



ESPE

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

DEPARTAMENTO DE ELÉCTRICA Y ELÉCTRONICA

**CARRERA DE INGENIERÍA EN ELECTRÓNICA E
INSTRUMENTACIÓN**

**TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL
TÍTULO DE INGENIERO ELECTRÓNICO EN
INSTRUMENTACIÓN**

**TEMA: “DISEÑO E IMPLEMENTACIÓN DE UN SIMULADOR
DE AUTOMATIZACIÓN DE PROCESOS INDUSTRIALES
COMPATIBLE CON LOS PLCS SIEMENS Y ALLEN BRADLEY
PARA EL DESARROLLO DE APLICACIONES DE
COMUNICACIÓN Y CONTROL DE PROCESOS DE LA
UNIVERSIDAD DE LAS FUERZAS ARMADAS-ESPE
EXTENSIÓN LATACUNGA”**

**AUTORES: OSCAR FABRICIO BAYAS SÁNCHEZ
HENRY PAUL COCHA APUPALO**

DIRECTOR: ING. EDWIN PRUNA

LATACUNGA

2016



DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA
CARRERA DE INGENIERÍA EN ELECTRÓNICA E INSTRUMENTACIÓN

CERTIFICACIÓN

Certifico que el trabajo de titulación, **“DISEÑO E IMPLEMENTACIÓN DE UN SIMULADOR DE AUTOMATIZACIÓN DE PROCESOS INDUSTRIALES COMPATIBLE CON LOS PLCS SIEMENS Y ALLEN BRADLEY PARA EL DESARROLLO DE APLICACIONES DE COMUNICACIÓN Y CONTROL DE PROCESOS DE LA UNIVERSIDAD DE LAS FUERZAS ARMADAS-ESPE EXTENSIÓN LATACUNGA”** realizado por los señores **Oscar Fabricio Bayas Sánchez** y **Henry Paul Cocha Apupalo**, ha sido revisado en su totalidad y analizado por el software anti-plagio, el mismo cumple con los requisitos teóricos, científicos, técnicos, metodológicos y legales establecidos por la Universidad de Fuerzas Armadas ESPE, por lo tanto me permito acreditarlo y autorizar a los señores **Oscar Fabricio Bayas Sánchez** y **Henry Paul Cocha Apupalo** para que lo sustenten públicamente.

Latacunga, Agosto de 2016



Ing. Edwin Patricio Pruna Panchi



**DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA
CARRERA DE INGENIERÍA EN ELECTRÓNICA E INSTRUMENTACIÓN**

AUTORÍA DE RESPONSABILIDAD

Nosotros, **Oscar Fabricio Bayas Sánchez**, con cédula de identidad N°1804484168 y **Henry Paul Cocha Apupalo**, con cédula de identidad N°1804228243, declaramos que este trabajo de titulación **“DISEÑO E IMPLEMENTACIÓN DE UN SIMULADOR DE AUTOMATIZACIÓN DE PROCESOS INDUSTRIALES COMPATIBLE CON LOS PLCS SIEMENS Y ALLEN BRADLEY PARA EL DESARROLLO DE APLICACIONES DE COMUNICACIÓN Y CONTROL DE PROCESOS DE LA UNIVERSIDAD DE LAS FUERZAS ARMADAS-ESPE EXTENSIÓN LATACUNGA”** ha sido desarrollado considerando los métodos de investigación existentes, así como también se ha respetado los derechos intelectuales de terceros considerándose en las citas bibliográficas.

Consecuentemente declaramos que este trabajo es de nuestra autoría, en virtud de ello nos declaramos responsables del contenido, veracidad y alcance de la investigación mencionada.

Latacunga, Agosto de 2016

Oscar Fabricio Bayas Sánchez
C.C.:1804484168

Henry Paul Cocha Apupalo
C.C.:1804228243



**DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA
CARRERA DE INGENIERÍA EN ELECTRÓNICA E INSTRUMENTACIÓN**

AUTORIZACIÓN

Nosotros, **Oscar Fabricio Bayas Sánchez** y **Henry Paul Cocha Apupalo**, autorizamos a la Universidad de las Fuerzas Armadas ESPE publicar en la biblioteca Virtual de la institución el presente trabajo de titulación “**DISEÑO E IMPLEMENTACIÓN DE UN SIMULADOR DE AUTOMATIZACIÓN DE PROCESOS INDUSTRIALES COMPATIBLE CON LOS PLC’S SIEMENS Y ALLEN BRADLEY PARA EL DESARROLLO DE APLICACIONES DE COMUNICACIÓN Y CONTROL DE PROCESOS DE LA UNIVERSIDAD DE LAS FUERZAS ARMADAS-ESPE EXTENSIÓN LATACUNGA**” cuyo contenido, ideas y criterios son de nuestra autoría y responsabilidad.

Latacunga, Agosto de 2016



Oscar Fabricio Bayas Sánchez
C.C. 1804484168



Henry Paul Cocha Apupalo
C.C. 1804228243

DEDICATORIA

Este proyecto es dedicado en primer lugar a Dios por permitirme estar vivo, por cada día darme su bendición para salir afrontar todo lo que se me presenta en el camino. A mis padres Martha y Pedro que han sido mi motor y ejemplo para salir adelante y luchar por mis sueños. A mis abuelitos quienes son como mis segundos padres y me han guiado por un buen camino de respeto y responsabilidad. A mis hermanos Vanessa y Eduardo por brindarme su apoyo incondicional.

Henry

DEDICATORIA

Dedico este trabajo primeramente a Dios, por darme la vida y salud. A mis maravillosos padres que me brindan su apoyo incondicional en mis objetivos, quienes supieron encaminarme por el camino del bien con su sabiduría y experiencia para llegar alcanzar este logro en mi vida profesional, a mis hermanos y novia por darme ánimos para seguir adelante. A mis familiares, por las palabras de apoyo para culminar esta importante etapa de mi vida.

Oscar

AGRADECIMIENTO

A Dios por darme la fuerza y la valentía para sobresalir en mi vida a pesar de dificultades que se me han presentado a lo largo de mi estancia en la universidad.

A mis padres por siempre brindarme su amor, cariño, comprensión y confianza todo lo que tengo y soy se los debo y estaré infinitamente agradecido con ellos por siempre.

A mis abuelitos por cuidarme en mi niñez y brindarme su amor, a mis hermanos por compartir momentos únicos junto a mí, a toda mi familia por estar siempre pendiente de mis pasos y alentarme a seguir adelante.

A todos mis amigos que han estado conmigo en este largo camino de estudio gracias por brindarme su amistad sincera e incondicional.

