



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

FACULTAD DE INFORMÁTICA Y ELECTRÓNICA

**ESCUELA DE INGENIERÍA ELECTRÓNICA EN CONTROL Y
REDES INDUSTRIALES**

**“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA SCADA PARA
EL ENVASADO Y TRANSPORTE DE LÍQUIDOS DE BAJA
DENSIDAD USANDO UN PLC SIEMENS S7-1200”**

TESIS DE GRADO

Previo a la obtención del título:

**INGENIERO EN ELECTRÓNICA CONTROL Y REDES
INDUSTRIALES.**

Presentado por:

WILMER RODRIGO FLORES ORNA

EDUARDO FRANCISCO GARCÍA CABEZAS

Riobamba - Ecuador

2012

AGRADECIMIENTO

A todas las personas que hicieron posible el cumplimiento de la meta más importante de mi vida, en especial a mis padres Eduardo y Norma, quienes me apoyaron en cada momento difícil de la carrera.

Eduardo Garcia

AGRADECIMIENTO

A nuestros padres pilares fundamentales de nuestra educación y superación, demostrando que a pesar de las adversidades siempre hay una luz al final del camino, los queremos.

A nuestros amigos, quienes siempre han permanecido junto a nosotros para escuchar, aconsejar y apoyarnos incondicionalmente.

Y a todas las personas que de una u otra manera influyeron para cumplir este sueño.

Wilmer Flores

A Dios por cada día que nos regala de vida para crear cada página del libro de nuestra existencia.

A nuestros padres pilares fundamentales de nuestra educación y superación, demostrando que a pesar de las adversidades siempre hay una luz al final del camino, los queremos.

A nuestros amigos, quienes siempre han permanecido junto a nosotros para escuchar, aconsejar y apoyarnos incondicionalmente.

Y a todas las personas que de una u otra manera influyeron para cumplir este sueño.

Eduardo Garcia

A Dios por cada día que nos regala de vida para crear cada página del libro de nuestra existencia.

A nuestros padres pilares fundamentales de nuestra educación y superación, demostrando que a pesar de las adversidades siempre hay una luz al final del camino, los queremos.

A nuestros amigos, quienes siempre han permanecido junto a nosotros para escuchar, aconsejar y apoyarnos incondicionalmente.

Y a todas las personas que de una u otra manera influyeron para cumplir este sueño.

Wilmer Flores

NOMBRE	FIRMA	FECHA
Ing. Iván Menes DECANO DE LA FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
Ing. Paúl Romero DIRECTOR DE LA ESCUELA DE INGENIERÍA ELECTRÓNICA CONTROL Y REDES INDUSTRIALES
Ing. Paúl Romero DIRECTOR DE TESIS
Ing. Fernando Chávez MIEMBRO DEL TRIBUNAL
Lcdo. Carlos Rodríguez DIRECTOR DPTO. DOCUMENTACIÓN
NOTA DE LA TESIS	

“Nosotros, **Wilmer Rodrigo Flóres Orna y Eduardo Francisco García Cabezas** somos responsable de las ideas, doctrinas y resultados expuestos en esta tesis; y, el patrimonio intelectual de la tesis de Grado pertenece a la ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO”

.....

Wilmer Rodrigo Flóres Orna

.....

Eduardo Francisco García Cabezas

ÍNDICE DE ABREVIATURAS

(En orden alfabético)

CMs	Módulos De Comunicación
DBs	Bloques De Datos
DDE	Intercambio Dinámico De Datos
DHCP	Protocolo De Configuración Dinámica De Host

ÍNDICE GENERAL

CAPITULO I.....	22
I. MARCO REFERENCIAL.....	22
1.1 ANTECEDENTES	22
1.2 JUSTIFICACION	24
1.3 OBJETIVOS	25
1.3.1 OBJETIVO GENERAL	25
1.3.2 OBJETIVOS ESPECIFICOS	25
1.4 HIPOTESIS	26
CAPITULO II.....	27
II. FUNDAMENTOS TEORICOS	27
2.1 SISTEMAS SCADA.....	27
2.1.1 LOS PERIFERICOS	27
2.1.1.1 SENSORES.....	27
2.1.1.1.1 Sensores Inductivos.....	28
2.1.1.1.1.1 Principio de Funcionamiento:	28
2.1.1.1.1.2 Aplicaciones.....	29
2.1.1.1.2 Sensores Magnéticos.....	29
2.1.1.1.2.1 Principio de funcionamiento:	29
2.1.1.1.2.2 Aplicaciones:.....	30
2.1.1.1.3 Sensores de nivel	30
2.1.1.1.3.1 Flotador o boya	30
2.1.1.1.3.2 Medida por diferencia de presiones	31
2.1.1.1.3.3 Medida por desplazadores de líquido.....	31
2.1.1.1.3.4 Medidor de nivel por ultrasonidos	31
2.1.1.1.4 SENSORES DE PRESIÓN.....	32
2.1.1.2 ACTUADORES.	33
2.1.1.2.1 ACTUADORES ELÉCTRICOS.....	33
2.1.1.2.1.1 Motores monofásicos de baja velocidad.....	33
2.1.1.2.1.2 Relés (Interruptores mecánicos).....	34
2.1.1.2.2 ACTUADORES NEUMÁTICOS.....	35
2.1.1.2.2.1 CILINDROS NEUMÁTICOS.....	35
2.1.1.2.2.1.1 CILINDROS DE SIMPLE EFECTO.	36
2.1.1.2.2.1.2 CILINDROS DE DOBLE EFECTO.	38
2.1.1.2.2.2 CONTROL DEL AIRE COMPRIMIDO: VÁLVULAS.	39
2.1.1.2.2.2.1 Parámetros de una válvula.	39
2.1.1.2.2.2.1.1 Vías y posiciones:.....	39

2.1.1.2.2.2.1.2	Accionamiento de la válvula:	39
2.1.1.2.2.2.1.3	Retorno de la válvula:	40
2.1.1.2.2.2.1.4	Simbología de Válvulas	40
2.1.1.2.2.2.1.5	Designación de una Válvula.....	41
2.1.1.2.2.2.2	VÁLVULAS DISTRIBUIDORAS.....	41
2.1.1.2.2.2.3	VÁLVULAS REGULADORAS DE CAUDAL.....	42
2.1.1.2.3	ELECTRONEUMÁTICA	43
2.1.2	PLC (Programmer Logic Controler).....	44
2.1.2.1	FUNCIONES BÁSICAS.....	45
2.1.2.2	ESTRUCTURA FISICA.....	46
2.1.2.2.1	Estructura Compacta.	46
2.1.2.2.2	Estructura Semimodular.	46
2.1.2.2.3	Estructura Modular.....	47
2.1.2.3	EL S7-1200 DE SIEMENS.....	47
2.1.2.3.1	CICLO DE OPERACIÓN	48
2.1.2.3.2	ESTADOS OPERATIVOS	49
2.1.2.3.3	ÁREAS DE MEMORIA, DIRECCIONAMIENTO Y TIPOS DE DATOS.....	49
2.1.2.3.4	TIPOS DE DATOS SOPORTADOS POR EL S7-1200.....	50
2.1.2.3.5	ÁREAS DE MEMORIA Y DIRECCIONAMIENTO.....	52
2.1.3	SISTEMA DE SUPERVISIÓN INTOUCH.....	54
2.1.3.1	Requerimientos del Sistema	54
2.1.3.2	Instalación.....	54
2.1.3.3	MENU WINDOWMAKER	58
2.1.3.3.1	Menú File (archivo).....	58
2.1.3.3.2	Menú Arreglar.....	59
2.1.3.3.3	Menú de Texto	60
2.1.3.3.4	Menú de Línea.....	60
2.1.3.3.5	Menú Especial	60
2.1.3.3.6	Menú de Ventana	61
2.1.3.4	Diccionario Tagname.....	61
2.1.3.4.1	Details.....	62
2.1.3.4.2	Alarms	63
2.1.3.4.3	Both	64
2.1.3.5	Ingresar Campos, Botones y Permitir al Usuario Opciones de Entrada	64
2.1.3.5.1	Tag Name:	64
2.1.3.5.2	Comment:.....	64
2.1.3.5.3	Read only	65
2.1.3.5.4	Read Write.....	65

2.1.3.5.5	Log Data.....	65
2.1.3.5.6	Log Events Priority 999.....	66
2.1.3.5.7	Retentive Value.....	66
2.1.3.5.8	Retentive Parameters.....	66
2.1.3.6	Tipos de Tag.....	66
2.1.3.6.1	Tagname tipo Memoria.....	66
2.1.3.6.2	Memory Discrete.....	67
2.1.3.6.3	Memory Integer.....	67
2.1.3.6.4	Memory Real.....	67
2.1.3.6.5	Memory Message.....	67
2.1.3.6.6	Tagname tipo DDE.....	67
2.1.3.6.7	DDE Discrete.....	67
2.1.3.6.8	DDE Integer.....	67
2.1.3.6.9	DDE Real.....	68
2.1.3.6.10	DDE Message.....	68
2.1.3.6.11	Tagnames de Tipo Misceláneo.....	68
2.1.3.6.12	Group Var.....	68
2.1.3.6.13	Hist Trend.....	68
2.1.3.6.14	Tag ID.....	68
2.1.3.6.15	Tagnames “Placeholder”.....	69
2.1.3.7	Links (Vínculos).....	70
2.1.3.7.1	Uniendo y Seleccionando un Vínculo.....	70
2.1.3.7.2	Prev Link y Next Link.....	71
2.1.3.7.3	Horizontal/Vertical Loc. Width/Height.....	71
2.1.3.8	Información común del Links... ..	72
2.1.3.8.1	La Paleta de Selección de Color.....	72
2.1.3.8.2	Asignando una tecla a un Vínculo de Animación.....	72
2.1.3.9	Links disponibles.....	72
2.1.3.9.1	Touch Links.....	72
2.1.3.9.2	Valor Slider Touch Links.....	73
2.1.3.9.3	Vínculos con Touch Pushbutton.....	73
2.1.3.9.4	Vínculo de Colores.....	73
2.1.3.9.5	Vínculos de Colores de Objetos a Estados de Alarma.....	74
	[Value Alarm Color Link].....	74
	[Deviation Alarm Color Link].....	74
2.1.3.9.6	Vínculos del Tamaño del Objeto.....	74
2.1.3.9.7	Vínculos de Localización.....	74
2.1.3.9.8	Vínculos del Porcentaje de Llenado.....	74

2.1.3.9.9	Vínculos Misceláneos	75
2.1.3.9.10	Vínculos de valores de salida	75
2.1.3.10	WINDOWVIEWER.....	75
2.1.3.10.1	Menú File	75
2.1.3.10.2	Controlador Logic	76
2.1.3.10.3	Menú Special	76
2.1.3.10.4	Special/Security	76
2.1.3.11	DISEÑO DE SCRIPTS EN INTOUCH.....	77
2.1.3.11.1	Scripts de Aplicación.....	78
2.1.3.11.2	Script de Ventana.....	79
2.1.3.11.3	Script de Teclas.....	79
2.1.3.11.4	Scripts de Condiciones	80
2.1.3.11.5	Scripts de Cambio de Datos	81
2.1.3.11.6	Ingreso de la Operación Lógica	82
2.1.3.11.7	Significado de cada uno de los Botones.....	82
2.1.3.11.8	Funciones específicas.....	83
2.1.3.11.9	Choose Function.....	84
2.1.3.12	ALARMAS Y EVENTOS	85
2.1.3.12.1	Alarmas.....	85
2.1.3.12.1.1	Tipos de Alarmas.....	85
2.1.3.12.1.2	Prioridades de las Alarmas	85
2.1.3.12.1.3	Grupos de Alarmas	85
2.1.3.12.1.4	Cómo Crear Grupos de Alarmas.....	86
2.1.3.12.1.5	Definición de una Condición de Alarma en un Tagname	86
2.1.3.12.1.6	Creación de un Objeto de Alarmas	87
2.1.3.12.1.7	Configuración de un Objeto de Alarmas	88
2.1.3.12.1.8	Configuración de Alarmas/Eventos	90
2.1.3.12.1.9	Creación de Condiciones de Reconocimiento	91
2.1.3.12.1.10	Campos de las Alarmas	91
2.1.3.12.2	Eventos.....	93
2.1.3.12.3	CONFIGURACION DE UN VISOR DE ALARMAS	93
2.1.3.12.4	COMUNICACIONES I/O	95
2.1.3.12.4.1	La Comunicación DDE	95
2.1.3.12.5	InTouch Access Name	95
2.1.4	El sistema de almacenamiento de datos.....	96
2.1.4.1	CONCEPTO DE BASE DE DATOS	96
2.1.4.2	COMPONENTES DE UN SISTEMA DE BASE DE DATOS	97
2.1.4.2.1	Datos	97

2.1.4.2.2	Equipo (Hardware).....	97
2.1.4.2.3	Programas (Software)	98
2.1.4.2.4	Personal	98
2.1.4.3	ESTRUCTURA GENERAL DEL SGBD.....	99
CAPITULO III.....		101
III.	SISTEMA DE ENVASADO VOLUMETRICO.	101
3.1	Módulo de envasado.	101
3.2	Diseño de la estructura.	102
3.3	Partes del equipo.....	103
3.4	Sistema de dosificación de recipientes.	103
3.5	Sistema de almacenamiento de líquido.	104
3.6	Sistema de Transporte.	105
3.7	Sistema de llenado.	105
3.8	Sistema de colocación de tapas.....	106
3.9	Funcionamiento del módulo de envasado volumétrico.....	108
3.10	Estado Inicial.	109
3.11	Detección de líquido y Detección de envase.....	110
3.12	Gusano Transporte.....	111
3.13	Dosificación del producto.....	112
3.14	Detección de tapa.....	113
3.15	Sellado.....	113
3.16	Dimensionamiento de elementos constitutivos del módulo.	114
3.17	Dimensionamiento del cilindro de empuje hacia gusano de transporte.	114
CAPITULO IV		116
IV.	DESARROLLO DEL SOFTWARE	116
4.1	PROGRAMACIÓN Y CONFIGURACION S7-1200 – Step 7 Basic v10.5.....	116
4.1.1	HERRAMIENTAS DE AUTOMATIZACIÓN: GRAFCET.....	122
4.1.1.1	METODOLOGÍA GRAFCET.....	122
4.1.1.1.1	Etapas.....	122
4.1.1.1.2	TRANSICIÓN	124
4.1.1.2	Divergencia	124
4.1.1.2.1	ESTRUCTURAS PRINCIPALES.....	125
4.1.1.2.1.1	SECUENCIA ÚNICA.....	125
4.1.1.2.1.2	SECUENCIAS SIMULTÁNEAS O PARALELISMO ESTRUCTURAL	126
4.1.2	DESARROLLO E IMPLEMENTACIÓN DEL DIAGRAMA GRAFCET DEL PROCESO.	127
4.1.2.1	DIAGRAMA GRAFCET PARA EL SISTEMA DE ENVASADO VOLUMETRICO.....	127

4.2	DESARROLLO DE LA APLICACIÓN EN INTOUCH.	128
4.2.1	ANÁLISIS DE REQUERIMIENTOS DE LA APLICACIÓN.	128
4.2.2	Creación de la aplicación.	130
4.3	Configuración del TOP SERVER 5.....	131
4.4	Enlace OPC-INTOUCH.....	137
4.5	Comunicación de INTOUCH con SLQ Server 2005	141
4.6	CREACIÓN DEL ODBC	142
4.7	PANTALLAS DE TRABAJO	148
4.7.1	Inicio.....	148
4.7.2	Ingreso.	148
4.7.3	Autenticación.....	149
4.7.4	Proceso.	150
	CAPITULO V	151
	V. PRUEBAS Y RESULTADOS	151
5.1	Sistema de dosificación.....	151
5.2	Pruebas de software.....	153
5.3	Tabulación de Datos de cada Pregunta realizada la encuesta.	156
	CONCLUSIONES	160
	RECOMENDACIONES	163
	RESUMEN	¡Error! Marcador no definido.
	SUMMARY	¡Error! Marcador no definido.
	ANEXOS.....	168
	BIBLIOGRAFIA	164

ÍNDICE DE TABLAS

Tabla II-I Ventajas y desventajas de los sensores inductivos	29
Tabla II-II Tipos de datos soportados por el S7-1200.....	50
Tabla II-III Datos adicionales para el S7-1200	52
Tabla II-IV Direccionamiento de datos	53
Tabla IV-I Niveles de acceso de la aplicación.	149
Tabla V-I Volúmenes obtenidos por tiempos de dosificación.....	152
Tabla V-II Tiempos de proceso.	153

ÍNDICE DE FIGURAS

Fig. II-1 Sensores inductivos	28
Fig. II-2 Funcionamiento del sensor inductivo.	28
Fig. II-3 Aplicaciones	29
Fig. II-4 Sensores Magnéticos.....	29
Fig. II-5 Funcionamiento del sensor magnético	30
Fig. II-6 Flotador o boya	31
Fig. II-7 Medidor de presión (Manómetro)	32
Fig. II-8 Motor monofásico de baja velocidad.....	33
Fig. II-9 Esquema básico de relé.....	34
Fig. II-10 Estructura de un cilindro neumático	35
Fig. II-11 Cilindro de simple efecto.....	36
Fig. II-12 Cilindro de membrana.....	37
Fig. II-13 Cilindro de membrana arrollable.	38
Fig. II-14 Cilindro de doble efecto	38
Fig. II-15 Partes del Cilindro de doble efecto	39
Fig. II-16 Foto de un cilindro de doble efecto.	39
Fig. II-17 Simbología de Posiciones para Válvulas.....	40
Fig. II-18 Simbología de Vías para Válvulas	40
Fig. II-19 Tipos de Accionamientos para Válvulas	41
Fig. II-20 Válvula cierra o abre el flujo de aire hacia el cilindro.....	42
Fig. II-21 Cilindro de simple efecto controlado por una válvula 3/2.....	42
Fig. II-22 Válvulas reguladoras de caudal	42
Fig. II-23 Estructura básica de un PLC	45
Fig. II-24 Ejemplos de PLC's compactos	46
Fig. II-25 PLC de estructura modular	47
Fig. II-26 Ciclo de ejecución del PLC	48
Fig. II-27 Estados operativos del PLC.....	49
Fig. II-28 Esquema de mapa de memoria.....	54
Fig. II-29 Pantalla inicial de la aplicación InTouch	55
Fig. II-30 Creación de la aplicación.....	56
Fig. II-31 Creación y definición de parámetros de una ventana	57
Fig. II-32 Barra de herramientas	57
Fig. II-33 Submenú Symbol Factory de la herramienta Wizard.....	58
Fig. II-34 Submenú File	59
Fig. II-35 Submenú arranque.....	59
Fig. II-36 Submenú Special	61
Fig. II-37 Tagname Dictionary	62
Fig. II-38 Cuadro de Diálogo Details	63
Fig. II-39 Cuadro de Diálogo de Alarmas	63
Fig. II-40 Cuadro combinado parámetros Details & Alarms.....	64
Fig. II-41 Ingresar Campos, botones y opciones de entrada	64

Fig. II-42 Tipos de Tags	65
Fig. II-43 Special/Substitute All Tags	69
Fig. II-44 Mensaje para crear tag no existe	70
Fig. II-45 Ítem Description	71
Fig. II-46 Parámetros animaciones longitud y posición	71
Fig. II-47 Touch Links	72
Fig. II-48 Touch Pushbutton	73
Fig. II-49 Submenú File/Windowviewer	75
Fig. II-50 Submenú Special/Windowviewer	76
Fig. II-51 Submenú Special/Security/Windowviewer.....	76
Fig. II-52 Special/Scripts (Submenú)	77
Fig. II-53 Cuadro de Diálogo para edición de Scripts	78
Fig. II-54 Modos de ejecución de los Scripts	78
Fig. II-55 Cuadro de Diálogo Window Script	79
Fig. II-56 Cuadro de Diálogo Key Script	79
Fig. II-57 Cuadro de Diálogo Condition Script	80
Fig. II-58 Cuadro de Diálogo Data Change Script	82
Fig. II-59 Cuadro de Diálogo para Operaciones Lógicas	82
Fig. II-60 Botones de Operaciones Lógicas y Comparaciones Matemáticas	82
Fig. II-61 Cuadro de Funciones Especiales pag1.....	84
Fig. II-62 Cuadro de Funciones Especiales pag2.....	84
Fig. II-63 Tipos de Alarmas	85
Fig. II-64 Asociacion de tags a grupos	86
Fig. II-65 Definición de una Alarma en un Tagname.	86
Fig. II-66 Posibilidades de Alarmas Analógicas.....	87
Fig. II-67 Cuadro de Alarmas	87
Fig. II-68 Cuadro de Diálogo para configuración de un Objeto de Alarma	88
Fig. II-69 Cuadro de Diálogo para configuración de un Objeto de Alarma /Message	89
Fig. II-70 Wizards/Alarms Display.....	93
Fig. II-71 Inserción de un Display de Alarmas	94
Fig. II-72 Estructura general del SGBD	100
Fig. III-1 Perfil de aluminio 25 Mk	102
Fig. III-2 Estructura del sistema.	102
Fig. III-3 Sistema de dosificación de recipientes.	104
Fig. III-4 Sistema de almacenamiento de líquido.	104
Fig. III-5 Gusano Transportador	105
Fig. III-6 Cilindros dosificadores	106
Fig. III-7 Sistema de colocación y sellado de tapas.	107
Fig. III-8 Diagrama de proceso de envasado volumétrico.....	108
Fig. III-9 Posiciones iniciales del sistema	110
Fig. III-10 Detección de líquido.....	110
Fig. III-11 Detector de envases.....	111
Fig. III-12 Gusano transportador.....	112

Fig. III-13 Cilindros dosificadores.	112
Fig. III-14 Dosificación de tapas.....	113
Fig. III-15 Cilindro de sellado.	114
Fig. IV-1 Ubicación de la Aplicación	116
Fig. IV-2 Pantalla Inicial TIA asignaciones por defecto.....	117
Fig. IV-3 Creación del Nuevo Proyecto.....	117
Fig. IV-4 Seleccionar Configura un Dispositivo	118
Fig. IV-5 Selección de CPU	119
Fig. IV-6 CPU cargado para el programa	119
Fig. IV-7 Inserción Módulo de Entradas y Salidas	120
Fig. IV-8 Establecimiento de IP para la PC.....	121
Fig. IV-9 Establecimiento de IP para el PLC.....	121
Fig. IV-10 Etapa grafcet.	123
Fig. IV-11 Representación de etapas.....	123
Fig. IV-12 Transiciones desde y hacia una etapa.....	123
Fig. IV-13 Representación de las transiciones.....	124
Fig. IV-14 Representación divergencia O	125
Fig. IV-15 Representación divergencia Y.....	125
Fig. IV-16 Secuencia única	126
Fig. IV-17 Secuencia simultáneas.	126
Fig. IV-18 Acciones que desarrollara el sistema.....	127
Fig. IV-19 Esquema de conexión con INTOUCH.	129
Fig. IV-20 Pantalla APLICATION MANAGER.	130
Fig. IV-21 Windows Maker diseño inicial de la HMI.....	131
Fig. IV-22 Directorio y Pantalla principal del top server 5.....	132
Fig. IV-23 Inicio de configuración de comunicación del PLC.....	132
Fig. IV-24 Pantalla de selección de driver de comunicación.	133
Fig. IV-25 Selección del adaptador de red para la comunicación.	133
Fig. IV-26 Pantalla modo de actualización de tags.....	134
Fig. IV-27 Pantalla final de configuración de red.	134
Fig. IV-28 Pantalla configuración de dispositivo.	135
Fig. IV-29 Escogemos el modelo de autómatas y asignamos la dirección IP.....	135
Fig. IV-30 Se establece el tiempo de scan para el proceso.	135
Fig. IV-31 Se establecen los parámetros de puerto de comunicación y modos de escritura. ..	136
Fig. IV-32 Establece el tipo de conexión y el bit más significativo en la comunicación.	136
Fig. IV-33 Datos finales de configuración.....	136
Fig. IV-34 Pantalla de creación de tags.	137
Fig. IV-35 Directorio de acceso a OPCLINK y ventana de configuración.....	137
Fig. IV-36 Creación de conexión opc y pantalla de configuración.	138
Fig. IV-37 Datos de configuración opc Link.	139
Fig. IV-38 Configuración del OPC Tag Creator.....	139
Fig. IV-39 Creación de variables en intouch.....	140
Fig. IV-40 Búsqueda del archivo de configuración del opclink.....	140

Fig. IV-41 Ubicación SQL SERVER MANAGEMENT STUDIO.	141
Fig. IV-42 Menú inicio (Seleccionamos Panel de Control).	142
Fig. IV-43 Ventana de Panel de Control (Seleccionamos Herramientas Administrativas).	142
Fig. IV-44 Submenú Herramientas Administrativas (Seleccionamos Origen de Datos ODBC). .	143
Fig. IV-45 Ventana Administrador de Origenes de Datos (Seleccionamos DSN de archivos) ...	143
Fig. IV-46 Selección controlador para el que se desea establecer Origen de De Datos (Seleccionamos SQL Server).	144
Fig. IV-47 Asignar un Nombre al Origen de Datos.	144
Fig. IV-48 Finalizar para Crear el Origen de Datos.	145
Fig. IV-49 Elección del Servidor con el q se desea Conectar el Origen de Datos.	145
Fig. IV-50 Elección de tipo de Autenticación del SQL Server.	146
Fig. IV-51 Asignación de la Base de Datos.	146
Fig. IV-52 Complementos de Información para creación del Origen de Datos.	147
Fig. IV-53 Datos del nuevo ODBC y finalización de la creación.	147
Fig. IV-54 51 Pantalla de inicio del proceso.	148
Fig. IV-55 Pantalla de ingreso.	149
Fig. IV-56 Pantalla de autenticación.	149
Fig. IV-57 Pantalla principal del Proceso.	150
Fig. V-1 primera posición del dosificador.	151
Fig. V-2 Corrección del sistema de cilindros dosificador.	152
Fig. V-3 Pantalla del proceso con señales del sistema físico.	154
Fig. V-4 Generación de reportes.	155
Fig. V-5 Reporte de producción por fecha.	¡Error! Marcador no definido.
Fig. V-6 Reportes diarios.	155
Fig. V-7 reporte con rango de fechas.	¡Error! Marcador no definido.
Fig. V-8 Reporte de alarmas.	156

INTRODUCCION

Los procesos de manufactura dentro de la industria de cualquier sector económico en la que esta se encuentre ubicada, han ido avanzando conforme avanza la tecnología de automatización y control de procesos. Haciendo que cada vez sean más importantes la implementación de sistemas completos en los que se puedan involucrar, tanto la parte operativa como la supervisión de los mismos. Para esto contamos con los sistemas de Supervisión, control y adquisición de datos (SCADA, pos sus siglas en inglés).

El presente proyecto representa el desarrollo de un sistema SCADA para el envasado y transporte de líquidos de baja densidad, como plataforma de aprendizaje de sistemas integrados de control, en los cuales además de los componentes del proceso se involucra también la parte de adquisición de datos y la integración de estos hacia sistemas superiores de administración basados en plataformas de bases de datos para ayudar a presentar la información en forma resumida que tenga como finalidad ayudar a la toma de decisiones con respecto al mantenimiento del proceso, así como también de la emisión de reportes que ayuden a identificar fallas y anomalías presentadas dentro del mismo.

Además cuenta con un sistema sensorizado que ayuda al control mismo del proceso y a la toma de decisiones por medio del controlador lógico programable ubicado como cerebro de la máquina. El cual mediante la tecnología OPC server se comunica con el computador y un sistema HMI, (intouch), para supervisar una a una las acciones desarrolladas por la maquinaria.

Se plantea un sistema de este tipo ya que como es de nuestro conocimiento los profesionales de la carrera de ingeniería electrónica en control y redes industriales necesitan conocer estos sistemas como parte de su formación integral, para poder involucrarse en el mundo laboral con un conocimiento mucho más pragmático de la realidad del mismo.

Este proyecto consta del sistema de control centralizado con el PLC s7-1200 de siemens, el software INTOUCH de wonderware, TOP SERVER 5 OPC server, y SQL

server 2005 como motor de bases de datos para el desarrollo de la parte administrativa del mismo.

El plc siemens s7-1200, ya que es de la gama alta de plc's de esta nueva generación, por sus ventajas a la hora de realizar la programación y por las potencialidades que este nos ofrece a la hora de centralizar el control de procesos dentro de una cpu de este gama de plc.

INTOUCH por la facilidad en su programación y por los retos que nos puede ofrecer a la hora de convertirse en un sistema didáctico y plataforma de aprendizaje en lo que tiene que ver a programación de sistemas SCADA.

SQL server 2005 por su facilidad de comunicación, por su alcance dentro del desarrollo de bases de datos, y el opc server por su potencialidad como traductor entre el PLC y el INTOUCH.

CAPITULO I

I. MARCO REFERENCIAL

1.1 ANTECEDENTES

Dentro del campo de la automatización de procesos industriales encontramos un sin número de problemas y retos a los cuales debemos prestar singular atención, este es el caso por ejemplo del tratamiento de líquidos de baja densidad, en lo que tiene que ver a sus mezclas, envasado, transporte dentro del proceso de producción.

Para esto contamos con diferentes etapas del procesamiento las cuales se encuentran comunicadas, ya sea por operadores que llevan el producto de una etapa a otra o por bandas transportadoras que realizan el mismo proceso, pero esto conlleva tiempo y retardos dentro del proceso productivo razón por la cual nos vemos en la necesidad de disminuir estos tiempos muertos de producción.

Todo esto lo podemos desarrollar dentro del entorno educativo en el cual todavía nos encontramos desarrollándonos con la adecuada implementación de procesos didácticos, que se puedan acoplar entre sí para simular un ambiente industrial real.

Estos sistemas deben contar por supuesto con la debida seguridad para evitar ser dañados y más que nada evitar causar daño a las personas que vayan a trabajar con estos, para esto lo que necesitamos es dotar a nuestros sistemas de cierta inteligencia y porque no decirlo así de sentidos que le permitan al sistema ser un tanto autónomo.

Los sentidos de los cuales vamos a dotar al proceso son mediciones de las variables involucradas dentro del proceso de producción, que serán las encargadas de

informarnos acerca de cómo se está desarrollando nuestro proceso, claro siempre que le dotemos de un sistema de control adecuado que pueda tomar en cierta forma decisiones para mantener a estas variables del proceso estables.

Por otro lado la técnica de control y regulación en el proceso de envasado de líquidos de baja densidad es un problema concerniente a diversas industrias tales como química, bebidas y alimentos, farmacéutica, pinturas en donde se producen productos mediante la mezcla de varios productos líquidos y en algunos casos acompañados de procesos térmicos controlados.

La búsqueda de soluciones automatizadas a estos procesos solo pueden ser desarrollados en laboratorios donde se pueda simular los procesos y en los cuales intervengan áreas multidisciplinarias tales como: mecatrónica, tecnología de calentamiento, electro neumática y sensórica, apoyadas por sistemas de hardware y software de control aplicados, que desarrollen potentes sistemas de control programas que den soluciones estándar o personalizados a los diferentes problemas planteados.

Estas soluciones deben ser soluciones integrales y acoplables a otras que ya se han desarrollado antes para formar sistemas completos de tratamiento de las materias primas. Contando estas con monitoreo en tiempo real, así como también con algoritmos de control óptimo que demuestren la capacidad técnica de desarrollo adquirida a lo largo de la carrera de **INGENIERÍA EN ELECTRÓNICA CONTROL Y REDES INDUSTRIALES.**

1.2 JUSTIFICACION

El desarrollo de este proyecto tiene como objetivo diseñar e implementar un sistema SCADA para el envasado y transporte de líquidos de baja densidad que involucrará tomando como base para el sistema de control un PLC siemens s7-1200, y como la parte más importante del control sensor de nivel, pistón de desplazamiento para el llenado del líquido a un volumen requerido, y demás dispositivos que realicen el control del proceso propiamente dicho.

El presente proyecto se plantea como una alternativa para la implementación de los laboratorios de control de procesos industriales, que ayuden a los estudiantes de la escuela de Ingeniería en Electrónica Control y Redes Industriales, a tener una formación integral en lo que tiene que ver al pensum de la carrera. Para lograr la formación de profesionales con criterios teórico-prácticos en lo que a procesos industriales se refiere.

El sistema de transporte que se plantea construir es un sistema de transporte por gusano el cual llevará los envases para el líquido de un lugar a otro como alternativa y una forma diferente de realizar un sistema de transporte a parte a las bandas transportadoras.

Por otro lado se busca implementar un proceso que sea directamente acoplable a otros sistemas que se han desarrollado como parte de los laboratorios de la escuela para brindar la posibilidad de generar procesos industriales completos y complejos que ayuden a la formación de los futuros profesionales y estudiantes de la EIE-CRI

Además el proyecto permitirá aprovechar los conocimientos de los estudiantes con los recursos tecnológicos y las competencias intelectuales, para el desarrollo de una herramienta didáctica basada en tecnologías de última generación para que se facilite el aprendizaje de los estudiantes en los laboratorios de la escuela de Ingeniería Electrónica, Control y Redes Industriales y sea parte de un programa piloto, modelo de innovación, empeño y muestra de colaboración para obtener como valor agregado el posicionamiento de nuestra escuela a nivel regional y nacional, aportando de este modo el desempeño brillante de sus egresados y el deseo de muchos Ecuatorianos de formarse y capacitarse en tan prestigiosa institución.

1.3 OBJETIVOS

1.3.1 OBJETIVO GENERAL

- Diseñar e implementar de un sistema SCADA para el envasado y transporte de líquidos de baja densidad usando un PLC siemens S71200 e INTOUCH como plataforma de aprendizaje de procesos industriales reales.

1.3.2 OBJETIVOS ESPECIFICOS

- Diseñar y construir un sistema de llenado volumétrico utilizando un pistón de desplazamiento para obtener un volumen ideal para ser depositado en los tarros del proceso.
- Diseñar y construir un sistema de alimentación de tarros de 1/8 de litro por medio de un gusano transportador que nos ubique los tarros en el lugar exacto donde se va a realizar el llenado.
- Diseñar e implementar el sistema de alimentación, colocación y sellado de tapas metálicas para cubrir el tarro que lleva el líquido para obtener un producto terminado.
- Desarrollar el programa del PLC para que nos permita controlar el proceso mediante la plataforma Tottally Integrated Studio V10.5
- Desarrollo del Sistema HMI en INTOUCH para visualizar el progreso del proceso y además realizar el control de usuarios, registro de variables del proceso, almacenamiento de datos y alarmas.
- Integrar el sistema HMI con el sistema de control del PLC para obtener un sistema SCADA completo que ayude a los estudiantes a visualizar un proceso industrial completo.
- Desarrollar un manual de usuario del sistema para dotar a los estudiantes de una información completa para el uso del sistema del laboratorio.

1.4 HIPOTESIS

Con la implementación del sistema SCADA de envasado volumétrico de líquidos de baja densidad se dotará a los estudiantes de la Escuela de Ingeniería en Electrónica Control y Redes Industriales de una plataforma de aprendizaje que permita realizar prácticas de laboratorio en lo que tiene que ver a las cátedras de Automatización Industrial, Redes Industriales, Control de Procesos Industriales, Neumática, para pragmatizar los conceptos adquiridos teóricamente y relacionarlos con la práctica.

CAPITULO II

II. FUNDAMENTOS TEORICOS

2.1 SISTEMAS SCADA

Los sistemas de supervisión, control y adquisición de datos SCADA por sus siglas en inglés, son sistemas desarrollados con la finalidad de tener una visión amplia del proceso y mediante el cual se puede integrar el control del mismo por la adquisición de datos que son dados por los sensores implementados, y además llevar un control de las variables involucradas dentro del sistema.

Estos sistemas están conformados por cuatro componentes principales:

- Los periféricos (sensores y actuadores).
- El controlador.
- El sistema de supervisión.
- El sistema de almacenamiento de datos.

2.1.1 LOS PERIFERICOS

Son los dispositivos o instrumentos que desarrollan las acciones del proceso en este caso los actuadores, motores, cilindros, válvulas, relés, contactores, etc., y también los sensores que son los que proporcionan señales sobre el estado del mismo, ubicación de los elementos que intervienen dentro de este, niveles de las variables involucradas, y demás situaciones susceptibles de ser medidas por las acciones realizadas por los actuadores dentro del proceso.

2.1.1.1 SENSORES

Un sensor es un dispositivo capaz de convertir un tipo de energía en una variación de energía eléctrica, ya sea de voltaje o corriente que nos permita conocer el estado de

un proceso, mediante una señal de encendido o apagado, o una variación de valores en un rango determinado.

2.1.1.1.1 Sensores Inductivos

le proximidad
s de posición
señal de salida
directo, estos
ipo de objetos



Los sensores de proximidad inductivos, son detectores de posición electrónicos, que dan una señal de salida sin contacto mecánico directo, estos sensores detectan todo tipo de objetos metálicos.

2.1.1.1.1.1 Principio de Funcionamiento:

Los sensores inductivos de proximidad han sido diseñados para trabajar generando un campo magnético y detectando las pérdidas de corriente de dicho campo generadas al introducirse en él, los objetos de detección férricos y no férricos. El

Fig. II-1 Sensores inductivos

sensor consiste en una bobina con núcleo de ferrita, un oscilador, un sensor de nivel de disparo de la señal y un circuito de salida.

Al aproximarse un objeto "metálico" o no metálico, se inducen corrientes de histéresis en el objeto. Debido a ello hay una pérdida de energía y una menor amplitud de oscilación. El circuito sensor reconoce entonces un cambio específico de amplitud y genera una señal que conmuta la salida de estado sólido o la posición "ON" y "OFF". La bobina detecta el objeto cuando se produce un cambio en el campo electromagnético y envía la señal al oscilador, luego se activa el disparador y finalmente al circuito de salida hace la transición entre abierto o cerrado.

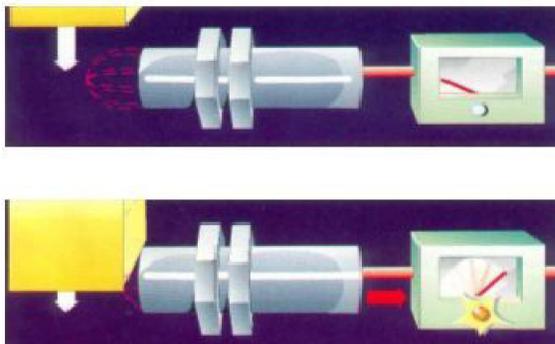


Fig. II-2 Funcionamiento del sensor inductivo.

Tabla II-I Ventajas y desventajas de los sensores inductivos

Ventajas	Desventajas
<ul style="list-style-type: none">• No entran en contacto directo con el objeto a detectar.• No se desgastan• Tienen un tiempo de reacción muy rápido.• Tienen una vida útil independiente del número de detecciones.• Son insensibles a la humedad y el polvo.• Incluyen indicadores LED de estado y tienen estructura modular.	<ul style="list-style-type: none">• Solo detectan la presencia de objetos metálicos.• Se pueden ver afectados por campos electromagnéticos externos.• Su rango de medición es más corto que otros sensores.

2.1.1.1.2 Aplicaciones

Estos sensores se utilizan en las condiciones de trabajo más difíciles donde hay presente aceites, líquidos, polvos y vibraciones, entre algunas que se mencionan están: herramientas, máquinas textiles, líneas transportadoras, sistema de transporte, equipos de empaquetado y paletizado, industria automotriz, etc.



Fig. II-3 Aplicaciones

2.1.1.1.2 Sensores Magnéticos



Los sensores magnéticos también se les denominan relés tipo “reed”, son utilizados en cilindros neumáticos para detectar la posición de fin de carrera a través del vástago.

2.1.1.1.2.1 Principio de funcionamiento:

Los sensores de proximidad magnéticos son caracterizados por la posibilidad de distancias grandes de la conmutación, disponible de los

Fig. II-4 Sensores Magnéticos

sensores con dimensiones pequeñas.

Detectan los objetos magnéticos (imanes generalmente permanentes) que se utilizan para accionar el proceso de la conmutación. Los campos magnéticos pueden pasar a través de muchos materiales no magnéticos, el proceso de la conmutación se puede también accionar sin la necesidad de la exposición directa al objeto. Usando los conductores magnéticos (ej. hierro), el campo magnético se puede transmitir sobre mayores distancias para, por ejemplo, poder llevarse la señal de áreas de alta temperatura.

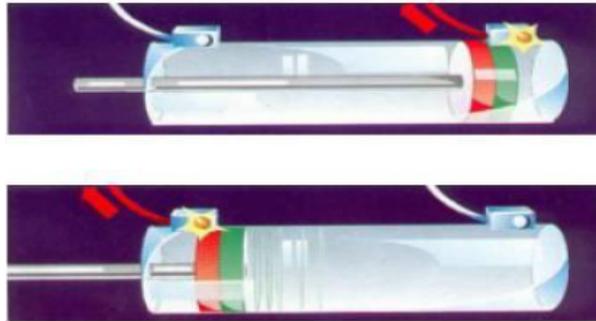


Fig. II-5 Funcionamiento del sensor magnético

2.1.1.1.2.2 Aplicaciones:

Automatismos, acondicionamiento, control de cadenas transportadoras, detección de posición de émbolo en cilindros neumáticos.

2.1.1.1.3 Sensores de nivel

Existen diferentes métodos para realizar la medición de nivel de líquidos, entre los que podemos citar:

2.1.1.1.3.1 Flotador o boya

El nivel de líquido eleva una boya o flotador que hace girar una polea mediante un hilo con un contrapeso en el extremo opuesto. El giro de la polea convenientemente traducido nos da el nivel del líquido.

Para el proyecto se utilizó este principio de medición con la diferencia de la implantación de un potenciómetro en la base giratoria del flotador para obtener una variación de resistencia, según la variación de nivel en el tanque reservorio.

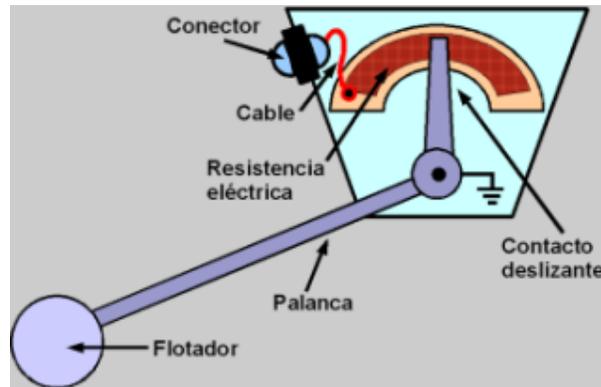


Fig. II-6 Flotador o boya

Existen otros métodos de detección de nivel que vale la pena mencionarlos.

