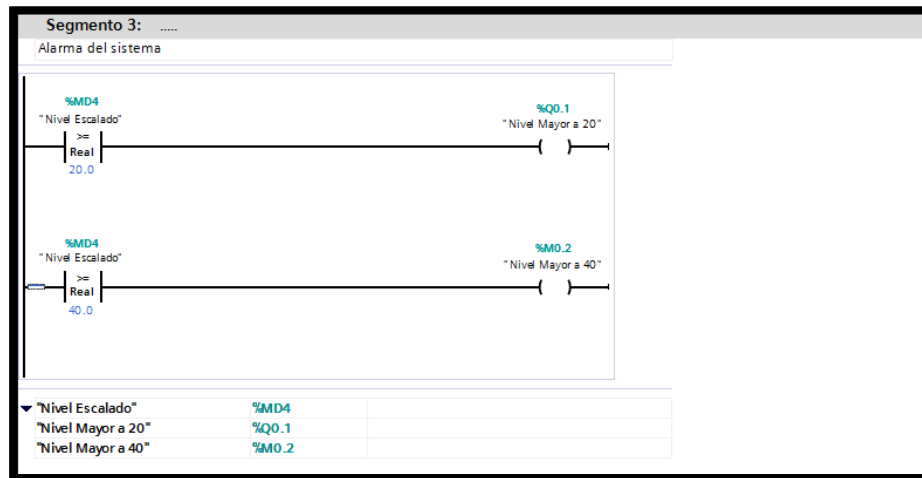


Figura 7.99 Segmento 3 del bloque Main editado.

Ahora la salida alarma de nivel que borramos del segmento 3 del Main la colocamos en la salida InputWarnig\_H del bloque PID\_Compact del cyclic Interrupt. Figura 7.100

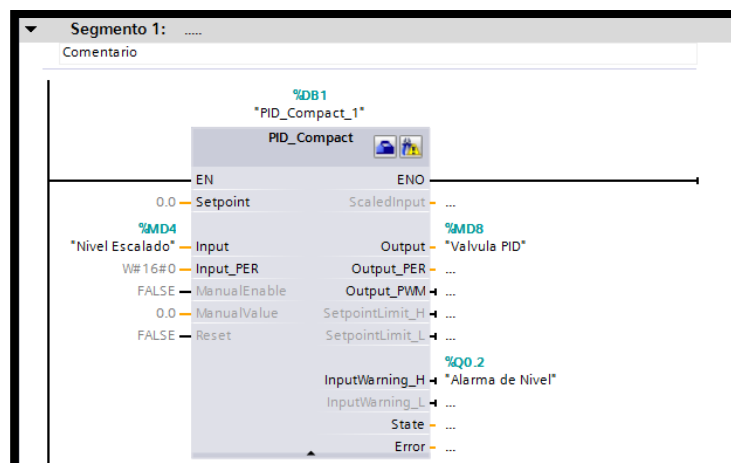


Figura 7.100 Bloque PID\_Compact con salida Alarma de Nivel

Creamos también una variable para ingresar el setpoint desde Labview de esta manera el setpoint no será fijo. La variable creada es Setpoint Labview %MD16 y la asignamos a la entrada Setpoint del Bloque PID\_Compact.

### 7.5.1 CONTROL MANUAL

En modo manual nosotros controlaremos directamente la apertura y cierre de la válvula. Para esto creamos dos variables una boolean que activa o desactiva el modo manual y una variable real para ingresar el valor de apertura desde la Interfaz de Labview. Estas variables son Manual % M0.4 y Apertura Manual %MD20. Tenemos entonces ya todas las variables que necesitamos asignadas en el bloque PID\_Compact. Figura 7.101.