De la misma forma agradezco a todos mis profesores que compartieron sus conocimientos y experiencias conmigo en especial a nuestro tutor el Ing. Edwin Pruna quien demostró ser una excelente persona y nos guio de manera oportuna a la culminación de este proyecto.

Henry

AGRADECIMIENTO

Agradezco a Dios, por la salud y la vida permitiéndome conocer personas maravillosas a lo largo de esta trayectoria profesional. A mis padres por su apoyo incondicional y a toda mi familia por su interés en mi superación.

Al tutor de tesis Ing. Edwin pruna, por la confianza y conocimientos impartidos para alcanzar la finalización de este proyecto y a mi compañero y amigo de tesis por la dedicación para cumplir con éxito este proyecto.

Oscar

ÍNDICE DE CONTENIDOS

CARÁTULA	i
CERTIFICACIÓN	ii
AUTORÍA DE RESPONSABILIDAD	iii
AUTORIZACIÓN	iv
DEDICATORIA	v
AGRADECIMIENTO	vii
ÍNDICE DE CONTENIDOS	ix
ÍNDICE DE FIGURAS	xv
ÍNDICE DE TABLAS	xix
RESUMEN	xx
ABSTRACT	xxi

CAPÍTULO I

FUNDAMENTACIÓN TEÓRICA

1.1. Antecedentes	1
1.2. Planteamiento del Problema	2
1.3. Justificación e Importancia	3
1.4. Introducción a los Simuladores	3
1.4.1. Definición de Simulación	4
1.4.2. Características de un Simulador	4
1.4.3. Ventajas de un Simulador	5
1.4.4. Campos de Aplicación.....	7
1.4.5. Aplicación de la simulación en la docencia	7
1.5. Simuladores de Automatización	7
1.5.1. DYN SIM	7
1.5.2. Tecnomatix Plant Simulation (SIEMENS)	8
1.5.3. ProModel.....	9
1.5.4. Simulador RSLogix Pro 500	10
1.6. Software necesario para el Desarrollo del simulador de automatización	12
1.6.1. Labview	12
1.6.2. Módulo DSC de Labview	14

1.6.3.	Módulo Control Design and Simulation de Labview	15
1.7.	Software de programación y simulación para PLC Allen Bradley ...	16
1.7.1.	RSLogix Emulate 500.....	16
1.7.2.	RSLogix 500.....	17
1.7.3.	RSLogix Emulate 5000.....	19
1.7.4.	RSLogix 5000.....	20
1.8.	Software de Programación para PLC Siemens.....	22
1.8.1.	TIA PORTAL	22
1.9.	OPC	26
1.9.1.	Historia	27
1.9.2.	Beneficios de OPC.....	27
1.10.	Servidores OPC	28
1.10.1.	RSLinx Classic.....	28
1.10.2.	KEPServerEx	29
1.11.	PLC SIEMENS S7-1200.....	29
1.11.1.	Introducción al PLC S7-1200	29
1.11.2.	Componentes del PLC Siemens S7-1200.....	30
1.11.3.	Características del PLC Siemens S7-1200	31
1.12.	PLC Allen Bradley CompactLogix 1769-L32E.....	31
1.12.1.	Introducción.....	31
1.12.2.	Características	32
1.12.3.	Partes del PLC CompactLogix 1769-L32E.....	32
1.13.	PLC Allen Bradley Micrologix 1200	34
1.13.1.	Descripción	34
1.13.2.	Características y ventajas clave.....	34

CAPÍTULO II

DISEÑO E IMPLEMENTACIÓN DEL SIMULADOR

2.1.	Introducción.....	35
2.2.	Requerimientos del Proyecto	35
2.3.	Diseño e Implementación del Simulador de Procesos Industriales .	36
2.4.	Diseño del HMI Principal del Simulador	37
2.4.2.	Pantalla Principal	38
2.4.3.	Menú Principal del Simulador.....	38

2.4.4.	Menús secundarios del Simulador	41
2.5.	Desarrollo de la simulación del Proceso Silo	46
2.5.1.	Diagrama de flujo del Proceso	46
2.5.2.	Descripción del Proceso.....	47
2.5.3.	Panel Frontal del Proceso Silo	47
2.5.4.	Programación del Proceso Silo	48
2.6.	Desarrollo de la simulación del Proceso Mezclador	57
2.6.1.	Diagrama de Flujo del Proceso	57
2.6.2.	Descripción del Proceso.....	57
2.6.3.	Panel Frontal del Proceso Mezclador.....	58
2.6.4.	Programación del Proceso Mezclador.....	59
2.7.	Desarrollo de la simulación del Proceso Semáforo	62
2.7.1.	Diagrama de Flujo del Proceso	62
2.7.2.	Descripción del Proceso.....	63
2.7.3.	Panel Frontal del Proceso Semáforo.....	63
2.7.4.	Programación del Proceso Semáforo.....	64
2.8.	Desarrollo de la simulación un Proceso Garaje	67
2.8.1.	Diagrama de Flujo del Proceso	67
2.8.2.	Descripción del Proceso.....	68
2.8.3.	Panel Frontal del Proceso Garaje	68
2.8.4.	Programación del Proceso Garaje	69
2.9.	Desarrollo Simulación de un Proceso de Nivel	72
2.9.1.	Diagrama de Flujo del Proceso	73
2.9.2.	Descripción del Proceso.....	73
2.9.3.	Panel Frontal del Proceso Nivel	74
2.9.4.	Etapas de programación de la simulación del Proceso Nivel.....	77
2.10.	Desarrollo Simulación de un Proceso de Flujo.....	81
2.10.1.	Diagrama de Flujo del Proceso	81
2.10.2.	Descripción del Proceso.....	81
2.10.3.	Panel Frontal del Proceso Flujo	82
2.10.4.	Etapas de programación de la simulación del Proceso Flujo	84
2.11.	Desarrollo de los SubVI Salto y Cerrar.....	87
2.11.1.	Desarrollo del sub-VI Salto.....	87
2.11.2.	Desarrollo del subVI Cerrar	88