2.1.1.1.3.2 Medida por diferencia de presiones

Se basa en que la presión estática en cualquier punto de un líquido es igual al producto de la distancia desde el punto a la superficie del líquido, su densidad y la aceleración de la gravedad. Si conocemos la densidad del líquido y la gravedad, se puede obtener relativamente fácil el nivel del líquido mediante una medida de presión (para lo que se usará el transductor correspondiente).

2.1.1.1.3.3 Medida por desplazadores de líquido

Se basa en el principio de Arquímedes. El empuje ejercido por el líquido sobre el elemento sumergido es igual al peso del volumen del líquido desalojado. Si el elemento sumergido tiene unas dimensiones constantes (cilindro) y le mantenemos siempre al mismo nivel, cuanto menor sea el líquido en el recipiente menor empuje se ejercerá sobre dicho elemento. En definitiva, mediremos el empuje que experimenta el elemento sumergido, convirtiéndolo en la correspondiente medida de nivel.

2.1.1.1.3.4 Medidor de nivel por ultrasonidos

Se basa en medir el tiempo que requiere una onda ultrasónica emitida por un transductor y reflejada en la superficie del líquido, en regresar al transductor. En la siguiente figura podemos ver esquematizado este principio. La relación entre el nivel del líquido, la velocidad de la onda ultrasónica (v) y el tiempo (t) que tarda en recogerse el ultrasonido viene dado por:

$$t \times v = 2(L - H)$$

Siempre que sea aire lo que haya en el "camino" del ultrasonido.

2.1.1.1.4 SENSORES DE PRESIÓN

Los elementos primarios de medición de presión son fundamentalmente de tres tipos:

1. Elementos Mecánicos
2. Elementos Electromecánicos
3. Elementos Electrónicos.

Elementos Mecánicos: estos se subclasifican en dos categorías: a) de medición directa y b) elementos primarios elásticos. Los de medición directa realizan su función, comparando la presión con la fuerza ejercida por una columna de líquido de densidad conocida, entre estos se encontraran: el barómetro de cubeta, el manómetro de tubo U, el manómetro de tubo inclinado, etc.

Los elementos primarios elásticos, miden la presión por deformación que estos sufren por efecto de ella misma. Los más empleados son: el tubo Bourdon, el elemento espiral, el diagrama y el fuelle.

El tubo de Bourdon es un elemento tubular de sección elíptica en forma de anillo casi completo, cerrado por un lado. Al aumentar la presión interna, el tubo tiende a enderezarse y este movimiento es transmitido por otros servomecanismos a una aguja indicadora o a un elemento transmisor.

Los materiales usualmente empleados son: acero inoxidable, aleaciones de cobre u otras aleaciones especiales como el Hastelloy/Monel.

El espiral se forma enrollando un tubo de Bourdon en forma de espiral alrededor de un eje común.

El helicoidal es similar al espiral con la diferencia de que las espiras se encuentran en planos diferentes y paralelos.

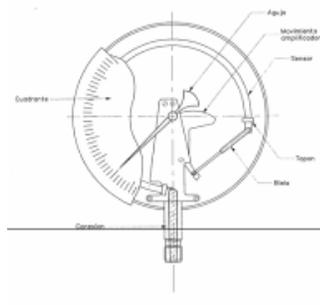


Fig. II-7 Medidor de presión (Manómetro)

2.1.1.2 ACTUADORES.

Son los elementos de los sistemas de control que transforman la salida de un PLC o de un sistema de control en una acción de control para una máquina o dispositivo. Para este caso serán quienes realicen el transporte, dosificación, sellado, etc., que permiten tener un sistema continuo para el envasado volumétrico de los líquidos y que son controlados a través del PLC. Teniendo presente en nuestro proyecto dos tipos de Actuadores:

Actuadores Eléctricos

Actuadores Neumáticos.

2.1.1.2.1 ACTUADORES ELÉCTRICOS

2.1.1.2.1.1 Motores monofásicos de baja velocidad.



Fig. II-8 Motor monofásico de baja velocidad

Los motores monofásicos, como su propio nombre indica son motores con un solo devanado en el estator, que es el devanado inductor. Prácticamente todas las realizaciones de este tipo de motores son con el rotor en jaula de ardilla. Suelen tener potencias menores de 1KW, aunque hay notables excepciones como los motores de los aires acondicionados con potencias superiores a 10KW.

Estos motores son utilizados en aplicaciones que no requieren mayor torque o velocidad, ya que su potencia no es muy alta por el hecho de tener un campo magnético de inducción estático no rotatorio, lo que genera que este no tenga un par de arranque.

Se utilizan fundamentalmente en electrodomésticos, bombas y ventiladores de pequeña potencia, pequeñas máquinas-herramientas, en los mencionados equipos de aire acondicionado, etc.

Se pueden alimentar entre una fase y el neutro o entre dos fases. No presentan los problemas de excesiva corriente de arranque como en el caso de los motores trifásicos de gran potencia, debido a su pequeña potencia, por tanto todos ellos utilizan el arranque directo.

Presentan los siguientes inconvenientes:

- "No arrancan solos", debido a que el par de arranque es cero.
- Se caracterizan por sufrir vibraciones debido a que la potencia instantánea absorbida por cargas monofásicas es pulsante de frecuencia doble que la de la red de alimentación.

2.1.1.2.1.2 Relés (Interruptores mecánicos)

Dispositivos de conmutación utilizados para controlar cargas de gran potencia o de potencia superior a la que el controlador pueda suministrar para su activación, por ejemplo si la salida digital de un controlador puede soportar un máximo de 0.5A, esta puede ser suficiente para accionar la bobina de un relé y a su vez conmutar el estado de la carga de potencia superior, por ejemplo motores, bombas, etc.

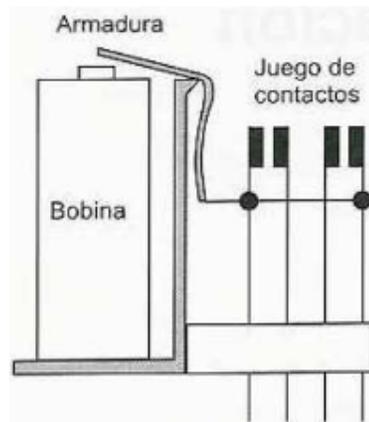


Fig. II-9 Esquema básico de relé.

Responden a una sencilla conmutación de ON/OFF, mediante la excitación de la bobina, al circular una corriente a través de ella. Esta bobina atrae un brazo movable, que es la armadura, la cual produce la apertura o cierre de contactos, por lo general

contienen dos juegos de contactos uno que se cierra con el accionar de la bobina y el otro que se abre respectivamente.

2.1.1.2.2 ACTUADORES NEUMÁTICOS

Los actuadores neumáticos son los elementos del circuito neumático que utilizan la energía del aire comprimido para desarrollar algún trabajo útil (fuerzas o desplazamientos). Los actuadores neumáticos más comunes son los cilindros neumáticos.

Transforman la energía potencial del aire comprimido (presión) en energía mecánica lineal (movimientos de avance y retroceso).

2.1.1.2.2.1 CILINDROS NEUMÁTICOS

El desplazamiento rectilíneo de vaivén es realizado con perfección y sencillez con energía neumática debido a la escasa inercia del aire comprimido. La utilización de la energía eléctrica para hacer este movimiento requiere un costo superior y sobre todo una mayor complejidad.

Los actuadores lineales, también denominados cilindros por tener su carcasa en la mayoría de los casos dicha figura geométrica, se clasifican en dos grupos: cilindros de simple efecto y cilindros de doble efecto, según realicen trabajo mecánico cuando se desplaza el émbolo con su vástago en un sentido solamente o en los dos, respectivamente.

El cilindro neumático normalmente consta de las partes fundamentales que se describen en la Fig.

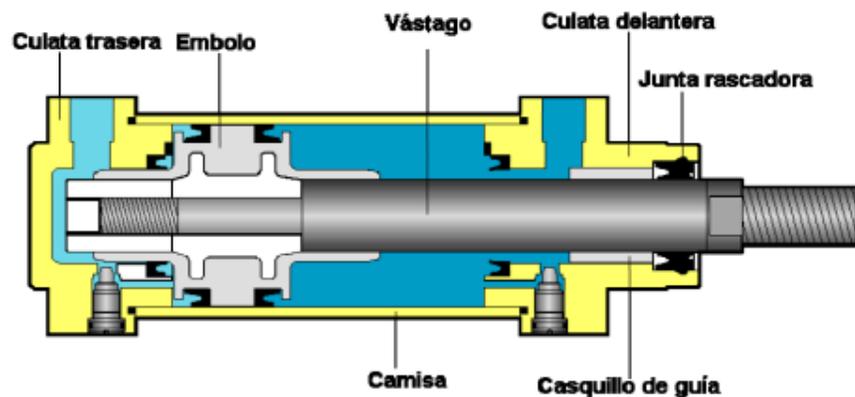


Fig. II-10 Estructura de un cilindro neumático

2.1.1.2.2.1.1 CILINDROS DE SIMPLE EFECTO.

Los cilindros de simple efecto pueden ser de émbolo o de membrana. Los primeros, los más usuales, constan de un cuerpo, generalmente cilíndrico, dentro del cual se desplaza un émbolo o pistón al que va solidariamente unido un vástago. En los cilindros de membrana el émbolo se sustituye por una superficie elástica. En ambos casos se tiene una sola conexión de aire comprimido por dónde entra el aire cuando se quiere hacer salir el vástago y por dónde sale cuando se desea que el vástago regrese. El émbolo o la membrana retornan por el efecto de un muelle incorporado o bien gracias a alguna fuerza externa. Están preparados para que sólo realicen trabajo en un sentido, ya sea entrando o saliendo el vástago. Se requiere aire comprimido sólo para el desplazamiento en un solo sentido.

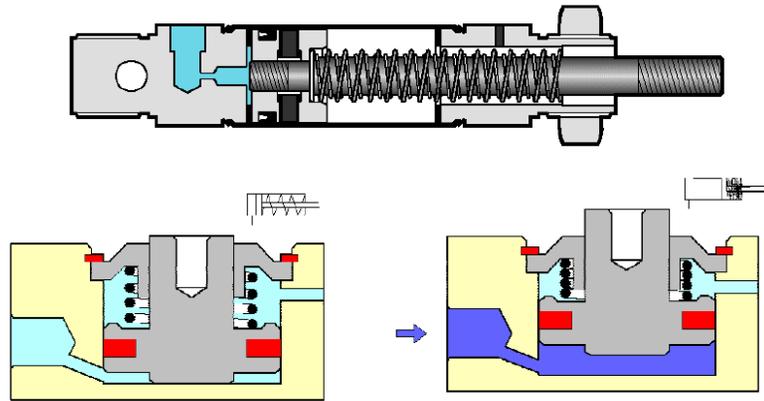


Fig. II-11 Cilindro de simple efecto

El resorte incorporado se calcula de modo que haga regresar el émbolo a su posición inicial a una velocidad suficientemente grande. La longitud de dicho muelle limita la carrera, no sobrepasando normalmente los 80 ó 100 mm. Los cilindros de simple efecto se utilizan principalmente para sujetar, expulsar, apretar, levantar, alimentar, etc. En la Fig. se pueden observar dos cilindros de este tipo así como su representación esquemática según la Norma ISO 1219, la cual simboliza exclusivamente su función, siendo independiente de los detalles constructivos del mismo.

La estanquidad entre las dos caras del émbolo se logra con una junta toroidal de material flexible, que se inserta en el émbolo, entre éste y el propio cilindro metálico.

Durante el movimiento del émbolo, los labios de la junta elastomérica se deslizan sobre la pared interna del cilindro.

En determinadas ocasiones el trabajo lo realiza el muelle en lugar del aire comprimido, mientras que gracias a éste se efectúa el movimiento de retorno.

En la figura.... se representa un actuador lineal de membrana donde una placa de goma, plástico o metal remplaza aquí al émbolo. El vástago, aunque prácticamente carece de él, se fija al centro de la membrana. Se consigue una estanqueidad total puesto que no hay piezas que se deslicen.

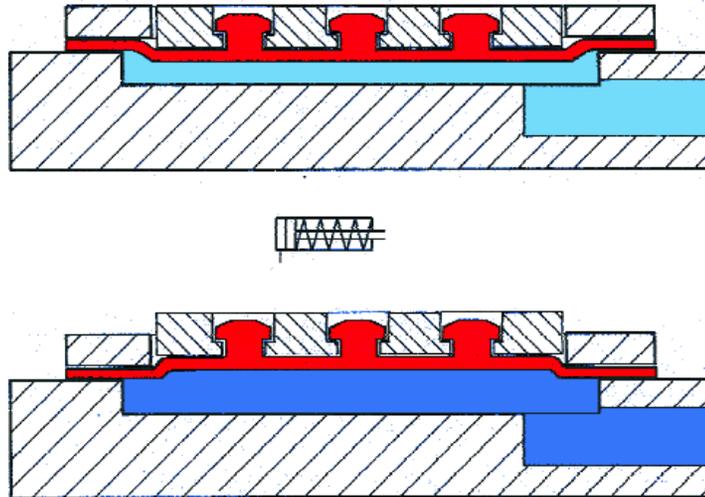


Fig. II-12 Cilindro de membrana

Su carrera está bastante limitada. El esfuerzo puede alcanzar valores importantes en función de la superficie de la membrana. Se emplean en la construcción de dispositivos y herramientas, así como para estampar, remachar, fijar y levantar pesos de cierta importancia.

Existen construcciones especiales que permiten obtener carreras superiores, como es el caso de los cilindros con membrana enrollable.

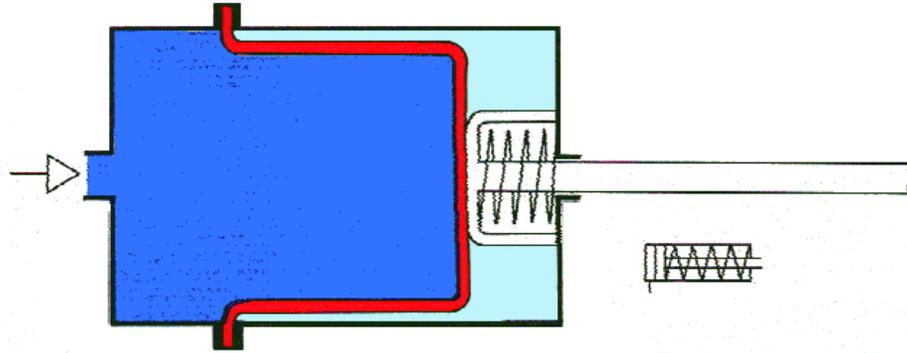


Fig. II-13 Cilindro de membrana arrollable.

2.1.1.2.1.2 CILINDROS DE DOBLE EFECTO.

La fuerza ejercida por el aire comprimido, en los cilindros de doble efecto, impulsa al émbolo en los dos sentidos, por lo que no necesita un muelle para realizar uno de los movimientos. Dispone, por lo tanto, de dos conexiones para la entrada y salida del aire. Realiza una fuerza útil tanto en la ida como en el retorno, pero por efecto del vástago, la sección útil es mayor en una sección que en la otra, por lo que la fuerza realizada será también mayor en un sentido que en el otro.

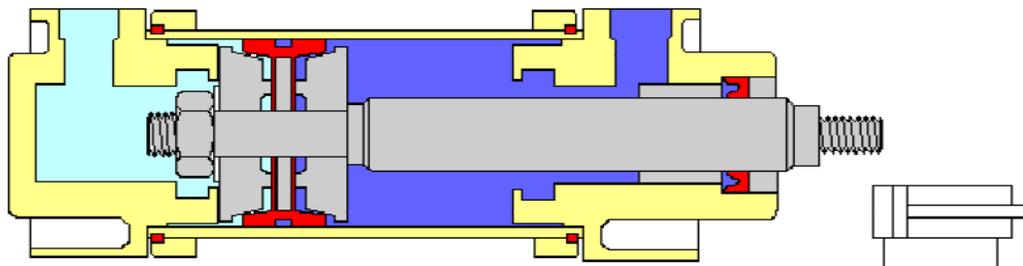


Fig. II-14 Cilindro de doble efecto

Al no llevar muelle pueden alcanzar mayores longitudes de carrera. En este caso la limitación vendrá impuesta por el pandeo y la flexión que pueda sufrir el vástago.

En el émbolo pueden instalarse imanes permanentes que sirvan para detectar sin contacto las posiciones finales del cilindro mediante detectores de proximidad.

Estos cilindros presentan dos entradas de aire comprimido, que hacen que el émbolo pueda ser empujado por el aire en los dos sentidos (avance y retroceso).

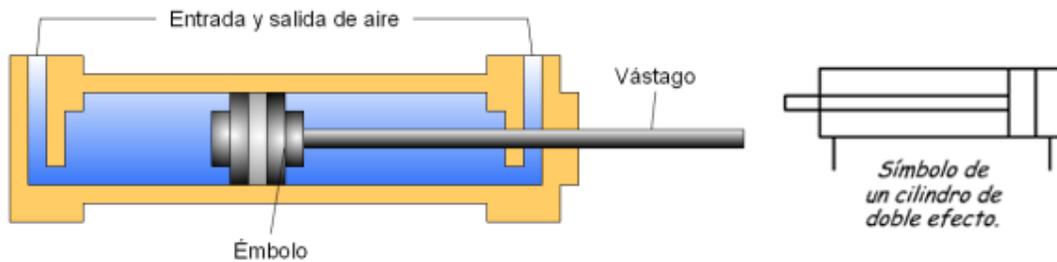


Fig. II-15 Partes del Cilindro de doble efecto

Ventajas: al tener dos tomas de aire puede realizar trabajo útil en ambos sentidos.

Desventajas: doble consumo de aire comprimido (en el avance y en el retroceso).



Fig. II-16 Foto de un cilindro de doble efecto.

2.1.1.2.2.2 CONTROL DEL AIRE COMPRIMIDO: VÁLVULAS.

Se requiere saber cómo controlar el funcionamiento de un circuito neumático (abrir o cerrar el circuito, dirigir el aire por diferentes conductos, ajustar presiones, etc.). De ello se encargan unos elementos neumáticos adicionales: las válvulas.

Las válvulas son dispositivos que controlan el paso del aire comprimido.

2.1.1.2.2.2.1 Parámetros de una válvula.

2.1.1.2.2.2.1.1 Vías y posiciones:

Las válvulas se nombran por el número de vías (orificios de entrada y salida) y por el número de posiciones (estados que puede adoptar, o movimientos que puede realizar).

Ejemplo: válvula 3/2 válvula con 3 vías y 2 posiciones.

2.1.1.2.2.2.1.2 Accionamiento de la válvula:

Un parámetro importante de las válvulas es cómo se accionan: la activación puede ser manual (por pulsador, por pedal, etc.), mecánica (por leva, por final de carrera, etc.), neumática (mediante aire comprimido), o eléctrica (mediante una señal eléctrica que activa un electroimán o un relé).

2.1.1.2.2.1.3 Retorno de la válvula:

Otra característica fundamental es la forma cómo una válvula vuelve a su posición inicial tras la activación. El retorno suele ser por muelle, pero también hay retornos neumático, eléctricos, etc.

2.1.1.2.2.1.4 Simbología de Válvulas

Para representar de forma sencilla una válvula se utilizan símbolos. Los símbolos muestran el funcionamiento de la válvula, pero no sus detalles constructivos internos. Para entender el símbolo de una válvula, hay que seguir estas indicaciones:

Cada posición de la válvula se representa con un cuadrado. Las vías de la válvula se representan por pequeñas líneas en la parte exterior de uno de los cuadrados.

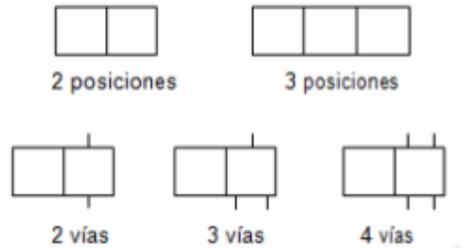


Fig. II-17 Simbología de Posiciones para Válvulas

Dentro de cada cuadrado se representan las conexiones internas entre las distintas vías o tuberías de la válvula, y el sentido de circulación del fluido se representa por flechas.

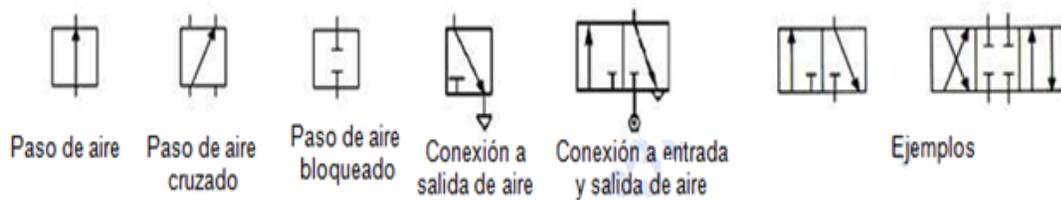


Fig. II-18 Simbología de Vías para Válvulas

En los extremos de los rectángulos se representa el accionamiento y el retorno de la válvula.

El accionamiento permite pasar de la posición de reposo a la posición de trabajo.

El retorno permite pasar de la posición de trabajo a la posición de reposo.

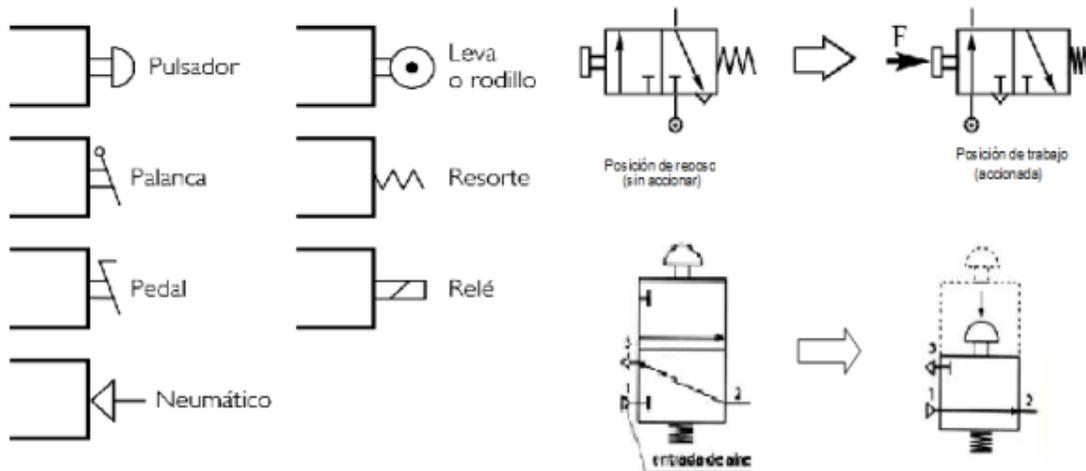
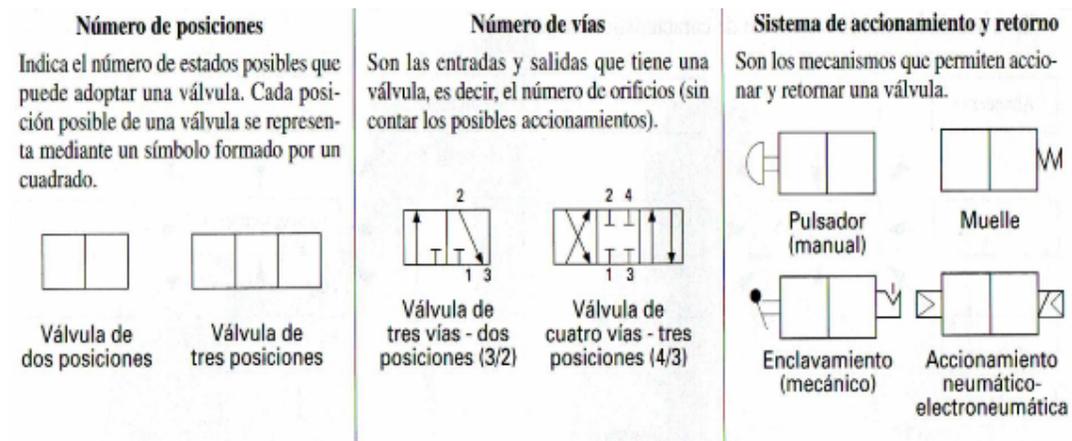


Fig. II-19 Tipos de Accionamientos para Válvulas

2.1.1.2.2.1.5 Designación de una Válvula



2.1.1.2.2.2 VÁLVULAS DISTRIBUIDORAS.

Las válvulas distribuidoras permiten activar o parar un circuito neumático. Su función es dirigir adecuadamente el aire comprimido para que tenga lugar el avance y el retroceso de los cilindros. Por tanto, las válvulas se pueden ver como los interruptores o conmutadores de los circuitos neumáticos.

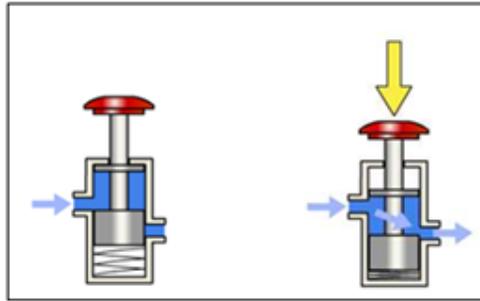
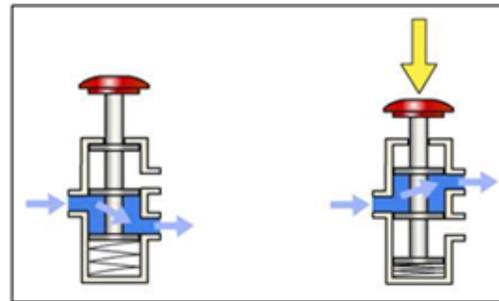


Fig. II-20 Válvula cierra o abre el flujo de aire hacia el cilindro



Válvula que dirige el aire en dos sentidos posibles

Controlando el flujo del aire mediante el accionamiento de una válvula se puede controlar la activación del cilindro.

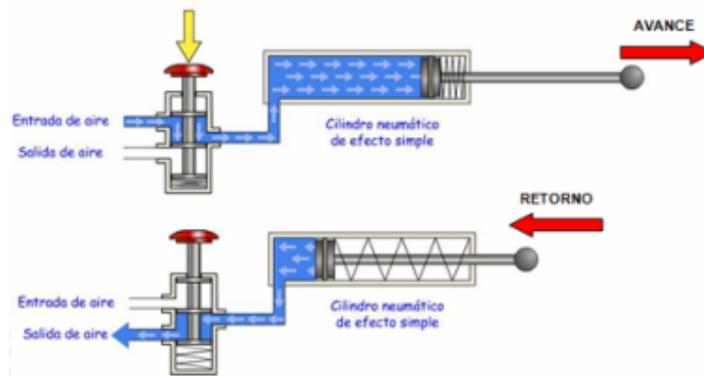


Fig. II-21 Cilindro de simple efecto controlado por una válvula 3/2

2.1.1.2.2.3 VÁLVULAS REGULADORAS DE CAUDAL.

Las válvulas reguladoras de caudal (o reguladoras de flujo) sirven para regular la cantidad de aire que circula por la tubería que llega a un cilindro. Se componen de un tornillo ajustable interno que estrangula el flujo de aire por el tubo. Se utilizan para regular la velocidad de salida y/o de retorno el vástago del cilindro.



Fig. II-22 Válvulas reguladoras de caudal

2.1.1.2.3 ELECTRONEUMÁTICA

La neumática básica o pura produce la fuerza mediante los actuadores o motores neumáticos, lineales o rotativos, pero además el gobierno de éstos y la introducción de señales, fines de carrera, sensores y captadores, se efectúa mediante válvulas exclusivamente neumáticas, es decir el mando, la regulación y la automatización se realiza de manera totalmente neumática.

Pues bien, esta manera de proceder se reserva a circuitos neumáticos muy sencillos y a casos en que, por cuestiones de seguridad, no se pueden admitir elementos eléctricos.

En la electroneumática los actuadores siguen siendo neumáticos, los mismos que en la neumática básica, pero las válvulas de gobierno mandadas neumáticamente son sustituidas por electroválvulas activadas con electroimanes en lugar de pilotadas con aire comprimido. Las electroválvulas son convertidores electroneumáticos que transforman una señal eléctrica en una actuación neumática. Por otra parte los sensores, fines de carrera y captadores de información son elementos eléctricos, con lo que la regulación y la automatización son, por tanto, eléctricas o electrónicas.

Las ventajas de la electroneumática sobre la neumática pura son obvias y se concretan en la capacidad que tienen la electricidad y la electrónica para emitir, combinar, transportar y secuenciar señales, que las hacen extraordinariamente idóneas para cumplir tales fines. Se suele decir que la neumática es la fuerza y la electricidad los nervios del sistema.

Teniendo en cuenta lo anterior se puede definir la electroneumática como la tecnología que trata sobre la producción y transmisión de movimientos y esfuerzos mediante el aire comprimido y su control por medios eléctricos y electrónicos.

La electroneumática es un paso intermedio entre la neumática básica y los autómatas programables, donde éstos por sí solos controlan el sistema con las ventajas singulares que conllevan.

No es estrictamente necesario saber electricidad y electrónica para entender la electroneumática, pues basta tomar los elementos eléctricos como cajas negras, de los que se conoce que con unos determinados estímulos proporciona unas respuestas concretas, es decir que ciertas entradas producen tales salidas. Sin embargo saber electricidad y electrónica es extraordinariamente útil pues la electroneumática es una

simbiosis donde se mezcla la neumática y la automática, con cierta preponderancia de ésta sobre aquella.

En la electroneumática la energía eléctrica (energía de mando y de trabajo) es introducida, procesada y cursada por elementos muy determinados. Por razones de simplicidad y vistosidad estos elementos figuran en los esquemas como símbolos que facilitan el diseño, la instalación y el mantenimiento.

Pero no es suficiente sólo la comprensión de los símbolos existentes en los esquemas de los circuitos electroneumáticos y el funcionamiento de los elementos que en él figuran para garantizar el correcto dimensionado de mandos y la rápida localización de errores o anomalías cuando aparecen, sino que el especialista en mandos debe conocer también las cuestiones y elementos más importantes y usuales de la electricidad y la electrónica.

Un sistema electroneumático consta de un circuito neumático simple y en paralelo circuitos eléctricos, en ocasiones bastante complejos, donde tiene una gran importancia la forma de representación de cada elemento.

2.1.2 PLC (Programmer Logic Control).

El Controlador Lógico Programable (PLC, por sus siglas en inglés), es un dispositivo, capaz de controlar un proceso industrial secuencial en tiempo real, mediante instrucciones que han sido previamente programadas y cargadas en la memoria del dispositivo.

Como se hace referencia en el párrafo anterior es necesario desarrollar un programa que permita al PLC, tomar decisiones según la lógica del mismo, tomando como datos de entrada al sistema, los sensores y captadores que llevan información básica del proceso, tales como posición de transportadores, velocidad de motores, presencia de objetos del proceso, nivel de líquidos, etc. Dependiendo del tipo de sensores que hayamos ubicado para el control del mismo.

El objetivo principal de la utilización de los controladores lógicos programables es el de reemplazar sistemas complejos que ocupan un gran espacio, desarrollados mediante lógica cableada, contactores, relés, secuenciadores, etc., ahorrando así la lógica constructiva, por una lógica de programación mas sencilla, y que cumple las mismas funciones que la implementación con relés y contactores, y que además aporta un sistema de seguridades y mensajería inteligente que ayuda al control del proceso.

Adicionalmente el PLC presenta la ventaja de ayudarnos a construir sistemas escalables, es decir ampliar el sistema sin la necesidad de realizar grandes cambios en el hardware sino más bien en la lógica del programa mediante la introducción de nuevas instrucciones que permitan a este controlar los nuevos periféricos colocados para esta acción.

Ahora el controlador actúa de tal forma en que al obtener las señales de entrada, (sensores, captadores, transductores), realiza un análisis de las mismas y mediante la lógica introducida en la memoria del dispositivo, toma las acciones necesarias y actualiza los módulos de salida, para que según estos se enciendan o se apaguen los actuadores.

En la figura 2.1 podemos visualizar de mejor forma la estructura básica de un PLC.

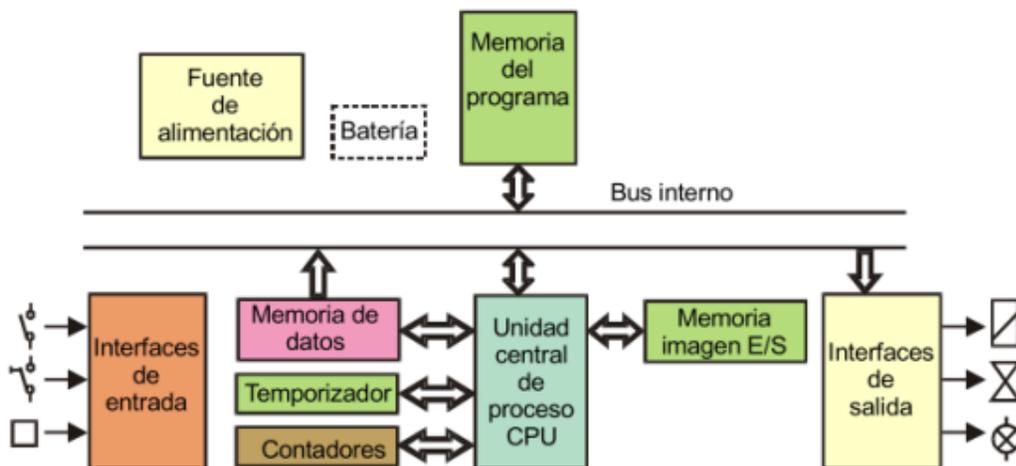


Fig. II-23 Estructura básica de un PLC

2.1.2.1 FUNCIONES BÁSICAS.

- **Detección:** Lectura de la señal de los captadores distribuidos por el sistema de fabricación.
- **Mando:** Elaborar y enviar las acciones al sistema mediante los accionadores y preaccionadores.
- **Diálogo** hombre máquina: Mantener un diálogo con los operarios de producción, obedeciendo sus consignas e informando del estado del proceso.

- **Programación:** Para introducir, elaborar y cambiar el programa de aplicación del autómeta. El diálogo de programación debe permitir modificar el programa incluso con el autómeta controlando la máquina.

2.1.2.2 ESTRUCTURA FISICA

Los PLCs actuales se pueden clasificar en función de dos criterios. Por un lado, según su configuración externa, la cual reflejará de qué manera y dónde están colocados los distintos elementos que componen el autómeta programable. Por otro lado, se distinguirán según su estructura interna, que reflejará cuáles son dichos componentes. El término estructura externa hace referencia al aspecto físico del PLC, como son sus elementos, en cuántos está dividido, etc. Actualmente son tres las estructuras más significativas del mercado:

2.1.2.2.1 Estructura Compacta.

Los autómetas que siguen esta estructura se caracterizan por llevar en un solo bloque todos sus elementos (fuente de alimentación, CPU, E/S,...). Generalmente son autómetas de gama baja o nano-autómetas. Su potencia de proceso suele ser muy limitada dedicándose a controlar maquinarias o procesos pequeños, aunque en la actualidad estos se han ido convirtiendo en autómetas de estructura semimodular, ya que han adquirido la potencialidad de implementar módulos de expansión de entradas/salidas binarias y analógicas. En la figura 2.2 se presentan ejemplos de autómetas de estructura compacta de diferentes marcas.



Fig. II-24 Ejemplos de PLC's compactos

2.1.2.2.2 Estructura Semimodular.

Se caracteriza por separar las E/S del resto del autómeta. Así pues, en un bloque compacto encontraremos la CPU, la memoria y la fuente de alimentación. Podríamos generalizar diciendo que los autómetas de gama media suelen responder a una estructura semi-modular.

2.1.2.2.3 Estructura Modular.

Existe un módulo para cada uno de los componentes del PLC. La sujeción de los mismos se hace mediante carril normalizado, placa perforada o sobre RACK, en donde va alojado el bus externo de unión de los distintos módulos que lo componen. Podemos generalizar diciendo que son autómatas de gama alta. El PLC utilizado en este proyecto corresponde a esta estructura.

En cuanto a la estructura interna, distinguiremos entre los componentes básicos y los necesarios para que el sistema sea operativo.

De los componentes básicos destacamos la CPU y las Entradas y Salidas. Teóricamente con dichos elementos ya se tendría un autómata, pero es intuitivo ver que sin otros componentes tales como una fuente de alimentación, los primeros no son operativos. La CPU es la parte inteligente del sistema e interpretará las instrucciones del programa usuario. Estará compuesta por un Procesador, una Memoria, y por Circuitos auxiliares (como una ALU, los Flags, el Contador de Programa, etc.).

Otros componentes como la Fuente de Alimentación, las interfaces de las E/S o los dispositivos periféricos complementan la estructura interna de un PLC y le hacen adoptar cierta operatividad, en muchos casos adaptable al tipo de proceso a controlar.



Fig. II-25 PLC de estructura modular

Una vez que ya hemos conocido las principales características de los controladores lógicos programables es hora de centrarnos en el PLC que vamos a utilizar en el proyecto.

2.1.2.3 EL S7-1200 DE SIEMENS

La CPU S7-1200 es un potente controlador que incorpora una fuente de alimentación y distintos circuitos de entrada y salida integrados. Para comunicarse con una proramadora, la CPU incorpora un puerto PROFINET integrado. Esta puede

comunicarse con paneles HMI o una CPU diferente en la red PROFINET. Para garantizar seguridad en la aplicación, todas las CPUs S7-1200 disponen de protección por contraseña, que permite configurar el acceso a sus funciones.

2.1.2.3.1 CICLO DE OPERACIÓN

En cada ciclo se escribe en las salidas, se leen las entradas, se ejecutan las instrucciones del programa de usuario y se realiza el mantenimiento del sistema o procesamiento en segundo plano. En inglés, el ciclo también se llama "scan cycle" o "scan". En condiciones estándar, todas las E/S digitales y analógicas se actualizan de forma síncrona con el ciclo, utilizando un área de memoria interna denominada memoria imagen de proceso. La memoria imagen de proceso contiene una instantánea de las entradas y salidas físicas de la CPU, de la Signal Board y de los módulos de señales.

La CPU lee las entradas físicas inmediatamente antes de ejecutar el programa de usuario y almacena los valores de entrada en la memoria imagen de proceso de las entradas. Así se garantiza que estos valores sean coherentes durante la ejecución de las instrucciones programadas.

La CPU ejecuta la lógica de las instrucciones programadas y actualiza los valores de salida en la memoria imagen de proceso de las salidas, en vez de escribirlos en las salidas físicas reales.

Tras ejecutar el programa de usuario, la CPU escribe las salidas resultantes de la memoria imagen de proceso de las salidas en las salidas físicas.

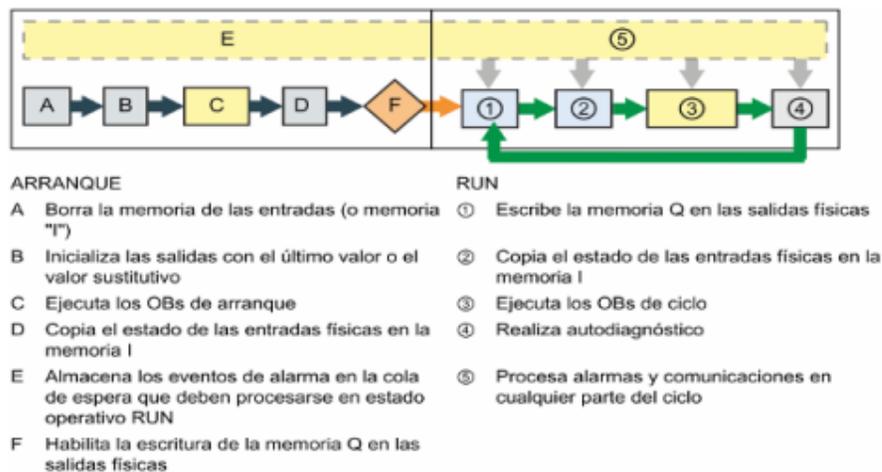


Fig. II-26 Ciclo de ejecución del PLC

2.1.2.3.2 ESTADOS OPERATIVOS

El PLC tiene tres estados operativos, a saber: STOP, ARRANQUE y RUN. Los LEDs de estado en el frente de la CPU indican el estado operativo actual.

- En estado operativo STOP, la CPU no ejecuta el programa. Entonces es posible cargar un proyecto en la memoria del PLC
- En el modo ARRANQUE, se ejecuta cualquier lógica de arranque existente. Los eventos de alarma no se procesan durante el modo de arranque.
- El ciclo se ejecuta repetidamente en estado operativo RUN. Los eventos de alarma pueden ocurrir y procesarse en cualquier fase del ciclo del programa.

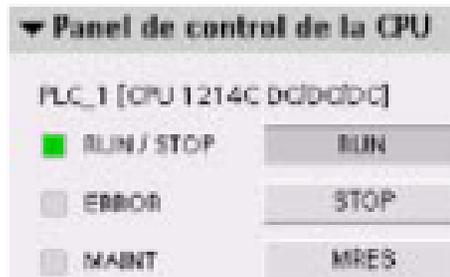


Fig. II-27 Estados operativos del PLC

2.1.2.3.3 ÁREAS DE MEMORIA, DIRECCIONAMIENTO Y TIPOS DE DATOS

La CPU provee las áreas de memoria siguientes para almacenar el programa de usuario, los datos y la configuración:

- La memoria de carga permite almacenar de forma no volátil el programa de usuario, los datos y la configuración. Cuando un proyecto se carga en la CPU, se almacena primero en el área de memoria de carga. Esta área se encuentra bien sea en una Memory Card (si está disponible) o en la CPU. Esta área de memoria no volátil se conserva incluso tras un corte de alimentación. La Memory Card ofrece mayor espacio de almacenamiento que el integrado en la CPU.
- La memoria de trabajo ofrece almacenamiento volátil para algunos elementos del proyecto mientras se ejecuta el programa de usuario. La CPU copia algunos elementos del proyecto desde la memoria de carga en la memoria de

trabajo. Esta área volátil se pierde si se desconecta la alimentación. La CPU la restablece al retornar la alimentación.

- La memoria remanente permite almacenar de forma no volátil un número limitado de valores de la memoria de trabajo. El área de memoria remanente se utiliza para almacenar los valores de algunas posiciones de memoria durante un corte de alimentación. Si ocurre un corte de alimentación, la CPU dispone de suficiente tiempo de retención para respaldar los valores de un número limitado de posiciones de memoria definidas. Estos valores remanentes se restablecen al retornar la alimentación.

Una Memory Card SIMATIC opcional proporciona una memoria alternativa para almacenar el programa de usuario, así como un medio para transferir el programa. Si se utiliza una Memory Card, la CPU ejecutará el programa desde allí y no desde la memoria de la CPU.

La CPU sólo soporta una SIMATIC Memory Card preformateada.