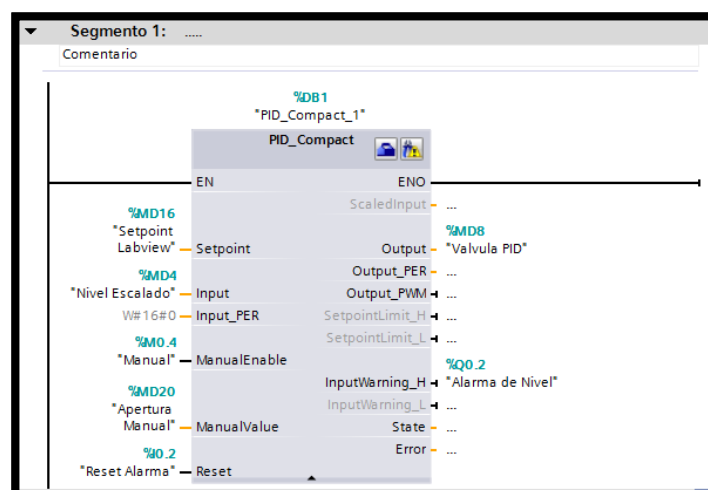


Figura 7.101 Bloque PID\_Compact con todas las variables

Para que el bloque PID\_Compact funcione tenemos que hacer la puesta en marcha del mismo. Dando clic sobre el icono de la parte superior derecha del bloque aparece la ventana de la figura 7.102 en la cual tenemos un botón para comenzar la medición y uno para la optimización. En optimización tenemos 2 opciones, optimización inicial y optimización fina. Los dos métodos son valederos ya que por pruebas realizadas no se nota mucha diferencia entre los 2 métodos.

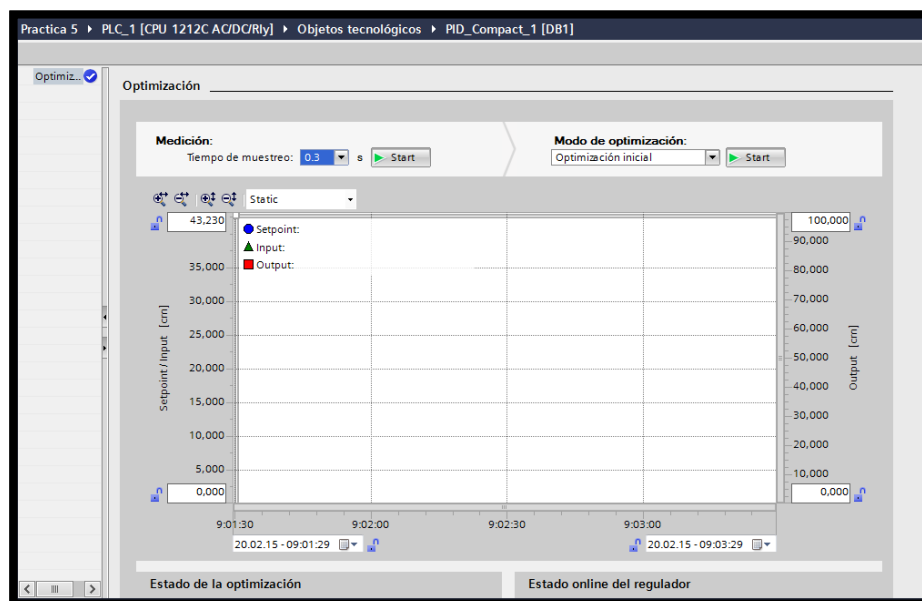
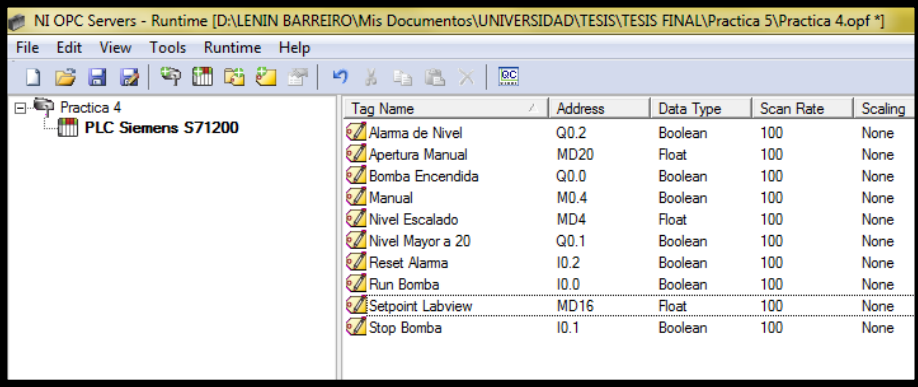