2.12.	Configuración y Programación de un PLC SIEMENS S7-1200.....	89
2.12.1.	Creación de un nuevo proyecto en TIA PORTAL.....	89
2.12.2.	Agregar y Configurar un nuevo dispositivo.....	89
2.12.4.	Cargar un programa en la CPU del PLC S7-1200	91
2.13.	Configuración y Programación del PLC Allen Bradley Micrologix 1200	92
2.13.1.	Configurar RSLinx para PLC Micrologix 1200.....	92
2.13.2.	Configurar RSLogix 500 para PLC Micrologix 1200.....	93
2.15.	Configuración y Programación del PLC Allen Bradley Micrologix Emulate	94
2.15.1.	Configurar RSLinx para PLC Micrologix Emulate.....	94
2.15.2.	Configurar RSLogix 500 para PLC Micrologix Emulate.....	95
2.15.3.	Configurar RSLogix Emulate 500.....	96
2.15.4.	Cargar programa en RSLogix 500	97
2.16.	Configuración y Programación del PLC Allen Bradley Compact Logix 1769-L32E	98
2.16.1.	Configurar RSLinx Classic para PLC Compact Logix 1769-L32E ...	98
2.16.2.	Configurar RSLogix 5000 para PLC CompactLogix 1769-L32E.....	99
2.17.	Configuración y Programación del PLC Allen Bradley Compact Logix Emulate	99
2.17.1.	Configurar RSEmulate 5000	100
2.17.2.	Configurar RSLinx Classic para PLC Compact Logix Emulate	101
2.17.3.	Configurar RSLogix 5000 para PLC CompactLogix Emulate	102
2.17.4.	Cargar programa en RSLogix 5000	103
2.18.	Programación de los controladores PID.....	104
2.18.1.	Programación de un control PID para un PLC Siemens S7 1200 .	104

CAPÍTULO III

RESULTADOS Y PRUEBAS EXPERIMENTALES

3.1.	Introducción.....	114
3.3.1.	Configuración del servidor OPC para PLC SIEMENS S7-1200	115
3.3.2.	Configuración del servidor OPC para PLC Allen Bradley.....	117
3.4.	Pruebas Experimentales del funcionamiento y Análisis de Resultados de la simulación del Proceso Silo.....	119

3.4.1.	Descripción del funcionamiento del proceso	119
3.4.2.	Pruebas de funcionamiento de la Simulación Silo	119
3.5.	Pruebas Experimentales del funcionamiento y Análisis de Resultados de la simulación del Proceso Semáforo	123
3.5.1.	Descripción del funcionamiento del proceso	123
3.5.2.	Pruebas de funcionamiento de la Simulación Semáforo	123
3.6.	Pruebas Experimentales del funcionamiento y Análisis de Resultados de la simulación del Proceso Garaje	127
3.6.1.	Descripción del funcionamiento del proceso	127
3.6.2.	Pruebas de Funcionamiento de la simulación Garaje	127
3.7.	Pruebas Experimentales del funcionamiento y Análisis de Resultados de la simulación del Proceso Mezclador	131
3.7.1.	Descripción del funcionamiento del proceso	131
3.7.2.	Pruebas de Funcionamiento de la simulación Mezclador	131
3.8.	Pruebas Experimentales del funcionamiento de la simulación del Proceso Nivel	135
3.8.1.	Descripción del funcionamiento del proceso	135
3.8.2.	Diagrama Ladder	135
3.8.3.	Sintonización del Proceso Nivel	136
3.8.4.	Pruebas de funcionamiento de la simulación Nivel	137
3.9.	Pruebas Experimentales del funcionamiento de la simulación del Proceso Flujo	140
3.9.1.	Descripción del funcionamiento del proceso	140
3.9.2.	Diagrama Ladder	140
3.9.3.	Sintonización del Proceso Flujo	141
3.10.	Alcances y Limitaciones	145
3.10.1.	Alcances.....	145
3.10.2.	Limitaciones	145

CAPÍTULO IV

CONCLUSIONES Y RECOMENDACIONES

4.1.	Conclusiones.....	146
4.1.1.	Simulador	146
4.2.	Recomendaciones.....	147

BIBLIOGRAFÍA.....	148
LINKOGRAFÍA.....	148
ANEXOS.....	151

ÍNDICE DE FIGURAS

Figura 1: Software DYN SIM.....	8
Figura 2: Software Tecnomatix	9
Figura 3: Software ProModel	10
Figura 4: Simulador RSLogix Pro.....	11
Figura 5: Componentes de la pantalla principal de LogixPro	11
Figura 6: Software Labview	12
Figura 7: Panel Frontal Labview	13
Figura 8: Diagrama de Bloques	14
Figura 9: Módulo DSC Labview	14
Figura 10: Toolkit Control Design and Simulation	15
Figura 11: RSLogix Emulate 500	16
Figura 12: RSLogix 500	19
Figura 13: RSLogix Emulate 5000	19
Figura 14: Módulos de Emulación.....	20
Figura 15: RSLogix 5000	21
Figura 16: Software TIA PORTAL.....	22
Figura 17: Vista de Red	23
Figura 18: Vista de Dispositivo.....	24
Figura 19: Vista Topológica	24
Figura 20: Lenguaje KOP	25
Figura 21: Modo Online	26
Figura 22. RSLinx Classic.....	28
Figura 23: OPC KEPServerEx	29
Figura 24: PLC Siemens S7-1200	30
Figura 25: Componentes del PLC S7-1200	30
Figura 26: PLC Allen Bradley CompactLogix	32
Figura 27: Partes de un PLC CompactLogix.....	33
Figura 28: PLC Allen Bradley Micrologix.....	34
Figura 29: Diseño del Simulador de Procesos Industriales.....	36
Figura 30: Implementación de Simulador de Procesos Industriales	36
Figura 31: Diagrama de Flujo del HMI del Simulador	37
Figura 32: Pantalla Principal del Simulador	38
Figura 33: Menú Principal del Simulador.	39
Figura 34: Diagrama de bloques del Menú Principal	39
Figura 35 :Segundo evento de la función "Event Structure".	40
Figura 36: Tercer evento de la función "Event Structure".	40
Figura 37: Acceso al Menú Siemens.	41
Figura 38: Panel Frontal Menú Siemens.....	42
Figura 39: Diagrama de bloques del Menú Siemens.	42
Figura 40: Acceso al Menú Allen Bradley Micrologix	44
Figura 41: Panel Frontal Menú de PLC Micrologix.....	44
Figura 42: Acceso al Menú Allen Bradley CompactLogix	45
Figura 43: Panel Frontal Menú de PLC CompactLogix.....	45
Figura 44: Diagrama de Flujo Programación Silo	46
Figura 45: Panel frontal del Proceso Silo.....	47
Figura 46: Diagrama de bloques General del Proceso Silo	48
Figura 47: Diagrama de bloques Simulación del Proceso Silo.....	49