2.1.2.3.4 TIPOS DE DATOS SOPORTADOS POR EL S7-1200

Los tipos de datos se utilizan para determinar el tamaño de un elemento de datos y cómo deben interpretarse los datos. Todo parámetro de instrucción soporta como mínimo un tipo de datos. Algunos parámetros soportan varios tipos de datos. Sitúe el cursor sobre el campo de parámetro de una instrucción para ver qué tipos de datos soporta el parámetro en cuestión.

Tabla II-II Tipos de datos soportados por el S7-1200

Tipos de datos	Tamaño (Bits).	Rango	Ejemplo de entradas de constantes.
Bool	1	0 a 1	True, False, 0, 1
Byte	8	16#00 a 16#FF	16#12, 16#AB
Word	16	16#0000 a 16#FFFF	16#ABCD, 16#0001
DWord	32	16#00000000 a 16#FFFFFFFF	16#02468ACE
Char	8	16#00 a 16#FF	'A', 't', '@'

Sint	8	-128 a 127	123, -123
Int	16	-32.768 a 32,767	123, -123
DInt	32	-2.147.483.648 a 2.147.483.647	123, -123
USInt	8	0 a 255	123
UInt	16	0 a 65.535	123
UDInt	32	0 a 4.294.967.295	123
Real	32	+/-1,18 x 10 ⁻³⁸ a +/-3,40 x 10 ³⁸	123.456
LReal	64	+/-2,23 x 10 ⁻³⁰⁸ a +/-1,79 x 10 ³⁰⁸	12345.123456789 -1.2E ⁺⁴⁰
Time	32	T#-24d_20h_31m_23s_648ms to T#24d_20h_31m_23s_647ms Almacenado como: -2,147,483,648 ms to +2,147,483,647 ms	T#5m_30s 5#-2d T#1d_2h_15m_30x_45 ms
String	Variable	0 a 254 caracteres en tamaño de byte	'ABC'
DTL	12 bytes	Mínima: DTL#1970-01-01-00:00:00.0 Máxima: DTL#2554-12-31-23:59:59.999 999 999	DTL#2008-12-16- 20:30:20.250

El tipo de datos DTL es una estructura de 12 bytes que almacena información de fecha y hora en una estructura predefinida. Un DTL se puede definir en la memoria temporal del bloque o en un DB.

Aunque no están disponibles como tipos de datos, las instrucciones de conversión soportan el siguiente formato numérico BCD.

Tabla II-III Datos adicionales para el S7-1200

Tipos de datos	Tamaño (bits).	Rango	Ejemplos
BCD16	16	-999 a 999	123, -123
BCD32	32	-9999999 a 9999999	1234567, -1234567

2.1.2.3.5 ÁREAS DE MEMORIA Y DIRECCIONAMIENTO

STEP 7 Basic facilita la programación simbólica. Se crean nombres simbólicos o "variables" para las direcciones de los datos, ya sea como variables PLC asignadas a direcciones de memoria y E/S o como variables locales utilizadas dentro de un bloque lógico. Para utilizar estas variables en el programa de usuario basta con introducir el nombre de variable para el parámetro de instrucción. Para una mejor comprensión de cómo la CPU estructura y direcciona las áreas de memoria, los siguientes párrafos explican el direccionamiento "absoluto" al que se refieren las variables PLC. La CPU ofrece varias opciones para almacenar datos durante la ejecución del programa de usuario:

- Memoria global: La CPU ofrece distintas áreas de memoria, incluyendo entradas (I), salidas (Q) y marcas (M). Todos los bloques lógicos pueden acceder sin restricción alguna a esta memoria.
- Bloque de datos (DB): Es posible incluir DBs en el programa de usuario para almacenar los datos de los bloques lógicos. Los datos almacenados se conservan cuando finaliza la ejecución del bloque lógico asociado. Un DB "global" almacena datos que pueden ser utilizados por todos los bloques lógicos, mientras que un DB instancia almacena datos para un bloque de función (FB) específico y está estructurado según los parámetros del FB.
- Memoria temporal: Cada vez que se llama un bloque lógico, el sistema operativo de la CPU asigna la memoria temporal o local (L) que debe utilizarse durante la ejecución del bloque. Cuando finaliza la ejecución del bloque lógico, la CPU reasigna la memoria local para la ejecución de otros bloques lógicos. Toda posición de memoria diferente tiene una dirección unívoca. El programa de usuario utiliza estas direcciones para acceder a la información de la posición de memoria.

Las referencias a las áreas de memoria de entrada (I) o salida (Q), como I0.3 o Q1.7, acceden a la memoria imagen del proceso. Para acceder inmediatamente a la entrada o salida física es preciso añadir ":P" a la dirección (p.ej. I0.3:P, Q1.7:P o "Stop:P"). El forzado permanente escribe un valor en una entrada (I) o una salida (Q). Para forzar permanentemente una entrada o salida, agregue una ":P" a la variable PLC o dirección. Para más información, consulte "Forzar valores permanentemente en la CPU".

Tabla II-IV Direccionamiento de datos

Área de Memoria	Descripción	Forzado Permanente	Remanente
I Memoria de imagen de proceso de las entradas I_:P (entrada física)	Se copia de las entradas físicas al inicio del ciclo.	NO	NO
	Lectura inmediata de entradas físicas de la CPU, SB y CM	SI	NO
Q Memoria de imagen de proceso de salidas Q_:P (salida física)	Se copia las salidas físicas al inicio del ciclo.	NO	NO
	Escritura inmediata en las salidas físicas de las CPU, SB y SM	Si	NO
M Área de marcas	Control y Memoria de datos	NO	SI (opcional)
L Memoria temporal.	Datos locales temporales de un bloque.	NO	NO
DB Bloque de datos	Memoria de datos y parámetros de FBs	NO	SI (opcional)

Toda posición de memoria diferente tiene una dirección unívoca. El programa de usuario utiliza estas direcciones para acceder a la información de la posición de memoria. La figura muestra cómo acceder a un bit (lo que también se conoce como direccionamiento "byte.bit"). En este ejemplo, el área de memoria y la dirección del byte (M = marca y 3 = byte 3) van seguidas de un punto (".") que separa la dirección del bit (bit 4).

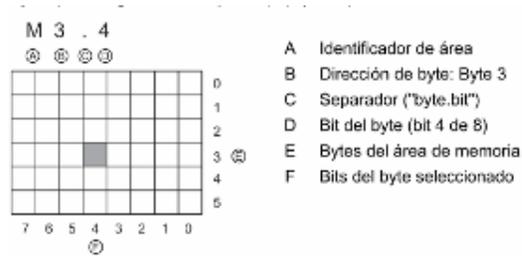


Fig. II-28 Esquema de mapa de memoria

2.1.3 SISTEMA DE SUPERVISIÓN INTOUCH

InTouch es un paquete de software utilizado para crear aplicaciones de interface hombre-máquina bajo entorno PC. InTouch utiliza como sistema operativo el entorno WINDOWS 95/98/NT/2000. El paquete consta básicamente de dos elementos: WINDOWMAKER y WINDOWVIEWER. WINDOWMAKER es el sistema de desarrollo. Permite todas las funciones necesarias para crear ventanas animadas interactivas conectadas a sistemas de e/s externos o a otras aplicaciones WINDOWS. WINDOWVIEWER es el sistema runtime utilizado para rodar las aplicaciones creadas con WINDOWMAKER.

En cualquier pantalla de WINDOWMAKER disponemos de una ayuda sensitiva pulsando la tecla F1.

2.1.3.1 Requerimientos del Sistema

- Cualquier PC compatible IBM con procesador Pentium 200 MHz o superior
- Mínimo 500 Mb de disco duro
- Mínimo 64 Mb RAM
- Adaptador display SVGA (recomendado 2 Mb mínimo)
- Puntero (mouse, trackball, touchscreen)
- Adaptador de red
- Microsoft Windows W95/98 SE o NT

2.1.3.2 Instalación

InTouch dispone de un sencillo programa de instalación que además detecta el sistema operativo sobre el que el programa se va a instalar. El CD-ROM dispone de un autoarranque.

Una vez instalado el paquete InTouch, ya podemos crear una aplicación. Para ello, es necesario entrar en InTouch desde WINDOWS pinchando dos veces con el ratón sobre el símbolo de InTouch. En su monitor aparecerá la siguiente pantalla:

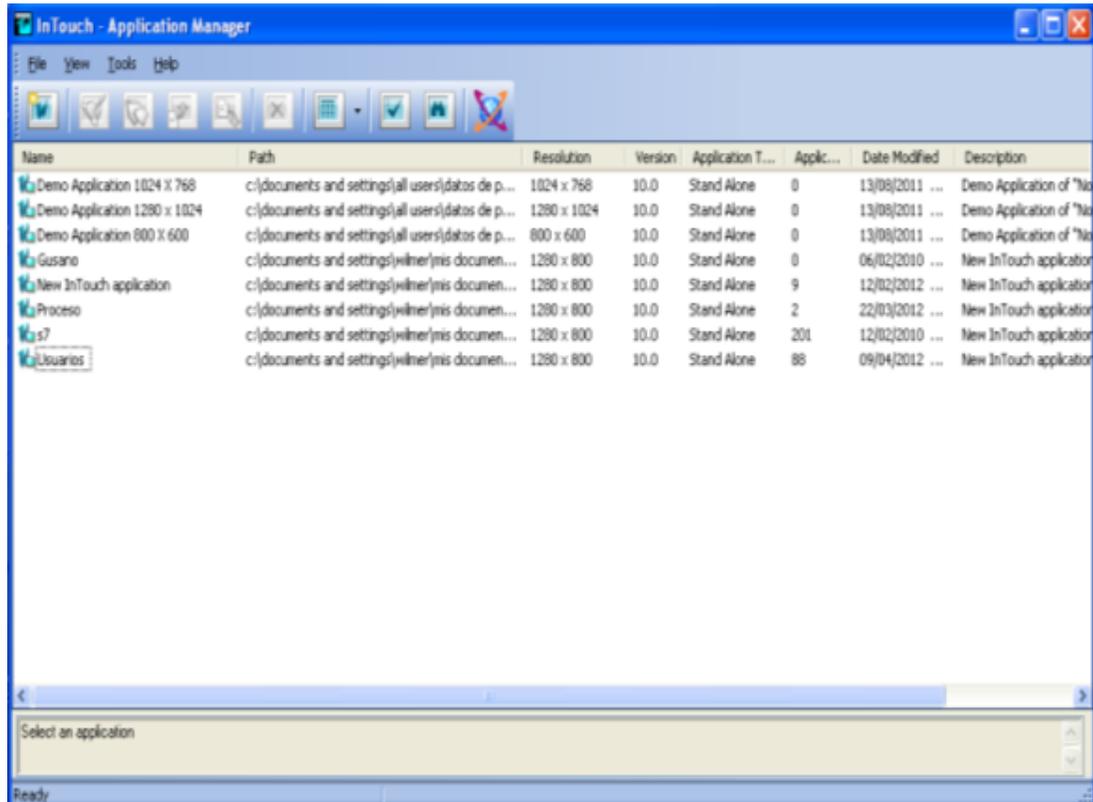


Fig. II-29 Pantalla inicial de la aplicación InTouch

Esta es la pantalla principal de Application Manager para la entrada a InTouch. Desde aquí podemos seleccionar cualquiera de las aplicaciones de nuestro ordenador previamente creadas, o bien crear una aplicación nueva. Para ello, seleccione FILE NEW para acudir al asistente de generación de aplicaciones, que le permitirá además dar un nombre y comentario a la nueva aplicación creada (muy útil tanto para documentación posterior, como para selección desde el application manager). InTouch volverá a la pantalla principal de Application Manager y mostrará en la lista el subdirectorio aplicación CURSILLO. Observe estos dos iconos en la barra de herramientas de Application Manager

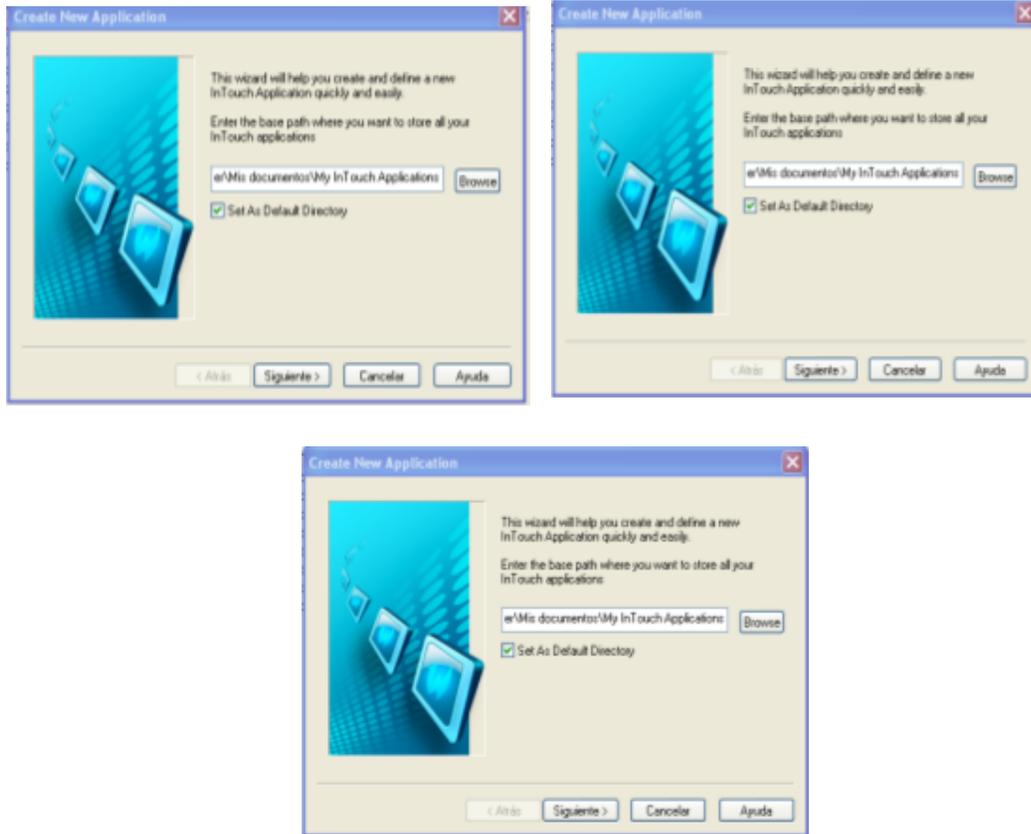


Fig. II-30 Creacion de la aplicacion

Tras pulsar Finalizar, la nueva aplicación quedará en la lista de aplicaciones de Application Manager. Ya dispone del icono de WindowMaker activado para poder crear la aplicación. InTouch se encarga de crear la carpeta con el nombre de la aplicación.



Este es el icono de WINDOWMAKER o creador de aplicaciones. Una vez seleccionada la aplicación que desea crear o modificar, pinche sobre este icono para llevar a cabo su trabajo.



Este es el icono de WINDOWVIEWER o runtime. Una vez la aplicación que desea monitorizar, pinche sobre este icono. Esta aplicación debe haber sido previamente creada, por lo que este icono no estará accesible cuando seleccione una nueva aplicación.

Al pinchar sobre el icono de WindowMaker, InTouch creará automáticamente un subdirectorio con este nombre, e incluirá en él los ficheros de trabajo.

Cuando se crea un directorio, tenemos la opción de hacer varias pantallas dentro de éste, al seleccionar el comando New Windows en el menú File aparece el cuadro Windows Properties (propiedad de pantalla), tal como se indica a continuación:

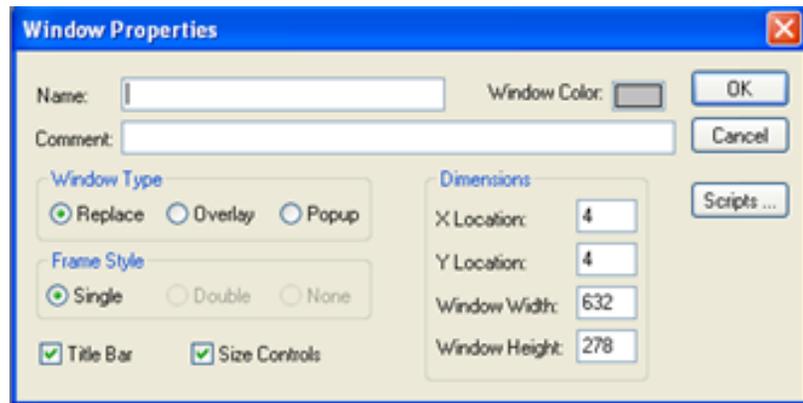


Fig. II-31 Creación y definición de parámetros de una ventana

Donde se coloca el nombre de la pantalla con un máximo de 32 caracteres, se selecciona el color de fondo de la pantalla en Window Color, se definen las dimensiones de la pantalla, también se define la visibilidad o no del título de pantalla. Luego de terminadas las propiedades de pantalla se presiona OK y se comienza a trabajar en la confección de la pantalla.

Cuando se está dentro de una nueva pantalla aparece la barra de herramientas Toolbox, es una caja de herramientas con una colección de objetos gráficos que son utilizados en la aplicación. La barra de herramientas queda normalmente visible al abrir Window Maker.

La barra de herramientas es tal como se muestra a continuación:



Fig. II-32 Barra de herramientas

Esta barra brinda herramientas que son usadas para crear objetos gráficos, escribir nombres, seleccionar objetos, etc. Los cuales pueden ser mostrados o animados en el modo de trabajo Runtime.

La herramienta Wizard  al seleccionarla aparece un cuadro con elementos básicos que se utilizan para hacer eficiente y rápido la construcción de las pantallas.



Fig. II-33 Submenú Symbol Factory de la herramienta Wizard

Basta con hacer doble click en el objeto seleccionado para que se de la opción de colocarlo donde se requiera en la creación de la pantalla, también se pueda variar su tamaño y dar Configuración de acuerdo a lo requerido en la construcción de la pantalla.

Encontrándose también en está cuadro opciones de gráficos en tiempo real, históricos y ventanas de alarmas.

2.1.3.3 MENU WINDOWMAKER

WindowMaker contiene una barra de menú provista con numerosas funciones. Esta barra de menú esta localizada en la parte superior de la pantalla y se puede acceder a ella solamente haciendo click sobre la opción que se desea utilizar.

2.1.3.3.1 Menú File (archivo)

El menú archivo contiene un Set de comandos que se utilizan para crear, abrir, salvar, imprimir, cerrar, borrar, exportar e importar archivos, también se cuenta con Exit que es salir del programa.

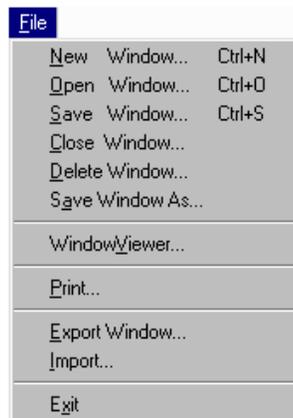


Fig. II-34 Submenú File

Además cuenta con **WindowViewer** que es un switch para cambio de WindowMaker a WindowViewer (pantalla de simulación).

2.1.3.3.2 Menú Arreglar

El menú Arreglar contiene un conjunto herramientas tales como rotar en sentido del reloj y en contra, transformar el objeto seleccionado en imagen de espejo vertical u horizontal, combina varios objetos seleccionados en uno solo o viceversa, habilitar o deshabilitar grid (grilla). Para usar cualquier de los comandos del menú arreglar, seleccione él o los objetos donde el comando va a ser aplicado y entonces seleccionado el comando desde el menú arreglar se aplica. Usted puede configurar el Toolbox para mostrar herramientas del menú arreglar. Muchas de estas herramientas se incluyen en el toolbox.

El Toolbox se configura por medio de Special / Configure.



Fig. II-35 Submenú arranque

2.1.3.3.3 Menú de Texto

El Menú de Texto es parte del Toolbox por lo tanto se maneja directamente de la barra de herramientas.

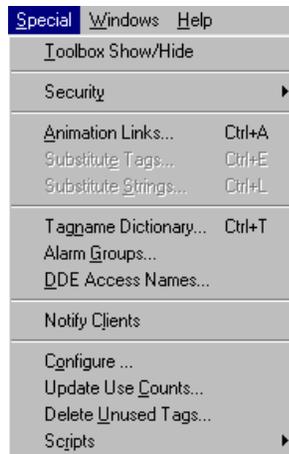
2.1.3.3.4 Menú de Línea

Los comandos de Menú de Línea se usan para cambiar el estilo y la anchura de líneas que son usados sobre objetos individualmente. Cualquier objeto seleccionado que contiene una línea (es decir, círculos, cuadrados, polígonos, líneas, etc.) será modificado. Después de seleccionar el o los objetos, y haciendo click sobre el tipo de línea deseado que muestra en el menú de línea, esta será cambiada.

2.1.3.3.5 Menú Especial

El Menú Especial contiene un conjunto de comandos y subcomandos permitiendo desempeñar funciones especiales tal como sustituir tagnames, strings, acceder al Diccionario de Tagname, cambiar los nombres DDE, etc. En este menú se encontraran comandos tales como mostrar y quitar el Toolbox de la pantalla (Show/hide toolbox), Vínculos de Animación (Animation Links), sustitución de tagnames, sustitución de string que son utilizados para cambiar el contenido de los String (botoneras), llamar el diccionario de definiciones de tagnames, crear nombres de grupos de alarmas o modificar viejos nombres de grupos, también se pueden crear subgrupos de alarmas, seleccionar el nombre de accesos DDE (Dynamic Data Exchange).

También existe el comando Configurar con el cual Intouch, provee la habilidad de personalizar completamente la funcionalidad y aspecto final de la aplicación seleccionando varias opciones. Estas opciones se obtienen desde este comando. Por ejemplo, puede colocarse las opciones que impidan al usuario salir de WindowViever, la barra de título puede personalizarse para mostrar el nombre de compañía, la barra de menú puede eliminarse, la tecla ALT puede estar incapacitado, etc.



**Fig. II-36 Submenú
Special**

El comando borrar tag inutilizados, permite mostrar una lista de todos los tagnames inutilizados en el Diccionario de Datos. Todos estos tagnames pueden borrarse desde la base de datos. Además se incluye una lista de Script que son editores de lógica, los cuales pueden ser usados por los usuarios. Dependiendo de que Script se edita, la lógica puede implementarse (Aplicación Script), cuando se selecciona el comando Script, los Scripts de Lógica son utilizados para crear simulaciones, contraseña de protección, cálculos de sistemas de variables o cambio de ventanas debido a los cambios en el proceso de variables, etc. Más adelante se profundizará más sobre los Scripts.

2.1.3.3.6 Menú de Ventana

El menú de Ventanas contiene las Propiedades de Ventana y una lista de todas las pantallas que se encuentran abiertas. Haciendo Click sobre cualquier de los nombres de las pantallas hará que las propiedades de la pantalla se activen.

2.1.3.4 Diccionario Tagname

El diccionario de datos tagname es el corazón de Intouch. En orden crea la rutina de la base de datos, Intouch requiere información aproximadamente de todas las entradas/tags que son creados. Cada entrada se le debe asignar un tagname. Un tagname es un nombre simbólico que se entra en el Diccionario de Tagname. Este nombre simbólico puede entonces configurar valores min., max., alarmas, etc. y también se define como un tipo específico, por ejemplo, un tag DDE. Este tag DDE

puede llegar a ser entonces un link entre Intouch, el servidor de I/O y el mundo real. El Diccionario de Tagname es el mecanismo usado para ingresar esta información sobre la variables/entradas en la base de datos. En el modo Runtime, contiene el valor actual de todos los artículos en la base de datos.

La creación de la base de datos o el Diccionario de Datos puede realizarse usando tres diferentes métodos. El primer método es la creación de un manual donde usted accesa el tagname de diccionario y define cada tag individualmente hasta completar la base de datos. El segundo método es el método automático donde usted crea un objeto gráfico, asignando vínculos de animación al objeto y el sistema rápidamente define el tagname. El tercer método es el método externo donde usted usa el utilitario de Wonderware DB de Basurero y DB carga para transferir la base de datos desde una aplicación de InTouch a otra aplicación de InTouch.

Si usted escoge el método automático o manual para crear la base de datos, aparece el cuadro de dialogo "Diccionario - TagName de Definición". En el método automático el cuadro de dialogo aparecerá una vez que se presiona OK, cuando pide definir los tagnames, asignando después los vínculos de animación del objeto. Para acceder al cuadro de diálogo cuando se usa el método manual de creación, primero se selecciona Special/Tag Name Dictionary

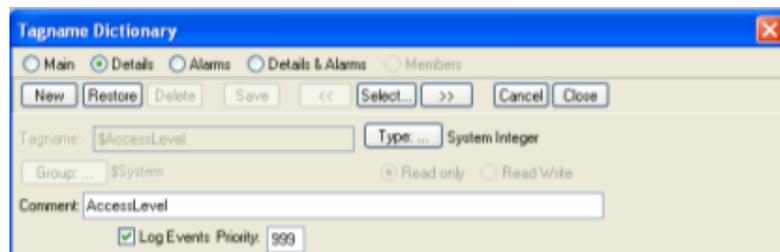


Fig. II-37 Tagname Dictionary

2.1.3.4.1 Details

Seleccione este botón para mostrar el cuadro de diálogo usado para ingresar los detalles del tipo específico de tag. El inicialmente cuadro de dialogo Dictionary – Tag Name Definition es usado para ingresar la información básica con respecto a un tagname. Muchos puntos, especialmente entradas y salidas, requieren grandes detalles. Para cada tipo de tagname especificado, existe un cuadro de dialogo “Details” específico para ingresar el detalle del tipo de tagname. Cuando un

tipo de tagname es seleccionado, aparece automáticamente un cuadro de diálogo "Details" respectivamente. Si el cuadro de diálogo "Details" para la definición del tagname actualmente mostrado en pantalla no aparece, seleccione Details en la parte superior del cuadro.

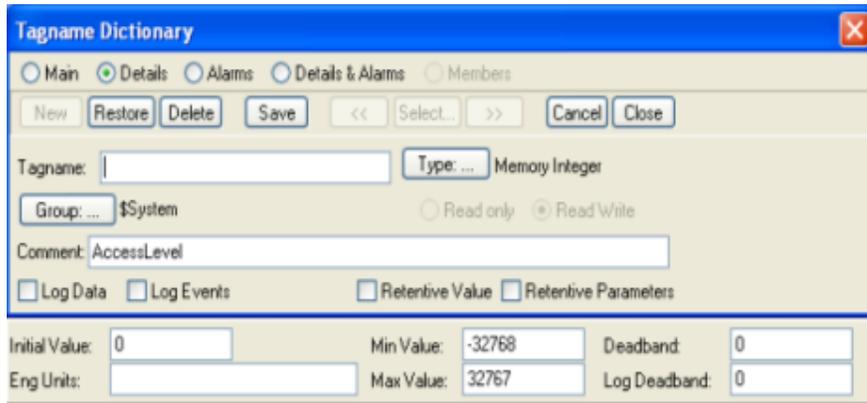


Fig. II-38 Cuadro de Diálogo Details

2.1.3.4.2 Alarms

Seleccione este botón para definir la condición de alarma del tagname. Cuando definimos un tagname análogo (entero o verdadero), el siguiente cuadro de diálogo se usa para seleccionar tipos de alarmas y entrar sus valores. Los valores de campos ingresados para cada tipo de alarma no aparecerán hasta que un tipo de alarma sea habilitado por un click en su respectivo cuadro de chequeo como se muestra más adelante:

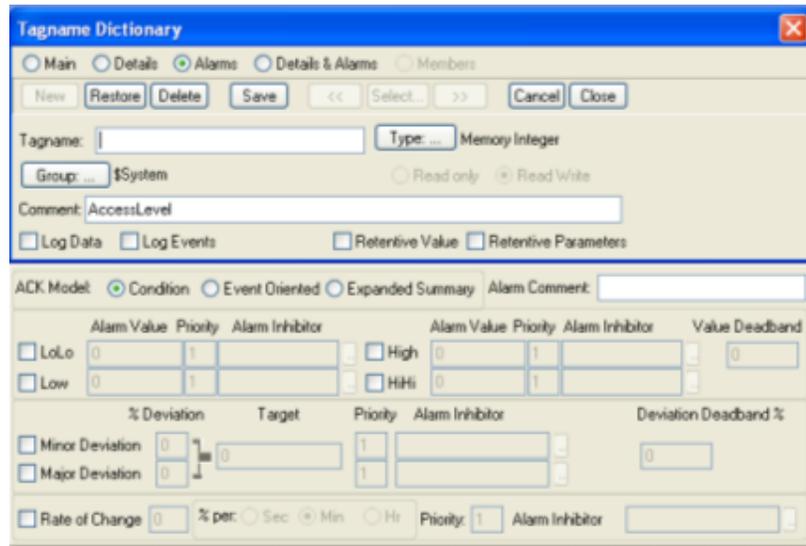
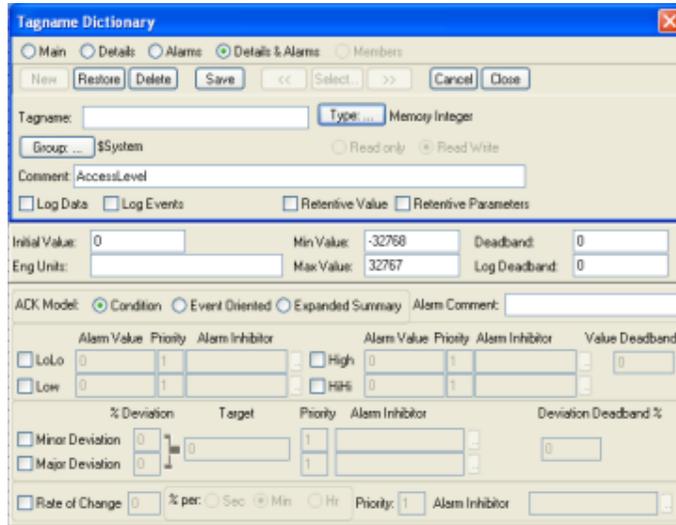


Fig. II-39 Cuadro de Diálogo de Alarmas

2.1.3.4.3 Both

Seleccione este botón para mostrar ambos cuadros de diálogos details y alarms simultáneamente. El siguiente es un ejemplo como la pantalla se verá cuando un tagname "Memoria Real" está siendo definido y este botón se selecciona:



2.1.3.5 Ir Fig. II-40 Cuadro combinado parámetros Details & Alarms

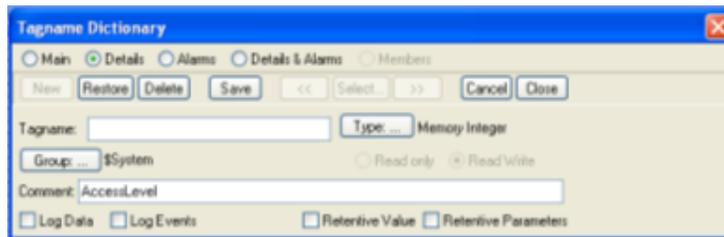


Fig. II-41 Ingresar Campos, botones y opciones de entrada

2.1.3.5.1 Tag Name:

Ingrese el nuevo tagname en este campo. Los Tagnames pueden ser hasta un largo de 32 caracteres y el primer carácter debe ser A-Z o a-z y puede seguirse con A-Z, a-z, 0-9, !, ¿, Q, -, ? , #, \$, %, _ y &.

2.1.3.5.2 Comment:

Los Comentarios son opcionales, pero pueden ser mostrados en una Ventana de Alarma.

Ingrese cualquier información opcional para documentar el tagname en este campo (hasta 50 - caracteres). (Estos comentarios pueden mostrarse en ventanas de alarmas.)

Type: ...

Hacer click sobre este botón para acceder al Escoger tipo de Tag..., el cuadro de diálogo selecciona el tipo de tagname:

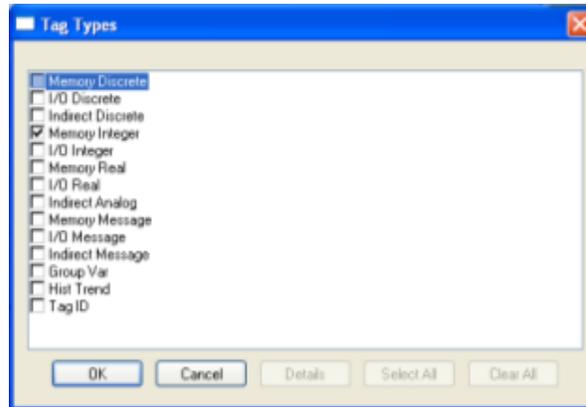


Fig. II-42 Tipos de Tags

Group: ...

Hacer click sobre este botón para acceder al cuadro de diálogo Escoger

Grupo de Alarmas... para asignar el tagname a un grupo de alarmas específico o crear un nuevo grupo de alarma.

2.1.3.5.3 Read only

Seleccione este botón para restringir a WindowViewer para que solamente sea capaz de leer el valor del tagname.

2.1.3.5.4 Read Write

Seleccione este botón para permitir a WindowViewer leer/escribir de/desde el valor del tagname.

2.1.3.5.5 Log Data

Registra automáticamente el tagname en el archivo Histórico Log cada vez que sus valores de unidades de ingeniería cambien, más el valor del Registro especificado Deadband.

2.1.3.5.6 Log Events Priority 999

Registra todos los cambios de valores del tagname que fueron establecidos por el operador, DDE, un script o por el sistema. Cuando esta opción es habilitada, el campo de Prioridad aparecerá. El valor ingresado en la Prioridad determina la preferencia de nivel del suceso para el tagname. Las entradas válidas en este campo son de 1 a 999 donde 1 es la más alta prioridad y 999 es la más baja.

2.1.3.5.7 Retentive Value

Esta opción es usada para retener el valor actual del tagname. Este valor, entonces es usado como el valor inicial para el tagname cuando **WindowViewer** se reinicie.

2.1.3.5.8 Retentive Parameters

No varía los cambios del valor de cualquier campo de alarma para el tagname. Este valor se usará como el valor inicial para las alarmas cuando **WindowViewer** es reiniciado.

2.1.3.6 Tipos de Tag

Cada tagname debe ser asignado a un tipo específico según el uso del tagname. Por ejemplo, si el tagname es para leer o escribir valores que vienen desde otra aplicación de Windows tal como un servidor DDE, este debe ser un tag tipo DDE. Se debe considerar si el Tag esta representando una señal Discreta, que es un solo bit (tal como una sola entrada del PLC) o una señal Análoga (tal como un registro de 16, 32, o 64 bits). Los tag analógicos son separados en dos de tipos, Enteros y Reales. A continuación se describe cada tipo de tag y su uso respectivo.

2.1.3.6.1 Tagname tipo Memoria

Estos tipos de tag existen solamente dentro del programa Intouch. Ellos pueden usarse para crear constantes de sistema, demostraciones y simulaciones. Estos también son útiles en la creación de cálculos de variables para ser accedados en otros Programas de Windows. Por ejemplo, si un tagname de memoria real es creado llamado "PI", podría asignarse a él, el valor inicial de 3.1416; o las fórmulas podrían almacenarse en grupos de tagname de memoria. En simulaciones; los tagnames de memoria pueden usarse para controlar las acciones de unos antecedentes de script. Por ejemplo un tagname de memoria "STEP" podría ser cambiado con una acción presionando un botón Script para provocar varios efectos de animación. Hay cuatro tipos de Memoria:

2.1.3.6.2 Memory Discrete

Tagname Discreto Interno con un valor de 0 (Falso, Fuera) o 1 (Verdadero, Habilitado).

2.1.3.6.3 Memory Integer

Es asignado un valor entero de 32 bit entre -2, 147, 483, 648 y 2, 147, 483, 647.

2.1.3.6.4 Memory Real

Tagname de memoria con punto flotante (decimal). El valor del punto flotante puede estar entre $\pm 3.4e38$. Todos los cálculos del punto flotante se realizan con 64 bit de resolución, pero el resultado se almacena en 32 bit.

2.1.3.6.5 Memory Message

El Texto del tagname de una fila puede ser de un largo de 131 caracteres.

2.1.3.6.6 Tagname tipo DDE

Todos los tagnames que leen o escriben sus valores hacia o desde otro programa de Windows son Tagname DDE. Esto incluye todas las entradas y salidas desde controladores programables, procesos de computadoras, otros programas de Windows y datos desde nodos de red. Los Tagnames DDE se ingresan por medio del Protocolo Dynamic Data Exchange (DDE) de Microsoft. Cuando el valor del tagname DDE de lectura/escritura cambia, éste es inmediatamente escrito en la aplicación remota por medio de DDE. El tagname también puede ser actualizado desde la aplicación remota cuando el ítem al cual el tagname es vinculado cambia a la aplicación remota. Por defecto **Leer / Escribir** es un conjunto de todos los tagnames DDE. Sin embargo, ellos pueden restringirse para Leer solamente al seleccionar el botón **Read Only** en el cuadro de diálogo Dictionary – Tag Name Definition. Hay cuatro Tipos de DDE:

2.1.3.6.7 DDE Discrete

Tagname de entrada/salida discreta con un valor entre 0 (Falso, Off) o 1 (Verdadero, On).

2.1.3.6.8 DDE Integer

Es asignado un valor entero de 32 bits entre -2, 147, 483, 648 y 2, 147, 483, 647.

2.1.3.6.9 DDE Real

Tagname de memoria con punto flotante (decimal). El valor del punto flotante puede estar entre $\pm 3.4e38$. Todos los cálculos del punto flotante se realizan con 64 bit de resolución, pero el resultado se almacena en 32 bit.

2.1.3.6.10 DDE Message

El Texto del tagname de una fila puede ser de un largo de 131 caracteres

2.1.3.6.11 Tagnames de Tipo Misceláneo

Estos son un tipo de tags especiales que son asignados a tagnames que realizan funciones complejas, tal como mostrar despliegues de alarmas dinámicas, históricos, monitoreo y / o cambio del tagname en cada trazado histórico que es dibujado y tagname indirectos que permiten reasignar los tagname a múltiples fuentes. Estos tipos de tags son los siguientes:

2.1.3.6.12 Group Var

Este tipo de tag es asignado a un tagname que tenga el nombre de un Grupo de Alarma asignado a él. Esto es muy útil en hacer despliegues de alarma, registros de discos e imprimir dinámicamente los registros. Las ventanas de alarmas o registros de alarmas pueden configurarse para mostrar todas las alarmas asociadas con un Group Var. Mediante asignación un grupo de Alarma diferente al nombre del Grupo de Variable, la selección de alarmas a mostrar o a registrar puede ser controlada. Un tagname de **Group Var** puede usarse para crear un botón que selectivamente muestre las alarmas de diferentes partes de una planta en la misma Ventana de Alarma. Todos los campos asociados con los Grupos de Alarma pueden ser aplicados a Group Var.

2.1.3.6.13 Hist Trend

Este tipo de tag es asignado a un tagname para ser usado como un diagrama Histórico de Tendencia. Cuando configuramos un mapa Histórico de Tendencia, InTouch requiere que un tipo de tagname HistTrend sea asignado a un gráfico.

2.1.3.6.14 Tag ID

Este tipo de tag es usado para obtener información sobre los tagnames que están siendo registrados en un mapa de Tendencia Histórica. El uso más común es mostrar el nombre del Tagname de la tendencia. Por ejemplo, Pen4 de un mapa Histórico de Tendencia se muestra la tendencia del tagname "Analog1". Una ventana

puede ser creada con un campo de salida que muestra el tagname actualmente asignado a Pen4.

Indirect Discrete

Indirect Analog

Indirect Messaga

El tagname de tipo Indirecto permite al operador crear una de ventana y reasignar el tagname en esa ventana a múltiples fuentes. Por ejemplo, uno Cambio de Datos de Script pueden ser creado para cambiar la fuente de todos los tagnames en una ventana basado sobre el cambio de un valor.

Cuando un tag indirecto es igual a otro tag de fuente, ambos llegan a ser iguales el uno al otro en cada aspecto. Por ejemplo, campos, scripts, etc., si el valor del tag de fuente cambia, en el tag indirecto se reflejará el cambio. Si el valor del tag indirecto cambia, el tag de fuente cambia consiguientemente. Los valores del tagname indirecto pueden establecer en la base de datos como retentivo y también se pueden modificar. Esto les permite tomar su última asignación de tagname en la partida.

2.1.3.6.15 Tagnames "Placeholder"

Cuando una ventana es importada o es exportada a una aplicación, todos los tagnames asociados con la ventana son transferidos con la ventana, pero no son agregados a la nueva base de datos de aplicación. Ellos serán automáticamente cambiados en el tagnames "placeholder" (índice) y deben ser convertidos y definidos por la nueva aplicación.

1. Para convertir el tagnames, seleccione el comando */Edit/Select All* (F2) luego el comando */Special/Substitute All Tags...*(Ctrl+U). El cuadro de diálogo *Sustitute Tag Names...* aparecerá listando todos los tagnames usados por todos los objetos en la ventana:

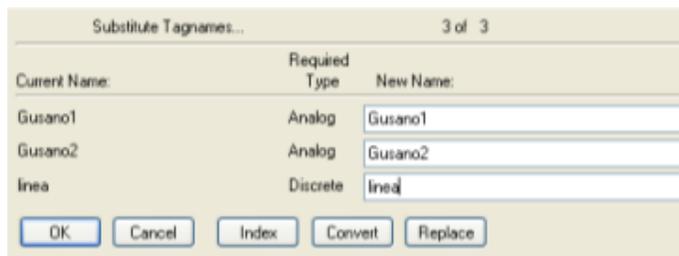


Fig. II-43 Special/Substitute All Tags

2. Note el " ?d: ", " ?h: ", etc. que precede el tagnames, estos son "placeholders". La letra "d" indica que el tagname se definió originalmente como un tipo "discreto", la "h" indica un "HistTrend", etc. haciendo Click sobre el botón Convert se removerá automáticamente el placeholders:

3. Una vez que los tagnames son convertidos, haga click sobre OK. Si el tagname no existe en la actual base de datos de la aplicación, el siguiente cuadro de mensaje aparecerá verificando la creación del nuevo tagname:



Fig. II-44 Mensaje para crear tag no existe

4. Haciendo Click sobre **OK** permitirá que el cuadro de diálogo **Dictionary - Tag Name Definition** aparezca y el tagname pueda definirse y agregarlo a la base de datos de la actual aplicación.

2.1.3.7 Links (Vínculos)

Una vez que un objeto gráfico o el símbolo se ha creado, éste puede ser animado uniendo vínculos de animación a él. Los vínculos de animación permiten que el objeto, símbolo o celda cambien su apariencia reflejando los cambios en el valor de una variable de la base de datos del tagname. Por ejemplo, el símbolo de una bomba puede ser de color rojo cuando esta apagada y de color verde cuando está encendida. El símbolo de la bomba también puede ser encendida por un pulsador que se usa para encender y apagar la bomba cuando se pulse. Estos y muchos otros efectos de animación son obtenidos cuando se definen los vínculos de animación para un objeto o símbolo seleccionado. Conjuntamente con los tagnames, algunos vínculos de animaciones también permiten el uso de expresiones lógicas usados para calcular variables (por ejemplo *integer1 + 100*, *discrete1 AND discrete2 AND NOT discrete3*, etc.).