Figura 7.102 Puesta en marcha del bloque PID\_Compact

Damos clic entonces en medición con un tiempo de muestreo de 0.3s y también lo hacemos en modo de optimización. Tenga en



cuenta que para hacer la optimización tiene que haber un valor en la entrada setpoint del bloque. Hágalo con 25cm. Una vez finalizada la optimización en la parte inferior derecha de la ventana aparece el mensaje proceso completado y cargar parámetros PID. Damos clic en esta opción y ya tenemos los parámetros PID para nuestro sistema. Las variables que creamos y que vamos a utilizar en la Interfaz de Labview tenemos que compartirlas por medio del NI OPC Servers. Figura 7.103.



Tag Name	Address	Data Type	Scan Rate	Scaling
Alarma de Nivel	Q0.2	Boolean	100	None
Apertura Manual	MD20	Float	100	None
Bomba Encendida	Q0.0	Boolean	100	None
Manual	M0.4	Boolean	100	None
Nivel Escalado	MD4	Float	100	None
Nivel Mayor a 20	Q0.1	Boolean	100	None
Reset Alarma	I0.2	Boolean	100	None
Run Bomba	I0.0	Boolean	100	None
Setpoint Labview	MD16	Float	100	None
Stop Bomba	I0.1	Boolean	100	None

Figura 7.103 Variables creadas en NI OPC Servers

Creamos las variables ligadas en Labview y las agregamos al VI de la práctica. Una vez hecho este procedimiento tenemos finalmente la estructura creada para el funcionamiento en modo manual y automático. El Block Diagram queda de la siguiente manera. Figura 7.104.