Figura 48: Control de motor en proceso silo.	50
Figura 49: Función “Case Structure” cuando el motor está activado.	51
Figura 50: Función “Case Structure” cuando el motor esta desactivado.	52
Figura 51: Indicadores led de la acción del proceso	53
Figura 52: Transformación de un indicador numérico a booleano	54
Figura 53: Localización de la función “One Button Dialog”	54
Figura 54: Activación de errores en Simulación Silo	54
Figura 55: Slide de 0 a 40 unidades tanque 1.	55
Figura 56: Slide de 20 a 40 unidades tanque 2.....	56
Figura 57: Slide de 20 a 40 unidades tanque3 lleno.....	56
Figura 58: Tanque de llenado de 0 a 10 unidades.....	56
Figura 59: Diagrama de Flujo Programación Mezclador	57
Figura 60: Panel frontal del Proceso Mezclador	58
Figura 61: Diagrama de bloques General del proceso Mezclador.....	59
Figura 62: Diagrama de bloques Simulación del Proceso Mezclador.....	60
Figura 63: Diagrama de Flujo Programación Semáforo	63
Figura 64: Panel frontal de proyecto semáforo.	64
Figura 65: Diagrama de Bloques General del Proceso semáforo.....	65
Figura 66: Diagrama de bloques Simulación del Proceso Semáforo.....	66
Figura 67: Diagrama de Flujo Programación Garaje.....	68
Figura 68: Panel frontal del Proceso Garaje.....	69
Figura 69: Diagrama de bloques completo del Proceso Garaje.....	70
Figura 70: Diagrama de bloques Simulación del Proceso Garaje.....	71
Figura 71: Diagrama de Flujo del Proceso Nivel.....	73
Figura 72: Presentación del Proceso Nivel	74
Figura 73: Ventana Sintonización de Proceso	75
Figura 74: Ventana Históricas.....	75
Figura 75: Ventana Alarmas	76
Figura 76: Diagrama de Bloque de Simulación Proceso Nivel.....	77
Figura 77: Función Blinking	78
Figura 78: Elementos del SubVI Plant Simulator	79
Figura 79: Configuración de subVi Plant Simulator.....	80
Figura 80: Diagrama de Flujo del Proceso Flujo	81
Figura 81: Presentación del Proceso Flujo	82
Figura 82: Ventana Sintonización de Proceso Flujo	83
Figura 83: Ventana Históricas de Proceso Flujo	83
Figura 84: Ventana Alarmas de Proceso Flujo.....	84
Figura 85: Diagrama de Bloques de Simulación Proceso Flujo	85
Figura 86: Configuración de subVi Plant Simulator.....	86
Figura 87: Apariencia del icono del Sub-VI salto.	87
Figura 88: Diagrama de bloques de Sub-VI salto.	87
Figura 89: Apariencia del icono del Sub-VI Cerrar.....	88
Figura 90: Función Cerrar VI	88
Figura 91: Ventana para crear un nuevo proyecto.....	89
Figura 92: Agregar un nuevo dispositivo.....	90
Figura 93: Selección CPU del Controlador	90
Figura 94: Asignación de dirección IP.....	91
Figura 95: Configuración Avanzada para cargar un programa en el PLC....	91
Figura 96: Configuración RSLinx para PLC Micrologix 1200	92
Figura 97: Configuración de comunicación entre RSLinx y PLC	93

Figura 98: Seleccionar el Tipo de Controlador en RSLogix 500	93
Figura 99: Propiedades del Controlador	94
Figura 100: Configuración RSLinx para Micrologix Emulate	94
Figura 101: Configuración driver del emulador SLC 500	95
Figura 102: Seleccionar el Tipo de Controlador en RSLogix 500	95
Figura 103: Configuración del Controlador en RSLogix	96
Figura 104: Abrir proyecto en RSLogix Emulate 500	96
Figura 105: Configuración RSEmulate 500	97
Figura 106: Cargar un programa en RSLogix	97
Figura 107: Configuración RSLinx Classic	98
Figura 108: Configuración del nuevo driver instalado	99
Figura 109: Configuración nuevo proyecto RSLogix 5000	99
Figura 110: RSLogix Emulate 5000	100
Figura 111: Añadir un nuevo módulo	100
Figura 112: Módulo de entradas y salidas	101
Figura 113: Configuración RSLinx	101
Figura 114: Configuración de un nuevo proyecto en RSLogix 5000	102
Figura 115: Crear Nuevo Módulo De Entradas y Salidas 1756-MODULE .	103
Figura 116: Parámetros de Conexión	103
Figura 117: Cargar el Programa en RSLogix 5000	104
Figura 118: Añadir un nuevo bloque	104
Figura 119: Crear variables para ser utilizadas por el PLC	105
Figura 120: Definición de Variables del Control PID	105
Figura 121: Agregar un nuevo bloque tecnológico	106
Figura 122: Objeto Tecnológico PID	106
Figura 123: Asignación de Variables al Bloque PID	107
Figura 124: Abrir la ventana de configuración del bloque PID	107
Figura 125: Tipo de Regulación	108
Figura 126: Parámetros de Entrada/Salida	108
Figura 127: Escala del valor de entrada	108
Figura 128: Escalar Variable de Salida	109
Figura 129: Parámetros PID	109
Figura 130: Asignación nombre instrucción PID	110
Figura 131: Creación de Variables para control PID	110
Figura 132: Variables de PID	111
Figura 133: Bloque PID	112
Figura 134: Configuración Bloque PID	112
Figura 135: Escala Bloque PID	113
Figura 136: Ajuste de Bloque PID	113
Figura 137: Ingreso al Simulador de Procesos	114
Figura 138: Configuración Canal	115
Figura 139: Configuración Dispositivo	116
Figura 140: Configuración Tag	117
Figura 141: Configuración OPC	117
Figura 142: Diagrama de Funcionamiento Silo	119
Figura 143: Pruebas de Funcionamiento del Proceso Silo	121
Figura 144: Programación Ladder Proceso Silo	122
Figura 145: Diagrama de Funcionamiento Semáforo	123
Figura 146: Pruebas de Funcionamiento del Proceso Semáforo	125
Figura 147: Programación Ladder del Proceso Semáforo	126

Figura 148: Diagrama de Funcionamiento Garaje	127
Figura 149: Pruebas de Funcionamiento del Proceso Garaje	129
Figura 150: Programación Ladder del Proceso Garaje	130
Figura 151: Diagrama de Funcionamiento del Proceso Mezclador.....	131
Figura 152: Pruebas de Funcionamiento del Proceso Mezclador.....	133
Figura 153: Programación Ladder del Proceso Mezclador	134
Figura 154: Diagrama Ladder de Control PID de Nivel.....	135
Figura 155: Resultados del controlador PID	137
Figura 156: Presentación del Proceso Nivel-PLC Siemens	138
Figura 157: Sintonización de Proceso Nivel-PLC Siemens	138
Figura 158: Históricos del Proceso Nivel-PLC Siemens	139
Figura 159: Alarmas del Proceso Nivel-PLC Siemens.....	139
Figura 160: Programación Ladder del Proceso Flujo.....	140
Figura 161: Resultados PID Flujo	142
Figura 162: Presentación del Proceso Flujo-Compactlogix	143
Figura 163: Sintonización del Proceso Flujo-Compactlogix	143
Figura 164: Históricos Variable Flujo-Compactlogix	144
Figura 165: Alarmas del Proceso Flujo-Compactlogix	144