2.1.3.7.1 Uniendo y Seleccionando un Vínculo

Cuando un objeto se selecciona y el comando */Special/Animation Links...* se ejecuta hay dos cuadros de diálogo que siempre aparecen; el *"Item Description"* y el *"Links selection"*. Ambos cuadros de diálogo son descritos más adelante.

Haciendo Doble – click sobre el que objeto deseado se llama directamente la pantalla de los *Links* de Animación.



Fig. II-45 Ítem Description

A continuación se describe cada campo en el cuadro previo de diálogo:

2.1.3.7.2 Prev Link y Next Link

Si varios vínculos han sido adjuntos al objeto, estos dos botones pueden usarse para avanzar y retroceder rápidamente mediante el cuadro de diálogo de detalle para cada Links adjunto al objeto sin salir del cuadro de diálogo de Vínculo de Animación.

En un objeto o símbolo pueden tener múltiples links definidos. La capacidad para combinar los vínculos de animación provee la capacidad para crear casi cualquier animación de pantalla y efecto imaginable.

2.1.3.7.3 Horizontal/Vertical Loc. Width/Height

Corresponde a la ubicación Horizontal y Vertical como también al Largo y Alto del objeto expresado en pixeles de la pantalla.

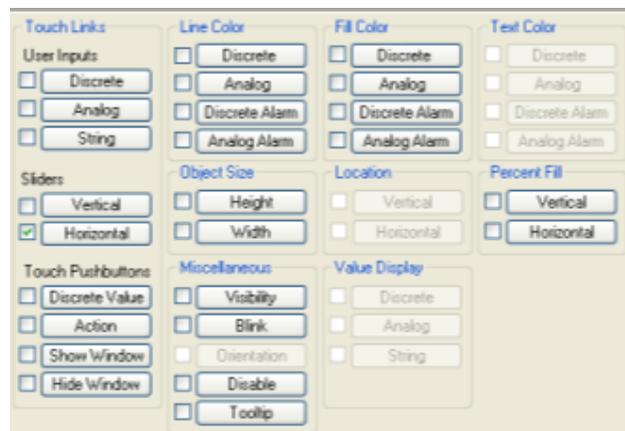


Fig. II-46 Parámetros animaciones longitud y posición

Este cuadro de diálogo se usa para seleccionar el *Link* (s) deseado (s) para el objeto, haciendo click sobre el botón con el nombre del *links*. Haciendo Click sobre el cuadro de chequeo, solamente selecciona o deshabita el *links*. Haciendo Click sobre el botón

del nombre del *links* selecciona y permite que aparezca el específico cuadro de diálogo de definición de detalle para el *links*.

2.1.3.8 Información común del Links...

2.1.3.8.1 La Paleta de Selección de Color

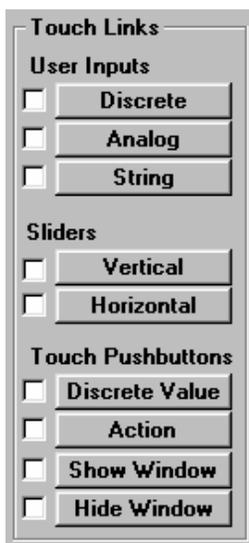
El cuadro de selección de color aparece cuando un color de *links* está siendo unido para animar el color de la línea, llenado de color, (y/o para atributos de color de texto de un objeto). El cuadro de selección de color ofrece una cantidad de 32 colores. La primera columna (desde la izquierda) muestra los ocho colores sólidos primarios. Las tres columnas que quedan muestran los ocho colores sólidos o colores derivados dependiendo de la tarjeta de gráficas instalada en la computadora.

2.1.3.8.2 Asignando una tecla a un Vínculo de Animación

Una Tecla específica o combinación de teclas en el teclado pueden ser asignado para activar ciertos links, cuando la tecla se presiona en **WindowViewer**. La tecla equivalente es solamente operacional cuando el objeto con el *link* es visible. Si una tecla se define en ventanas múltiples, la definición en la ventana abierta más recientemente será la que se active primero. Para asignar una tecla, haga click sobre el botón **Key...** en el cuadro de diálogo de detalle de animación. El cuadro de diálogo "Choose Key..." aparecerá rápidamente para que usted seleccione la tecla deseada.

2.1.3.9 Links disponibles

2.1.3.9.1 Touch Links



Touch Link convierte cualquier objeto o símbolo en un botón de acción. Un botón de acción puede ser activado al presionar el botón del mouse sobre él, tocando la imagen en la pantalla (si la pantalla es sensible al tacto), presionando alguna letra asignada o presionando la tecla [Enter] (sí el objeto está marcado alrededor). Hay nueve de tipos de *Touch Links* que pueden ser definidos, que se describen a continuación.

Nota: Si el objeto o símbolo utilizado para estos vínculos contiene un campo de texto, todos atributos diseñados para el texto

Fig. II-47 Touch Links

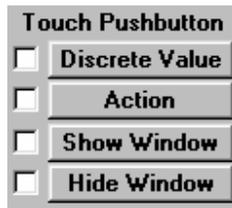
aparecerán cuando el objeto aparezca en el modo Runtime. El valor de salida que usted haya ingresado será mostrará en el campo de a menos que la opción Input Only sea habilitada en el cuadro de diálogo de Details.

2.1.3.9.2 Valor Slider Touch Links



El *Slider* (Deslizador) *Touch Links* se usa para crear un objeto o símbolo que pueda moverse con el mouse. Como el objeto o símbolo se mueve, este altera el valor de una variable vinculada a él. Esto permite la capacidad de crear dispositivos para colocar valores en el sistema. Un objeto puede ser un *Slider* Horizontal, un *Slider* Vertical, o ambos. Para usar ambos *Link* sobre un objeto único, el valor de las dos variables análogas pueden ser cambiadas simultáneamente.

2.1.3.9.3 Vínculos con Touch Pushbutton



Los Touch Pushbutton son vínculos de objeto que ejecutan inmediatamente una operación cuando son activados. Estas operaciones pueden ser cambios de Valores *Discreto* (*Discrete Value*), ejecución de *Script* (*Action*), mostrar Ventanas (*Show Window*) u Ocultar Ventanas (*Hide Windows*).

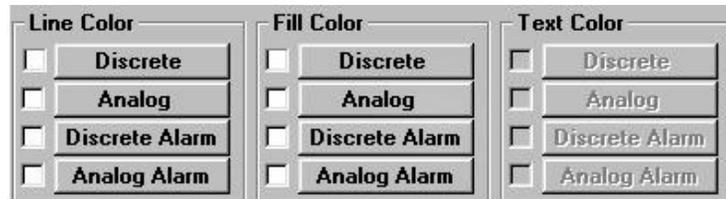


Fig. II-48 Touch Pushbutton

2.1.3.9.4 Vinculo de Colores

Los vínculos de Colores permiten la animación de atributos de color de línea, llenado, y/o texto de un objeto. Cada uno de estos atributos pueden ser creados dinámicamente al definir un color vinculado al atributo. El atributo de color puede vincularse al valor de una expresión discreta, expresión análoga, condición discreta de alarma o condición análoga de alarma.

2.1.3.9.5 Vínculos de Colores de Objetos a Estados de Alarma



El color de texto, línea, y llenado de un objeto pueden ser vinculados al estado de alarma de un *Tagname*, *Group Name*, o *Group Variable*. Hay dos tipos básicos de alarmas orientados al color. El primero es el Estado del tag de Alarma Discreta (Discrete Alarm) que puede usarse con cualquier tipo de variable. El segundo tipo es el Estado de Alarma Análoga (*Analog Alarm*) que es únicamente aplicable a las variables de tipo análogas.

[Value Alarm Color Link]

Provee la capacidad para seleccionar hasta cinco diferentes colores basados en los límites de alarma definidos para ese tagname.

[Desviation Alarm Color Link]

Proveen la capacidad para seleccionar hasta tres diferente colores dependiendo de los estados de las alarmas de desviación que se definen para el punto.

[Rate-Of-Change Alarm Color Link]

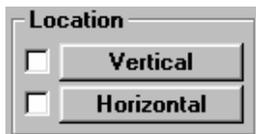
Provee la capacidad para seleccionar dos colores diferentes dependiendo del estado de la razón de cambio de la alarma definido para el punto.

2.1.3.9.6 Vínculos del Tamaño del Objeto



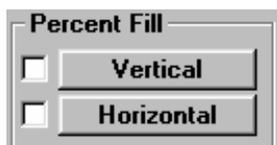
Este vínculo se usa para variar la altura y/o ancho de un objeto según el valor de una expresión analógica. Provee la capacidad para controlar la dirección en que el objeto se agranda en altura y/o anchura seteando el **Anchor** (*Top*, *Middle*, o *Botton*) del vínculo. Ambos vínculos, de anchura y altura, pueden ser usados en el mismo objeto.

2.1.3.9.7 Vínculos de Localización



Este vínculo se usa para hacer que un objeto se mueva horizontalmente, verticalmente o en ambas direcciones automáticamente con respecto a cambios en el valor de una expresión.

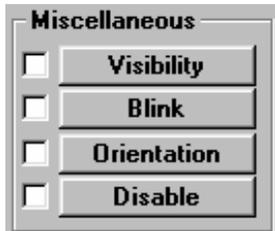
2.1.3.9.8 Vínculos del Porcentaje de Llenado



Este vínculo provee la capacidad para variar el nivel de relleno de una forma o un símbolo que se llenan según el valor de una

expresión analógica. Por ejemplo, este vínculo puede usarse para mostrar el nivel de líquidos en una embarcación. Un objeto o símbolo puede ser llenado en forma horizontal, vertical, o ambos.

2.1.3.9.9 Vínculos Misceláneos



Existen cuatro vínculos misceláneos, *Visibility*, *Blink*, *Orientation* y *Disable*. Los vínculos de visibilidad controlan la visualización de un objeto dependiendo del valor de un tagname discreto o expresión. Este vínculo puede usarse para hacer que aparezcan y desaparezcan objetos. Los vínculos *Blink* proveen la capacidad para hacer que un objeto parpadee de acuerdo al valor de un tagname discreto o expresión. (Cada uno de los atributos de color del texto, Línea y relleno pueden ser seleccionado para que un objeto parpadee). Los vínculos de orientación proveen la capacidad para hacer que un objeto gire sobre un eje especificado de acuerdo al valor de un tagname o expresión. El vínculo *Disable* provee la capacidad de impedir a un usuario particular o un grupo de usuarios tener acceso a presionar un botón en particular si sus *AccessLevel* no es lo suficientemente alto por ejemplo (si el vínculo *disable* es verdadero, el texto sobre el botón 3-d no se destacará indicando que esta deshabilitado).

2.1.3.9.10 Vínculos de valores de salida



Estos vínculos proveen la capacidad para usar un texto objeto para mostrar el valor de una variable discreta (*Discrete*), análoga (*Analog*), o alfanumérica (*String*).

2.1.3.10 WINDOWVIEWER

2.1.3.10.1 Menú File

El menú File de WindowViewer es muy similar a cualquier menú de una aplicación en común. Permitiendo abrir, cerrar, salir, etc. ventanas de la aplicación.

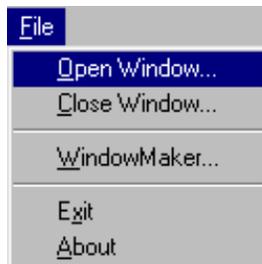


Fig. II-49 Submenú File/Windowviewer

2.1.3.10.2 Controlador Logic



El Menú de *Logic* contiene los comandos de partida (*Start Logic*) y parada (*Stop Logic*) de la ejecución de los scripts que se crearon en el **WindowMaker**.

2.1.3.10.3 Menú Special

Este menú de **WindowViewer** contiene los comandos que son utilizados para iniciar o reiniciar conversaciones DDE, reanudar Registros de Alarmas, iniciar/detener registros Históricos y acceder al **Wonderware Logger** para ver información de error.

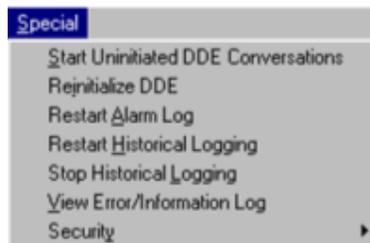


Fig. II-50 Submenú Special/Windowviewer

2.1.3.10.4 Special/Security

Los comandos de seguridad de este menú son usados para iniciar y cerrar la aplicación, cambiar passwords o contraseñas y para configurar la lista de usuarios, passwords y niveles de acceso. Cuando éste comando es seleccionado aparece un menú secundario de comandos:

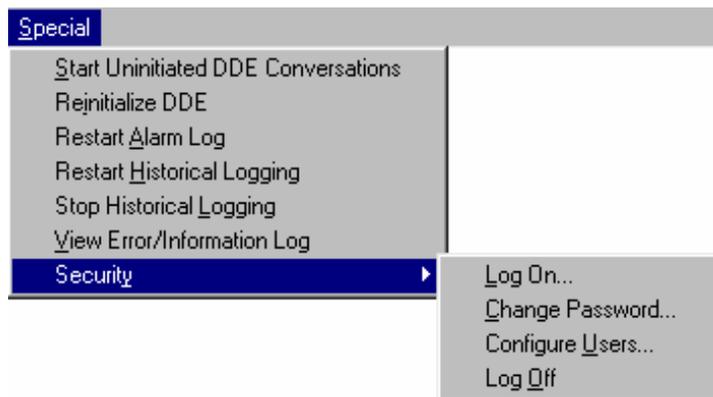


Fig. II-51 Submenú Special/Security/Windowviewer

2.1.3.11 DISEÑO DE SCRIPTS EN INTOUCH

La variedad de scripts de Wonderware expande las capacidades de InTouch para proporcionar la ejecución de comandos y operaciones lógicas según las necesidades. Por ejemplo presionar una tecla, abertura de ventanas, cambio de valores, etc. Al usar Scripts, se pueden crear una gran variedad de funciones y sistemas automáticos personalizados.

Existen seis tipos de Scripts disponibles:

Scripts de Aplicación.

Scripts de Ventanas.

Scripts de Teclas

Scripts de Condiciones

Scripts de Cambio de Datos

Scripts de Accionamiento de Botones (Touch Pushbotton)

Los comandos para crear estos *Scripts* (a excepción del Accionamiento de Botones) se ubican en el menú **/Special/Scripts**

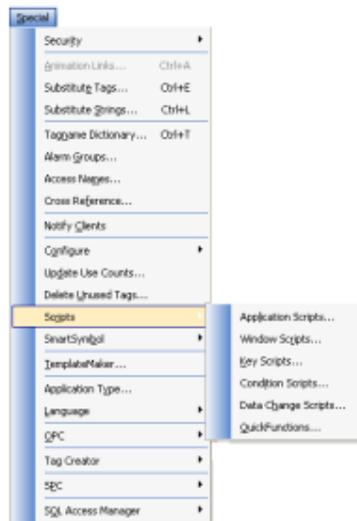


Fig. II-52 Special/Scripts (Submenú)

2.1.3.11.1 Scripts de Aplicación

Los Scripts de Aplicación son vinculados a una aplicación completa y se pueden usar para correr otras aplicaciones, crear simulación de procesos, cálculo de variables. Al seleccionar este comando aparece el siguiente cuadro de diálogo:

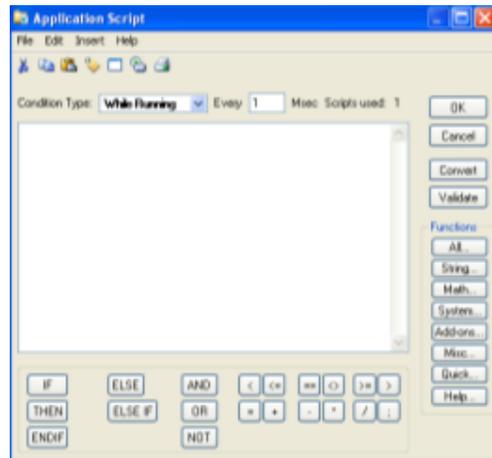


Fig. II-53 Cuadro de Diálogo para edición de Scripts



“On Startup” permite cre Fig. II-54 Modos de ejecución na vez cuando la aplicación comienza.

“While Running” crea un Script para que se ejecute continuamente cada cierta cantidad de tiempo mientras corre una aplicación. Este tiempo se ingresa en milisegundos.

“On Shutdown” ejecuta el Script una vez que la aplicación termina.

Una vez que el botón Script es presionado, aparecerá en pantalla el cuadro de diálogo de selección de script.

Los tres tipos de scripts pueden ser utilizados en una misma aplicación. Un Script While Running se ejecutará después de transcurridos el tiempo establecido. Para una ejecución inmediata se debe crear un script On Starup.

2.1.3.11.2 Script de Ventana

Estos tipos de scripts son vinculados a una ventana específica. La selección de este comando causará la aparición de la ventana de diálogo **Window Action Script for “Nombre de Ventana”**.

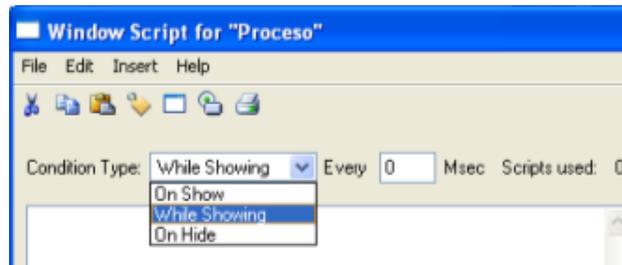


Fig. II-55 Cuadro de Diálogo Window Script

“**On Show**” permite crear un Script que se ejecute una vez cuando la ventana se abre por primera vez.

“**While Showing**” crea un Script para que se ejecute continuamente cada cierta cantidad de tiempo mientras la ventana está abierta. Este tiempo se ingresa en milisegundos.

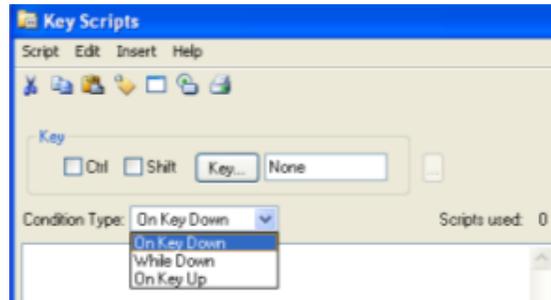
“**On Hide**” ejecuta el Script una vez que la ventana se cierra.

Una vez que el botón Script es presionado, aparecerá en pantalla el cuadro de diálogo de selección de script.

Los tres tipos de scripts pueden ser utilizados en una misma ventana. Un Script While Showing se ejecutará después de transcurridos el tiempo establecido. Para una ejecución inmediata se debe crear un script On Show.

2.1.3.11.3 Script de Teclas

Estos tipos de scripts son vinculados a una tecla específica del teclado. Pueden muy útiles para crear teclas globales para la aplicación, por ejemplo, volver a una ventana del menú principal, salida del operador (log off). Seleccionando este comando aparecerá el cuadro de diálogo **Keyboard Action Script**:



“**On Key Down**” permite crear un Script que se ejecute una vez cuando la tecla inicialmente es presionada por primera vez.

“**While Down**” crea un Script para que se ejecute continuamente cada cierta cantidad de tiempo mientras la tecla está presionada. Este tiempo se ingresa en milisegundos.

“**On Key Up**” ejecuta el Script una vez que la tecla es dejada de presionar.

Una vez que el botón Script es presionado, aparecerá en pantalla el cuadro de diálogo de selección de script.

Los tres tipos de scripts pueden ser utilizados en una misma ventana. Un *Script While Down* se ejecutará después de transcurridos el tiempo establecido. Para una ejecución inmediata se debe crear un script *On Key Down*.

Si un objeto o botón de acción en la ventana activa (abierta) es asignado a una misma tecla usada para un Script de teclas, el vínculo en la tecla de la ventana tendrá preferencia sobre la ejecución del Script.

2.1.3.11.4 Scripts de Condiciones

Estos Scripts están vinculados a un tagname o expresión discretos. Cuando este comando es seleccionado aparecerá el cuadro de diálogo **Condition Action Script**:

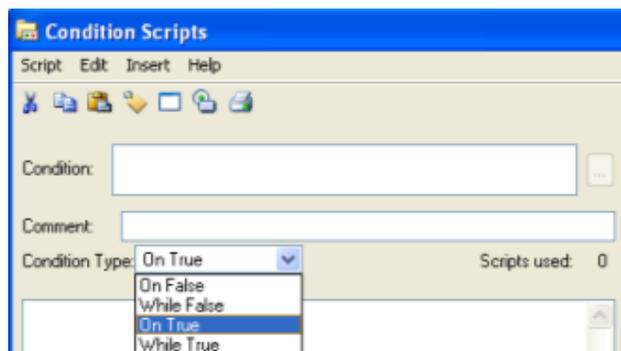


Fig. II-57 Cuadro de Diálogo Condition Script

“**On True**” ejecuta un Script una vez cuando la condición pasa a ser verdadera.

“**On False**” se ejecuta cuando la condición pasa a ser falsa.

“**While True**” ejecuta el Script continuamente a una frecuencia especificada mientras la condición es verdadera.

“**While False**” ejecuta el Script continuamente a una frecuencia especificada mientras la condición es falsa.

El valor para la condición debe pasar a la transición verdadera o falsa antes de que el Script se ejecute. Por ejemplo, si el valor inicial cuando parte el WindowViewer es verdadero, el valor debe transformarse en falso y nuevamente a verdadero para que se ejecute un Script On True.

 Establece el tiempo (en milisegundos) en que los Scripts **WhileTrue** y **While False** se ejecutarán en forma repetitiva.

Una vez que un botón de Script es pulsado aparecerá en pantalla el cuadro de diálogo del Script.

Los cuatro tipos de Scripts pueden ser usados a la misma condición. Ambos Scripts, While True y While False, se ejecutarán después de transcurridos el tiempo en milisegundos establecido. Para una ejecución inmediata, se crea o se crean los Scripts On True y/o On False.

Los tagnames que son modificados (escritos) en un Script de Condición o Script de Cambio de Dato, no deberían ser usados como el tagname para un Script de Cambio de Datos o en la expresión de un Script de Condición. Ejemplo, un Script de Cambio de Dato ejecuta un valor A cambiando el contenido lógico de $B=B+1$. El tagname B no debería ser usado como el tagname para un Script de Cambio de Dato o ser parte de la expresión de un Script de Condición.

2.1.3.11.5 Scripts de Cambio de Datos

Estos Scripts se vinculan a un tagname o tagname field. Se ejecuta una vez cuando el valor del tagname o campo de un tagname cambien por un valor mayor que la banda muerta (dead band) definida para él, el ítem /point en el Tagname Dictionary.

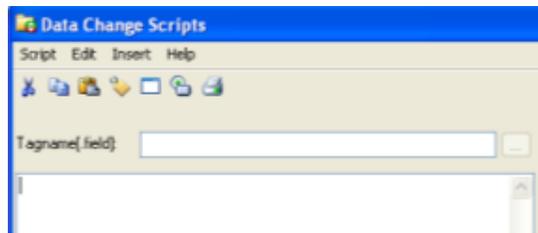


Fig. II-58 Cuadro de Diálogo Data Change Script

Los tagname que son modificados en un Script de Condición o Script de Cambio de Dato no deben ser usados con el tagname para un Script de Cambio de Datos o en la expresión de un Script de Condición.

2.1.3.11.6 Ingreso de la Operación Lógica

En la mayoría de los casos, una vez que el tipo de Script es seleccionado en su respectivo cuadro de diálogo, aparecerá también el siguiente cuadro de diálogo en la pantalla:

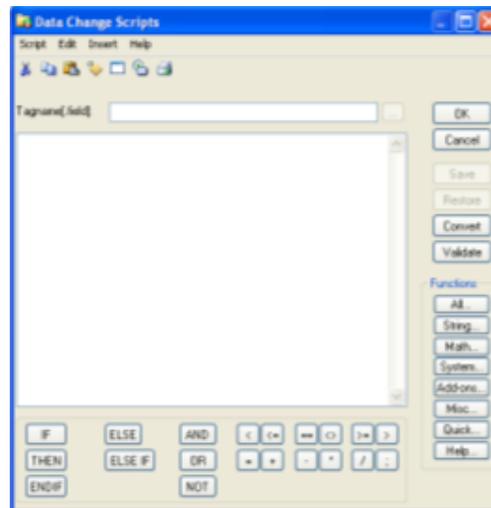


Fig. II-59 Cuadro de Diálogo para Operaciones Lógicas

2.1.3.11.7 Significado de cada uno de los Botones

Pulsando alguno de estos botones aparecerá la palabra escrita de ellos.

Pulsando alguno de estos botones de equivalencia o matemática ingresa la función correspondiente a él.



Fig. II-60 Botones de Operaciones Lógicas y Comparaciones Matemáticas

- Del** El texto seleccionado (resaltado) es eliminado del Script. El texto eliminado no es guardado en la carpeta de Windows.
- Cut** El texto seleccionado (resaltado) es borrado del Script. El texto cortado es guardado en la carpeta de Windows.
- Copy** El texto seleccionado (resaltado) es copiado a la carpeta de Windows. El texto copiado no es borrado del Script.
- Paste** Pega el contenido de la carpeta de Windows en la posición del cursor en el Script.
- Undo** Se devuelve a la última operación de edición.
- TAB** Remueve el primer carácter tabulado a la izquierda de la posición del cursor.
- +TAB** Inserta una tabulación a la derecha de la posición del cursor.
- Clear** Borra completamente el texto del Script. (El Script no es borrado de la aplicación).
- Validate** Valida el Script ingresado. Si existen errores aparecerá el correspondiente cuadro de mensaje.
- Help...** Aparece en pantalla los tópicos de ayuda de WindowMaker para todas las funciones del editor de Script.



2.1.3.11.8 Funciones específicas

Están creadas internamente en el sistema y pueden ser seleccionadas e insertadas al Script seleccionando uno de esos botones. Al seleccionar uno de ellos aparecerá el cuadro de diálogo.

2.1.3.12 ALARMAS Y EVENTOS

2.1.3.12.1 Alarmas

InTouch permite la visualización de alarmas distribuidas (gestión de las alarmas bajo una estructura cliente/servidor en una red de ordenadores).

InTouch soporta la visualización, archivo (en disco duro o en base de datos relacional) e impresión de alarmas tanto digitales como analógicas, y permite la notificación al operador de condiciones del sistema de dos modos distintos: Alarmas y Eventos. Una alarma es un proceso anormal que puede ser perjudicial para el proceso y que normalmente requiere de algún tipo de actuación por parte del operador. Un evento es un mensaje de estado normal del sistema que no requiere ningún tipo de respuesta por parte del operador.

2.1.3.12.1.1 Tipos de Alarmas

Las alarmas pueden dividirse en los siguientes tipos: Condición de Alarma

Condición de Alarma	Tipo
Discrete	DISC
Deviation - Major	LDEV
Deviation - Minor	SDEV
Rate-Of-Change (ROC)	ROC
SPC	SPC
Value - LoLo	LOLO
Value - Lo	LO
Value - Hi	HI
Value - HiHi	HIHI

Cada alarma se asocia a un tag. Dependiendo del tipo de tag podremos crear uno u otro tipo de alarma.

Fig. II-63 Tipos de Alarmas

2.1.3.12.1.2 Prioridades de las Alarmas

A cada alarma de cada tag puede asociarse un nivel de prioridad (importancia) de 1 a 999 (Prioridad 1 es más crítica). Ello permite filtrar alarmas en displays, en impresora o en disco duro.

2.1.3.12.1.3 Grupos de Alarmas

InTouch dispone de un cómodo sistema para prioridades de alarmas.

Cuando creamos un tagname de alarma, le asignamos un grupo de alarmas. Estos grupos o "jerarquía" de alarmas permiten significar qué alarmas son más importantes, a la vez que permiten reconocer un grupo de alarmas en lugar de todas a la vez.

Al crear un tag, lo asociamos a un grupo (si no lo hacemos, el tag de alarma queda automáticamente asociado al grupo principal, llamada *\$SYSTEM*).

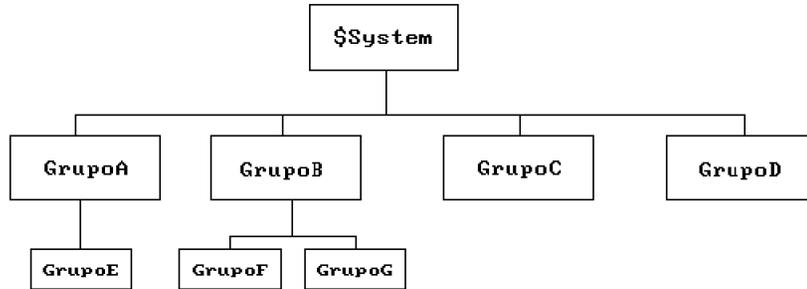


Fig. II-64 Asociación de tags a grupos

Estos grupos y subgrupos funcionan de un modo análogo al sistema de jerarquía utilizado por el MsDOS.

2.1.3.12.1.4 Cómo Crear Grupos de Alarmas

Los grupos de alarmas se crean desde /Special/Alarm Groups.

Desde aquí también "emparentamos" unos grupos con otros, hasta crear toda la "jerarquía"

2.1.3.12.1.5 Definición de una Condición de Alarma en un Tagname

Para definir un tagname debemos seleccionar ALARMS en el momento de definir el tagname.

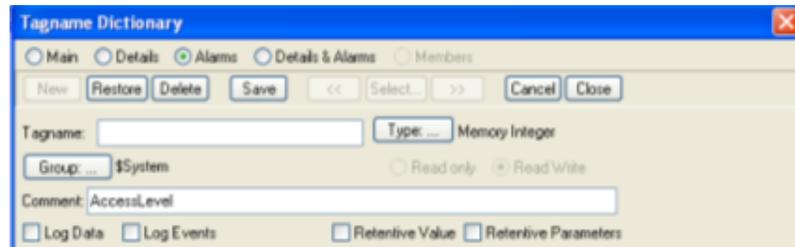


Fig. II-65 Definición de una Alarma en un Tagname.

Para las alarmas analógicas, disponemos de las siguientes posibilidades:

Fig. II-66 Posibilidades de Alarmas Analógicas

Alarm Value: Valores límite de la alarma. 4 niveles

Pri: Prioridad de la alarma

Minor/Major Deviation: Se utiliza para detectar cuándo el valor analógico es una desviación mayor o menor del valor especificado en Target.

% Deviation: Porcentaje de desviación permitido al tagname con respecto al Target para que se produzca una alarma por desviación mayor o menor.

Target: Valor de referencia para los porcentajes mayor/menor de desviación.

Rate of Change: Este tipo de alarma detecta cuándo el valor de la alarma varía en exceso de acuerdo a una cantidad de tiempo.

2.1.3.12.1.6 Creación de un Objeto de Alarmas

Dentro del icono de *WIZARDS* de la *toolbox* (Caja de Herramientas) encontraremos el objeto *ALARMAS*. Para crear una ventana de alarmas, basta con seleccionar el objeto como si se tratara de un rectángulo o un círculo. Definimos su tamaño, y la ventana de alarmas quedará creada. Esta ventana puede ser tan grande como toda la pantalla.

Date	Time	State	Class	Type	Priority	Name	Group
11 abr	21:00	UNACK	Value	HHI	1	Alam1	GroupName
11 abr	21:00	UNACK	Value	HI	250	Alam2	GroupName
11 abr	21:00	UNACK	Value	LO	500	Alam3	GroupName
11 abr	21:00	UNACK	Value	LOLD	750	Alam4	GroupName
11 abr	21:00	ACK	Dev	Minor	1	Alam5	GroupName
11 abr	21:00	ACK	Dev	Major	250	Alam6	GroupName
11 abr	21:00	ACK	ROC	1	500	Alam7	GroupName
11 abr	21:00	ACK	Custom	1	750	Alam8	GroupName

Fig. II-67 Cuadro de Alarmas

2.1.3.12.1.7 Configuración de un Objeto de Alarmas

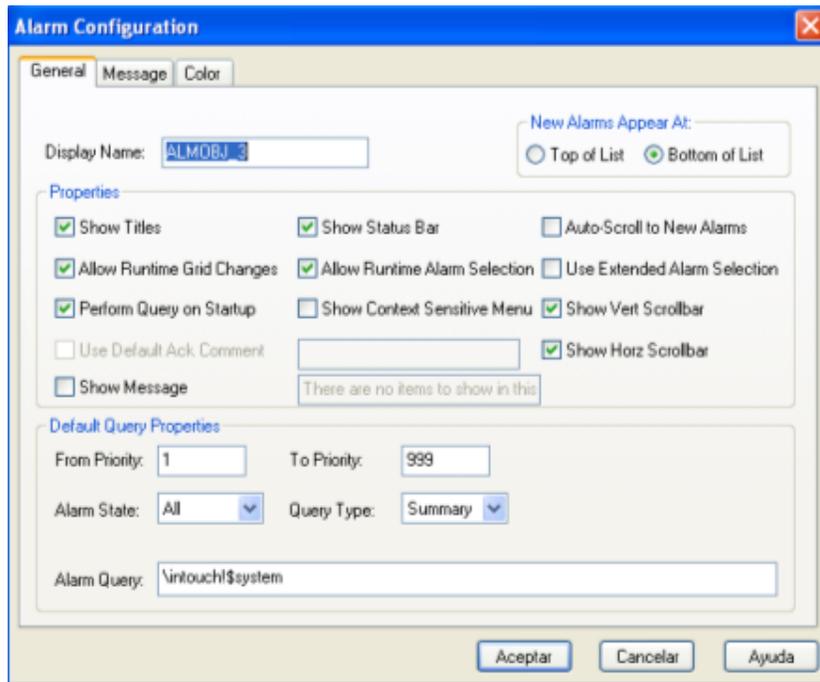


Fig. II-68 Cuadro de Diálogo para configuración de un Objeto de Alarma

Display Name: Nombre unívoco del display que estamos parametrizando.

New Alarms Appear At: Alarmas nuevas al principio del display (Top), o al final (Bottom).

Show Titles: Muestra o no títulos.

Show Vert Scroll Bar: Muestra o no la barra de scroll vertical.

Show Horz Scroll Bar: Muestra o no la barra de scroll horizontal.

Show Status Bar: Muestra o no la barra de estado.

Allow Runtime Grid Changes: Permite o no cambios en la grilla de visualización durante el runtime.

Perform Query on Startup: Ejecuta consulta sobre alarmas al arrancar.

Auto Scroll to New Alarms: Se posiciona automáticamente en la nueva alarma.

Allow Runtime Alarm Selection: Permite selección de alarmas en runtime.

Use Extended Alarm Selection: Usa selección extendida de alarmas.

From ... To Priority: Niveles de prioridad de alarma mayor y menor que se displayarán en el objeto de alarmas.

Alarm State: Define si queremos ver todas las alarmas (all) o sólo las no reconocidas (unack).

Query Type Seleccione Alarm Summary para displayar las alarmas activas actualmente. Seleccione Alarm History para displayar las alarmas activas y también aquellas que han dejado de serlo.

Alarm Query: Tipo de query. Por defecto es \InTouch!\$system (alarmas del propio nodo, del grupo \$system). Es posible cambiar el nodo del que llegan las alarmas (\nodo1\InTouch!\$system), o el grupo (\InTouch!grupo1).

Utilice las pestañas superiores de Mensaje y Color para seleccionar los campos que desea visualizar en cada display de alarma, así como los colores de la alarma en estado de Set, Reconocimiento, Reset y Evento.

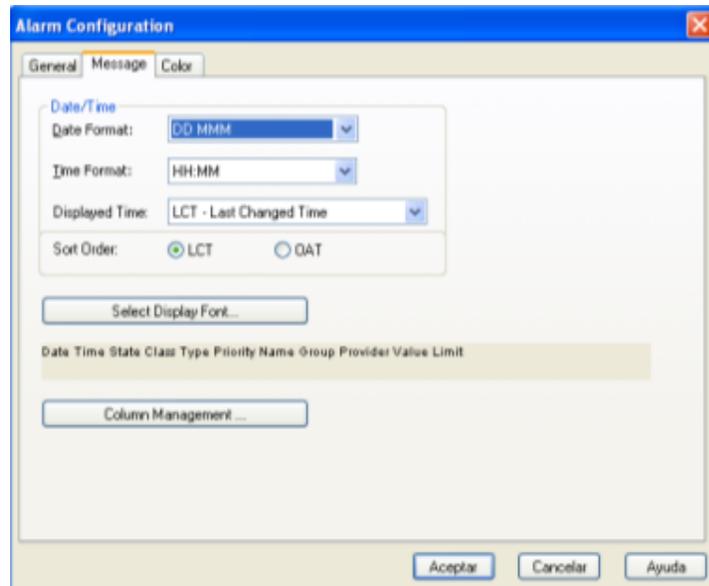


Fig. II-69 Cuadro de Diálogo para configuración de un Objeto de Alarma /Message

Date: Definimos si queremos que aparezca la fecha, y en caso positivo su formato

Time: Definimos si queremos que aparezca la hora, y en caso positivo su formato

Alarm State: Estado de la Alarma

Alarm Class: Clase de la Alarma

Alarm Type: Definimos si queremos ver el tipo de alarma: HIHI, LO, etc Tipo de condición de alarma

LDEV: Para condiciones de desviación menor de alarma

SDEV: Para condiciones de desviación mayor de alarma

OPR: Cuando el operador provoca con un cambio una alarma

Priority: Definimos si queremos ver la prioridad de la alarma.

Alarm Name: Nombre de la Alarma.

Group Name: Definimos si queremos ver el grupo de alarma asociado

Alarm Provider: Nodo Proveedor de la Alarma.

Value at Alarm: Definimos si queremos ver el valor current del registro dentro de la pantalla de alarmas.

Limit: Definimos si queremos ver el valor límite de la alarma del registro.

Operator: Seleccionar si queremos asociar el número de identificación del operador con la alarma.

Comment: Definimos si queremos ver el comentario asociado.

2.1.3.12.1.8 Configuración de Alarmas/Eventos

Disponemos de tres menús para la configuración de las alarmas. A ellos se accede desde el menú Special/ÆConfigure

GENERAL: Configuración de parámetros de las alarmas en pantalla

ALARM LOGGING: Configuración del fichero de alarmas

ALARM PRINTING: Configuración de la impresión de alarmas

2.1.3.12.1.9 Creación de Condiciones de Reconocimiento

Podemos crear pulsadores de reconocimiento utilizando el campo .Ack en un pulsador.

Es también posible reconocimiento por grupo, por tag seleccionado, por display de alarmas, etc.

2.1.3.12.1.10 Campos de las Alarmas

.Ack: Campo discreto de lectura/escritura para reconocimiento de alarmas.

Tagname.Ack=1;

.AckNombreGrupo;

.AckVariableGrupo;

.Alarm: Campo discreto de lectura que se activa cuando se activa el tagname o grupo.

GrupoAlarma.Alarm;

.AlarmClass: Devuelve la clase de la alarma.

.AlarmComment: Devuelve el comentario de la alarma.

.AlarmDate: Devuelve la fecha de la alarma.

.AlarmDevDeadband: Campo analógico de lectura/escritura que controla el porcentaje de desviación mayor y menor de las alarmas. Por ejemplo, para cambiar el porcentaje de desviación al 25%, utilizaremos:

Tagname.AlarmDevDeadband=25;

.AlarmEnable: Discreto de lectura/escritura, que activa/desactiva eventos y alarmas de un tagname o grupo.

GroupVariable.AlarmEnable=1;

.AlarmGroupSel: Devuelve el grupo al que pertenece la alarma.

.AlarmLimit: Devuelve el límite de la alarma.

.AlarmName: Devuelve el nombre de la alarma.

.AlarmOprName: Devuelve el nombre del operador de la alarma.

.AlarmOprNode: Devuelve el nombre del operador del nodo de la alarma.

.AlarmPri: Devuelve la prioridad de la alarma.

.AlarmProv: Devuelve el proveedor de la alarma.

.AlarmState: Devuelve el estado de la alarma.

.AlarmTime: Devuelve la hora de la alarma.

.AlarmType: Devuelve el tipo de la alarma.

.AlarmValDeadband: Analógico de lectura/escritura que controla el deadband de una alarma o grupo.

Tagname.AlarmValDeadband=10;

.AlarmValue: Devuelve el valor de la alarma.

.DevTarget: Campo analógico de lectura/escritura que controla la desviación mayor/menor.

Tagname.DevTarget=500;

.HiHiLimit, .HiLimit, .LoLimit, .LoLoLimit: Discreto de lectura y escritura que indica los límites de una alarma

.HiHiStatus, .HiStatus, .LoStatus, .LoLoStatus: Discreto sólo de lectura que indica el estado en que se encuentra una alarma.

.MajorDevPct: Entero de lectura/escritura para controlar el porcentaje de desviación mayor de la alarma.

.MinorDevPct: Entero de lectura/escritura para controlar el porcentaje de desviación menor de la alarma.

.MajorDevStatus: Discreto que indica si la alarma se encuentra en estado de desviación mayor por porcentaje.

.MinorDevStatus: Discreto que indica si la alarma se encuentra en estado de desviación menor por porcentaje.

.NameDevuelve: El nombre actual de la alarma

.Normal: Discreto de lectura que cuando vale 1 indica que no hay alarmas para el tag especificado

.ROCPct: Lectura/escritura. Monitoriza y controla el valor de Radio de cambio de una alarma.

.ROCStatus: Indica si una alarma se encuentra en estado ROC

2.1.3.12.2 Eventos

Los eventos representan mensajes de estado normal del sistema y no requieren respuesta por parte del operador. Un evento se produce cuando se produce alguna condición del sistema, por ejemplo cuando un operador entra en el sistema.