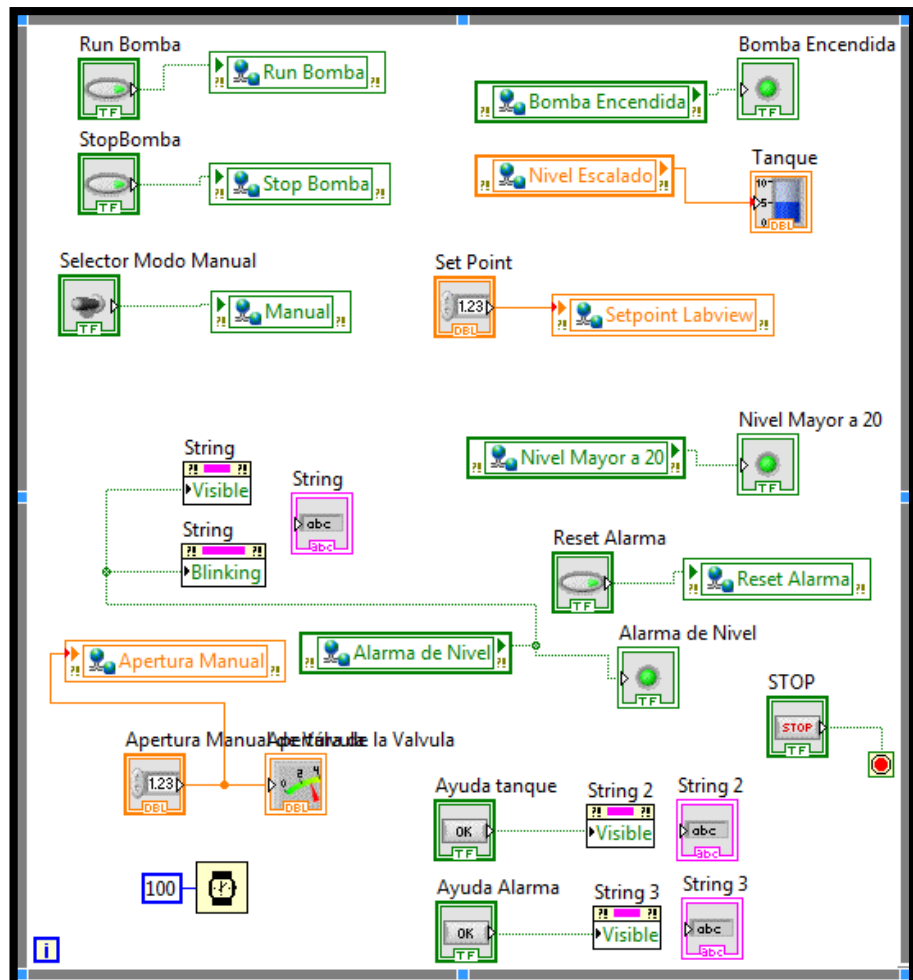


Figura 7.104 Block Diagram con lógica para funcionamiento manual y automático

### 7.5.2 CONTROL PID

En el modo PID (automático) el funcionamiento es diferente. El valor de la salida para apertura de la válvula depende de la regulación que realiza el bloque PID\_Compact. Para que el bloque

trabaje de modo automático el selector Selección de Modo en la Interfaz de Labview debe estar del lado de automático. Con esta variable booleana activada el regulador busca mantener el nivel que está en la entrada setpoint del bloque PID\_Compact.

Se puede pasar de modo automático a modo manual solo moviendo el selector en la Interfaz de Labview.

**TAREA:**

- Haga un print de pantalla de los parámetros PID de bloque PID\_Compact.
- Añada a la interfaz de Labview un Waveform Chart que muestre dos entradas, la variación del nivel en función del tiempo y el valor de setpoint. Realice un print del proceso cuando se varíe de un setpoint de 0cm a un setpoint 25cm y que el valor de nivel quede estable con respecto a este setpoint.
- Con ayuda del print anterior calcule los valores de sobre nivel porcentual y tiempo de estabilización

## **CONCLUSIONES Y RECOMENDACIONES**

### **CONCLUSIONES**

1. El control para Planta de Nivel diseñado, y basado en el bloque PID\_COMPACT de TIA PORTAL, mostró resultados, más que aceptables para diferentes puntos de operación comparado con otras técnicas de sintonización de parámetros PID como Ziegler Nichols, además el algoritmo de autotuning que utiliza dicho bloque, es de ejecución rápida y nos permite ahorrar mucho tiempo en esta tarea comparado con métodos tradicionales.
2. La comunicación establecida entre la Interfaz de Labview y el PLC Simatic S7-1200, por medio de NI OPC Servers, fue estable y nos

permitió realizar las pruebas y puesta en marcha sin mayores inconvenientes, además; de lo analizado durante el desarrollo del proyecto, el NI OPC Servers cuenta con soporte para comunicar a Labview la mayoría de las marcas comerciales disponibles por lo que se concluye que es una herramienta que nos permite superar el problema que surge al querer integrar equipos de diversos protocolos propietarios.

3. La estructura de programación de TIA PORTAL de tipo modular, integrada por bloques de organización (OB), bloque de función (FB), funciones (FC) y bloque de datos (DB), nos permite trabajar en forma más eficiente que la de los software de programación para PLCs tradicionales, pues reduce la posibilidad de cometer errores, al permitir la edición en cada bloque o función en forma independiente, además al tener la opción de asignar un OB para procesos supervisar condiciones para el arranque , nos permite monitorear en modo online que estas condiciones se cumplan en un solo sector de la programación, estas son algunas de las características positivas del software TIA PORTAL que nos permiten concluir que dicho recurso nos facilita trabajar de forma organizada y nos ahorra tiempo en la edición de un programa.

## RECOMENDACIONES

1. En vista de los resultados positivos obtenidos con este proyecto se recomienda implementar, equipamiento similar en el resto de las plantas de Laboratorio de Instrumentación Industrial, pues el entorno gráfico de TIA PORTAL, ofrece facilidad para el aprendizaje en la automatización de procesos.
2. Se recomienda analizar la opción de implementar un control que involucre las plantas de nivel y de caudal, pues estos 2 procesos en la práctica se relacionan frecuentemente y con la estructura de PLC y módulo que deja el presente proyecto esta opción es factible.
3. Recomendamos adicionar a los actuales manuales de laboratorio de la Planta GUNT RT450, los datasheets de los elementos que las integran a fin de conocer más detalles técnicos de los mismos e identificar posibles alternativas, cuando el estudiante más tarde en su vida profesional, se encuentre con procesos similares sepa la instrumentación adecuada que debe seleccionar.