ÍNDICE DE TABLAS

Tabla 1 Beneficios de DYN SIM.....	8
Tabla 2 Características de RSLogix 500.....	18
Tabla 3 Características de RSLogix 5000.....	21
Tabla 4 Partes del PLC Siemens S7-1200	31
Tabla 5 Tecnologías de PLC compatibles con el simulador	38
Tabla 6 Indicadores y Controles del Panel Frontal Silo	48
Tabla 7 Indicadores y Controles del Panel Frontal Mezclador.....	59
Tabla 8 Indicadores y Controles del Panel Frontal Semáforo	64
Tabla 9 Indicadores y Controles del Panel Frontal Garaje.....	69
Tabla 10 Variables del Panel Frontal del Proceso Nivel	76
Tabla 11 Variables del Panel Frontal del Proceso Flujo	84
Tabla 12 Tags del Bloque PID	111
Tabla 13 Nombres de código de conexión de los procesos al OPC.	118
Tabla 14 Nombres de código de conexión de los procesos al OPC.	118
Tabla 15 Variables del Proceso Silo	120
Tabla 16 Variables del Proceso Semáforo.....	124
Tabla 17 Variables del Proceso Garaje	128
Tabla 18 Variables del Proceso Mezclador.....	132
Tabla 19 Constantes de sintonización de la simulación Nivel.....	136
Tabla 20 Variables del Proceso Nivel	137
Tabla 21 Constantes de sintonización de la simulación Flujo.....	141
Tabla 22 Variables del Proceso Flujo	142

RESUMEN

El desarrollo de este proyecto consiste en el diseño e implementación de un simulador de automatización de procesos industriales, este permite desarrollar aplicaciones de comunicación y control, dicho simulador se lo realizó en Labview, un software muy común para los estudiantes de ingeniería, en el cual se desarrolló la simulación de varios procesos industriales para realizar prácticas sobre la programación de PLC'S en el laboratorio de Control de Procesos y Redes Industriales el mismo que permitirá a estudiantes desplegar sus conocimientos en el ámbito de comunicaciones industriales, instrumentación industrial, control de procesos y controladores lógicos programables. Los procesos que se simularon son: 4 procesos de automatización tales como Silo, Semáforo, Mezclador, Garaje y el control de dos procesos Nivel y Flujo. El desarrollo de la programación del PLC se lo realiza en un software determinado dependiendo con que controlador se va a trabajar, si se trabaja con PLC Siemens se programa en TIA Portal, con tecnología Allen Bradley se trabaja con RSLogix 500 o RSLogix 5000 para la programación de Micrologix y CompactLogix respectivamente, una vez programado el controlador se procede a comunicar el PLC con el servidor OPC para poder realizar las respectivas pruebas en cada uno de los procesos industriales disponibles en el simulador desarrollado en este proyecto.

PALABRAS CLAVE:

- **AUTOMATIZACIÓN INDUSTRIAL**
- **LABVIEW**
- **PROCESOS INDUSTRIALES**
- **CONTROLADOR LÓGICO PROGRAMABLE**

ABSTRACT

The development of this project consists in the design and implementation of a simulator of industrial process automation, this project allows to develop communication and control applications, this simulator was made in Labview, a common software for engineering students, in which was developed the simulation of various industrial processes for programming PLC'S practice on laboratory of process control and Networking industrial, the same that allows students display their knowledge in the field of industrial communications, industrial instrumentation, process control and programmable logic controllers. The processes that has been simulated are: 4 automation processes such as Silo, Stoplight, Mixer, garage and two processes control Level & Flow. The development of PLC programming is done in a particular software depending on that controller will work, if you work with Siemens PLC is programmed in TIA Portal, with Allen Bradley technology works with RSLogix 500 or RSLogix 5000 for the programming of CompactLogix and Micrologix respectively, once programmed the controller proceeds to the PLC communicate with the OPC server to perform the respective tests in each of the available industrial processes in the simulator developed in this project.

KEYWORDS:

- **INDUSTRIAL AUTOMATION**
- **LABVIEW**
- **INDUSTRIAL PROCESSES**
- **PROGRAMMABLE LOGIC CONTROLLER**

CAPITULO I

FUNDAMENTACIÓN TEÓRICA

1.1. Antecedentes

En los últimos años, una de las actividades importantes dentro de la industria es la automatización de procesos industriales, para lo cual se recurre a equipos tecnológicos de altas prestaciones que faciliten la comunicación y control de dichos procesos. Hoy en día los computadores son capaces de almacenar, procesar y mostrar una gran cantidad de información de forma confiable, la tendencia que se presenta en la industria es enlazar sus procesos automatizados a programas HMI's (Human Machine Interface), los cuales posean ambientes que simulen a cabalidad el funcionamiento y como está formado el proceso.

La simulación de procesos es muy importante tanto en el campo industrial como en el educativo, ya que ofrece una representación gráfica de un proceso, pero de una forma más simple y entendible. Existen simuladores desarrollados en Lenguaje Visual C++, que permiten realizar el monitoreo y control de un proceso, pero son compatibles con cierto grupo de dispositivos y no abarca una gran cantidad de aplicaciones.

Los dispositivos comúnmente que se utilizan para realizar el control de un proceso son los PLC'S, existe simuladores que se acoplan a diferentes tecnologías de controladores lógicos programables, por ejemplo, para controladores Micrologix de la marca Allen Bradley se tiene el simulador RSLogix Pro, este permite simular procesos de automatización tales como un silo, mezclador, semáforo, etc. En este software se programa mediante lenguaje ladder para comprobar el correcto funcionamiento de los procesos que contiene este simulador.

Para los PLC'S Siemens 1200 se puede utilizar el paquete TIA Portal, que permite simular los programas que se realicen en estos controladores. Sin

embargo, no se cuenta con la simulación de procesos industriales cuando se realiza la programación de ciertos controladores tales como los ControlLogix, CompactLogix entre otros, con lo que no se puede comprobar de una manera idónea su funcionamiento, la necesidad de ofrecer un simulador de procesos industriales que brinde la opción de trabajar con estos PLC es alta por lo que este proyecto se enfoca en ello.