Eventos y Condiciones que se pueden dar:

ACK Se ha reconocido una alarma

ALM Se ha producido una alarma

EVT Se ha producido un evento

RTN El tagname ha vuelto a su estado normal desde el estado de alarma

SYS Evento de sistema

USER Ha cambiado \$Operador

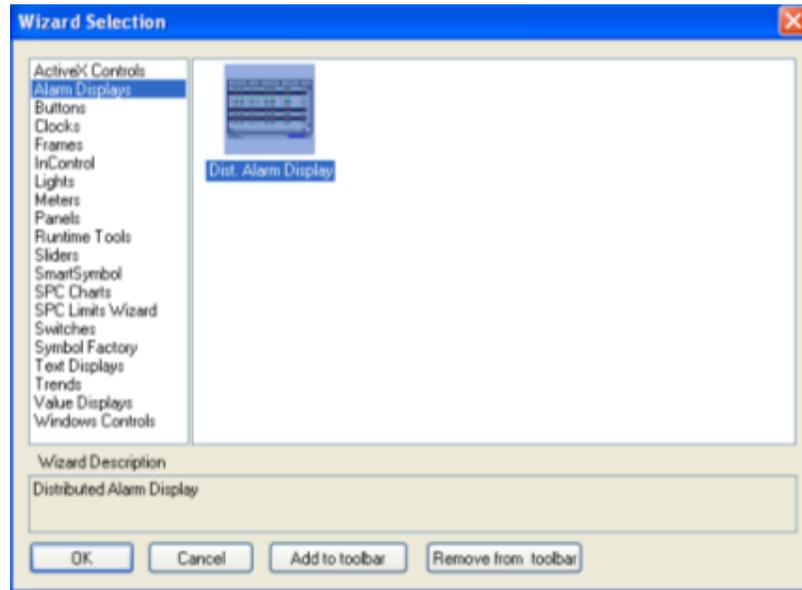
DDE Un cliente DDE ha hecho un POKE sobre un tagname

LGC Una Quicksript ha modificado el valor de un tagname

OPR Un operador ha modificado el valor de una tagname usando un Value Input (entrada de teclado).

2.1.3.12.3 CONFIGURACION DE UN VISOR DE ALARMAS

Seleccionaremos el símbolo de *Dist. Alarm Display* de entre los *WIZARDS* (categoría ALARM DISPLAYS).



Lo colocamos en la pantalla en una posición libre de la misma. Recuerde que puede mover o cambiar el tamaño del objeto.

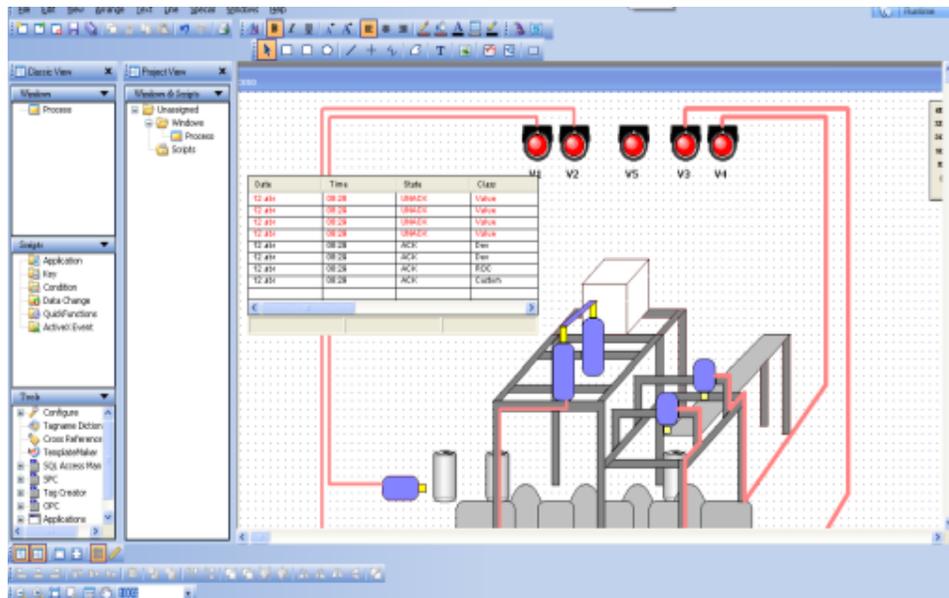


Fig. II-71 Inserción de un Display de Alarmas

Hacemos doble clic sobre el objeto de alarmas para configurarlo y aparecerá el cuadro de diálogo mostrado en la figura 2.70 anteriormente mostrada.

2.1.3.12.4 COMUNICACIONES I/O

2.1.3.12.4.1 La Comunicación DDE

DDE es un protocolo de comunicaciones desarrollado por Microsoft para intercambio de datos entre aplicaciones Windows.

DDE es un sistema estándar en WINDOWS de muy sencillo uso, en el que la comunicación se establece automáticamente entre programas que contemplan la estructura DDE (cliente- servidor). Un programa que puede mandar datos al bus DDE es un programa servidor. Un programa cliente puede recibir datos DDE. Ello permite que podamos muy fácilmente crearnos programas con gestiones especiales en VBASIC, EXCEL, etc., y pasar los datos a **InTouch** sin necesidad de crear un programa de comunicaciones.

Así ocurre con los servidores de autómatas que dispone **WONDERWARE**, entre los que se incluyen la práctica totalidad de los PLCs más conocidos del mundo con comunicación tanto punto a punto como en red. Por supuesto, podemos direccionar un servidor DDE a un puerto de comunicaciones y otro servidor a otro puerto, con lo que podemos compartir información que venga de distintos PLCs o sistemas de campo.

La comunicación DDE se basa en una convención con estos tres parámetros:

- Aplicación
- Tópico
- Elemento

Cuando queremos desde otra aplicación (p.e. Excel), enlazar vía DDE un dato **InTouch**, los parámetros son:

APLICACIÓN	VIEW
TOPICO	TAGNAME
ELEMENTO	{nombre del tag que queremos enlazar}

2.1.3.12.5 InTouch Access Name

Para poder enlazar datos vía DDE de otras aplicaciones Windows a **InTouch**, debemos crear un Access Name. A cada Access Name podemos asociar una Aplicación y un Tópico.

2.1.4 El sistema de almacenamiento de datos.

2.1.4.1 CONCEPTO DE BASE DE DATOS

Una base de datos es un conjunto, colección o depósito de datos almacenados en un soporte informático de acceso directo. Los datos deben estar interrelacionados estructurados.

Dada la importancia que tienen en el mundo real las interrelaciones entre los datos, es imprescindible que la base de datos sea capaz de almacenar éstas interrelaciones, al igual que hace con otros elementos (como las entidades y atributos), siendo ésta una diferencia esencial respecto a los ficheros donde no se almacenan las interrelaciones.

La redundancia de los datos debe ser controlada, de forma que no existan duplicidades perjudiciales ni innecesarias, y que las redundancias físicas, convenientes muchas veces a fin de responder a objetivos de eficiencia, sean tratadas por el mismo sistema, de modo que no puedan producirse incoherencias. Por tanto, un dato se actualizará lógicamente por el usuario de forma única, y el sistema se preocupará de cambiar físicamente todos aquellos campos en los que el dato estuviese repetido, en caso de existir redundancia física.

La actualización y recuperación en las bases de datos debe realizarse mediante procesos bien determinados, incluidos en un conjunto de programas que se encargan de la gestión de la base de datos y que se denominan sistemas gestores de bases de datos (**S.G.B.D**); procedimientos que han de estar diseñados de modo que se mantenga la integridad, seguridad y confidencialidad de la base.

El concepto de base de datos ha ido cambiando y configurándose a lo largo del tiempo, en la actualidad, y de acuerdo con estas características que acabamos de analizar, podemos definir la base de datos como:

"Colección o depósito de datos integrados con redundancia controlada y con una estructura que refleje las interrelaciones y restricciones existentes en el mundo real; los datos, que han de ser compartidos por diferentes usuarios y aplicaciones, deben mantenerse independientes de éstas, y su definición y descripción, únicas para cada tipo de datos, han de estar almacenadas junto con los mismos. Los procedimientos de actualización y recuperación comunes y bien determinados, habrán de ser capaces de conservar la integridad, seguridad y confidencialidad del conjunto de los datos"

2.1.4.2 COMPONENTES DE UN SISTEMA DE BASE DE DATOS

Un sistema de base de datos es algo más que simples datos o que un conjunto de datos en combinación con unos programas de gestión. Un sistema de base de datos está formado por los siguientes componentes:

2.1.4.2.1 Datos

Las características más importantes de la información en estos sistemas es que va a estar integrada y compartida.

Integrada: La Base de datos puede considerarse como una unificación de varios ficheros de datos, que son tratados como uno solo, y en el que se ha eliminado totalmente, o en parte, la redundancia de datos.

Compartida: Los datos pueden compartirse entre varios usuarios distintos. Es posible que varios de estos usuarios accedan al mismo tiempo al mismo elemento de información (acceso concurrente).

2.1.4.2.2 Equipo (Hardware)

Conjunto de dispositivos físicos utilizados para almacenar y procesar los datos.

Ordenadores, utilizados para procesar los datos de la Base de datos: pueden ser mainframe, miniordenador u ordenador personal. El mainframe y los miniordenadores fueron utilizados tradicionalmente para soportar el acceso de varios usuarios a una base de datos común. Los ordenadores personales eran empleados, inicialmente, para manejar bases de datos autónomas controladas y manipuladas por un usuario único. No obstante, actualmente, también pueden conectarse a una red *cliente/servidor*, garantizando el acceso de varios usuarios a una base de datos común almacenada en unidades de disco y controladas por un ordenador servidor. El servidor puede ser otro ordenador personal más potente, o bien, un miniordenador o un mainframe.

Volúmenes de almacenamiento. Generalmente son unidades de disco que constituyen el mecanismo de almacenamiento principal para las bases de datos.

Otros dispositivos, como unidades de cinta, terminales, impresoras, etc.

2.1.4.2.3 Programas (Software)

Un sistema de base de datos incluye dos tipos de programas:

El *software de propósito general*, para la gestión de la base de datos, comunmente llamado *Sistema Gestor de Bases de Datos (S. G.B.D., o también D. B. M S, en inglés)*. El S.G.B.D maneja todas las solicitudes de acceso a la base de datos formuladas por los usuarios y los programas de aplicación.

El *software de aplicación*, que usa las facilidades del S.G.B.D para manipular la base de datos con el fin de llevar a cabo una función específica en la gestión de la empresa (por ejemplo: la gestión de almacén). Puede estar desarrollado en un lenguaje de programación estándar, tal como COBOL o C, o en un lenguaje propio de los S.G.B.D denominados lenguajes de cuarta generación (4GL).

2.1.4.2.4 Personal

En un sistema de base de datos intervienen un número importante de usuarios, que podemos clasificar en tres grupos:

Administrador de la base de datos (A.B.D.). Son los encargados de diseñar la estructura de la base de datos y los responsables de que el sistema funcione correctamente. El A.B.D. se encarga de autorizar el acceso a la base de datos, de coordinar y vigilar su utilización y de adquirir los recursos necesarios de software y hardware. El A.B.D. es el responsable cuando surgen problemas como violaciones de seguridad o una respuesta lenta del sistema. El A.B.D. tiene, entre otras, las siguientes funciones:

Definición del esquema: Decidir el contenido de la base de datos, eligiendo cuales son los datos que interesa tener almacenados y organizarlos de la mejor forma posible, creando el esquema conceptual, que se escribirá mediante un lenguaje de definición de datos (DDL).

Definición de las estructuras de almacenamiento y método de acceso: Debe decidir sobre la forma en que se van a almacenar los datos sobre los soportes físicos en los que se grabará la base de datos y la correspondencia entre esta estructura de almacenamiento y el esquema conceptual.

Modificación del esquema y de la organización física si los requerimientos cambian.

Decidir los controles de autorizaciones para el acceso a los datos: Es el que concede diferentes tipos de autorizaciones al resto de los usuarios de la base de datos.

Especificar las restricciones de integridad: Debe definir los procedimientos de validación que habrán de ejecutarse cada vez que se actualiza la base de datos. Estas restricciones son consultadas por el SGBD cada vez que se realiza una actualización de los datos.

Programadores de aplicaciones, que se encargan de desarrollar las aplicaciones que manejan datos de la base de datos. Estas aplicaciones contendrán solicitudes de datos al S.G.B.D que luego serán procesados por los programas de la aplicación que tendrán como finalidad resolver problemas específicos de la empresa.

Usuarios finales, que son personas que no tienen por que tener conocimientos informáticos y que pueden manipular los datos (examinarlos y actualizarlos) con la ayuda de las aplicaciones, o bien de lenguajes de consulta no procedimentales (no es necesario indicar el algoritmo de acceso a los datos), tipo SQL, o bien, mediante herramientas basadas en sistemas de menús. Se distinguen tres tipos de usuarios finales:

Usuarios especializados: Aquellos que son capaces de escribir ciertas aplicaciones para la BD, para su uso propio.

Usuarios casuales: Aquellos que realizan consultas a través de un procesador de consultas. Esas consultas pueden ser creadas por ellos mismos o por otras personas.

Usuarios ingenuos: Aquellos que solo acceden a través de aplicaciones previamente escritas por otros usuarios.

2.1.4.3 ESTRUCTURA GENERAL DEL SGBD

Gestor de ficheros: Se encarga de las estructuras de datos que se emplean para representar la información almacenada en disco. Además es el encargado de asignar espacio en disco.

Gestor de BD: Es el interface entre los datos de bajo nivel almacenados en la BD y los programas de aplicaciones y consultas de datos.

Procesador de consultas: traduce las proposiciones del lenguaje de consultas a instrucciones de bajo nivel para que las entienda el gestor de BD. Además el

procesador de consultas trata de traducir una petición del usuario a otra equivalente pero más eficiente.

Compilador DDL: Convierte las proposiciones DDL en un conjunto de tablas que contienen metadatos.

Ficheros de datos: Almacenan la información (los datos).

Diccionario de datos: Almacena información referida a la estructura de la BD. Se usa continuamente. Se debe poner un gran énfasis en un buen diseño y una implementación adecuada del diccionario de datos.

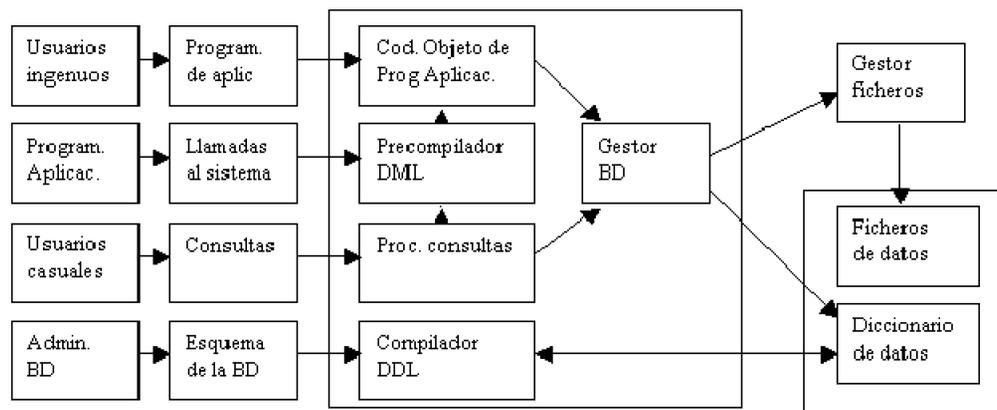


Fig. II-72 Estructura general del SGBD

CAPITULO III

III.SISTEMA DE ENVASADO VOLUMETRICO.

En este capítulo se detallarán las partes componentes del sistema mecánico y estructural del sistema de envasado volumétrico, sus criterios de selección, dimensiones y capacidades que permiten que el sistema cuente con las especificaciones didácticas para ser usada dentro de un centro educativo de nivel superior.

3.1 Módulo de envasado.

Para la construcción de la estructura del módulo, hay que considerar las dimensiones horizontales y verticales, ya que sobre esta base se colocan los dispositivos que desarrollarán las operaciones dentro del proceso, sensores, actuadores, gusano transportador, dispensador y los cilindros dosificadores, que debido al trabajo que van a realizar necesitan tener una base fija y robusta, capaz de soportar las fuerzas de reacción a las acciones que los actuadores realizan.

Teniendo en cuenta que es un sistema didáctico la estructura soporta bajos esfuerzos, pero debe estar constituida por un material de fácil adaptabilidad y también fácil de mecanizar, entre las opciones existentes se ha elegido el perfil de aluminio serie 25 mk, de la figura 3.1, que son utilizados para la construcción de stands de feria, la

electrónica y la construcción de aparatos, por su bajo costo y facilidad de montaje se decidió utilizar este tipo de perfil para la construcción de la bancada del proyecto



Fig. III-1 Perfil de aluminio 25 Mk

3.2 Diseño de la estructura.

Una vez decidido el material utilizado hay que tener claro el proceso que se va a llevar a cabo en el sistema de envasado, para esto desarrollamos un esquema general del proceso y los elementos que intervendrán.

Por el hecho de que es un sistema didáctico y como se mencionó anteriormente, es necesario crear una estructura transparente, fácil de manipular y que pueda ser acoplada a otras estructuras desarrolladas para formar procesos continuos y completos se decidió construir la estructura mostrada en la figura 3.2, de la cual se proporciona mayor información en el anexo 1.



Fig. III-2 Estructura del sistema.

Como se puede visualizar la bancada fue diseñada de tal forma que presenta la facilidad de reconocer de forma fácil y rápida los componentes que fueron instalados en ella para que el estudiante o la persona que vaya a hacer uso de la misma los pueda manipular y de ser el caso desensamblar y ensamblar los mismos dentro de las prácticas de laboratorio a desarrollarse con el equipo.

El ensamblaje de los elementos del sistema se lo realizó de forma sencilla gracias al perfil de aluminio seleccionado, ya que presenta la facilidad de adaptar a los elementos del equipo mediante bases fáciles de colocar y sujetar mediante pernos hexagonales que actúan como prisioneros para dotar a la bancada de estabilidad y modularidad.

3.3 Partes del equipo.

El proceso cuenta con los siguientes sistemas:

- Sistema de almacenamiento del líquido.
- Sistema de dosificación de recipientes.
- Sistema de Transporte.
- Sistema de llenado.
- Sistema de colocación de tapas.

3.4 Sistema de dosificación de recipientes.

La función de este sistema es la de entregar al proceso los recipientes (tarros metálicos de 1/8 de litro), en los cuales se realiza la dosificación del líquido, por medio de un cilindro en el cual se colocan los tarros formando una columna de los mismos, que va soltando un recipiente a la vez hacia el piso de la bancada para que el cilindro colocado en la parte delantera empuje al recipiente hacia el transportador de la máquina, como se puede observar en la figura 3.3.

Una vez que el tarro ha sido colocado en el piso, un sensor de presencia inductivo colocado a la entrada del sistema de transporte nos avisa su posición, en caso de no existir recipientes, el sistema envía una alarma que indica la ausencia de los mismos y detiene el proceso hasta que los recipientes sean colocados en el dispensador, una vez corregido el problema el proceso continua.



Fig. III-3 Sistema de dosificación de recipientes.

3.5 Sistema de almacenamiento de líquido.

Este sistema provee al equipo del líquido a ser dosificado, es un recipiente de 6 litros de capacidad en el cual es depositado el líquido que será almacenado hasta que sea necesaria la dosificación del mismo.

El líquido almacenado es medido por medio de un sensor flotador al cual ha sido adaptado un potenciómetro que nos da una variación de resistencia conforme el nivel del líquido varíe con lo que provee de información al controlador para que este tome las acciones necesarias.

Como se puede apreciar en la figura 3.4 este sistema está ubicado en la parte superior de la bancada, soportado sobre las bases de aluminio y justo al lado de los cilindros dosificadores para evitar pérdidas en el transporte del líquido.

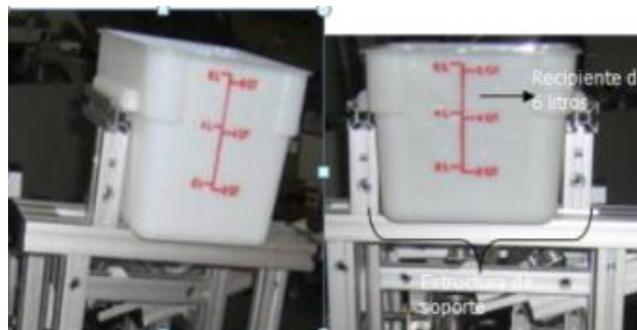


Fig. III-4 Sistema de almacenamiento de líquido.

3.6 Sistema de Transporte.

El transporte de los recipientes hacia las diferentes estaciones del proceso es realizado por un tornillo sin fin, gusano transportador, que se encuentra conectado por medio de una banda al motor monofásico de baja velocidad. El cual por medio de un eje sujeto a la estructura realiza un movimiento circular que al introducirse el recipiente entre las ranuras del gusano genera un desplazamiento lineal del mismo hacia las estaciones donde se realizan los procesos de llenado, colocación de tapa, y sellado de tapa, también es el que saca el recipiente del sistema.

En la figura 3.5 se puede apreciar el gusano transportador colocado sobre la estructura del proceso.

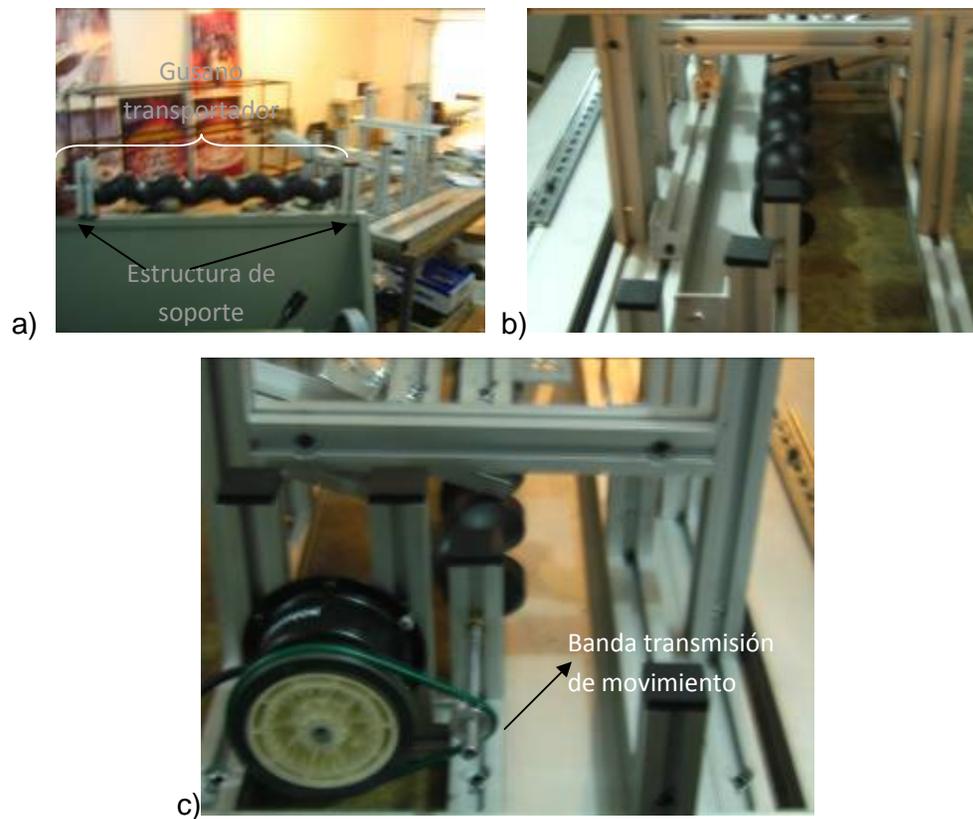


Fig. III-5 Gusano Transportador

3.7 Sistema de llenado.

El sistema de dosificación propiamente dicho está compuesto por dos cilindros conectados mecánicamente, y controlados por una misma válvula 5/2 como se puede ver en la figura 3.6, esto para realizar en la etapa 1 el movimiento que proporciona la succión del líquido a dosificarse y en la etapa 2 la descarga del líquido hacia el recipiente contenedor del mismo.

Se toma como referencia este método ya que el cuerpo del cilindro dosificador proporciona una capacidad definida, por su radio y la altura a la que puede llevar el líquido lo que hará que se tenga un volumen constante en la dosificación, prácticamente sin pérdidas.



Fig. III-6 Cilindros dosificadores

Este sistema trabaja conjuntamente con el sistema de almacenamiento de líquidos, ya que si no existe líquido a dosificar, los cilindros dosificadores deberán suspender su trabajo para evitar que ingrese aire en la línea de líquidos y produzca una contaminación.

3.8 Sistema de colocación de tapas.

Una vez dosificado el líquido dentro de los tarros contenedores, es necesario sellar el contenido de los mismos para asegurar que este se conserve con la mayor frescura posible, es por esto que es necesario el sellado de los tarros, para esto contamos con un sistema de dos cilindros ubicados uno a continuación del otro, el primero realiza una contención de las tapas hasta cuando el sensor ubicado en la parte inferior después del sistema de dosificación, nos indique que el tarro está posicionado para colocar la tapa.

Una vez que el contenedor se encuentra en esta posición se suelta la tapa y se la coloca en la boca del mismo por efecto de la gravedad, y se realiza el prensado de la tapa con el cilindro ubicado perpendicular al tarro en la misma posición.

Todo este proceso se lo puede apreciar de mejor forma en la fig. III.7.

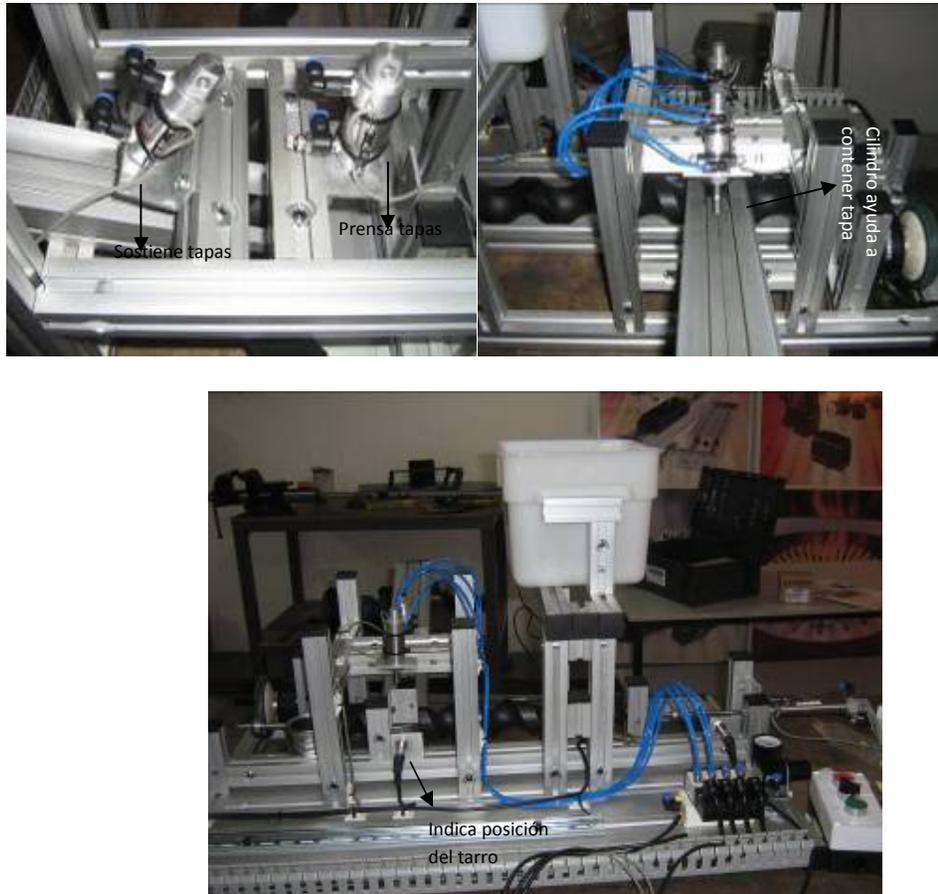


Fig. III-7 Sistema de colocación y sellado de tapas.

Una vez realizado el proceso de colocación y sellado de tapas, es necesario que el recipiente cumpla con el ciclo y se llevado hasta el final de la línea de producción, para lo cual es accionado una vez más el gusano transportador.

Todos los sistemas antes mencionados convergen para hacer que el sistema de envasado volumétrico para líquidos de baja densidad tome forma y realice un trabajo conjunto para obtener un producto terminado y además pueda ser acoplado a otros sistemas que han sido construidos como sistemas didácticos también.

3.9 Funcionamiento del módulo de envasado volumétrico.

Una vez que hemos visto las partes constitutivas del módulo, es necesario realizar una explicación detallada del proceso de envasado, para esto es necesario la implementación del diagrama del proceso como se muestra en la fig III.8.

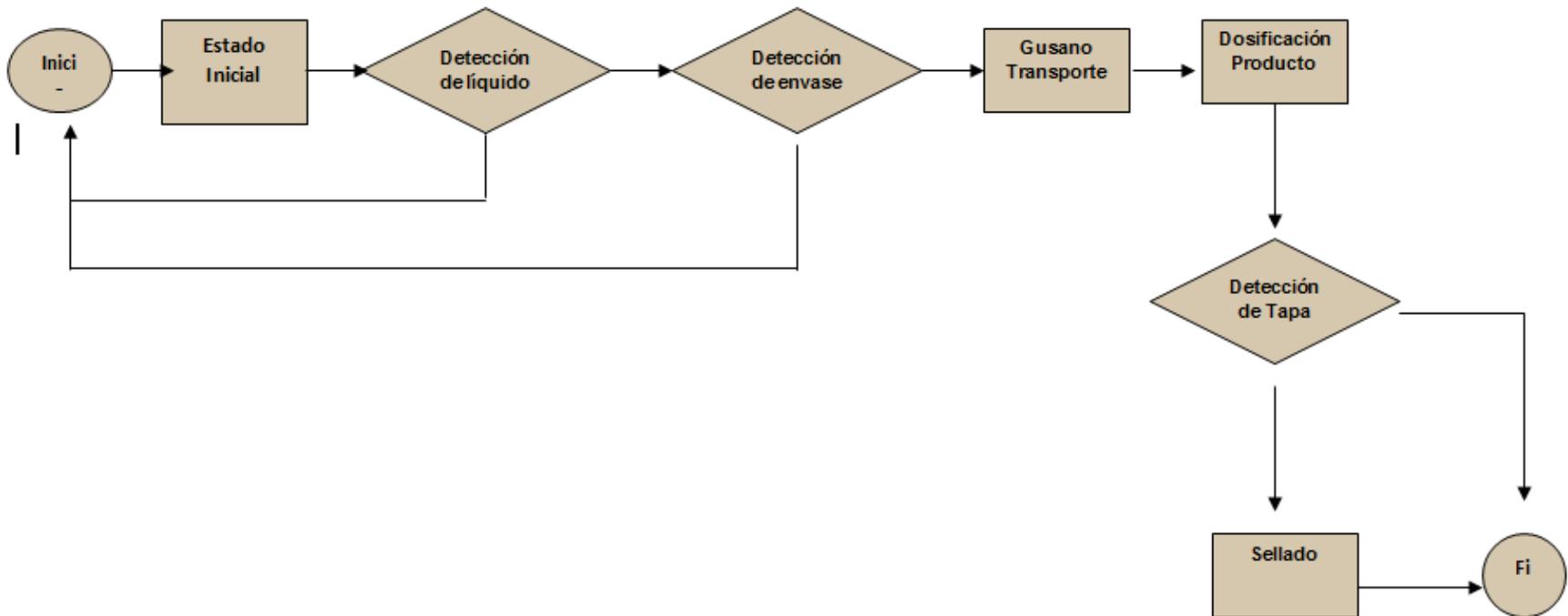


Fig. III-8 Diagrama de proceso de envasado volumétrico.

Como se puede apreciar en el diagrama del proceso el envasado volumétrico consta de 7 etapas definidas que son las siguientes:

3.10 Estado Inicial.

Este es el estado del que parte la máquina al iniciar el proceso, en el cual el operador o manipulador debe estar pendiente de la cantidad de recipientes y tapas, además de verificación de alarmas y posibles eventos que pudieran ocurrir para que el proceso no pueda arrancar, una vez corregidos todos estos inconvenientes estamos listos y la maquina se inicializa con los siguientes estados en sus actuadores:

1. El motor principal que da movimiento al sistema gusano transportador, debe encontrarse apagado, para asegurar que cuando se produzca el empuje del recipiente hacia el gusano transportador, este tenga la posición adecuada que es el con la primera ranura de frente hacia el dosificador de recipientes.
2. El cilindro A que hace el empuje para llevar al recipiente hacia el gusano transportador, necesita estar desactivado, retraído A⁻, para dar lugar a que se posicione el recipiente en el cual se va a dosificar. Figura 3.9
3. Los cilindros dosificadores que se encuentran conectados mecánicamente y actúan con una sola válvula deben estar en posición extendida, para tener desde el inicio una cantidad de líquido prevista para la dosificación B+ y C+, para ayudar a contener el líquido dentro del cilindro dosificador, es necesario que la válvula de paso para la dosificación se encuentre cerrada en la posición inicial.
4. El cilindro que ayuda a realizar la contención de tapas cilindro D debe estar desactivado en posición extendida D⁺, lo que nos asegura que las tapas están siendo retenidas por este.
5. Y por último el cilindro de sellado de tapas debe estar en posición retraída para actuar únicamente cuando se detecte el recipiente en su posición y la tapa haya sido colocada correctamente sobre este y necesite realizar la acción de sellado, si no existe tapa para realizar el sellado este no realiza ninguna acción y continúa con el resto del proceso.

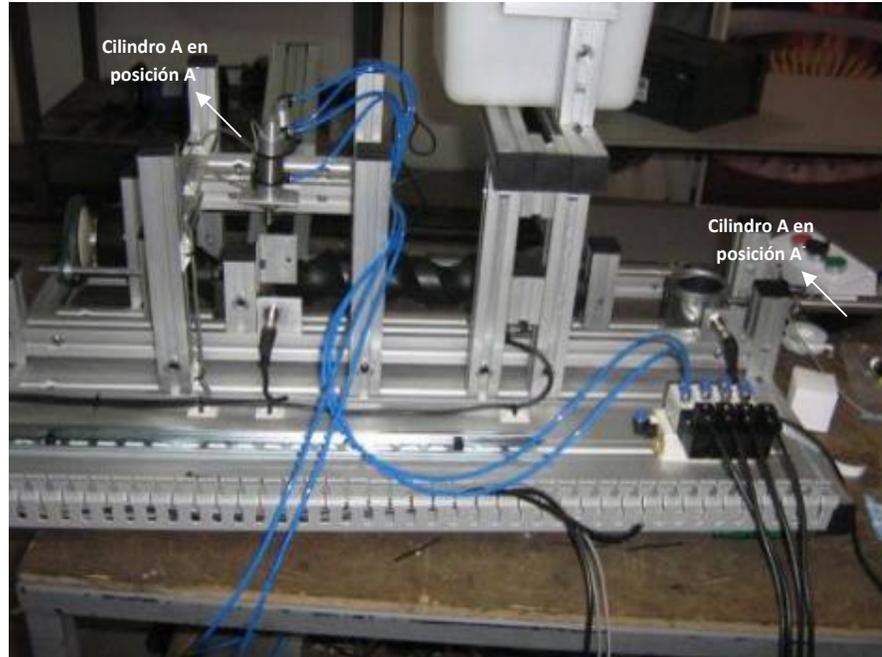


Fig. III-9 Posiciones iniciales del sistema

3.11 Detección de líquido y Detección de envase.

Para la detección del líquido a dosificar utilizamos una boya, flotador (S10) que está conectado a un potenciómetro el cual nos da una variación de resistencia que acondicionada por medio una tarjeta electrónica se ha convertido esta a una variación de voltaje normalizada de 0 a 10Vcd, y está conectado hacia una entrada analógica del plc, para darnos una medida de de nivel real dependiendo de la señal recibida desde el flotador.

En caso de que el nivel del líquido se encuentre en el mínimo, el proceso no iniciará se quedará a la espera de que el líquido sea dosificado en el contenedor de líquido para poder iniciar el proceso.



Fig. III-10 Detección de líquido.

Al igual que en el caso anterior, para detectar la presencia de recipientes para el envasado es utilizado un sensor inductivo (S9), ubicado en el cilindro contenedor de recipientes el cual nos debe estar enviando una señal continua de existencia de envases, caso contrario el proceso no iniciará.



Fig. III-11 Detector de envases.

En caso de que el proceso este en funcionamiento y se incumpla con los requerimientos de estos sensores, se continuará con la dosificación del último recipiente ingresado a la línea y a continuación se bloqueara y detendrá la producción hasta que se cumplan estas dos condiciones, además presentará una alarma en el sistema de supervisión que nos indique la falta de cualquiera de los dos elementos.

3.12 Gusano Transporte.

El sistema de transporte como se mencionó anteriormente está compuesto por un motor que está acoplado a un eje el cual se encuentra inmerso en el gusano transportador, tornillo sin fin que forma un sistema de engranaje que con el recipiente emulando un sistema de transmisión lineal, y desplaza al mismo hacia los lugares de dosificación y colocación de tapas, este inicia su movimiento cuando el sensor magnético del cilindro A (S1), se encuentra activado.

Esto quiere decir que el recipiente está en la posición inicial, lista para empezar con el transporte del mismo hacia las estaciones siguientes, lo que hace encender al gusano transportador. Se detiene cuando los recipientes se han posicionado en las respectivas estaciones para realizar las operaciones siguientes. O cuando se para el proceso por completo.



Fig. III-12 Gusano transportador.

3.13 Dosificación del producto.

Una vez que el sensor inductivo colocado debajo de los cilindros dosificadores (S6), ha detectado la presencia de un recipiente, el gusano transportador se detiene ubicando al recipiente en la posición exacta para realizar la descarga del líquido a ser dosificado.

Cuando esto sucede se abre la válvula de apertura de líquidos y el cilindro B empieza a bajar hasta retraerse completamente, lo que hace que el cilindro C (dosificación), vacíe todo el contenido succionado hacia el recipiente colocando un volumen definido por el diámetro y la altura del cilindro.



Fig. III-13 Cilindros dosificadores.

3.14 Detección de tapa.

En esta etapa se realiza también la comprobación de la existencia de tapas en el dispensador de las mismas mediante un sensor inductivo (S9), que se encuentra ubicado en la estructura de la parte posterior al cilindro que realiza la acción de sellado de las tapas.

Una vez que se verifica la tapa el cilindro D pasa hacia su posición D⁻, dejando caer la tapa hacia el tarro que ya ha sido colocado por medio del gusano transportador.

De no existir tapa para realizar el sellado, el proceso continúa, y saca el tarro sin sellarse, para efectos de corrección del problema se emite una alarma en el sistema de supervisión que avisa al operador que es necesaria la colocación de tapas en el sistema.



Fig. III-14
Dosificación de
tapas

3.15 Sellado.

Una vez que se ha realizado la colocación de la tapa sobre el recipiente con el contenido dentro, el cilindro E cambia su posición de E⁻ a E⁺, realizando así una fuerza perpendicular hacia el recipiente sellando el recipiente con la tapa metálica y dejando así el producto terminado.

Una vez que se ha sellado el recipiente nuevamente se prende el gusano transportador para sacar el tarro sellado hacia el final del proceso.

De no existir tapa para el sellado el sistema da una alarma de esto y previene al operador que el recipiente saldrá sin tapa y que se realice la colocación de las mismas

en el dosificador de tapas, de ser este el caso el cilindro E no realiza ninguna acción y se queda en la posición E'.

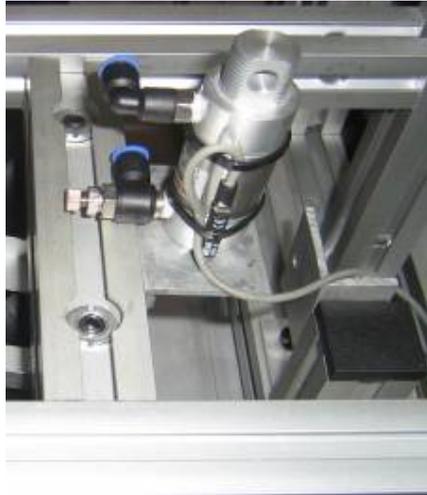


Fig. III-15 Cilindro de sellado.

3.16 Dimensionamiento de elementos constitutivos del módulo.

Una vez descritos cada uno de los elementos y sistemas que conforman el módulo de envasado volumétrico se describen las condiciones y los cálculos que se han tomado en cuenta para su respectiva selección y dimensionamiento, que hace que el equipo funcione de forma óptima.

3.17 Dimensionamiento del cilindro de empuje hacia gusano de transporte.

La fuerza que pueden desarrollar los cilindros neumáticos es la característica más importante por la cual comienza la etapa de selección. El valor de la fuerza depende exclusivamente del diámetro del pistón y de la presión del aire comprimido con que se alimenta el cilindro. Los cilindros realizan su acción en posiciones determinadas como por ejemplo el cilindro del brazo de ajuste, su posición es vertical y a la vez su desplazamiento.

La elección de cilindros puede ser resuelta con ayuda de gráficos, los cuales muestran los valores teóricos de fuerza según las diferentes presiones y diámetros de cilindros.

La ecuación que lo sustenta es:

$$F = P * A = 10 * P * \pi \left(\frac{d^2}{4} \right) * n$$

Donde:

F: Fuerza.

P: Presión de trabajo.

d: Diámetro del pistón.

n: Fuerza de rozamiento (idealmente 0.9).

Para el caso del cilindro de empuje hacia el transportador se utilizaron los siguientes datos.

CAPITULO IV

IV. DESARROLLO DEL SOFTWARE

4.1 PROGRAMACIÓN Y CONFIGURACION S7-1200 – Step 7 Basic v10.5

El procedimiento para la programación del PLC S7-1200 en el STEP7 Basic v10.5, es una serie de pasos nada complejos, a la vez que por medio del presente se detallará paso a paso la configuración y programación de dicho dispositivo facilitando el aprendizaje.

Inicialmente ejecutaremos la aplicación del STEP 7 Basic v10.5 que la podemos ubicar en el Menú de Inicio – Programas – Siemens Automation localizando aquí la aplicación Totally Integrated Automation Portal v10 (TIA) que será la que ejecutemos para iniciar con nuestro proceso.