# ANEXOS

## ANEXO 1

### DATASHEET DE SENSOR DE NIVEL CAPACITIVO ELOBAU

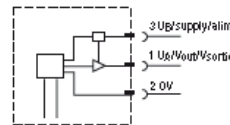
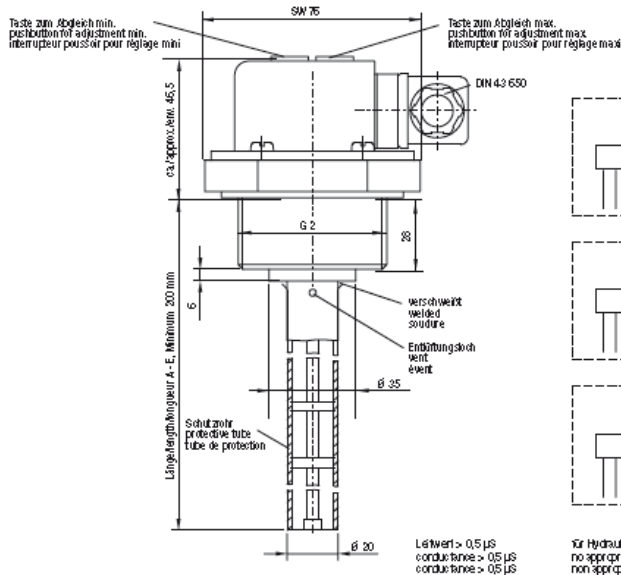
**Kapazitiver Füllstandsgeber**  
**Capacitive level sensors**  
**Détecteurs de niveau capacitifs**

www.elobau.com

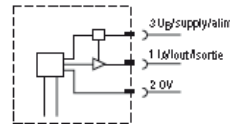


**212 KK. 0.**

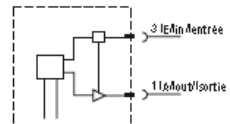
**Koaxial für leitende Medien**  
**Coaxial for conducting fluids**  
**Coaxial pour liquides conducteurs**



212 KK. 00  
 212 KK. 02  
 212 KK. 03



212 KK. 01  
 212 KK. 04



212 KK. 05

Typen Nr. type n.o. référence	Betriebsspannung operating voltage tension d'alimentation	Ausgangssignal output signal signal de sortie	Schutzart protection class protection	Temperaturbereich** temperature range** plage de température**	Material* material* matériau*
212 KK. 00	12 - 30V DC	1 - 5 V	IP 65	-20...+80°C	PVC/PA/Viton/PPVA stainless steel acier inoxydable
212 KK. 01	12 - 30V DC	4 - 20 mA			
212 KK. 02	15 - 30V DC	0 - 10 V			
212 KK. 03	12 - 30V DC	0 - 5 V			
212 KK. 04	12 - 30V DC	0 - 20 mA			
212 KK. 05	12 - 30V DC	4 - 20 mA (2-Draht/2-wire/2 fils)			

## ANEXO 2

# DATASHEET DE POSICIONADOR ELECTRO-NEUMÁTICO SAMSON 3760

Posicionador electro neumático y  
posicionador neumático  
Tipo 3760



MODEL 3760  
00121100000.02

### Aplicación

Posicionador de simple efecto, para montaje directo en válvulas de control neumáticas. La magnitud guía es una señal eléctrica normalizada de 4 a 20 mA o una señal neumática de 0,2 a 1 bar (3 a 15 psi).

Para carreras nominales de 5 a 15 mm



JIS

Los posicionadores garantizan una relación prefijada entre la posición de la válvula (magnitud regulada  $x$ ) y la señal de mando (magnitud guía  $w$ ). Comparan la señal de mando procedente de un aparato de regulación o de mando con la carrera de la válvula y envían como magnitud de salida ( $y$ ) una presión de mando (psi). Los posicionadores tienen las siguientes características:

Adecuados para servicio normal y de rango partido (split-range), sentido de actuación reversible, buen comportamiento dinámico, bajo consumo de energía auxiliar, escasa influencia de las vibraciones, ejecución compacta y exenta de mantenimiento. A elección con un final de carrera inductivo y sobre demanda con limitación de la presión de salida.