1.2. Planteamiento del Problema

Actualmente existen varios simuladores que son licenciados a un costo elevado y la función de cada uno de ellos es para un solo tipo de PLC, tal es el caso de los Micrologix de la marca Allen Bradley cuyo simulador es Logix Pro, el cual permite simular sistemas de automatización tales como silo, semáforo, etc. Sin embargo, su uso es limitado tan solo para estos equipos, además no se permite la conexión física de este simulador con los PLC Allen Bradley, lo que provoca que muchos equipos no se pueda demostrar su funcionamiento de una manera real.

En la institución existen poca cantidad de procesos industriales donde se pueda automatizar mediante tareas de comunicación y control automático, sin embargo, en contraste a ello existen muchos PLC'S, que no pueden ser utilizados a menudo debido a la falta de procesos industriales. Hay que añadir que las estaciones de procesos que dispone la institución son muy costosas y el poco espacio físico donde puedan ser colocadas provoca que no se implementen un mayor número de ellas, por lo que imposibilita de alguna manera que se desarrolle de forma apropiada el aprendizaje del manejo correcto de ciertos controladores.

Además otra problemática que se presenta es la gran cantidad de estudiantes que utilizan los laboratorios de la institución, por lo que a la hora de realizar sus prácticas de PLC'S se tiene un panorama un tanto problemático ya que en una misma estación de algún proceso se encuentran muchos estudiantes lo que reduce el grado de comprensión y manipulación de la estación y de los equipos con los cuales se realiza una práctica, de este modo es necesario la creación de un software que facilite la comprensión de

la técnica de programación de los PLC de una mejor manera y en menor tiempo. Entonces integrando recursos humanos, tecnológicos y competencias intelectuales se podrá crear dicho simulador para el aprendizaje de control de un PLC.

1.3. Justificación e Importancia

La motivación principal para el desarrollo de este simulador de automatización de procesos industriales, es que permitiría tener un entorno de trabajo ideal a la hora de realizar pruebas con los controladores lógicos programables disponibles en la institución, para que la actividad de aprendizaje de esta tecnología tenga un sustento práctico y no solo se quede en teoría, esto permitirá a estudiantes y maestros adquirir conocimientos sólidos sobre este tema.

El profesor guía tendrá un software en el cual se simulará procesos industriales, que por diferentes circunstancias no pueden ser implementados de manera real en la institución, con esto se logrará realizar prácticas que anteriormente no fueron efectuadas por la falta de procesos industriales, además evitará que en un mismo proceso se encuentren gran cantidad de alumnos ya que solamente hará falta tener un computador con el simulador instalado y un PLC para realizar la práctica.

Hay que destacar que el desarrollo de productos basados en el estándar OPC, así como en las metodologías actuales de diseño de software, garantizarán en un futuro cercano una alta compatibilidad entre plataformas tecnológicas en el área de la automatización industrial, ya que se podrá adaptarse a otros dispositivos con lo que permitirá estar acorde a los nuevos estándares en la comunicación y control de algún proceso industrial. El proyecto ofrece un simulador con grandes ventajas, bajo costo, fácil de utilizar, ideal para prácticas en la institución.

1.4. Introducción a los Simuladores

En la siguiente sección se va detallar diferentes conceptos útiles para entender el desarrollo de este proyecto de titulación.

1.4.1. Definición de Simulación

Mediante la simulación es posible verificar el desempeño de cualquier sistema que posea un modelo de comportamiento, además se puede distinguir los parámetros que caracterizan a dicha simulación. (Salas Ramón, 1995)

Un simulador es un paquete de software, este permite el desarrollo y representación de un sistema. Gracias a un simulador se logra representar efectos y experiencias que pueden suceder en la realidad.

1.4.2. Características de un Simulador

Un Simulador posee algunas características de las cuales se destacan las siguientes:

a. Los entornos gráficos: Permiten trabajar con imágenes de alta resolución y representación gráfica todo esto gracias a la ayuda de sistemas operativos modernos y lenguajes de programación, esto brinda un entorno de simulación con grandes ventajas.

b. La posibilidad de conexión con el exterior: Da un valor significativo a un simulador debido a que es posible desarrollar entornos de simulación que se acercan a la realidad, la conexión se realiza mediante el uso de módulos de adquisición de datos.

c. La incorporación de módulos de planificación del aprendizaje: Dentro de un simulador permite tener una herramienta con grandes prestaciones para facilitar la correcta conducción del aprendizaje, actualmente estos módulos son muy poderosos debido a que integran lenguajes de autor dentro de entornos de simulación coordinados por un Sistema Experto Planificador.

d. Posibilidad de conexión con otros programas: Algo que destaca de manera notable a cualquier programa simulador es la capacidad de enlazar, interactuar con otros paquetes de software, existen diferentes

técnicas que un simulador debe aprovechar al máximo para lograr el intercambio dinámico de datos e incorporar estándares de gestión de funciones y librerías escritas en diferentes tipos de lenguajes de programación.

e. Lenguajes de Programación Gráfica: Existe una gran cantidad de herramientas de simulación que permiten trabajar mediante programación gráfica. Para lo cual cuando se va a realizar un proyecto de simulación se programa mediante objetos gráficos que puedan vincularse configurando un esquema de bloques jerarquizado por categorías los cuales mostrarán la información de la simulación.

f. Posibilidad de ampliación de Bibliotecas: Esta característica única permite a los usuarios crear y modificar sus propios bloques de programación e incorporarlos en sus propias librerías cuyo objetivo es adaptar la herramienta a su campo de trabajo.

g. Interfaces Hombre-Máquina: El término HMI (Human Machine Interface) permite distinguir las características más importantes dentro de los sistemas de control y supervisión. Estas interfaces graficas contienen objetos los cuales son configurados de manera que muestren al usuario de una forma más intuitiva toda la información necesaria del proceso que se desea trabajar.

h. Instrumentación Virtual: Para el desarrollo de un proyecto de simulación es necesario el uso de instrumentos electrónicos. Actualmente los entornos poseen módulos que simulan de manera idónea estos instrumentos. (Gutiérrez, 2008)

1.4.3. Ventajas de un Simulador

Un Simulador posee ventajas de las cuales se pueden destacar las más importantes tales como:

El Aprendizaje por Descubrimiento: Es cuando alguien es capaz de aprender de manera autónoma. Se plantea una hipótesis en la cual el alumno se embarca en la búsqueda de causa y efecto de los problemas planteados. Además, el alumno debe ser capaz de analizar y probar el comportamiento de un modelo con varios factores y aplicando varios métodos. Uno de los métodos de aprendizaje más utilizado es el de ensayo-error.