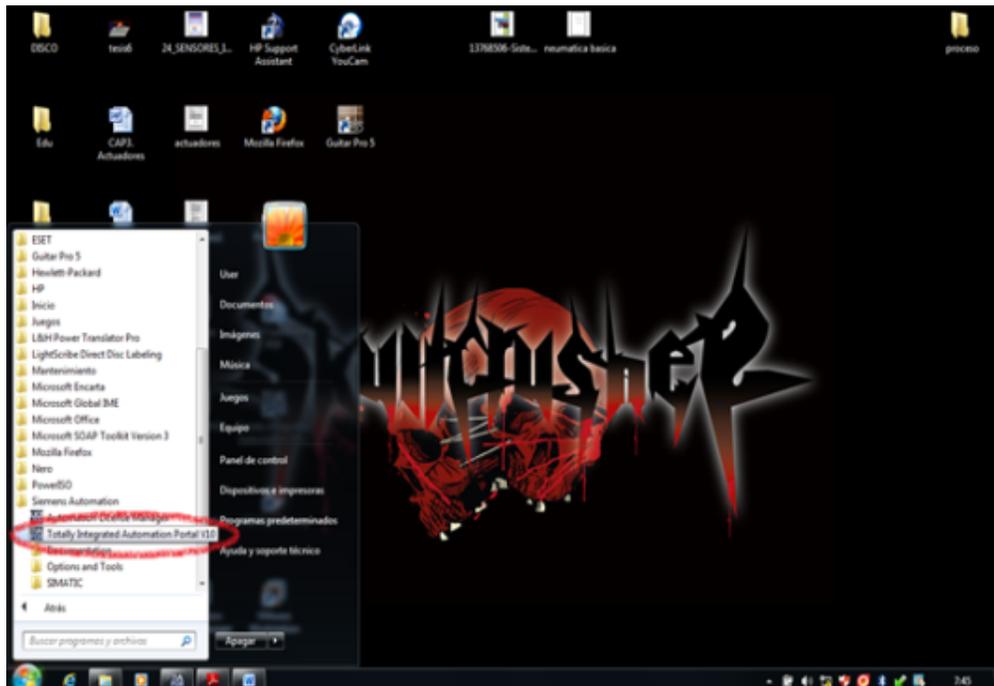


Fig. IV-1 Ubicación de la Aplicación

Al ejecutar la aplicación nos encontramos con la pantalla inicial del TIA en la que por defecto nos aparecerá seleccionada la opción de “Abrir Proyecto”, además de la tabla de proyectos existentes en el PG/PC.

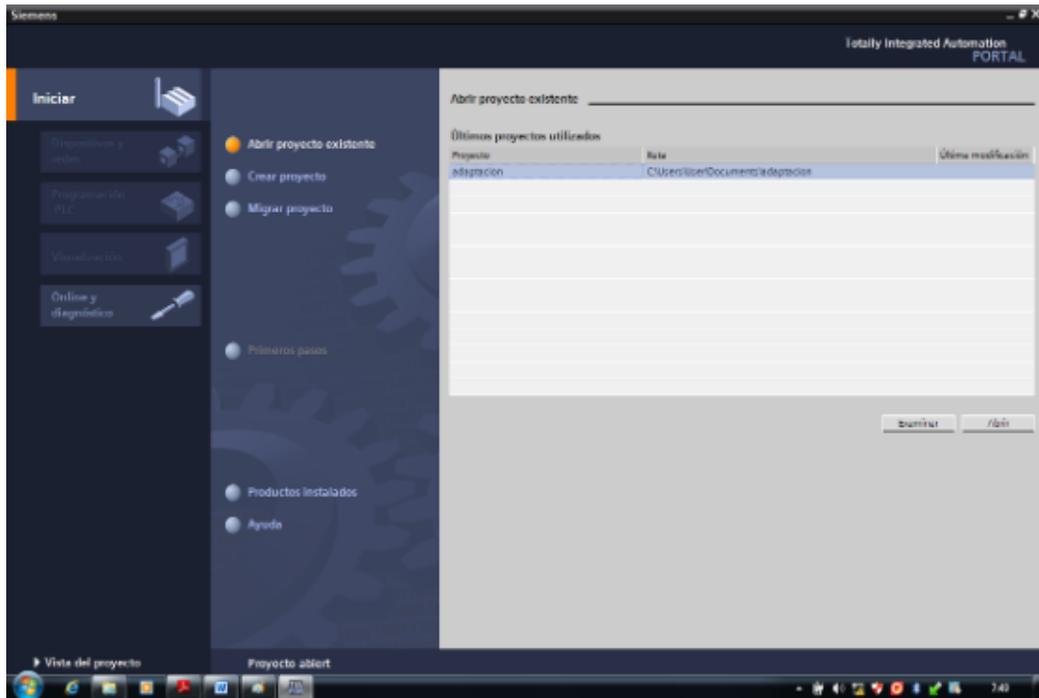


Fig. IV-2 Pantalla Inicial TIA asignaciones por defecto.

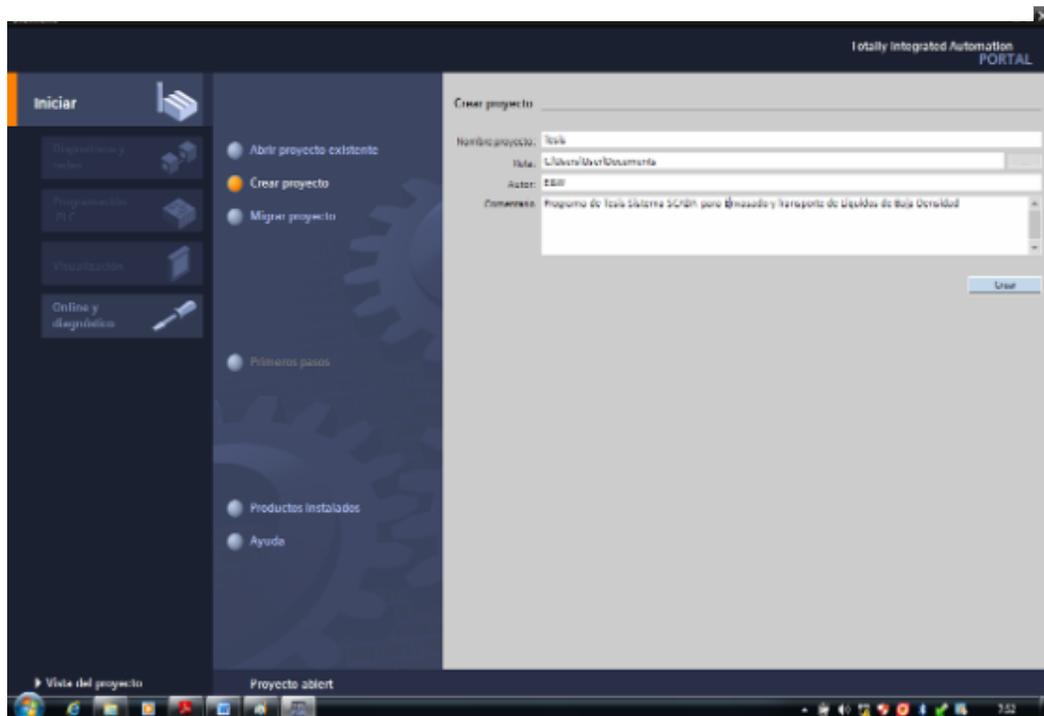


Fig. IV-3 Creación del Nuevo Proyecto

Para nuestro caso seleccionaremos la opción “Crear Proyecto” ya que empezaremos de cero para la configuración y programación de nuestro dispositivo.

En este paso procederemos a llenar los datos del nuevo proyecto tales como Nombre, Ruta, Autor e Información adicional que se la pueda plasmar en un comentario.

Al darle clic en “Crear” luego de haber llenado los datos del programa, aparecerá una nueva ventana en la que por defecto tendremos asignado “Primeros Pasos”, Aquí tenemos las opciones “Configurar Dispositivo”, “Crear un Programa PLC” y “Configurar una Imagen HMI”. Nosotros escogeremos “Configurar un Dispositivo” para iniciar configurando nuestro hardware.

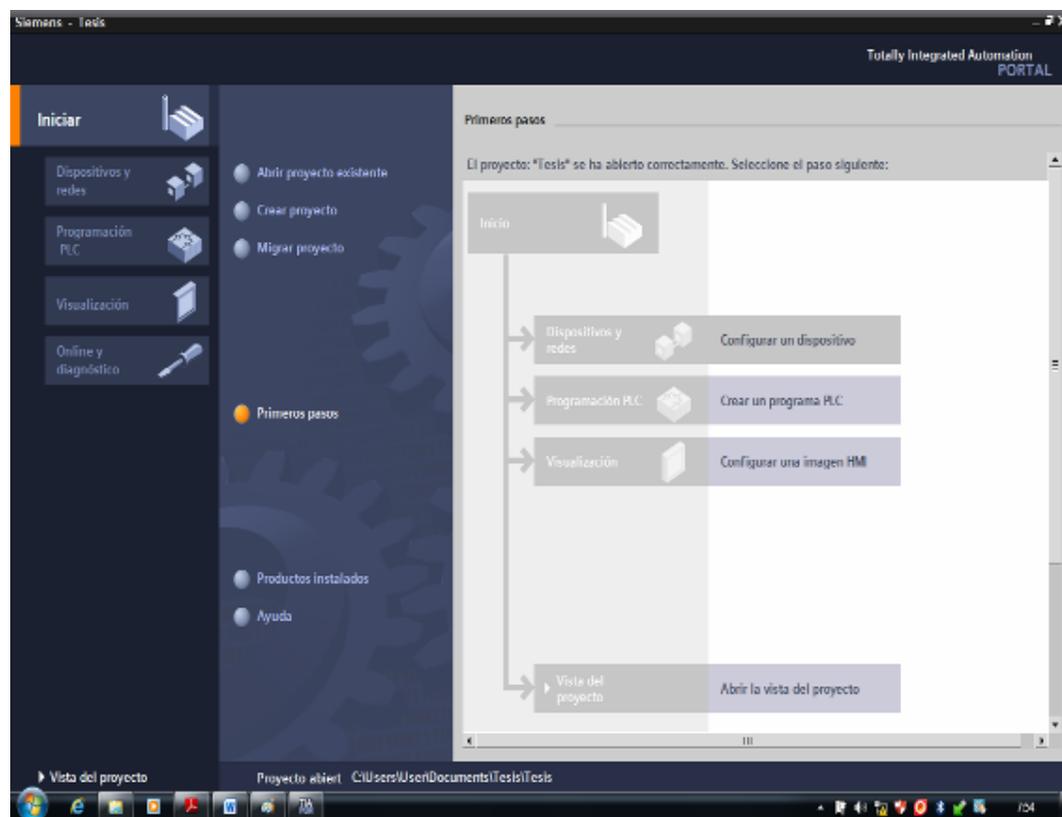


Fig. IV-4 Seleccionar Configura un Dispositivo

Al seleccionar “Agregar el dispositivo” nos aparecerán dos opciones “SIMATIC PLC”, “SIMATIC HMI”, para nuestro caso elegiremos “SIMATIC PLC” y nos aparecerá en una ventana a la derecha todas las CPU de PLC’s que podemos agregar en nuestro caso elegiremos CPU 1212C - 6ES7- 212 – 1BD30 – OXBO

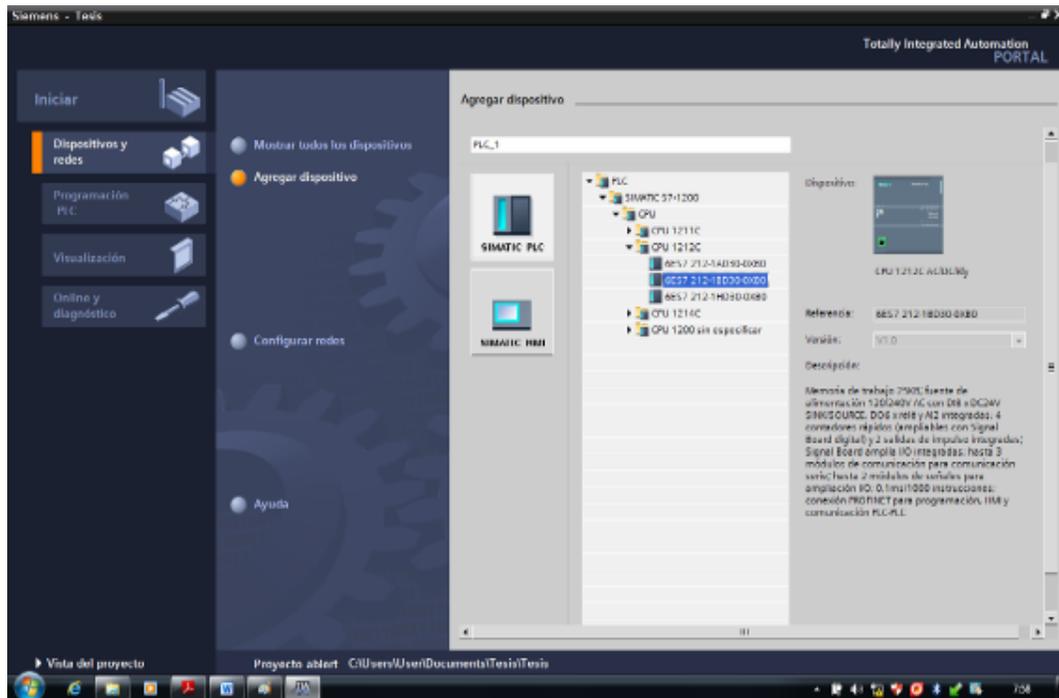


Fig. IV-5 Selección de CPU

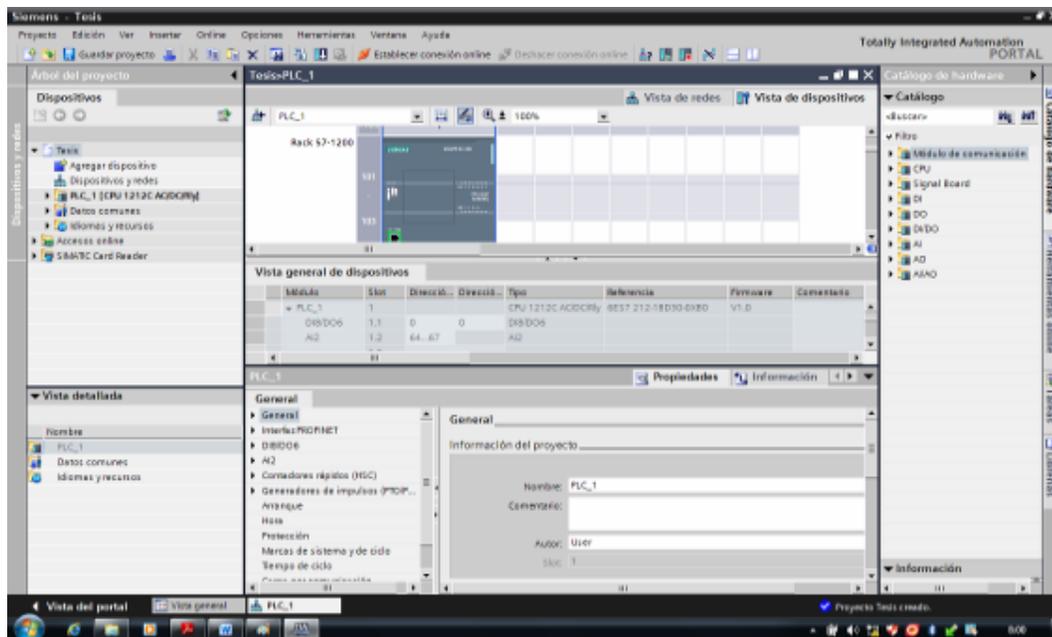


Fig. IV-6 CPU cargado para el programa

Posteriormente nos aparece la ventana de configuración del equipo. Lo que tenemos que hacer ahora es meter los módulos que tenemos en nuestro equipo físicamente: módulos de entradas y salidas (I/O), módulos de comunicación, etc... Para ello

seleccionaremos del catálogo de la derecha los módulos correspondientes y los iremos arrastrando y soltando en su posición correcta.

En el SIMATIC S7-1200 los módulos de comunicación se insertan a la izquierda de la CPU y los módulos de I/O se meten a la derecha. Como máximo puede haber 3 módulos de comunicación y 8 de I/O.

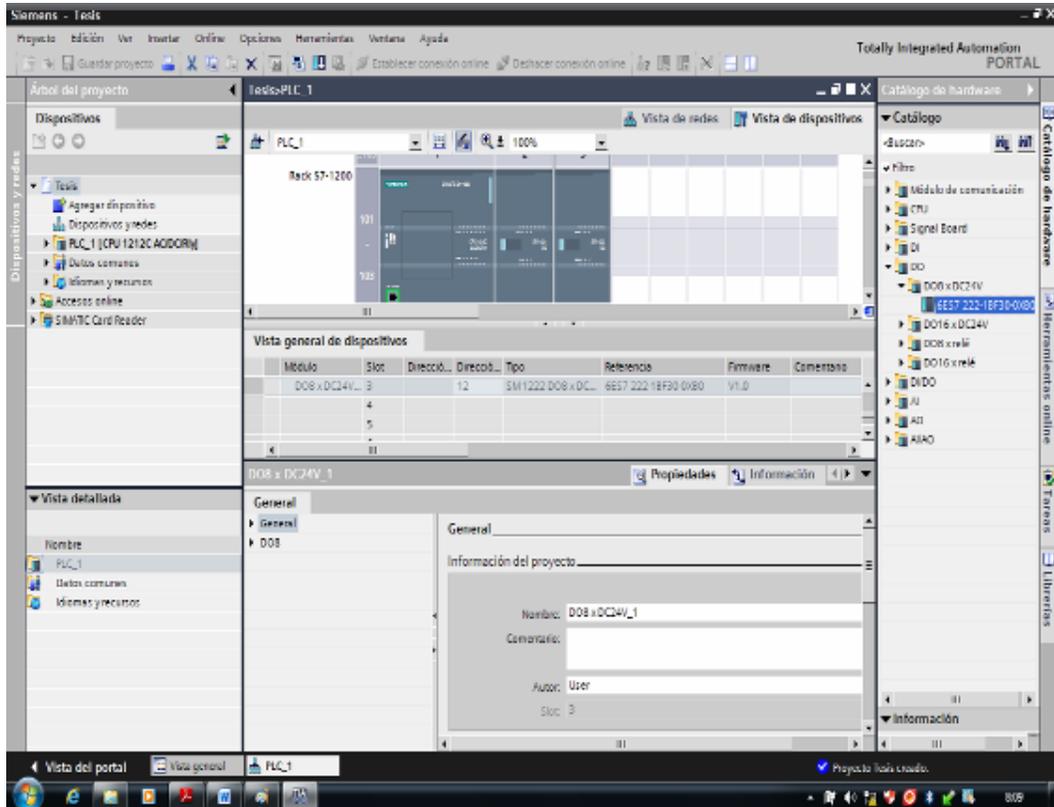


Fig. IV-7 Inserción Módulo de Entradas y Salidas

En la parte de debajo de la pantalla según vamos insertando los módulos nos podemos meter en las propiedades del módulo y ver sus propiedades, el direccionamiento, etc...

Para transferir la configuración seleccionamos la CPU y se nos habilita el icono



que es para transferir pero antes de esto comprobaremos la dirección IP

del PC y del PLC.

Primero la IP del PC le ingreso como 192.168.0.1 o la que sea del rango que no coincida con la del PLC. Por defecto el rango suele ser de 192.168.0.xxxx.

En el PLC la dirección IP se la da pinchando sobre la CPU y en la ventana de propiedades de la parte inferior dentro de la opción PROFINET interface. Ahí es donde le vamos a asignar la IP y máscara correspondiente a nuestro PLC.

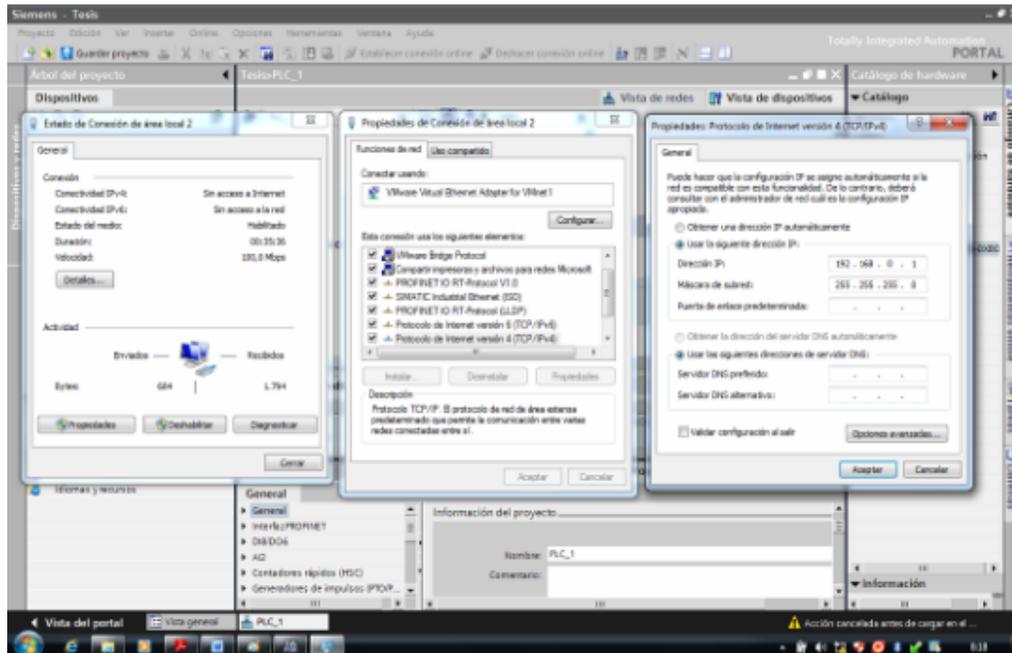


Fig. IV-8 Establecimiento de IP para la PC.

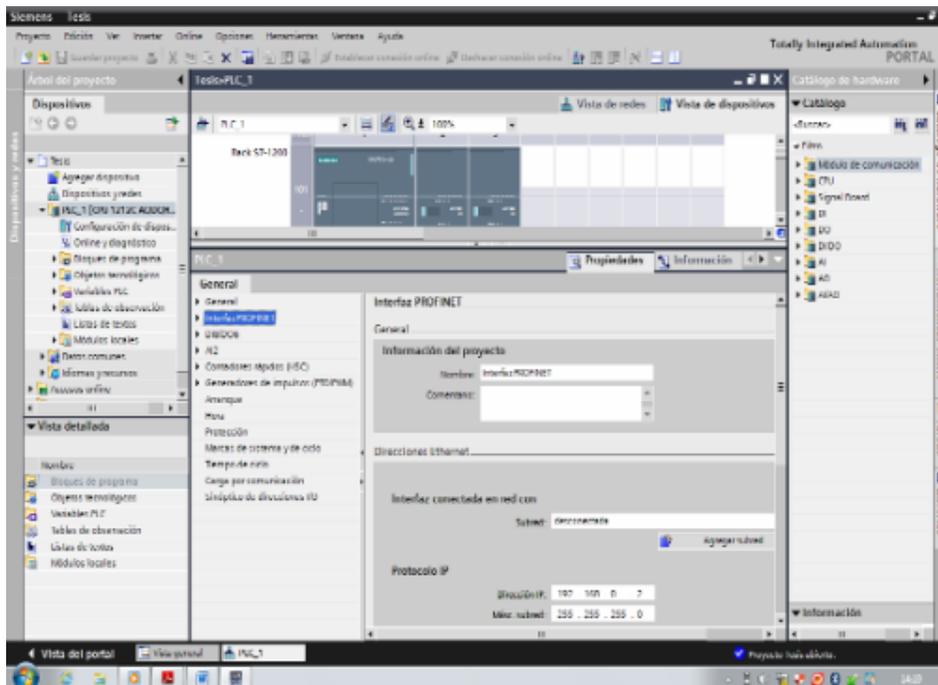


Fig. IV-9 Establecimiento de IP para el PLC.

Una vez realizada la conexión y configuración del hardware nos centramos en la programación del PLC, para esto realizamos una introducción a lo que tiene que ver a la programación estructurada. Las opciones que se tienen para esto es la programación directa mediante lenguaje escalera (KOP), ladder, lenguaje de funciones (FUP), pero si se quiere tener una respuesta óptima del sistema es necesario realizar el diagrama grafset del programa.

4.1.1 HERRAMIENTAS DE AUTOMATIZACIÓN: GRAFCET

Este diagrama funcional permite describir los comportamientos del automatismo en relación a las informaciones que recibe, imponiendo un funcionamiento riguroso, evitando de esta forma incoherencias, bloqueos o conflictos en el funcionamiento. En cada nivel de descripción, este diagrama puede ser modificado o corregido, sin necesidad de volver a partes ya estudiadas.

4.1.1.1 METODOLOGÍA GRAFCET

El Grafset se compone de un conjunto de:

Etapas o Estados a las que van asociadas acciones.

Transiciones a las que van asociadas receptividades.

Uniones Orientadas que unen las etapas a las transiciones y las transiciones a las etapas.

4.1.1.1.1 Etapa.

Una etapa se caracteriza por un comportamiento invariable en una parte o en la totalidad de la parte de mando.

En un momento determinado, y según sea la evolución del sistema:

- Una etapa puede estar activa o inactiva.
- El conjunto de las etapas activas definen la situación de la parte de mando.
- Las etapas se representan por un cuadrado con un número en su parte superior como identificación. La entrada y salida de una etapa aparece en la parte superior e inferior, respectivamente, de cada símbolo.
- El conjunto formado por el cuadrado y la extensión de las entradas y salidas constituye el símbolo completo de la etapa

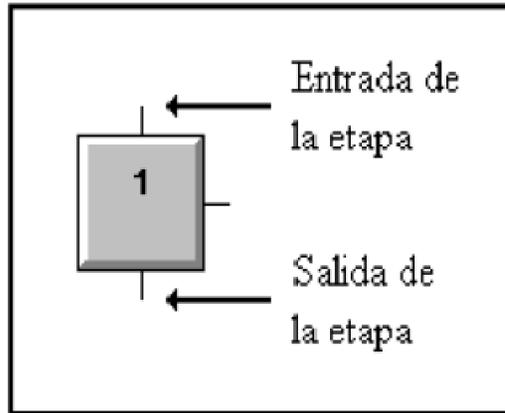


Fig. IV-10 Etapa graficet.

En la depuración del programa, es necesario realizar la identificación de las etapas activas e inactivas, esto se puede realizar mediante el uso de una marca cuadrada negra dentro del símbolo de la etapa correspondiente al estado, Fig. a), esto para etapas que se encuentren activas, para las etapas inactivas se las deja sin marca para realizar su representación Fig. b).

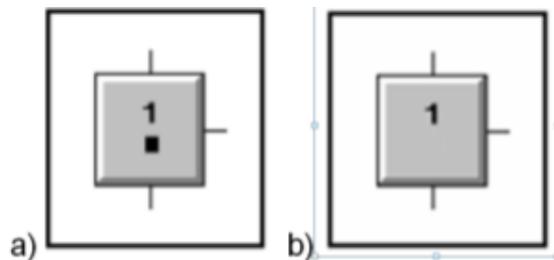


Fig. IV-11 Representación de etapas.

Cuando tenemos transiciones que parten o llegan hacia una misma estas deben converger en un punto antes o después de la etapa designada para ubicar una única transición ya sea de entrada o salida.

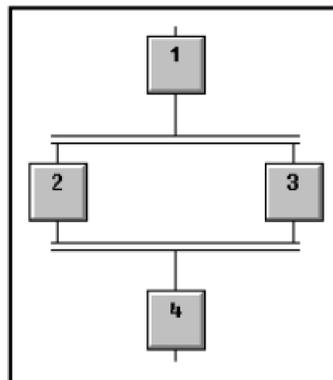


Fig. IV-12 Transiciones desde y hacia una etapa.

4.1.1.1.2 TRANSICIÓN

Una transición indica el posible cambio entre etapas, este cambio se produce cuando existe el flaqueo de la transición.

El flaqueo representa la evolución del proceso, y puede darse cuando las condiciones de la transición anterior son verdaderas y las etapas anteriores se han activado.

Esta secuencia provoca el paso en la parte de mando de un evento a otro. Una transición puede considerarse validada o no validada.

- Validada.- Es cuando todas las etapas inmediatamente unidas a esta transición están activas y las condiciones de transición están activas.
- No validada.- cuando las etapas anteriores no han cumplido los respectivos ciclos, o cuando las condiciones de transición no se han cumplido.

Una transición entre dos etapas se representa mediante una línea perpendicular a las uniones orientadas, también puede llevar una línea paralela a las uniones orientadas. Para facilitar la comprensión del Grafcet cada transición puede ir numerada a la izquierda de la línea perpendicular.

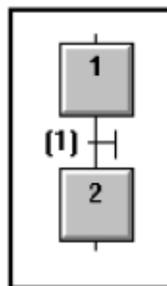


Fig. IV-13 Representación de las transiciones.

Como en toda metodología de programación en los diagramas grafcet pueden añadirse estructuras previamente definidas que nos ayudan con el desarrollo del proyecto, entre estas tenemos.

4.1.1.2 Divergencia

Cuando la etapa 1 está activa, según se cumpla la receptividad asociada a la transición a o la receptividad asociada a la transición b, pasará a ser activa la etapa 2 o bien la etapa 3 respectivamente.

Si la etapa activa es la 2 debe cumplirse la receptividad asociada a la transición a para pasar a la etapa 4 a activa. Si la etapa activa es la 3 debe cumplirse la receptividad asociada a la transición b, para que la etapa 4 pase a estar activa

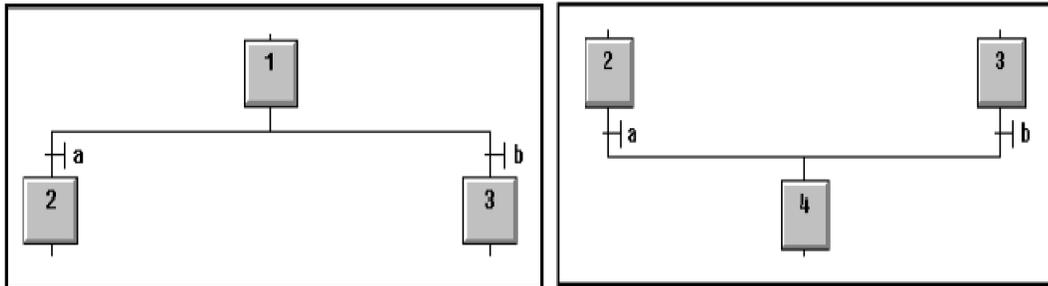


Fig. IV-14 Representación divergencia O

Divergencia en Y.

Estando activa la etapa 1 y si se cumple la receptividad asociada a la transición C, pasan a estar activas las etapas 2 y 3. Y hará que se activa la etapa 4 deben estar activas las etapas 2 y 3 y cumplirse la receptividad asociada a la transición D.

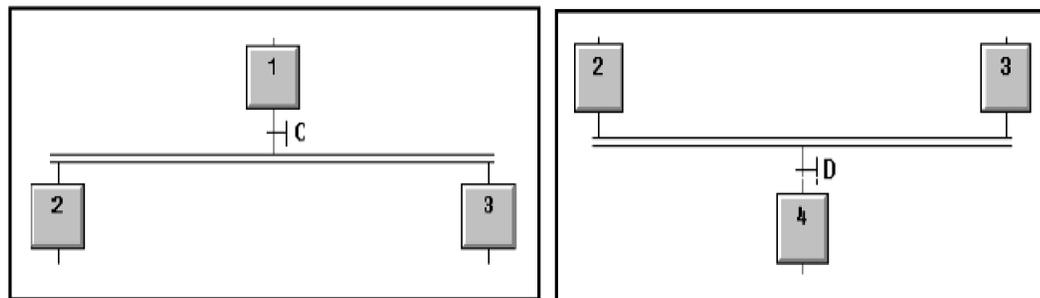


Fig. IV-15 Representación divergencia Y

Una vez conocidas las partes que conforman un diagrama grafcet, así como también las estructuras posibles podemos analizar las estructuras principales.

4.1.1.2.1 ESTRUCTURAS PRINCIPALES

Estas estructuras pueden variar dependiendo de la complejidad del proyecto o de la cantidad eventos que tengamos dentro del sistema. Entre estas tenemos.

4.1.1.2.1.1 SECUENCIA ÚNICA

Conocida también como secuencia lineal o secuencia plana, y se puede considerar que se compone de una sucesión de etapas que son activadas una a continuación de otra. A cada Etapa le sigue solamente una transición y cada transición es validada por

una sola etapa, se puede considerar como la secuencia básica de un diagrama grafcet.

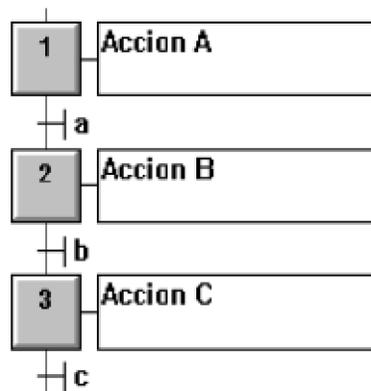


Fig. IV-16 Secuencia única.

Se considera que la secuencia se encuentra activa si al menos una de las etapas se encuentra activa, por el contrario se considera una secuencia inactiva.

4.1.1.2.1.2 SECUENCIAS SIMULTÁNEAS O PARALELISMO ESTRUCTURAL

Cuando el flaqueo de una transición se produce y esta conduce a la activación de varias etapas al mismo tiempo, podemos decir que son secuencias simultáneas.

Después de la activación de estas secuencias, las evoluciones de las etapas activas en cada una de las secuencias son independientes. Para asegurar la sincronización de la desactivación de varias secuencias al mismo tiempo, generalmente se ponen etapas de espera recíproca.

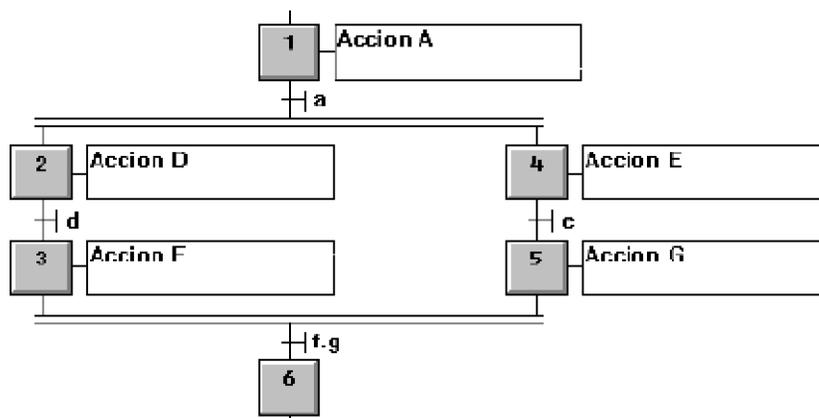


Fig. IV-17 Secuencia simultáneas.

Estas activaciones simultáneas pueden presentarse una o varias veces durante el proceso o sistema que se esté implementando.

4.1.2 DESARROLLO E IMPLEMENTACIÓN DEL DIAGRAMA GRAFCET DEL PROCESO.

Una vez conocidos las pautas para el desarrollo de un programa con metodología grafcet podemos iniciar con el desarrollo del programa del para el sistema de envasado volumétrico de líquidos de baja densidad.

4.1.2.1 DIAGRAMA GRAFCET PARA EL SISTEMA DE ENVASADO VOLUMETRICO

El siguiente esquema representa las acciones que desarrollara el sistema.

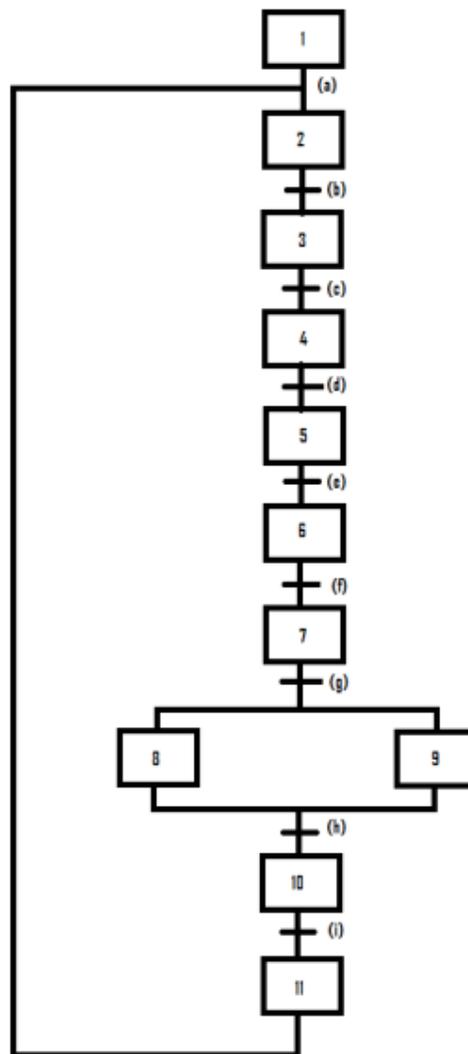


Fig. IV-18 Acciones que desarrollara el sistema.

4.2 DESARROLLO DE LA APLICACIÓN EN INTOUCH.

4.2.1 ANALISIS DE REQUERIMIENTOS DE LA APLICACIÓN.

En esta etapa es necesario definir los requerimientos necesarios para que el módulo didáctico sea funcional para el desarrollo de las prácticas que se van a desarrollar en este. En especial asignación de permisos y recursos para cada uno de los tipos de usuarios definidos que utilizarán el sistema.

Como se mencionó anteriormente la HMI debe ser un sistema amigable con el usuario, lo más completa, simplificada y fácil de navegar.

Como el proceso simula un sistema industrial real de envasado volumétrico, es necesario que el sistema pueda realizar la autenticación de usuarios que nos permita asignar las funciones y permisos definidos para cada uno de los perfiles asignados.

Para este caso se tienen planteados 3 perfiles de usuarios.

Invitado: Como su nombre lo indica es un perfil que simula una persona que no tendrá permisos para realizar acciones de control o supervisión sobre el proceso, a lo único que este tendrá acceso es a la visualización del desarrollo del proceso, pero no podrá cambiar ningún parámetro del mismo.

Operador: Este usuario puede ingresar a la etapa de supervisión, y realizar los cambios operativos dentro del proceso, estos permisos incluyen cambios en el volumen que será depositado en el recipiente, reconocimiento de alarmas, encendido y apagado del proceso.

Ingeniero: Este perfil de usuario tiene los permisos para acceder a todas las opciones del proceso y desarrollar cambios operativos así como también acceder a las opciones de seguridad del sistema (bypass), así como también cambiar de modo manual a modo automático y viceversa.

Una vez definidos los perfiles de usuarios podemos ver la estructura del programa, y en esto tenemos.

- Autenticación de usuarios tres niveles de permisos (invitado, operadores e ingenieros)
- Visualización de datos de nivel de líquido a dosificar, recipientes, tapas, velocidad del proceso, cantidad de producto terminado, fallos del proceso, etc.
- Generación de reportes en Excel
- Una base de datos que guarde la información de alarmas, eventos, usuarios y productos terminados desde el INTOUCH.

De acuerdo con los requisitos previstos para el sistema, es necesario realizar la conexión desde el sistema de almacenamiento de datos, así como también desde el dispositivo central de control en este caso el PLC.

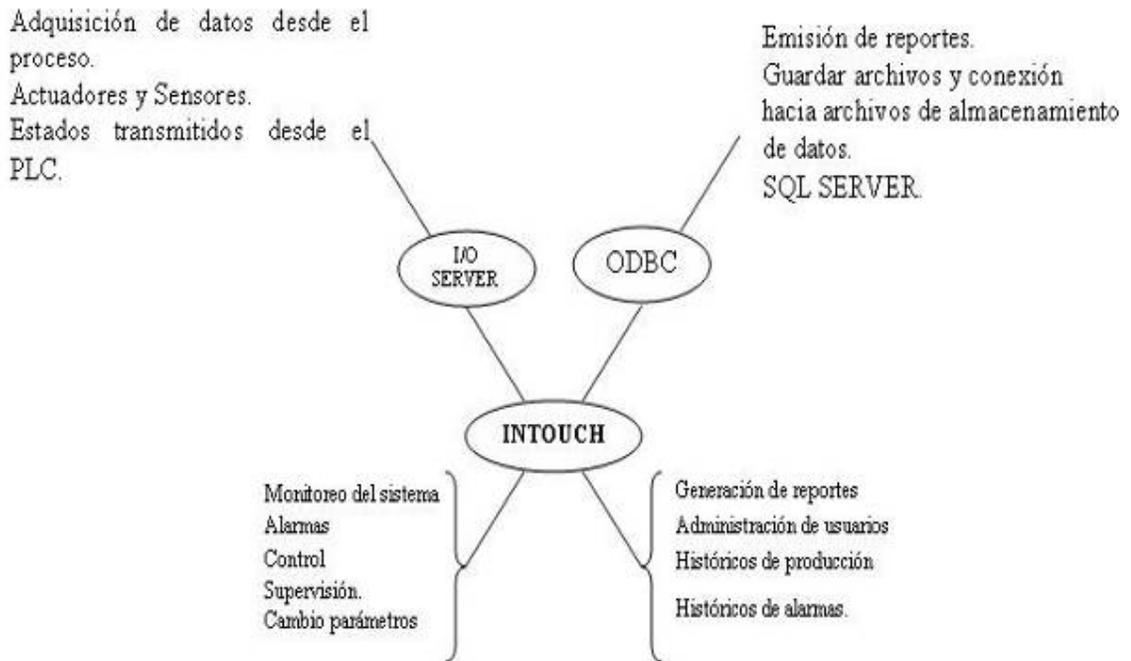


Fig. IV-19 Esquema de conexión con INTOUCH.

Como se puede observar en el diagrama de la figura anterior contamos con dos intermediarios, si se los puede llamar así, para realizar las acciones de enlace entre el intouch y el respectivo gestor de base de datos SQL SERVER 2005, para este caso el ODBC, (Origin Data Base Connection), por otro lado la conexión entre la máquina y el intouch se realiza mediante la conexión de los I/O servers, para nuestro caso el OPC Server, Top Server 5.

4.2.2 Creación de la aplicación.

Tomando en cuenta los requerimientos del sistema descritos al inicio de este capítulo, se tienen que crear las respectivas ventanas y darles los atributos a los botones y scripts, que nos englobará las aplicaciones necesarias para el monitoreo y control del proceso de envasado volumétrico de líquidos de baja densidad.

Para esto y como se describió en la parte teórica del INTOUCH es necesario ingresar al programa, escoger la aplicación que deseamos editar.