Ejecución para lugares con peligro de explosión con protección de seguridad intrínseca (i) II 2 G EEx ia IIC Tó según ATEX para el circuito de corriente del contacto de proximidad, así como en el posicionador electro neumático para el circuito de corriente de la señal (ver en la última página la relación de certificados de protección Ex concedidos).

Protección EEx d con el convertidor i/p Tipo 6116 (fig. 2).

Montaje directo en los accionamientos Tipo 3277 con superficie útil de membrana de 120, 240 y 360 cm<sup>2</sup> (para más detalles sobre los accionamientos consultar la hoja técnica T 8311).

A elección con manómetro para la presión de mando (escala de 0 a 6 bar y de 0 a 90 psi); caja del manómetro de acero inoxidable, pieza de conexión niquelada o de acero inoxidable.

### Ejecuciones

Tipo 3760 - Posicionador electro neumático (fig. 1)

Ejecución para lugares sin peligro de explosión. Margen de presión de mando de 0 a 6 bar (0 a 90 psi); aire de alimentación de 1,4 a 6 bar (20 a 90 psi). Magnitud guía: ejec. normal 4 a 20 mA.

Tipo 3760 - Posicionador electro neumático como el anterior, pero en ejecución para lugares con peligro de explosión, con protección (i) II 2 G EEx ia IIC Tó para el circuito de corriente de señal.

Protección EEx d con el convertidor i/p Tipo 6116 (fig. 2).

A elección también con un contacto de proximidad en ejecución de seguridad.

Tipo 3760 - Posicionador neumático - Magnitud guía: 0,2 a 1 bar (3 a 15 psi); margen de presión de mando de 0,2 a aprox. 6 bar (3 a aprox. 90 psi); aire de alimentación de 1,4 a 6 bar (20 a 90 psi).

A elección también con un contacto de proximidad en ejecución de seguridad.

En la última página se encuentra la información necesaria para pedidos.



Fig. 1 - Posicionador Tipo 3760



Fig. 2 - Válvula para microcaudales con posicionador EEx d (Tipo 3760 con convertidor i/p Tipo 6116)



### Funcionamiento

El posicionador electroneumático (i/p) se distingue del posicionador neumático únicamente en que el primero lleva incorporado un módulo convertidor (2).

Los posicionadores garantizan la relación entre la posición de la válvula y la señal de mando. La magnitud regulada  $x$  es la carrera de la válvula, la magnitud guía  $w$  es en el posicionador i/p una señal de corriente continua (i) y en el posicionador neumático (p/p) una señal neumática ( $p_e$ ) procedente de un dispositivo de regulación o de mando. La señal de mando  $y$  es la presión de salida ( $p_{st}$ ) del posicionador.

Los posicionadores Tipo 3760 están previstos para el montaje directo a accionamientos SAMSON Tipo 3277.

En el posicionador i/p fluye la señal de mando  $i$  procedente del regulador al módulo convertidor (2) y allí se transforma en una presión de aire proporcional  $p_e$  de 0,2 a 1 bar o de 3 a 15 psi. En el posicionador neumático (p/p) se conduce la señal de mando  $p_e$ , disponible en forma de una presión de aire, directamente a la membrana de medición (3).

La señal de presión  $p_e$  produce una fuerza sobre la membrana de medición (3) que se compara con la fuerza del resorte de medición (7). El movimiento de la membrana de medición (3) se transmite por medio de la palanca (4) al obturador doble (13) del piloto (12), dando así salida a una presión de mando  $p_{st}$  correspondiente. Variaciones de la señal de mando  $p_e$  o de la posición de la válvula producen una variación de la presión de mando  $p_{st}$ . Con ello el vástago del obturador de la válvula se sitúa en la posición correspondiente a la magnitud guía.

Los posicionadores pueden utilizarse para servicio normal o de rango partido (split-range). El margen de la presión de mando (margen de la presión de salida  $p_{st}$ ) debe corresponder al margen de los resortes del accionamiento.

En el servicio de rango partido se divide la señal de salida del regulador para el mando de dos válvulas, de forma que cada accionamiento alcanza su carrera total con la mitad del span de entrada: p. ej. el primer equipo se ajusta de 0,2 a 0,6 bar, y el segundo de 0,6 a 1 bar.