Fomentar la Creatividad: La posibilidad de disponer de cajas de herramientas en los entornos permite la disponibilidad de un laboratorio, taller, o mesa de diseño con la que el alumno pueda no sólo simular modelos que se le den hechos, sino que pueda construir los suyos propios. Por definición una herramienta de simulación debe permitir de modo sencillo la edición de diversos escenarios, esquemas o plantas.

Ahorrar tiempo y dinero: El ahorro que representa el uso de herramientas de simulación es evidente todo es gracias a que se sustituye los equipos de entrenamiento, laboratorios y plantas de entrenamiento por un entorno virtual. Procesar la información no es fácil, tanto la adquisición, ordenación, tratamiento y análisis de la información son aspectos muy importantes durante el proceso de aprendizaje. La cantidad de conocimientos que hay que aprender hace necesario el utilizar técnicas de aprendizaje que aceleren el proceso. La simulación es una de ellas ya que facilita la construcción de los modelos y el tratamiento repetitivo de los datos.

La Enseñanza Individualizada: Es importante recalcar que las herramientas de simulación permiten al alumno lleve su propio método de aprendizaje de tal modo que afronte de modo individual el proceso de elaboración de sus propias conclusiones con relación a los fenómenos que va a simular. La gran ventaja de los simuladores es que el alumno puede repetir cuantas veces quiera la simulación de un mismo fenómeno o proceso hasta que tenga la seguridad de haber captado las ideas.

La autoevaluación: Algo destacable de la simulación es que permite al alumno realizar acciones orientadas a su propia autoevaluación mediante el planteamiento de guiones y cuestionarios orientados al tema que está

estudiando. Esta posibilidad ha permitido que se puedan establecer tutorías a distancia y aprendizaje no presencial. (Gutiérrez, 2008)

1.4.4. Campos de Aplicación

Los campos de aplicación de las herramientas de simulación se destacan los siguientes:

- Simulación de Procesos, Circuitos y Sistemas
- Control de maquetas
- Adquisición de Datos y medidas de Variables físicas.
- Entorno de Entrenamiento con Operadores Técnicos.
- Laboratorio Virtual.
- Interacción con el medio externo. (Gutiérrez, 2008)

1.4.5. Aplicación de la simulación en la docencia

Mediante los simuladores se puede desarrollar prácticas simuladas. La simulación permite el aprendizaje de un proceso real. Tanto el profesor como el alumno interactúan con hipótesis para trabajar en base a una estación real. Es una gran ventaja para el desenvolvimiento del alumnado implementar la simulación de procesos reales para conocer sus características sin necesidad de tener acceso a dichos procesos de forma física. (Calvo Guadaño, 2013)

1.5. Simuladores de Automatización

Existen algunos simuladores de automatización que se encuentran disponibles para la industria actualmente como son:

1.5.1. DYNSIM

El software DYNSIM (**Figura 1**) permite diseñar una planta procesadora de forma segura. Además, es posible realizar el diseño, análisis operacional, simulación, para reducir la inversión y mejorar la producción de procesos. (DYNSIM, 2015)

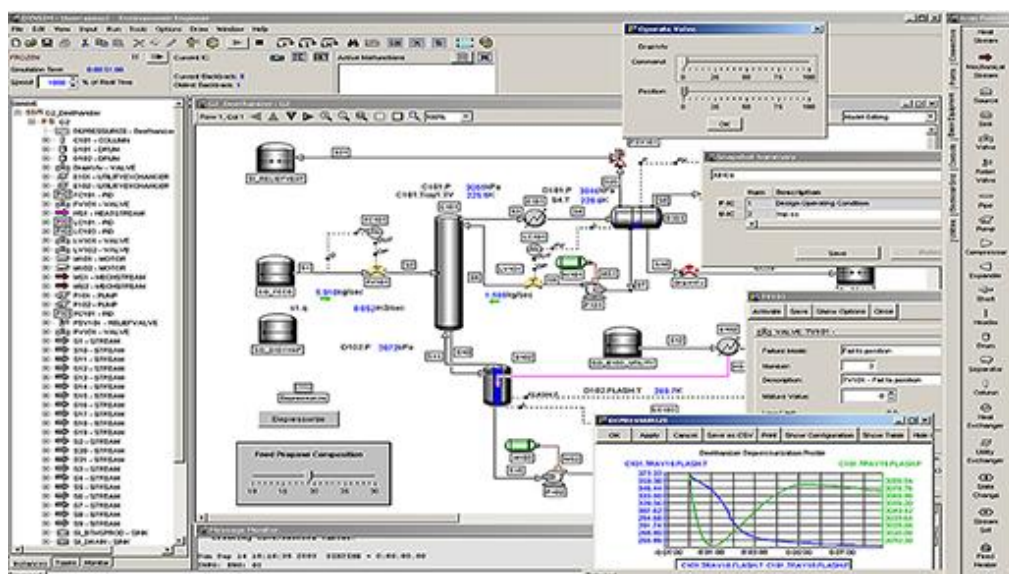


Figura 1: Software DYNsIM.

Fuente: (DYNsIM, 2015)

1.5.1.1. Beneficios de DYNsIM

Los principales beneficios que ofrece este simulador son los que se presentan en la (Tabla 1).

Tabla 1

Beneficios de DYNsIM

Beneficios de DYNsIM	
Reducción de costos en la producción de la planta	Aumento de Seguridad
Reducción de Riesgos	Procedimientos de Puesta en Marcha y Cierre
Aumento de la eficiencia de la planta	Evaluación de Procesos

Fuente: (DYNsIM, 2015)

1.5.2. Tecnomatix Plant Simulation (SIEMENS)

Mediante el uso Plant Simulation se puede modelar y simular los sistemas de producción de procesos. Además, mediante estos modelos digitales es posible simular las características de la planta. Tecnomatix Plant Simulation (Figura 2) es un paquete de software que permite crear modelos digitales de

sistemas para la verificar el correcto funcionamiento del mismo y analizar su rendimiento. Además, se puede ahorrar el uso de recursos y la logística en los niveles de planificación de la planta. (SIEMENS, 2016)