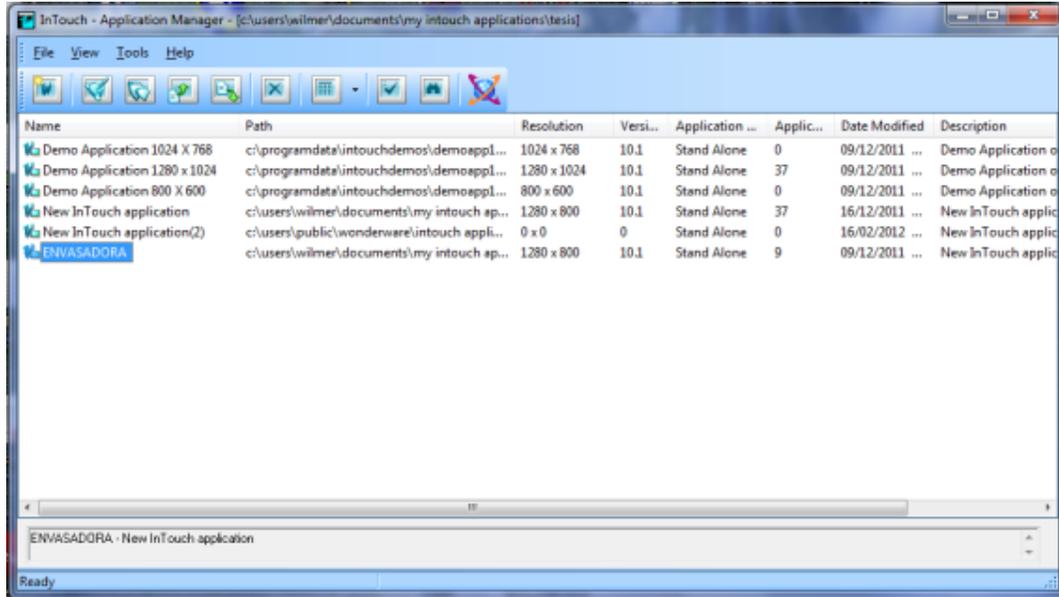


Fig. IV-20 Pantalla APLICACION MANAGER.

Una vez escogida la aplicación esta se abre en el Windows maker para hacer los cambios respectivos, en la interfaz HMI.

Dentro de esta interfaz del programa podemos realizar los cambios respectivos al aspecto de las ventanas, inserción de botones y controles que nos ayudan a la ejecución y supervisión del proceso, mediante la programación de sentencias en los animation links de cada uno de los objetos insertados, la siguiente figura muestra la pantalla principal diseñada para el sistema.



Fig. IV-21 Windows Maker diseño inicial de la HMI.

Como podemos observar es aquí donde creamos todas las pantallas de acceso hacia los diferentes niveles del proceso y de las respectivas opciones de usuario.

Una vez realizada la pantalla inicial es necesario realizar la conexión del PLC con la HMI, para esto se necesita crear la conexión OPC mediante el software TOP SERVER 5.

4.3 Configuración del TOP SERVER 5.

El top server 5 es un servidor OPC, que cuenta con las facilidades de conexión para la CPU S7-1200 de Siemens, es uno de los pioneros en presentar esta ventaja de conexión para este PLC.

Es necesario realizar la configuración de este software y levantar el servicio para establecer la conectividad de la HMI con el PLC.

Para realizar este proceso es necesario seguir el siguiente protocolo de conexión para obtener resultados satisfactorios a la hora de crear una referencia de datos de memoria del PLC, con los datos que serán representados dentro de la interfaz del proceso.

Abrimos el programa ubicado en el directorio de programas del menú inicio.



Fig. IV-22 Directorio y Pantalla principal del top server 5.

En el menú File de la pantalla principal, escogemos la opción **NEW** y nos aparecerá la siguiente pantalla, en la cual nombraremos a la configuración del PLC, podemos darle cualquier nombre o dejarlo con el nombre por defecto.

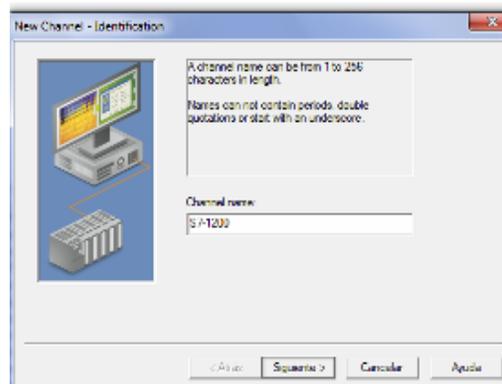


Fig. IV-23 Inicio de configuración de comunicación del PLC.

Como es un OPC certificado por OPC Foundation, este cuenta con varios drivers de comunicación pero el que a nosotros nos interesa es el driver de siemens. Por lo cual damos clic en siguiente y escogemos el driver de siemens TCP/IP ethernet, ya que este será el que nos reconozca la configuración de Profibus, vía Ethernet.

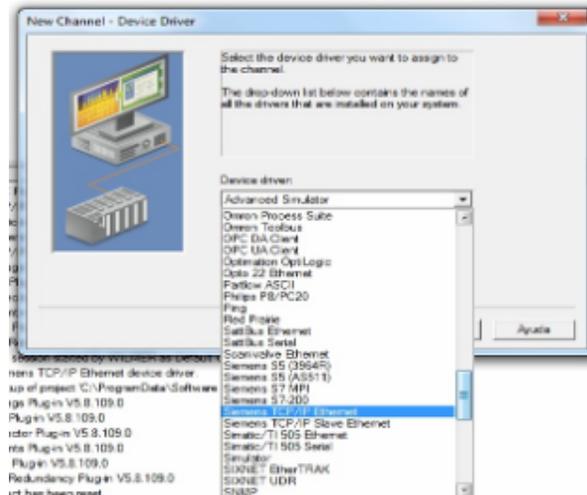


Fig. IV-24 Pantalla de selección de driver de comunicación.

- Una vez escogido el driver de comunicación damos clic en siguiente y pasamos a escoger el adaptador de red con el que vamos a realizar la comunicación para nuestro caso lo dejamos por default.

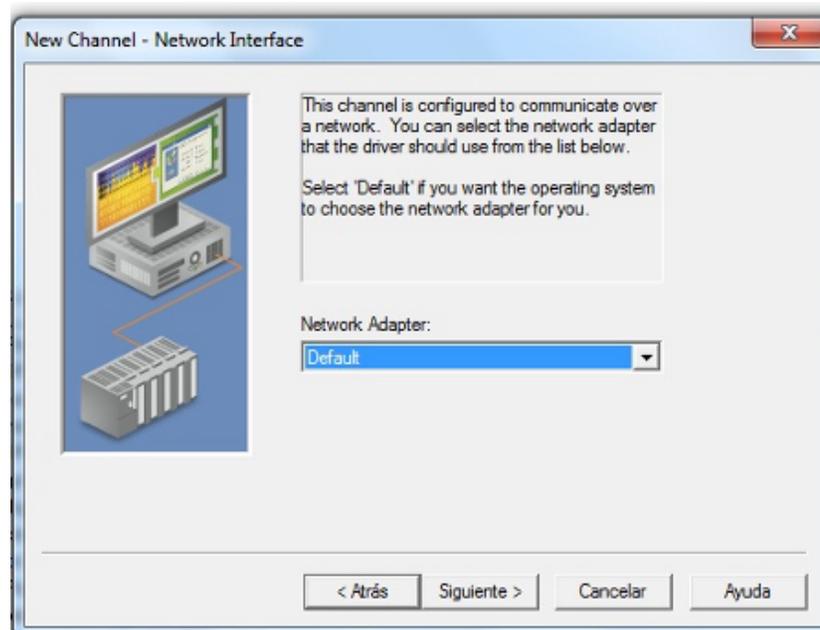


Fig. IV-25 Selección del adaptador de red para la comunicación.

- Una vez escogido el adaptador para la comunicación del sistema pasamos a configuración el modo de actualización de las tags del sistema, para esto seleccionamos el siguiente perfil de configuración.

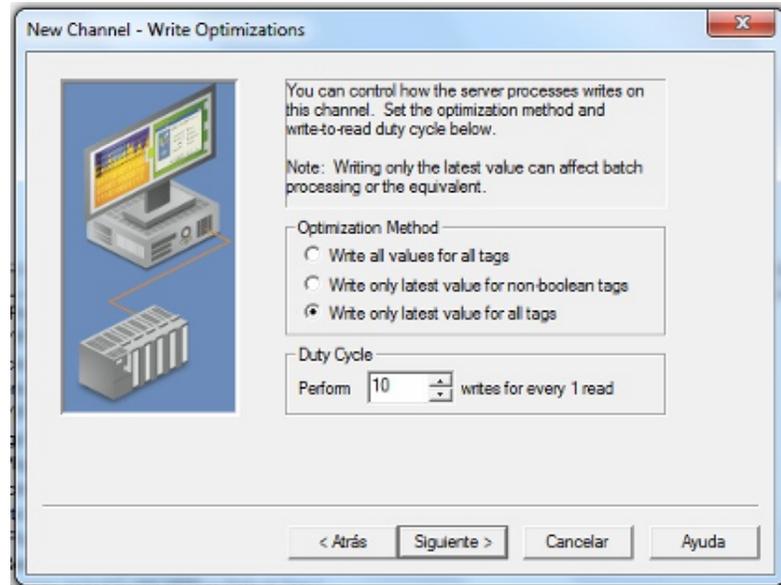


Fig. IV-26 Pantalla modo de actualización de tags.

- Ahora que ya hemos escogido el tiempo y los modos de actualización, damos clic en siguiente, para probar la conexión y finalizar con el proceso.

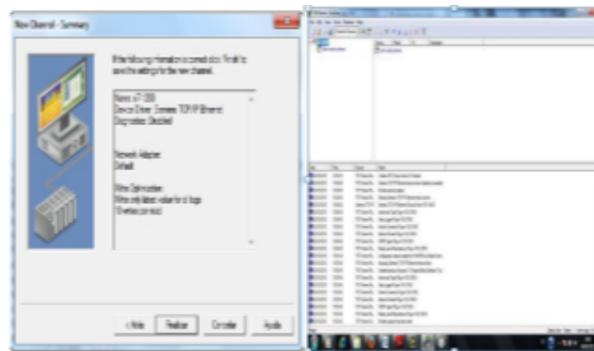


Fig. IV-27 Pantalla final de configuración de red.

- Una vez configurado el driver de comunicación es necesario configurar el dispositivo con el que nos vamos a conectar, para el proyecto la CPU S7-1200.

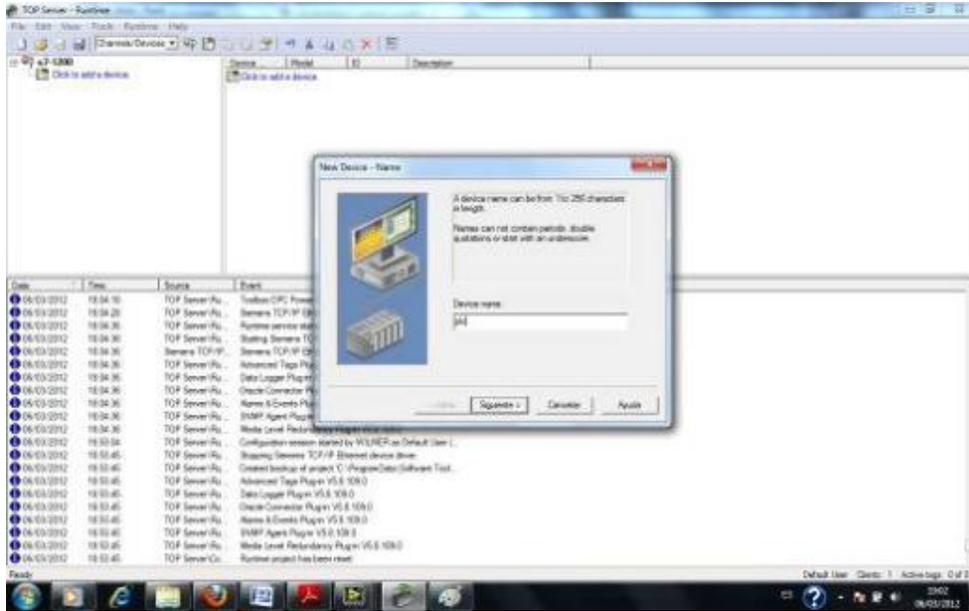


Fig. IV-28 Pantalla configuración de dispositivo.

- Como se puede ver en la figura ... Es necesario crear el dispositivo al cual vamos a conectarnos, para esto tenemos la pantalla

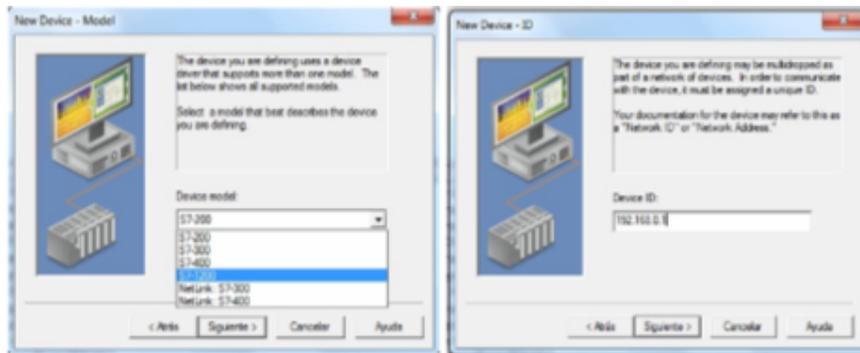


Fig. IV-29 Escogemos el modelo de automática y asignamos la dirección IP.

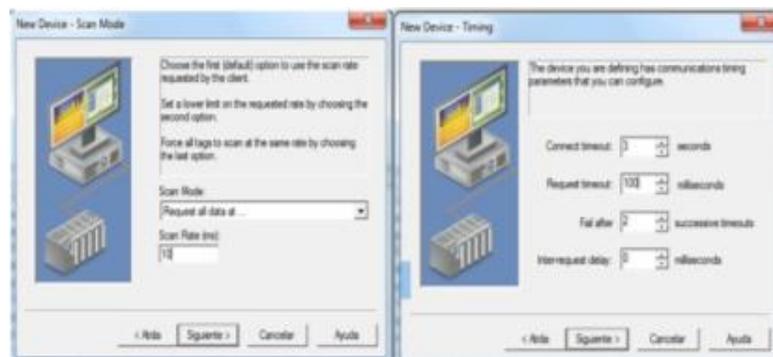


Fig. IV-30 Se establece el tiempo de scan para el proceso.



Fig. IV-31 Se establecen los parámetros de puerto de comunicación y modos de escritura.

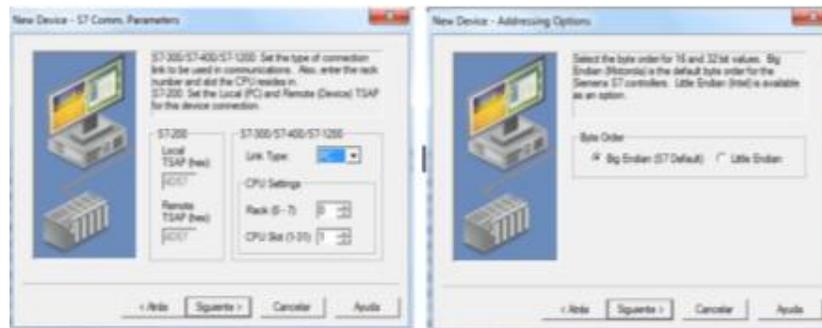


Fig. IV-32 Establece el tipo de conexión y el bit más significativo en la comunicación.

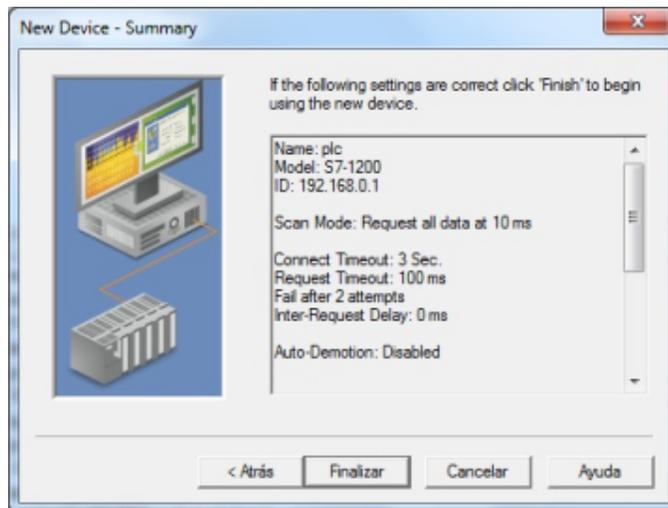


Fig. IV-33 Datos finales de configuración.

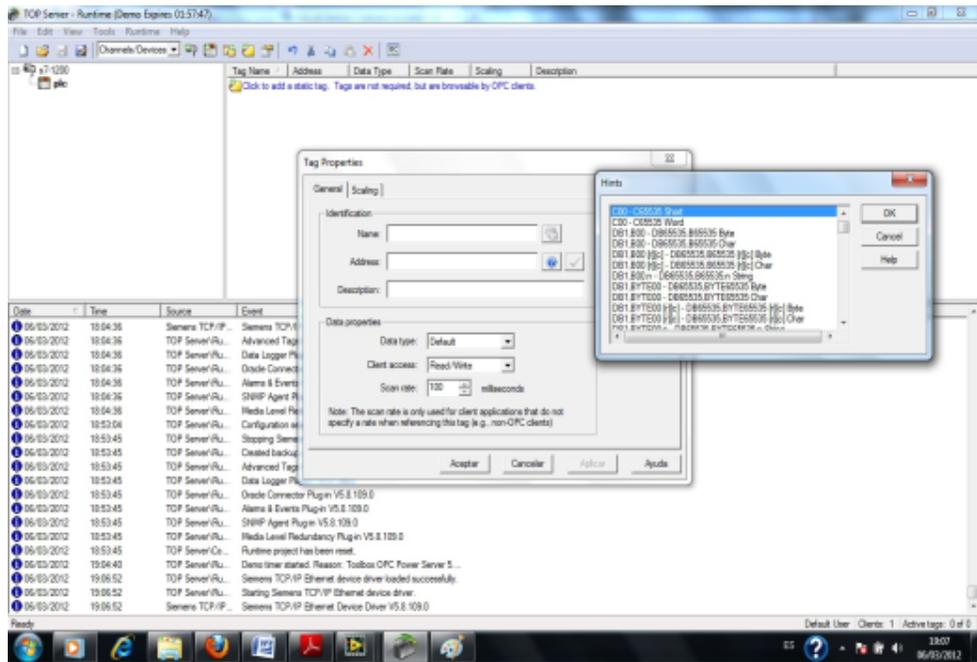


Fig. IV-34 Pantalla de creación de tags.

Ya creado el dispositivo y configurado su modo de comunicación es necesario crear los tags respectivos, como para cualquier servidor opc conocido.

Lo que nos interesa en el momento es la configuración del opclink del paquete wonderware development Studio.

4.4 Enlace OPC-INTOUCH.

Para poder desarrollar la comunicación entre la interfaz HMI y el PLC es necesaria la creación de la configuración de la pasarela de intouch el OPCLINK este nos permitirá comunicarnos entre el OPC e intouch, y nos dirá si la comunicación está activa y que variables estamos utilizando dentro de la misma.

Para esto buscamos en el directorio de wonderware la aplicación OPCLINK y la abrimos.

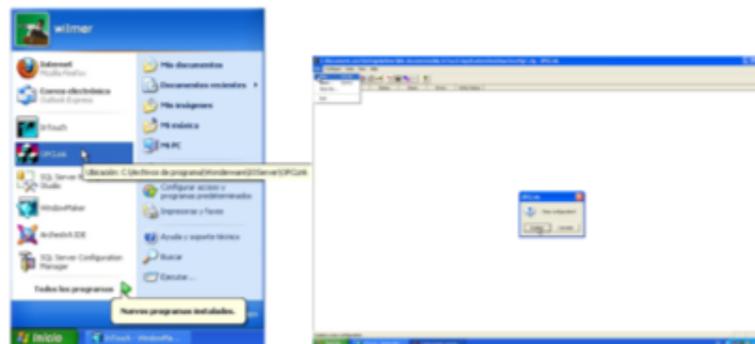


Fig. IV-35 Directorio de acceso a OPCLINK y ventana de configuración.

Una vez abierta la aplicación vamos al menú File, y escogemos la opción New, luego de esto y si existe una configuración previa creada, nos preguntará si queremos limpiar esta configuración, damos clic en aceptar para crear una nueva configuración. Ya con la nueva configuración establecida, vamos hacia el menú configure damos clic sobre la opción Topic definition, después de lo cual nos aparece una pequeña pantalla en donde se visualizarán los dispositivos que se han configurado anteriormente, pero como en nuestro caso es un nuevo dispositivo, nos debe aparecer en blanco, luego de esto damos clic sobre la pestaña new, para crear la configuración de la conexión que necesitamos.

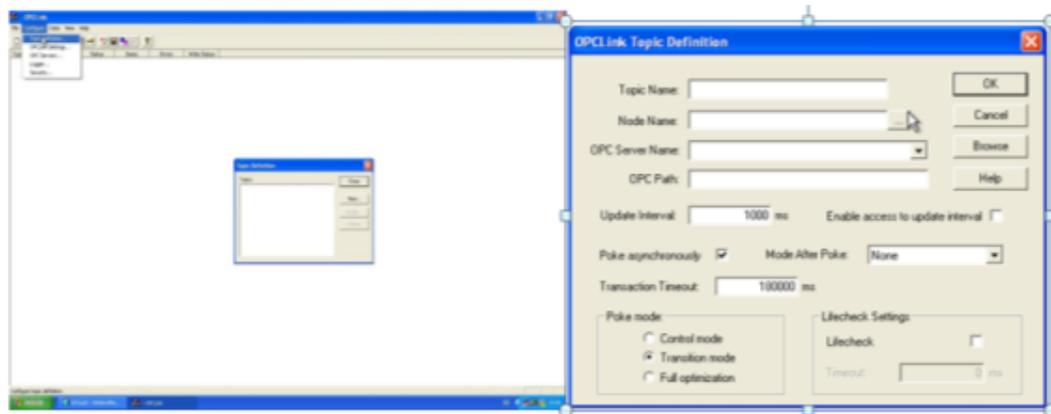


Fig. IV-36 Creación de conexión opc y pantalla de configuración.

En la pantalla de configuración, llenamos los datos requeridos respectivamente, nombre de la conexión, nodo de la conexión (dirección del dispositivo en la red), nombre del servidor OPC (top server five), Intervalo de actualización (100 ms), poke mode (full optimization).

Con estos datos llenos damos clic en aceptar para luego guardar la configuración en un archivo físico, que servirá para rescatar la configuración del opclink cuando lo la aplicación lo requiera y realizar las pruebas de conexión del dispositivo con el intouch.

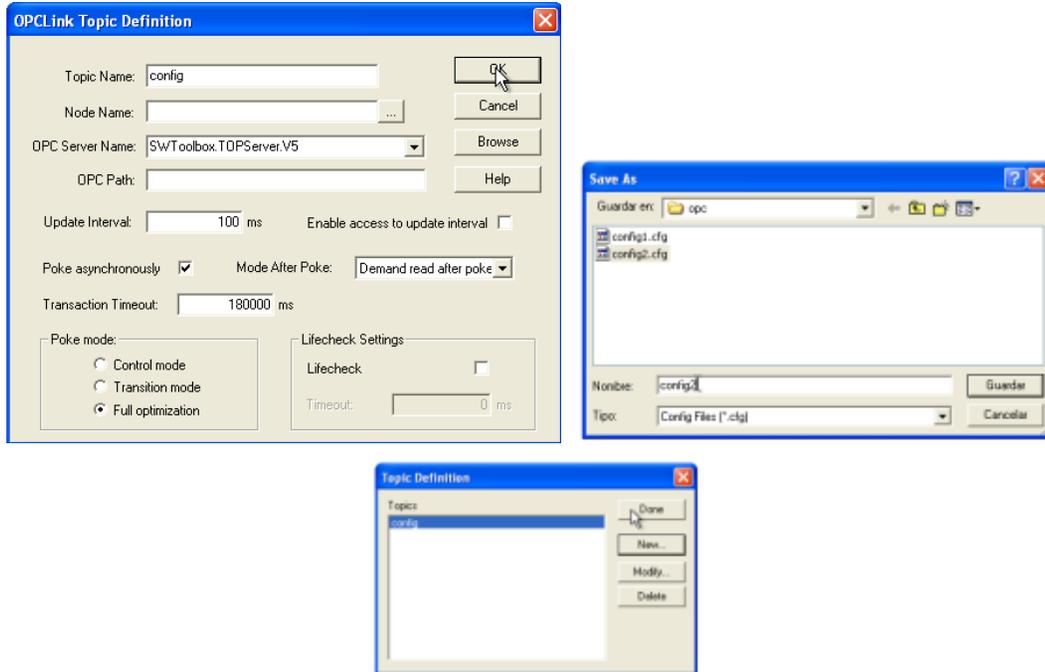


Fig. IV-37 Datos de configuración opc Link.

Cuando hayamos terminado de configurar y guardemos los cambios realizados, aparecerá la pantalla pequeña mostrada al pie de la figura anterior en la cual damos clic en el botón done y terminamos con la configuración del opclink.

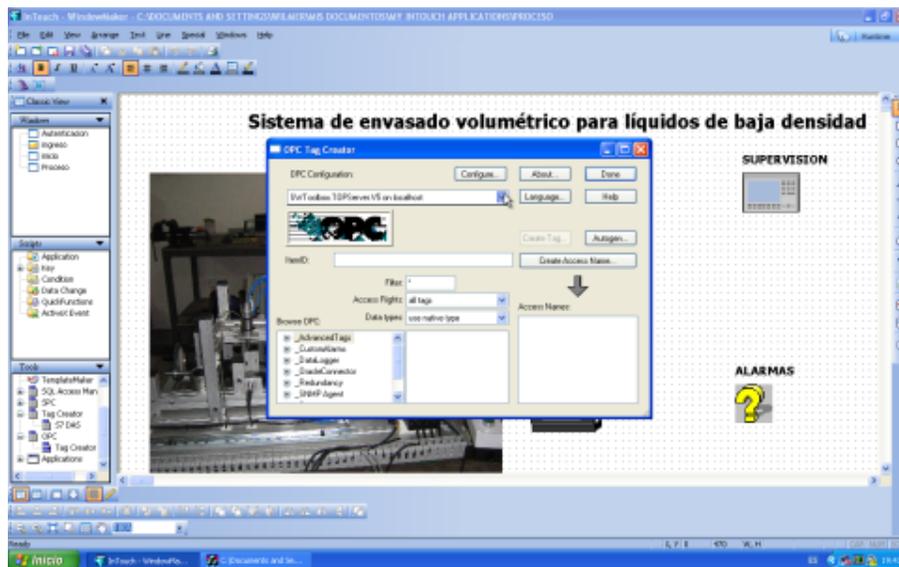


Fig. IV-38 Configuración del OPC Tag Creator.

Una vez establecida la configuración del opclink, es necesario enlazar las variables físicas del plc con las variables lógicas creadas en el opc para pasarlas al intouch, para esto es necesario configurar el OPC tag creator, en este paso buscamos el opc

que estamos utilizando, y en el cual aparecerá el dispositivo configurado, con sus respectivas variables del sistema y variables creadas por el usuario como se muestra en la siguiente fig.

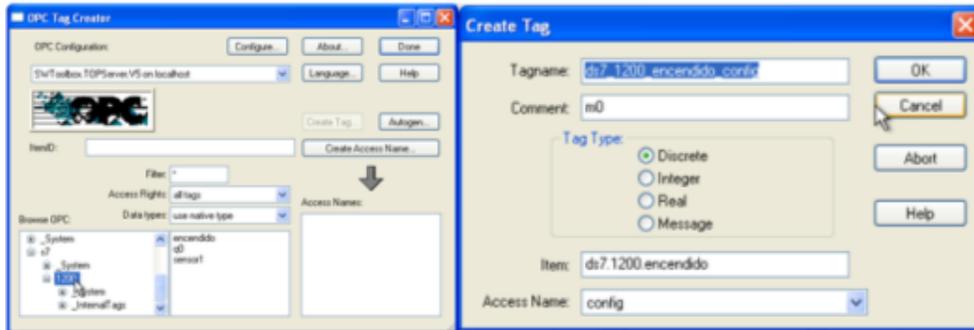


Fig. IV-39 Creación de variables en intouch.

Escogemos las variables que deseamos crear una por una, con un nombre que se puede asignar por parte del usuario, o se puede asignar el nombre que el sistema proporciona.

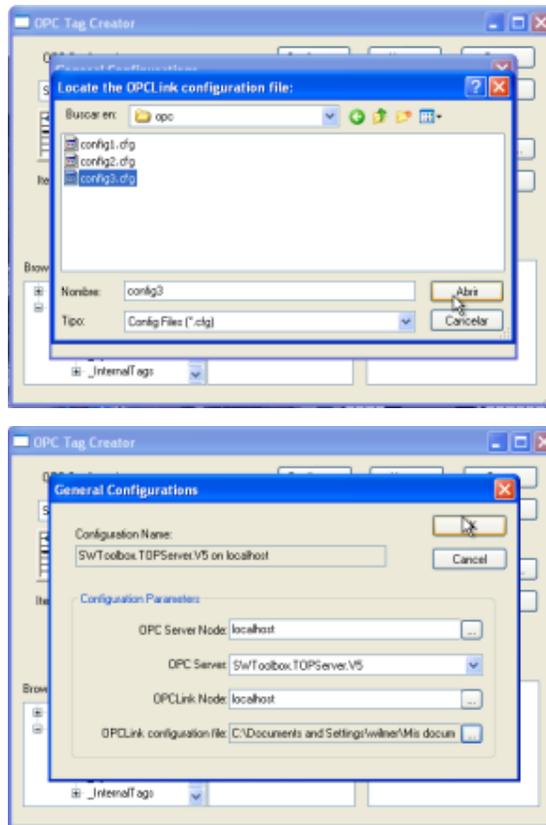


Fig. IV-40 Búsqueda del archivo de configuración del opclink.

En el caso de que no se haya cargado el archivo de configuración correcto para el funcionamiento del opclink, es necesario cargarlo en la interfaz del OPC Tag Creator, y también es necesario cerciorarnos que el archivo cargado es el correspondiente al dispositivo físico que tenemos ya que de no ser así podemos tener problemas de funcionamiento del equipo.

Una vez configurado el OPC Tag Creator y creadas las variables ya podemos utilizarlas dentro de la aplicación, después de realizada esta acción las variables traídas desde el PLC están en el tag dictionary del Intouch y se las puede utilizar como cualquier variable de usuario.

4.5 Comunicación de INTOUCH con SLQ Server 2005

Como en cualquier lenguaje de programación, el intouch requiere de un driver para realizar la conexión e interpretación de datos entre él y el gestor de base de datos SQL Server 2005, en esta sección se detallará paso a paso la conexión entre estos dos software de desarrollo.

Primero necesitamos la creación de la base de datos. Para esto es necesario realizar correctamente la instalación del motor de base de datos para nuestro caso el SQL server 2005, el mismo que después de la instalación se encuentra ubicado en INICIO/TODOS LOS PROGRAMAS/ MICROSOFT SQL SERVER2005/SQL SERVER MANAGEMENT STUDIO, como lo podemos ver en la figura.

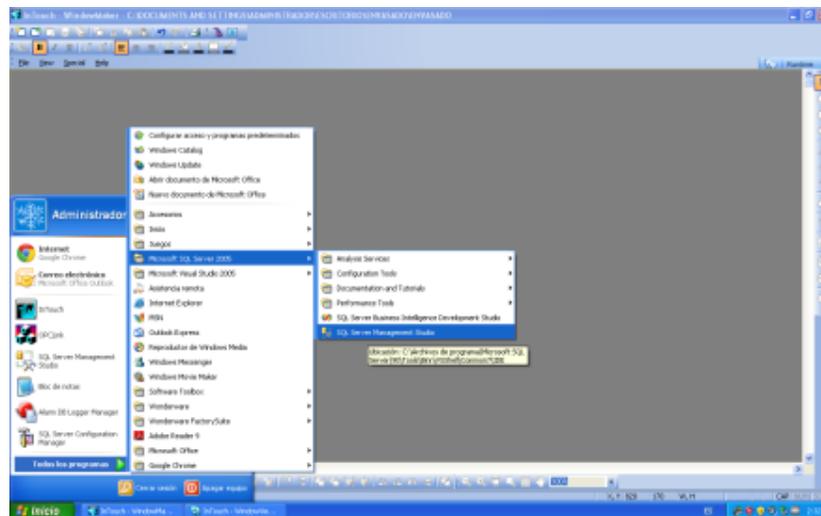


Fig. IV-41 Ubicación SQL SERVER MANAGEMENT STUDIO.

Una vez que hemos ingresado al entorno de trabajo de SQL procedemos a crear la base de datos

4.6 CREACIÓN DEL ODBC

El ODBC es el enlace que nos permite conectarnos a la base de datos. De acuerdo a nuestro requerimiento necesitamos crear un ODBC para lo cual inicialmente nos ubicamos en el menú de inicio y seleccionamos Panel de Control.



Fig. IV-42 Menú inicio (Seleccionamos Panel de Control).

Enseguida nos aparecerá una nueva pantalla con varios iconos opción de los cuales seleccionaremos Herramientas Administrativas.

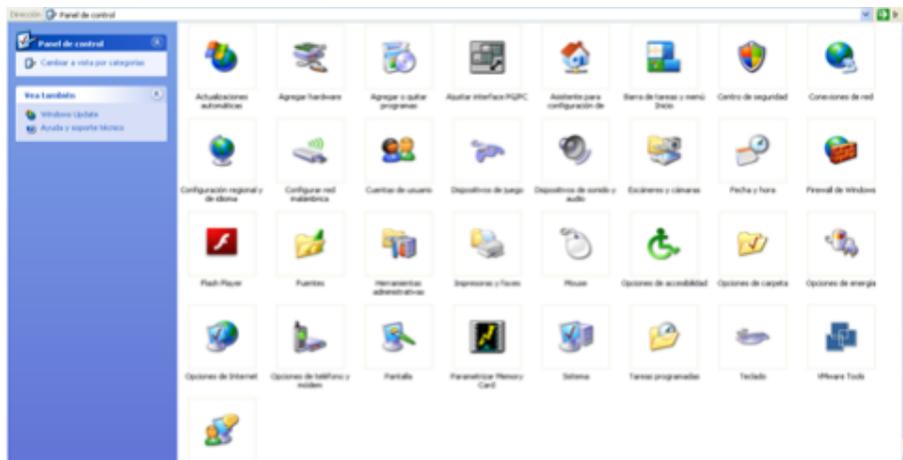


Fig. IV-43 Ventana de Panel de Control (Seleccionamos Herramientas Administrativas).



Fig. IV-44 Submenú Herramientas Administrativas (Seleccionamos Origen de Datos ODBC).

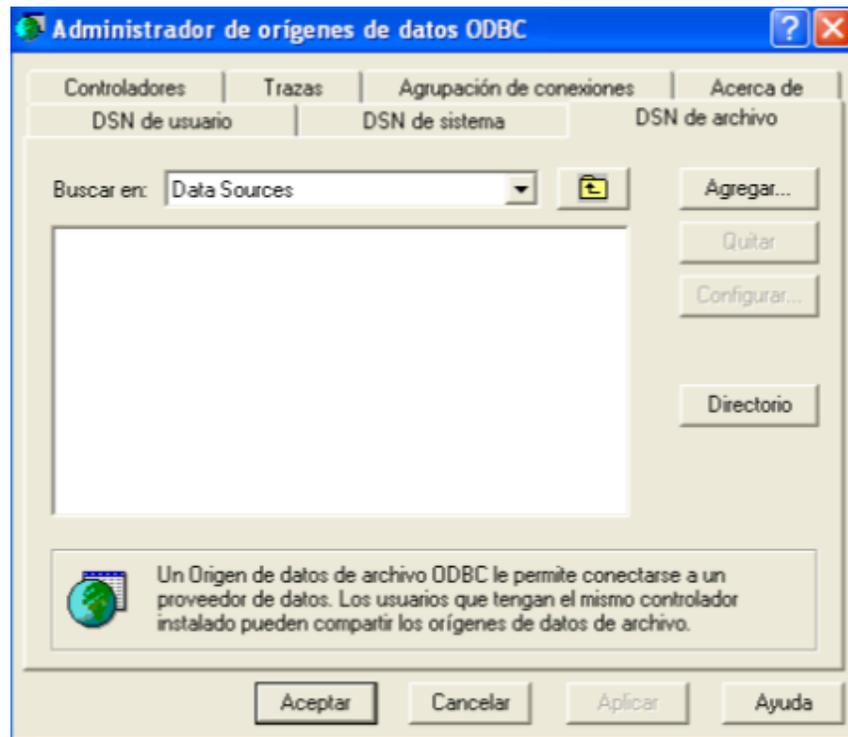


Fig. IV-45 Ventana Administrador de Orígenes de Datos (Seleccionamos DSN de archivos).

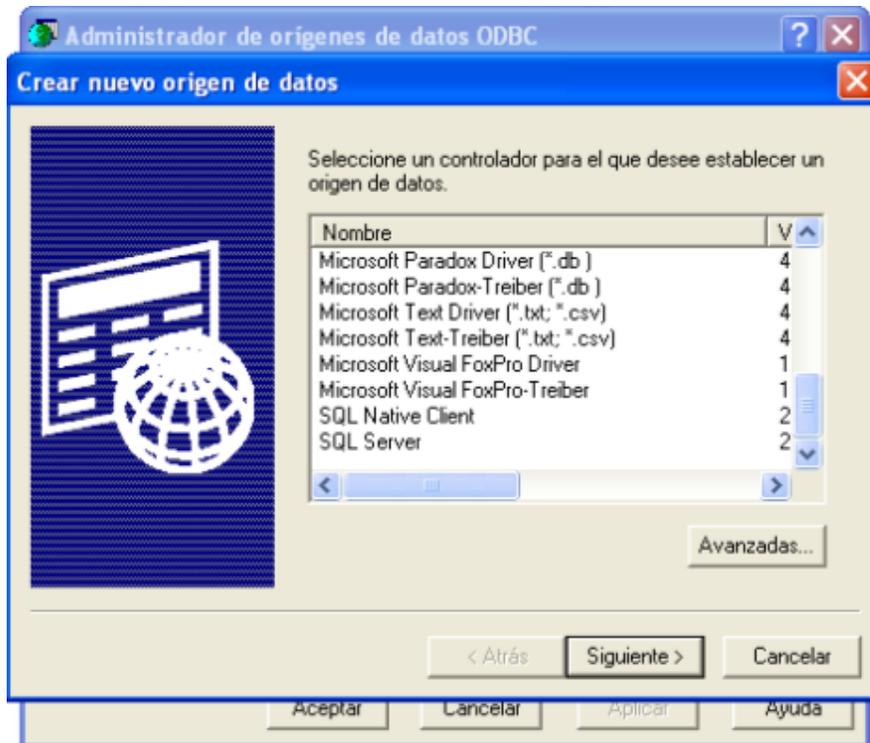


Fig. IV-46 Selección controlador para el que se desea establecer Origen de De Datos (Seleccionamos SQL Server).



Fig. IV-47 Asignar un Nombre al Origen de Datos.



Fig. IV-48 Finalizar para Crear el Origen de Datos.

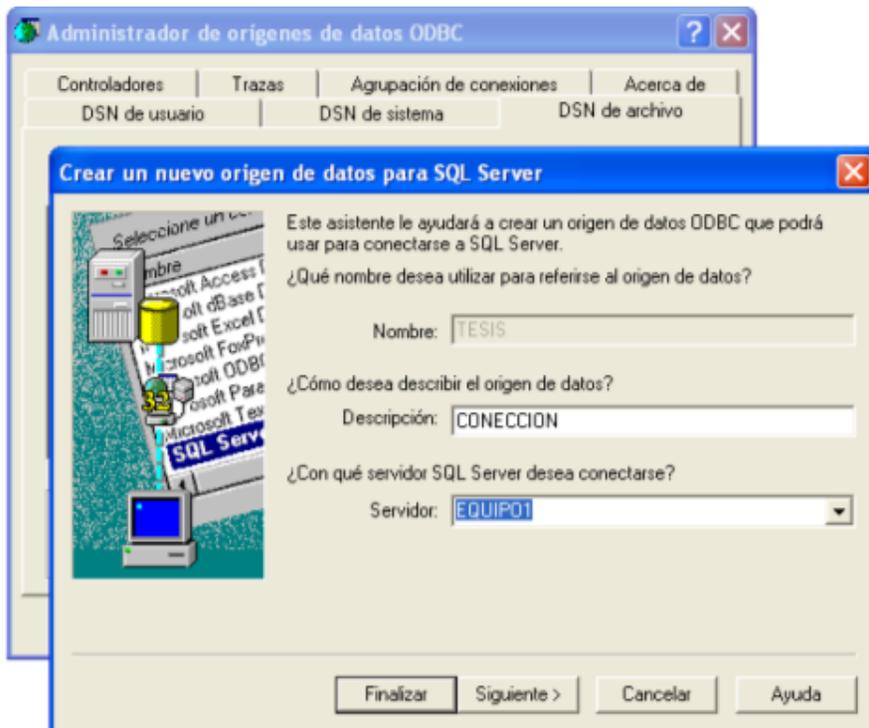


Fig. IV-49 Elección del Servidor con el que se desea Conectar el Origen de Datos.



Fig. IV-50 Elección de tipo de Autenticación del SQL Server.

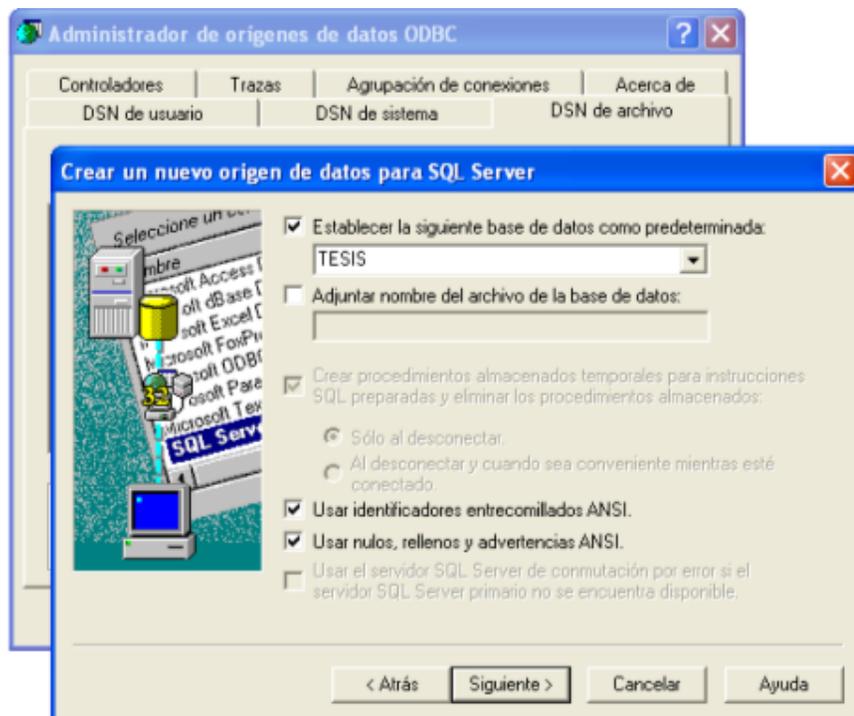


Fig. IV-51 Asignación de la Base de Datos.