Los valores inicial y final de la señal de entrada se ajustan por medio de los tornillos de ajuste (5) para el cero y (8) para el span. El resorte de medición (7) deberá elegirse de acuerdo con la carrera nominal de la válvula y el span nominal de la magnitud guía.

### Sentido de actuación

Al aumentar la magnitud guía  $p_e$ , la presión de mando  $p_{st}$  puede aumentar (acción directa >>) o disminuir (acción inversa <<). El sentido de actuación se determina mediante la posición del piloto (12) y puede modificarse in situ.

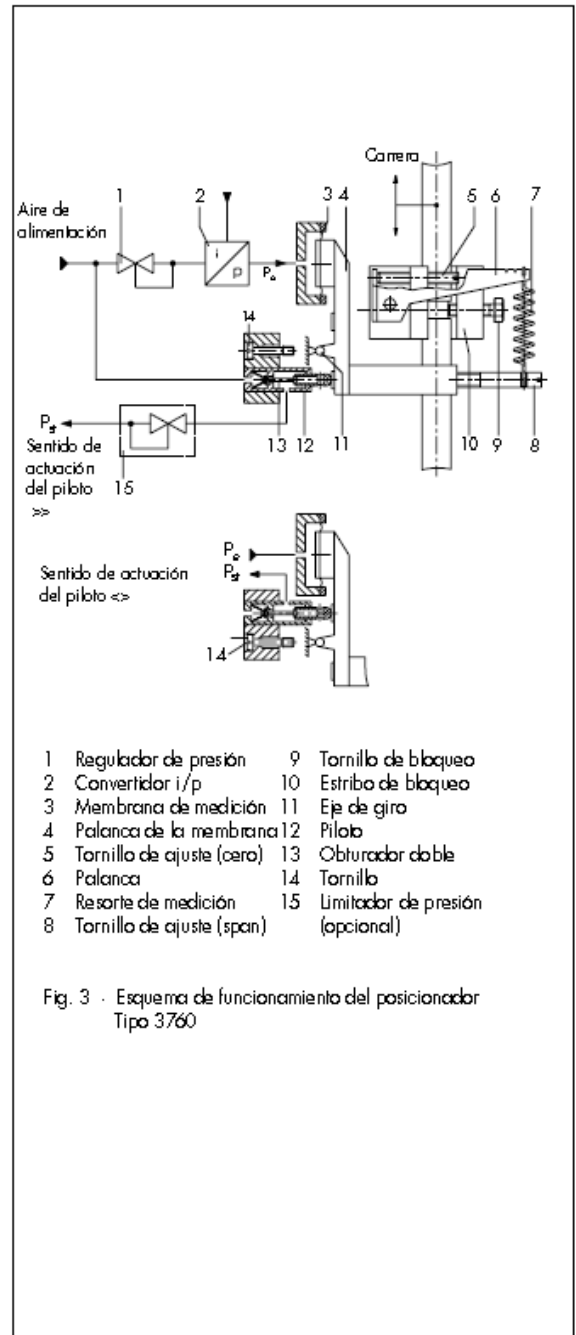


Fig. 3 - Esquema de funcionamiento del posicionador Tipo 3760

**Tabla 1 · Datos técnicos**

Carrera		5 a 15 mm (ver también la tabla 2, resortes de medición)
Magnitud guía Span para servicio rango partido 0 a 50 % o 50 a 100 %  (R <sub>i</sub> = resistencia de la bobina a 20 °C)	neumática	0,2 a 1 bar (3 a 15 psi)
	eléctrica	4 a 20 mA (sólo Ex) · R <sub>i</sub> ≅ 250 Ω ± 7 %
		4 a 20 mA (no Ex) · R <sub>i</sub> ≅ 200 Ω ± 7 %
		0 a 20 mA · R <sub>i</sub> ≅ 200 Ω ± 7 %
	1 a 5 mA · R <sub>i</sub> ≅ 880 Ω ± 7 %	
Energía auxiliar (aire de alimentación)		aire de alimentación de 1,4 a 6 bar (20 a 90 psi)
Presión de mando p <sub>st</sub> (salida)		máx. 0 a 6 bar (0 a 90 psi)
Característica		lineal, desviación de la característica con ajuste punto fijo ≤ 1,5 %
Histéresis		≤ 0,5 %
Sensibilidad de reacción		< 0,1 %
Sentido de actuación		reversible
Consumo de aire en régimen estable		con presión de mando de 0,6 bar y alimentación hasta 6 bar ≤ 100 l <sub>n</sub> /h
Suministro de aire		con Δp = 1,4 bar 1600 l <sub>n</sub> /h · con Δp = 6 bar 5000 l <sub>n</sub> /h
Tiempos de recorrido con accionamiento Tipo 3277 (carrera 15 mm, presión de mando 0,2 a 1 bar)		120 cm <sup>2</sup> ≤ 2 s · 240 cm <sup>2</sup> ≤ 6 s · 350 cm <sup>2</sup> ≤ 8 s
Temperatura ambiente admisible		-20 a +70 °C
		hasta -30 °C con rácores metálicos hasta -40 °C con rácores metálicos y convertidor i/p Tipo 6112 para equipos Ex son válidos los valores del Certificado UE de prueba de tipo -40 a 70 °C con el posicionador neumático Tipo 3760 sin contacto inductivo
Influencia	temp. punto cero	≤ 0,03 %/°C
	span	≤ 0,03 %/°C
	vibraciones	entre 5 y 120 Hz y 2g ≤ 0,5 %
	energía auxiliar	≤ 1 % entre 1,4 y 6 bar
Dependencia de la posición con giro de 180°		< 3,5 %
Clase de protección		IP 54 (IP 65 como ejecución especial)
Peso		0,6 kg
Materiales		caja de poliamida, piezas externas en acero inoxidable
Equipamiento adicional		
Final de carrera inductivo		tipo SJ2 – SN
Circuito de maniobra		valores según relé transistorizado conectado a continuación
Rango de conmutación con carrera nominal		≤ 1 %

## BIBLIOGRAFÍA

[1] Ogata, Katsuhiko, Ingeniería de Control Moderna, Pearson Educación 3ra Ed, 1998.

[2] Universidad Técnica Federico Santa María (Chile), Números en Punto Flotante, <http://profesores.elo.utfsm.cl/~tarredondo/info/comp-architecture/apuntes-lsb/cap9a.pdf>, fecha de consulta enero de 2015.

[3] GUNT HAMBURG, Manual de Experimentos RT450 para Sistema Didáctico Modular para la Automatización de Procesos, GUNT HAMBURG, 2005.

[4] SIEMENS, Simatic S7 Controlador Programable S7-1200 Manual de Sistema, SIEMENS, 2009.

[5] National Instruments, Introduction to OPC, <http://www.ni.com/white-paper/7451/en/pdf>, fecha de consulta diciembre de 2014.

[6] SIEMENS, The TIA PORTAL configuring instead of programming, <http://www.industry.siemens.com/topics/global/es/tia-portal/pages/default.aspx> , fecha de consulta febrero de 2015.

[7] ELOBAU, Capacitive level sensors 212 Datasheet, ELOBAU, 2010.

[8] SAMSON, Posicionador electroneumático y posicionador neumático datasheet, SAMSON, 2003.

[9] OPC FOUNDATION, What is OPC?, <https://opcfoundation.org/about/what-is-opc/>, fecha de consulta agosto de 2014.

[10] Kominek, Darek, OPC: De que se trata y cómo funciona, [http://www.infoplc.net/files/documentacion/comunicaciones/infoplc\\_net\\_guia\\_para\\_entender\\_la\\_tecnologia\\_opc.pdf](http://www.infoplc.net/files/documentacion/comunicaciones/infoplc_net_guia_para_entender_la_tecnologia_opc.pdf) , fecha de consulta noviembre de 2014.

[11] Halvorsen, Hans-Petter, OPC and Real Time Systems in Labview, Telemark University College (Noruega), 2012.

[12] Wikipedia, NI Labview, <http://es.wikipedia.org/wiki/LabVIEW>, fecha de consulta enero de 2015