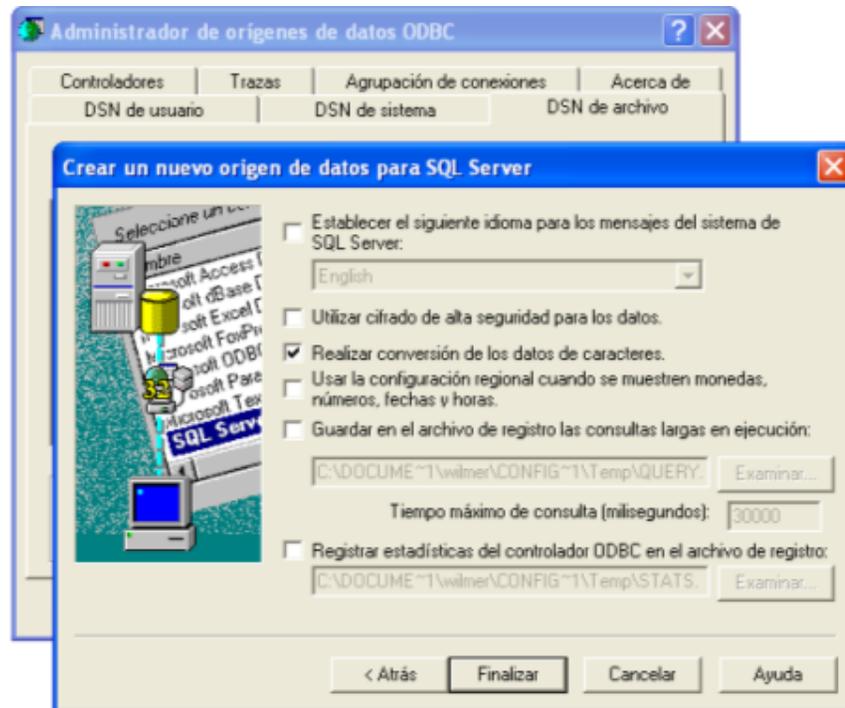


Fig. IV-52 Complementos de Información para creación del Origen de Datos

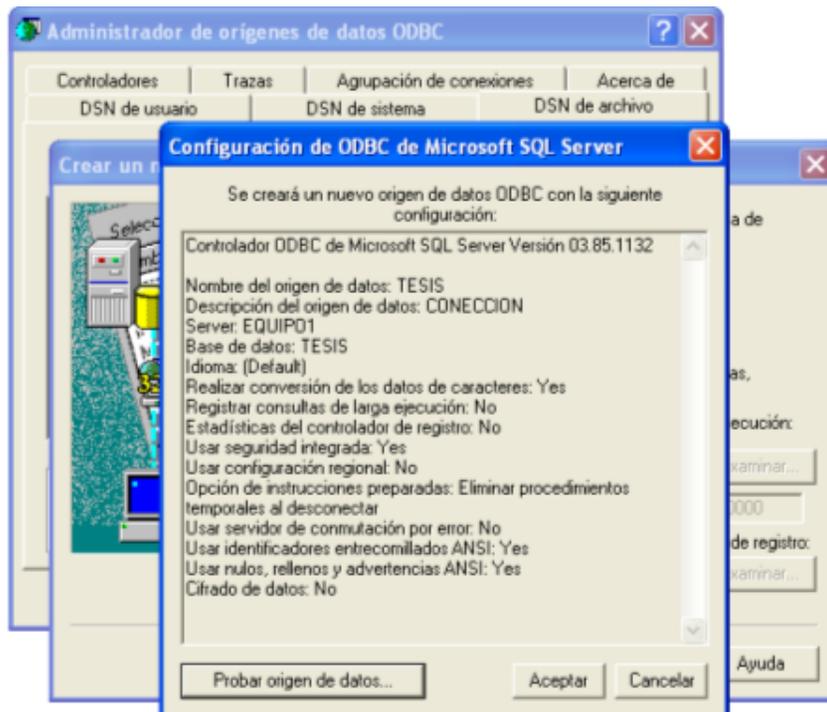


Fig. IV-53 Datos del nuevo ODBC y finalización de la creación.

4.7 PANTALLAS DE TRABAJO

4.7.1 Inicio.

Esta pantalla muestra una vista general del proceso y permite el acceso hacia las opciones de usuario. Una vez que demos clic sobre el botón proceso se habilitará la siguiente pantalla en donde podremos realizar la autenticación de usuario. Esta pantalla está disponible para todos los usuarios del sistema.



Fig. IV-54 51Pantalla de inicio del proceso.

4.7.2 Ingreso.

En esta interfaz realizamos tenemos la opción de autenticarnos como usuarios para luego ingresar a las opciones del proceso. Dando clic sobre la opción autenticación aparecerá la siguiente pantalla.



Fig. IV-55 Pantalla de ingreso.

4.7.3 Autenticación.

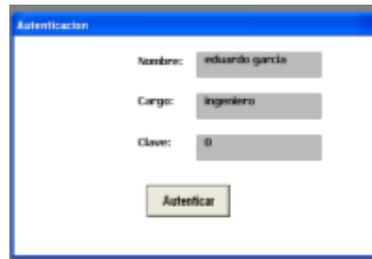


Fig. IV-56 Pantalla de autenticación.

En esta pantalla realizamos la autenticación de usuarios con sus nombres y los diferentes niveles de usuario, para esto tenemos que ingresar los datos de usuario como en este caso un nombre, el cargo o rango de operador, y una clave que se les ha sido asignada para dar los permisos respectivos. Para nuestro caso tenemos 3 niveles de usuarios en la siguiente tabla se detalla los niveles de acceso y los rangos de claves que se les han sido asignadas.

Tabla IV-I Niveles de acceso de la aplicación.

Tipo de Usuario	Acceso	rango de claves
Ingeniero	Puede ingresar a todas las opciones de control y supervisión del proceso, tiene todos los privilegios en el sistema.	1001 -1999

Operador	Puede realizar las acciones de supervisión, reconocimiento de alarmas, cambio de parámetros básicos del proceso, se han restringido ciertas opciones.	2001-2999
Invitado	Solo puede verificar parámetros básicos del proceso y visualizar el desarrollo del mismo, además del reconocimiento de alarmas.	3001-3999

4.7.4 Proceso.

Una vez autenticado el usuario se regresa hacia la pantalla de ingreso y puede ingresar al proceso con el perfil de usuario ya definido dependiendo del nivel de acceso.



Fig. IV-57 Pantalla principal del Proceso.

CAPITULO V

V. PRUEBAS Y RESULTADOS

Las pruebas de funcionamiento del equipo fueron desarrolladas con la finalidad de visualizar la respuesta de los componentes principales del proceso, para nuestro caso los cilindros dosificadores, establecer el nivel de líquido que podemos dosificar, carrera de los cilindros que sirven de empuje y sujeción de piezas, motor del transportador de gusano, y la capacidad de detección de los sensores.

5.1 Sistema de dosificación.

Para realizar las pruebas del hardware es necesario realizar la conexión del PLC con el TIA v10.5, para descargar el programa que contiene la lógica que será ejecutada como lógica del proceso.

Para esto necesitamos descargar el programa hacia el controlador, siguiendo los pasos vistos en la parte teórica del capítulo II.

Una vez descargado el programa hacia el controlador, podemos realizar la prueba de capacidad de volumen de los cilindros.

En un inicio se colocó los cilindros dosificadores en forma vertical y con la succión hacia arriba, pero por efecto de la gravedad en los cilindros no se pudo realizar una dosificación exacta del líquido, se tenía un volumen mínimo del mismo como se puede observar en la figura.



Fig. V-1 primera posición del dosificador.

Para corregir este problema se decidió invertir los cilindros para que nos ayude el efecto de gravedad a obtener un volumen de líquido con mayor exactitud, con esto conseguimos que el tarro contenedor se llene completamente. Como podemos observar en la figura.

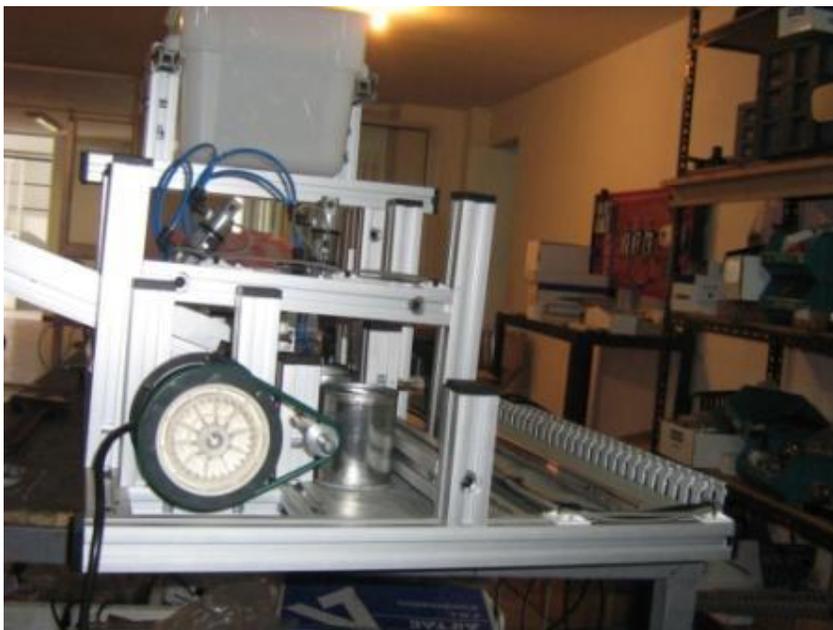


Fig. V-2 Corrección del sistema de cilindros dosificador.

En la tabla .. se muestra los resultados del volumen dosificado en las diferentes posiciones de los cilindros y por tiempo de dosificación especificado, para ser ubicado el líquido en los tarros contenedores.

Tabla V-I Volúmenes obtenidos por tiempos de dosificación

Tiempo	Numero de accionamientos	Volumen	Posicion del cilindro
4seg	1	10ml	Vertical Arriba
8 seg	2	30ml.	Vertical Arriba
4seg	1	80ml	Vertical Abajo
8seg	2	160ml	Vertical Abajo
4seg	1	62ml	Horizontal
8seg	2	124ml	Horizontal.

Cabe recalcar lo que se busca es realizar la dosificación exacta en los tarros de 1/8 de litro, para esto se han determinado el tiempo de 8seg y la posición horizontal de los cilindros dosificadores.

Para el mejor desempeño del proceso hemos determinado los tiempos que son necesarios para ejecutar las etapas de llenado del empuje del tarro hacia el proceso, llenado del líquido, colocación y sellado de tapa y salida del recipiente dosificado, los cuales podemos verificar en la tabla V-II

Tabla V-II Tiempos de proceso.

Tiempo	Transicion
1seg	Empuje - Transportador
3seg	Inicio Trans - dosficador
10seg	Dosificación
8seg	Dosificacion – Sellado
12seg	Tapa – Sellado
5.2seg	Sellado – Salida prod.

Total 37.2seg de dosificación del sistema, los tiempos han sido probados antes de estandarizar el proceso.

5.2 Pruebas de software.

Para realizar estas pruebas se hizo necesario conectar el proceso hacia la hmi desarrollada en intouch, con la finalidad de comprobar que el sistema de supervisión sea lo más parecido al sistema real, y muestre datos reales del producto del proceso.

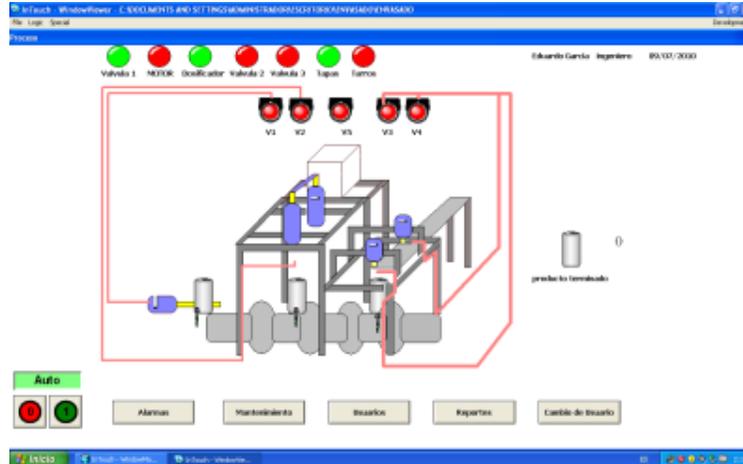


Fig. V-3 Pantalla del proceso con señales del sistema físico.

Como se muestra en la figura las variables del proceso se muestran de forma clara y en tiempo real en el sistema de supervisión desarrollado.

Ahora es necesario probar la generación de reportes de producción y alarmas generadas en el sistema.

La generación de reportes es una herramienta importante y vital dentro de un sistema scada, ya que nos ayuda a verificar el funcionamiento del proceso. La generación de reportes se ha diseñado en una herramienta como es Microsoft Excel, para esto los datos guardados en la base de datos son transferidos en forma secuencial y ordenada hacia una hoja de Excel que puede ser visualizada de forma fácil, para generar esto es necesario ingresar a la pantalla de reportes y dar clic sobre el botón generar reportes.

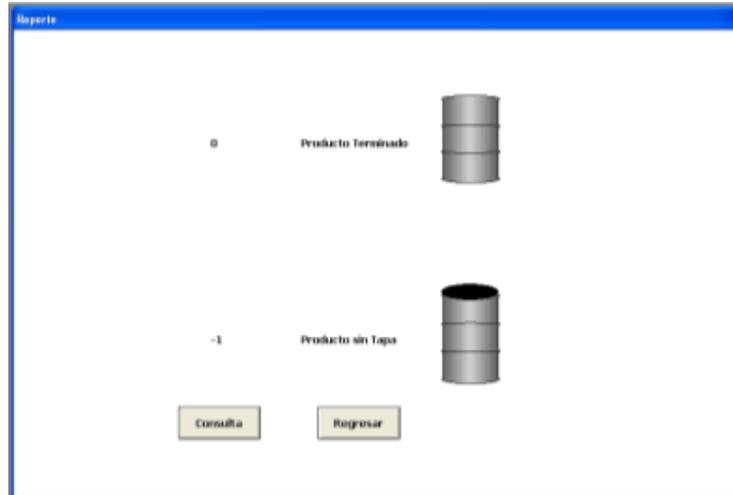


Fig. V-4 Generación de reportes.

Una vez que damos clic sobre este botón se llama al activeX que genera la salida de datos hacia Excel.

El reporte consta de los valores de los recipientes que se han producido sin errores durante el proceso y también de los productos que no han completado el proceso de dosificación, como

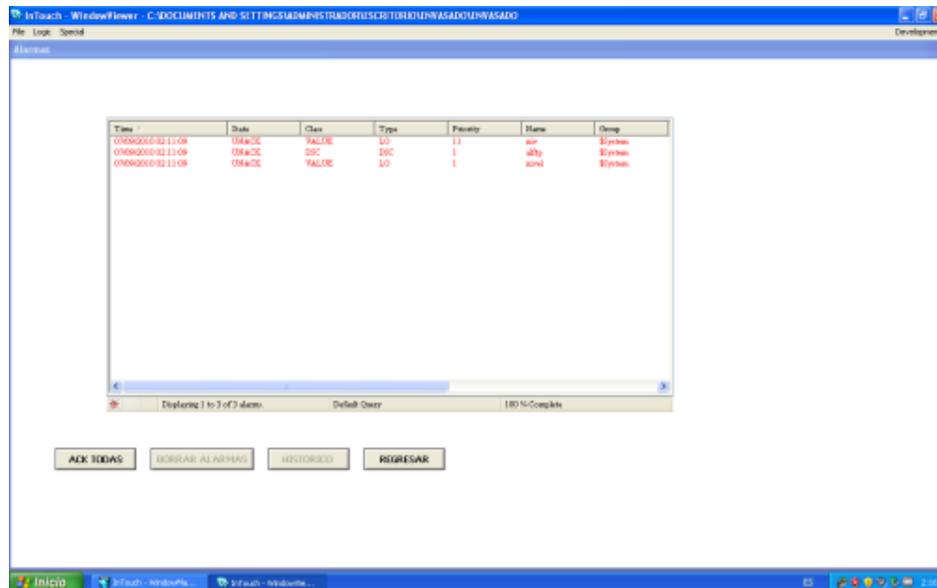
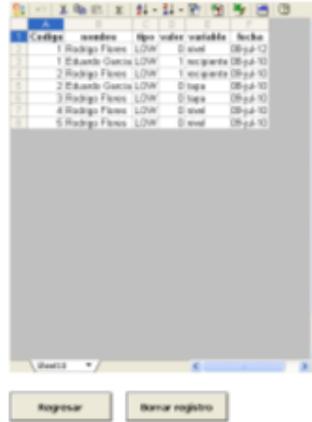


Fig. V-5 Reportes diarios

El reporte de alarmas se genera con la finalidad de detectar los problemas frecuentes que se vienen dando diariamente en el proceso y nos ayuda a tomar decisiones sobre mejoras del proceso, o mejoras hacia el personal operativo en este caso estudiantes que realizaran pruebas en el sistema.



Código	nombre	tipo	valor	variable	fecha
1	Redigo Fases	LOW	0	swt	09-pa-10
1	Estado Genera	LOW	1	no genero	09-pa-10
2	Redigo Fases	LOW	1	no genero	09-pa-10
2	Estado Genera	LOW	0	sepa	09-pa-10
3	Redigo Fases	LOW	0	swt	09-pa-10
4	Redigo Fases	LOW	0	swt	09-pa-10
6	Redigo Fases	LOW	0	swt	09-pa-10

Registrar Borrar registro

Fig. V-6 Reporte de alarmas.

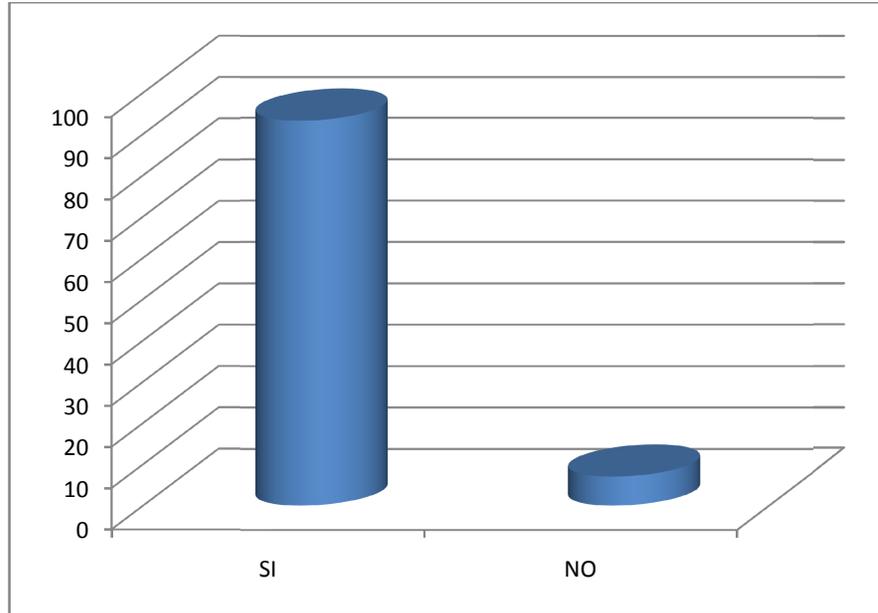
5.3 Tabulación de Datos de cada Pregunta realizada la encuesta.

Haciendo uso de la encuesta como herramienta para la adquisición de datos estadísticos planteamos lo siguiente:

Como estudiante de la Escuela de Ingeniería Electrónica en Control y Redes Industriales ¿Piensa que se cuenta con los suficientes y equipados laboratorios para la realización de prácticas que permitan fortalecer la teoría adquirida en las aulas?

SI 5%

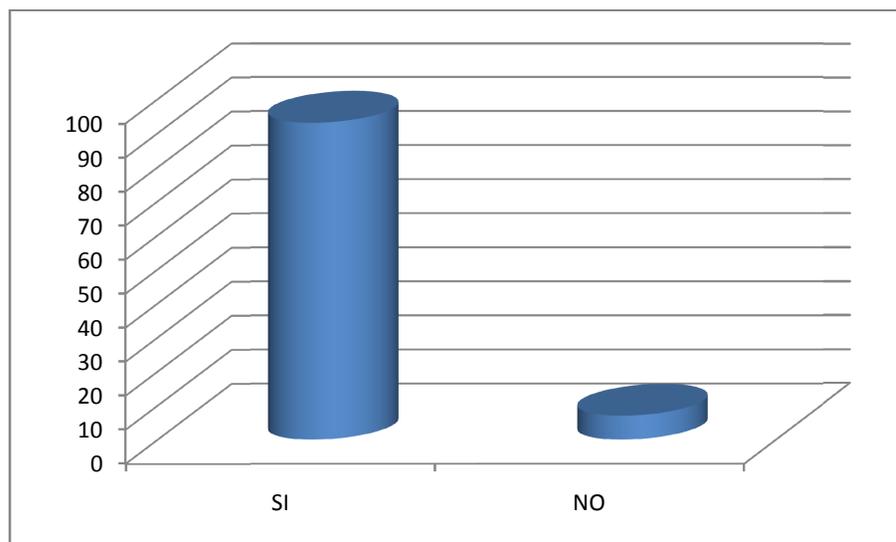
NO 95%



¿Cree Ud. Que la implementación de Procesos Industriales como Plataformas de Aprendizaje desarrollados como temas de Tesis, ayudan a la formación profesional de los alumnos de la Escuela de Ingeniería Electrónica en Control y Redes Industriales?

SI 98%

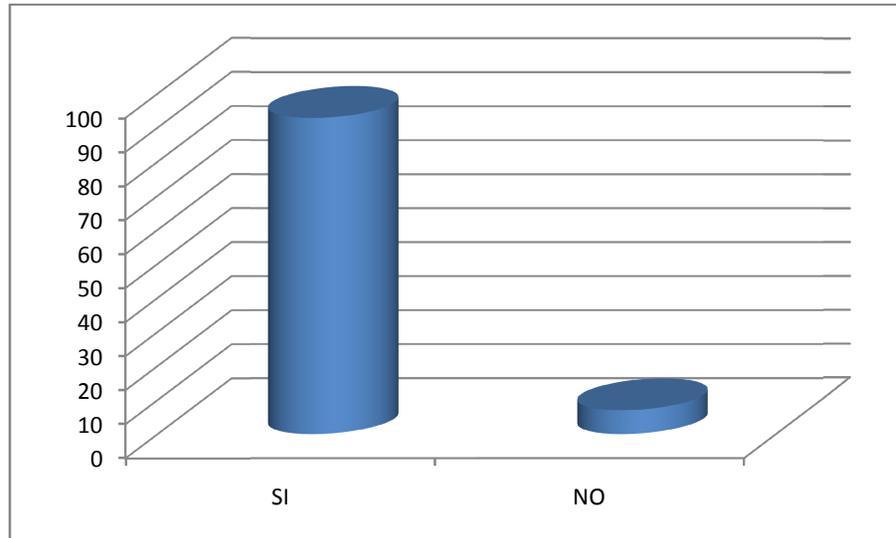
NO 2%



Al tener la facilidad de experimentar el campo real con un proceso de envasado y transporte de líquidos de baja densidad en un laboratorio ¿Piensa que facilitará el desenvolvimiento de los nuevos profesionales que ofrezca la Escuela de Ingeniería Electrónica en Control y Redes Industriales en el campo laboral?

SI 92%

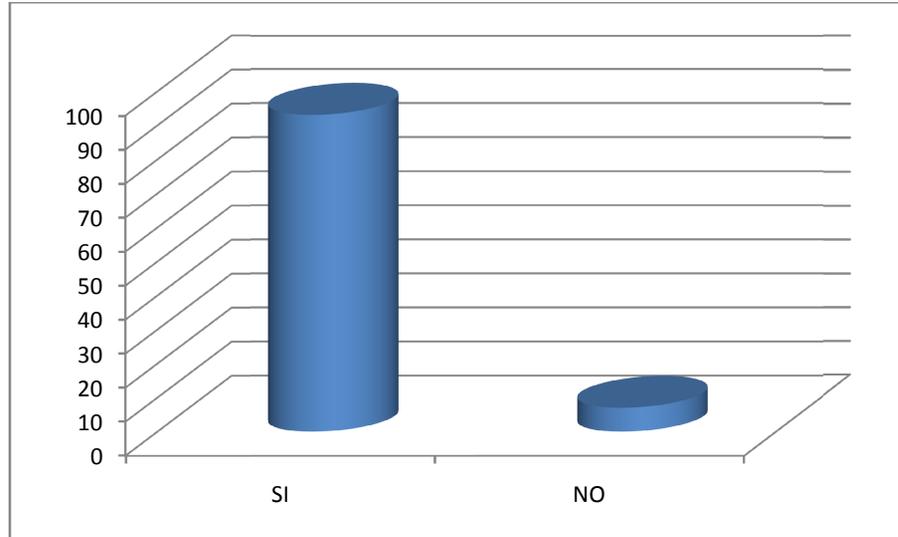
NO 8%



Los procesos ya implementados ¿Han sido de ayuda para su preparación profesional?

SI 93%

NO 7%



El sistema de dosificación tubo una aceptación del 92% dentro de los estudiantes de los últimos niveles de la carrera de Ingeniería Electrónica Control y Redes Industriales.

CONCLUSIONES

1. Existen numerosos sistemas de dosificación para líquidos entre los cuales se presentan los sistemas que toman como referencia variables como el peso o densidad del líquido que pueden ser un tanto imprecisos al momento de la medición, por esto la dosificación volumétrica con recipiente constante que contenga el líquido se torna en una forma muy segura de obtener un volumen constante de dosificación ya que siempre tendremos la misma succión en expulsión del líquido presentando mínimas variaciones en las medidas del volumen del líquido.
2. Así como los sistemas de dosificación, en los sistemas de transporte no estamos ceñidos a un solo tipo de sistemas como es el caso de las bandas transportadoras en el proyecto se consideró el desarrollo de un sistema alternativo a este y es un gusano transportador que cuenta con un tornillo sin fin que realiza una transmisión de movimiento lineal para desplazar a los recipientes de una estación a otra, este se vuelve un sistema bastante preciso a la hora de posicionar al recipiente en un lugar específico de la máquina mediante la rotación del eje que permite hacer una traslación lineal del recipiente como un sistema de engrane.
3. El shunt construido para la dosificación de tapas, nos muestra un sistema sencillo de colocación de las tapas para realizar el sellado del recipiente dosificado, ya que es sostenido por un cilindro que evita la caída de la tapa hasta cuando sea necesaria la ubicación de la tapa sobre el recipiente, lo libera y la tapa cae por gravedad, aprovechando el peso de la tapa y la diferencia de altura de la superficie del dispensador de tapas con el recipiente para realizar la ubicación correcta de la misma.
4. La plataforma de programación TIA V10.5 de Siemens es una poderosa herramienta para el desarrollo de aplicaciones que involucren autómatas programables de la familia S7-1200. Ya que además de la lógica escalera acostumbrada para la programación de PLC, cuenta con un sistema de programación modular que involucra los Bloques de Funciones (FB), y los bloques de datos (DB), que hacen que la programación del proceso se vuelva descomplicada, sencilla y ahorra espacio en la memoria del programa optimizando de esta forma los recursos de la CPU seleccionada.

5. Para realizar el diseño de la HMI es necesario contar con el proceso constituido en su mayor parte mecánica para tener una idea clara de la máquina y realizar una representación bastante real de la misma, para esto INTOUCH se ha convertido en una herramienta bastante útil ya que es un sistema de supervisión que cuenta con un entorno amigable para el usuario, un nivel de seguridad bastante confiable para el proceso y los datos del mismo, además de la generación de reportes y alarmas que ayudan al operador a realizar una detección de fallos inmediata ya que trabaja en tiempo real.
6. La integración del proceso con la HMI, máquina real con el proceso virtual, es necesario contar con un traductor entre el PLC y la computadora o servidor en el cual está instalada la aplicación, para obtener un proceso transparente para el usuario, en este caso los servidores OPC Toolbox OPC server 5 es la opción más viable ya que cuenta con el protocolo de comunicaciones y reconocimiento para las instrucciones del S7-1200, herramienta con la que otros servidores OPC no cuenta. Además es necesario realizar la configuración exacta del OPCLINK (Wonderware), ya que este en cambio es el servidor DDE que transmite los datos entre aplicaciones si uno de los dos servidores no se encuentran bien configurados el proceso presenta conflictos.
7. Como para todas las aplicaciones o maquinarias construidas esta se debe dotar de un manual de usuario para las personas que vayan a hacer uso del mismo ya que por mala manipulación o desconocimiento de las partes constructivas que forman parte del proceso puede ocasionarse daños a la máquina misma o a la persona que se encuentra haciendo uso de ella. Por esto el manual de usuario debe contar con la información más detallada y con ilustraciones de cómo utilizar el proceso.
8. La implementación de maquinaria didáctica ayuda a la formación y refuerzo de los conocimientos adquiridos dentro de nuestra formación como profesionales técnicos que estaremos inmersos en el campo industrial.
9. El PLC Siemens S7-1200 es una potente plataforma para el desarrollo de aplicaciones industriales ya que cuenta con un sistema operativo potente así como también con un sistema robusto, y un lenguaje de programación amigable para el usuario.

10. Es necesario realizar pruebas del equipo antes ponerlo en marcha sobre todo en la parte mecánica y la parte lógica del cableado y programa para no cometer errores que pueden costar daños en el equipo o la persona que va a hacer uso del mismo.
11. Antes de implementar una solución directamente en el PLC es necesario crear una solución mediante las herramientas conocidas de lógica de programación, para el caso el método GRAFSET, que nos permita verificar las acciones de la lógica que se va a programar para el control del proceso.
12. Los sistemas SCADA nos ayudan a tener una visualización y control sobre el proceso tratado, mediante interfaces amigables para el usuario que pueden convertirse en un centro o estación de trabajo fácil de utilizar.
13. Es necesario conocer todas las variables involucradas en el proceso para desarrollar la HMI (Interfaz Hombre-Máquina, por sus siglas en inglés), ya que de solo de esta forma podremos involucrar todos los pasos del proceso y se desarrollará una solución completa ajustadas a las necesidades del usuario.
14. El sistema SCADA debe estar compuesto por una estación de mando local, para el sistema de envasado volumétrico, el tablero de control, y una estación de mando remota, Computador donde se encuentra desarrollada el HMI, que puede conectarse hacia una red e involucrar la parte de proceso con la parte de control y la parte administrativa de la maquinaria.
15. La arquitectura del sistema SCADA es bastante flexible, es decir, se puede desarrollar un sistema SCADA tan sencillo como un enlace entre un PC y un PLC, o tan complejo que integre muchos sub-procesos como los presentes en un proceso industrial grande e incluso geográficamente separado.

RECOMENDACIONES

1. Es necesario crear una lista de materiales y conocer el funcionamiento y conexión de los equipos a utilizarse cuando vamos a realizar un sistema de control.
2. Cuando se desarrolla el sistema SCADA, es necesario buscar la alternativa mas conveniente para el problema a solucionarse, que contenga las facilidades en programación y conectividad para tener un sistema completo.
3. Es mejor realizar la programación del PLC S7-1200 cpu 1212C utilizando las funciones incorporadas como es la programación estructurada, para reducir los tiempos de SCAN durante el proceso, esto minimiza el tiempo del proceso y agiliza la carga de transacciones dentro de la CPU del controlador.
4. Intouch representa una de las soluciones mas completas a la hora de desarrollar sistemas SCADA ya que contiene la mayoría de servidores OPC y de datos para dinamizar el uso de este con las distintas marcas de controladores existentes en el mercado.
5. Se recomienda cambiar la IP de controlador dirección de red que tenga un direccionamiento optimizado y que no desperdicie direcciones que pueden ser utilizadas con otros controladores.
6. Es necesario realizar un mantenimiento preventivo y predictivo dentro de lo que es la estructura mecánica del módulo de envasado volumétrico con la finalidad de prolongar la vida util del equipo.
7. Ceñirse al manual de usuario para evitar que se produzcan daños o deterioro en las partes constitutivas de la máquina.

BIBLIOGRAFÍA

1. **BLANCHARD M.**, EL GRAFCET Principios y Conceptos., 4ª.ed., Roma – Italia., ADEPA., 1999., pp. 14-32.
2. **CREUSS A.**, Instrumentación Industrial., 5ª.ed., Antioquia – Colombia., Colecciones gráficas., 1992., pp. 35-58.
3. **CUZCO S.**, Control y Medición de Nivel de Líquidos mediante Intouch., Guayaquil – Ecuador., Corrales., 2001, pp285.
4. **GONZÁLEZ I.**, Los PC's en la Industria., 3ª.ed., Barcelona – España., Marcombo., 1995., pp. 76-78.
5. **KATSUHIKO O.**, Ingeniería de Control Moderna., 2ª.ed., México DF – México., Calypso S.A., 1985., pp. 2-16.
6. **MAYOL A.**, Autómatas Programables., 3ª.ed., México DF – México., Productiva., 1987., pp. 5-31.
7. **SORIANO A.**, Autómatas Programables frente a Computadores Personales en Sistemas de Control., 2ª.ed., Barcelona –España., Marcombo., 1989., Revista nº196 pp. 811.
8. **SCHARF S.**, Electroneumática nivel básico., 5ª.ed., México DF–México., Festodidactic., 2005., pp. 16 – 18.

BIBLIOGRAFÍA INTERNET

9. SISTEMAS SCADA

www.personal.redestb.es/efigueras/memoria.htm.

www.colsein.com.co

2010-09-01

10. ACTUADORES NEUMATICOS

<http://es.scribd.com/doc/2684435/ACTUADORES-NEUMATICOS>

2011-03-15

<http://www.festo.com/argentina/104.htm>

2011-04-20

<http://www.sapiens.itgo.com/neumatica/neumatica19.htm>

2011-05-29

11. SENSORES

<http://www.gmelectronica.com.ar/catalogo/pag48.html>

2011-04-10

<http://profesormolina.com.ar/tecnologia.html>

2011-05-10

RESUMEN

Sistema de supervisión Control y Adquisición de Datos (SCADA por sus siglas en inglés), para el envasado y transporte de líquidos de baja densidad usando un PLC siemens s7-1200 e INTOUCH, en la escuela de ingeniería Electrónica en Control y Redes Industriales de la ESPOCH.

Para obtener los estándares del proceso se utilizó el método experimental obteniendo a prueba y error, los tiempos de carga, descarga del líquido para la dosificación de 125 mililitros, así como también la posición ideal de los cilindros dosificadores para que hagan uso de la gravedad como herramienta en su funcionamiento; además de conseguir establecer los tiempos de interacción entre cada estación, obteniendo un tiempo total para la dosificación de 37.2 segundos.

La bancada del proceso por tratarse de un sistema didáctico fue construida en aluminio y consta del sistema de control centralizado con el PLC s7-1200, el software INTOUCH, TOP SERVER 5 OPC Server, y SQL Server 2005 como motor de base de datos.

El sistema de dosificación para la realización de las prácticas de laboratorio, tuvo una aceptación del 95% dentro de los estudiantes de los últimos niveles de la carrera de Ingeniería Electrónica en Control y Redes Industriales.

En conclusión se logró obtener un sistema didáctico que permite a los usuarios en este caso los estudiantes, realizar prácticas de laboratorio que servirán como modelos de procesos industriales reales que ayuden a que su formación no sea estrictamente teórica si no que se complemente con la parte práctica.

Se recomienda antes de hacer uso del proceso revisar el manual de usuario del sistema de dosificación de líquidos de baja densidad y seguir las instrucciones de las prácticas para la conservación del mismo.

SUMMARY

System of Supervision, Control and Data Acquisition (SCADA by its abbreviation in English) in order to bottle and transport low density liquids by using a PLC Siemens 57 – 1200 and INTOUCH at the Electronic Engineering and Control and Industrial Networking at the ESPOCH.

To obtain standards of the process, the experimental method was used, getting a test and mistake of load times, unloading of liquid to dose 125 milliliters as well as the ideal position of the dosed cylinders, to make use of gravity as a tool in its operating, furthermore, to establish interaction times between each station, getting a total time of 37.2 seconds per doses.

The process was built in aluminum and it consists of a centralized control with the PLC 57- 1200, software INTOUCH TOP SERVER 5 OPC SERVER and SQL SERVER 2005 like an engine of data bases.

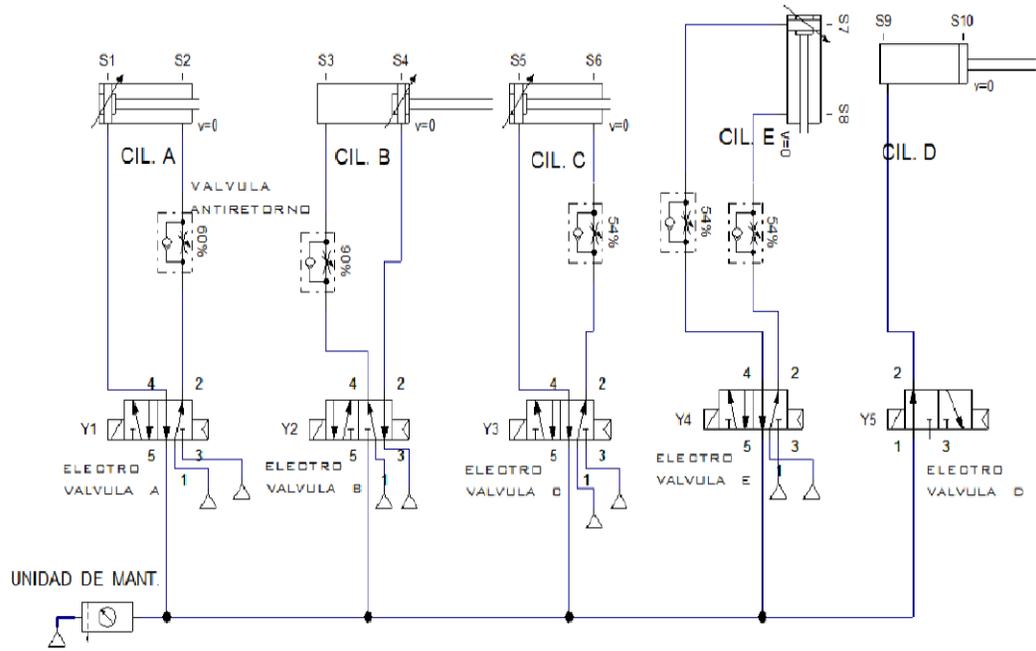
The dose system to make the laboratory practices had an acceptance of 95% in the students of last levels at the Control and Industry Network Electronic Engineering carrier.

As a conclusion a didactic system has been achieved to permit the users, in the case of students to make practices at the laboratory that will be useful as true patterns of industrial process which help students in their education, not just in theory, but in a practical education.

It is recommended to make use of the process before the system dosed of low density checking the user manual and follow the instructions of practices for its conservation.

ANEXOS

ANEXO 1



Circuito neumático del proceso de envasado volumétrico.

ANEXO 2

TAGS UTILIZADOS EN EL PROGRAMA DEL PLC.

Nombre	Direccion	Tipo de dato	Direccion Logica
Mmotor	Tabla de variables estándar	Bool	%M0.0
Motor	Tabla de variables estándar	Bool	%Q0.0
mv1	Tabla de variables estándar	Bool	%M0.1
v1	Tabla de variables estándar	Bool	%Q0.1
mv2	Tabla de variables estándar	Bool	%M0.2
V2	Tabla de variables estándar	Bool	%Q0.2
mv3	Tabla de variables estándar	Bool	%M0.3
V3	Tabla de variables estándar	Bool	%Q0.3
mv4	Tabla de variables estándar	Bool	%M0.4
V4	Tabla de variables estándar	Bool	%Q0.4
mluz verde	Tabla de variables estándar	Bool	%M0.5
luz verde	Tabla de variables estándar	Bool	%Q0.5
mluz roja	Tabla de variables estándar	Bool	%M0.6
luz roja	Tabla de variables estándar	Bool	%Q12.1
mluz amarilla	Tabla de variables estándar	Bool	%M0.7
luz amarilla	Tabla de variables estándar	Bool	%Q12.2
mboton verde	Tabla de variables estándar	Bool	%M1.0
boton verde	Tabla de variables estándar	Bool	%I0.6
boton rojo	Tabla de variables estándar	Bool	%I0.7
mboton rojo	Tabla de variables estándar	Bool	%M1.1
Mselector	Tabla de variables estándar	Bool	%M1.2
Selector	Tabla de variables estándar	Bool	%I8.0
d+	Tabla de variables estándar	Bool	%I8.1
Sensor tarro	Tabla de variables estándar	Bool	%I0.0
msensor tarro	Tabla de variables estándar	Bool	%M1.7
Sensor envasado	Tabla de variables estándar	Bool	%I0.1
msenvasado	Tabla de variables estándar	Bool	%M2.0
sensor tapas	Tabla de variables estándar	Bool	%I8.3
metapa sellado	Tabla de variables estándar	Bool	%M2.1
a+	Tabla de variables estándar	Bool	%I0.3
ma+	Tabla de variables estándar	Bool	%M1.3
b+	Tabla de variables estándar	Bool	%I0.4
mb+	Tabla de variables estándar	Bool	%M1.4
c+	Tabla de variables estándar	Bool	%I0.5
mc+	Tabla de variables estándar	Bool	%M1.5
md+	Tabla de variables estándar	Bool	%M1.6
all	Tabla de variables estándar	Bool	%M2.3

Sellado	Tabla de variables estándar	Bool	%I0.2
Mstapa	Tabla de variables estándar	Bool	%M2.2
al2	Tabla de variables estándar	Bool	%M2.4
Nivel	Tabla de variables estándar	Word	%IW64
Mnivel	Tabla de variables estándar	Word	%MW4
Nivelb	Tabla de variables estándar	Bool	%M2.5
Nivela	Tabla de variables estándar	Bool	%M2.6
mv5	Tabla de variables estándar	Bool	%M2.7
v5	Tabla de variables estándar	Bool	%Q12.3
M10	Tabla de variables estándar	Bool	%M10.4
M1	Tabla de variables estándar	Bool	%M9.3
M2	Tabla de variables estándar	Bool	%M9.4
M3	Tabla de variables estándar	Bool	%M9.5
M4	Tabla de variables estándar	Bool	%M9.6
M5	Tabla de variables estándar	Bool	%M9.7
M6	Tabla de variables estándar	Bool	%M10.0
M7	Tabla de variables estándar	Bool	%M10.1
M9	Tabla de variables estándar	Bool	%M10.3
M8	Tabla de variables estándar	Bool	%M10.2
M11	Tabla de variables estándar	Bool	%M10.5
Tag_1	Tabla de variables estándar	Word	%MW5
Tag_2	Tabla de variables estándar	Time	%ID0
M7_1	Tabla de variables estándar	Bool	%M10.6