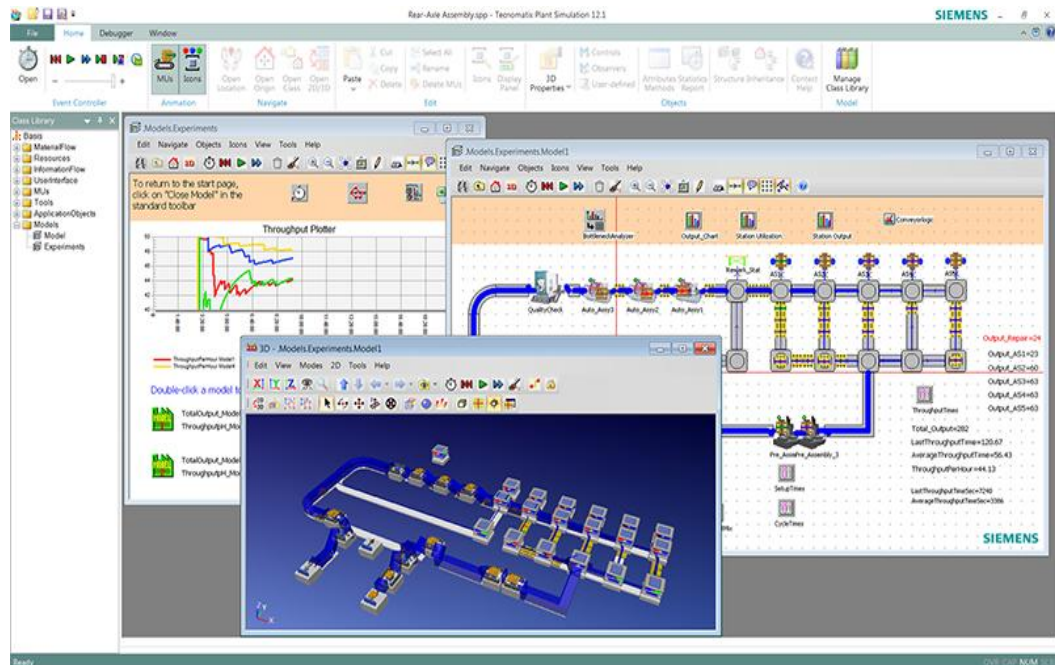


Figura 2: Software Tecnomatix

Fuente: (SIEMENS, 2016)

1.5.3. ProModel

ProModel es un software que posee animaciones para computadoras. Con este es posible simular cualquier tipo de sistemas de manufactura, logística, etc. Se puede simular bandas transportadoras, grúas, ensamble, corte, talleres, etc. ProModel (**Figura 3**) que no requiere que se realice la programación. Otras características que destacan es la facilidad de uso y flexibilidad para aplicaciones complejas. El módulo de optimización nos ayuda a encontrar rápidamente la solución óptima, en lugar de solamente hacer prueba y error. (ProModel, 2014)

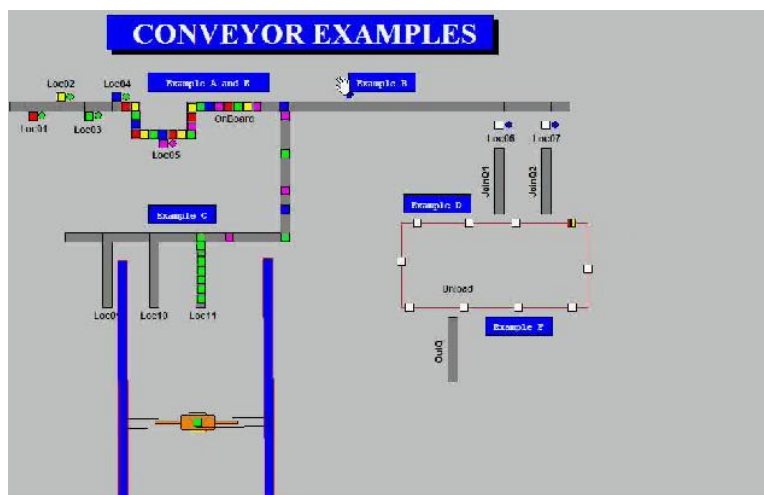


Figura 3: Software ProModel

Fuente: (ProModel, 2014)

1.5.4. Simulador RSLogix Pro 500

LogixPro 500 es una herramienta ideal para el aprendizaje de los fundamentos de la programación lógica de escalera. La apariencia, el funcionamiento del editor de escalera de LogixPro realmente imita muy bien el software de edición de PLC de renombre mundial de Rockwell. Este programa posee una ventana que contiene simulaciones de varios procesos industriales, tales como transportadores, plantas embotelladoras, etc. En software, la naturaleza sincrónica e interactiva de los procesos industriales reales, presenta al estudiante una experiencia de programación mucho más realista y desafiante.

Tanto si eres un instructor asignado para RSLogix, un estudiante que estudia la programación de PLC o simplemente alguien interesado en aprender RSLogix de Allen Bradley, entonces LogixPro (**Figura 4**) es un programa sumamente ideal.

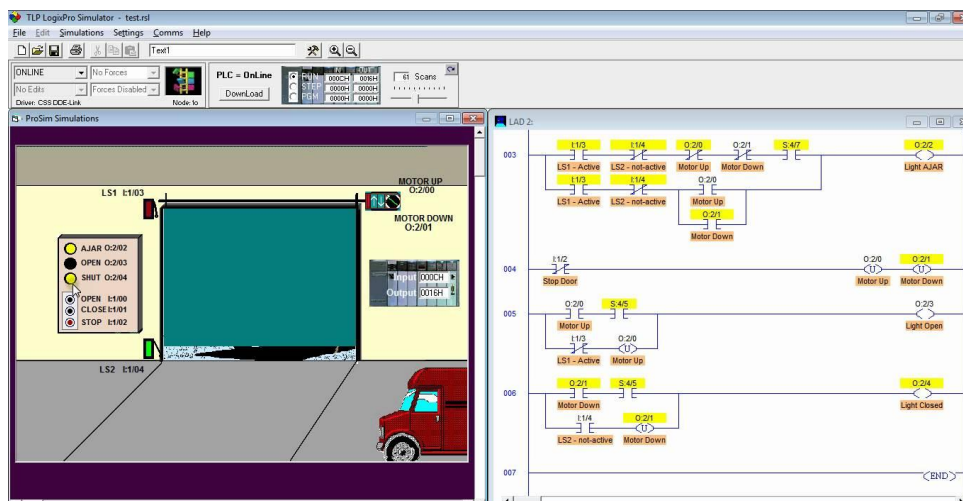


Figura 4: Simulador RSLogix Pro

Fuente: (Thelearningpit, 2012)

LogixPro ha sido diseñado para aumentar y mejorar los programas de formación del PLC, no para sustituirlas. La principal ventaja de LogixPro es que elimina la dependencia de tener un PLC físico, RSLogix y una serie de componentes eléctricos con el fin de llevar a cabo la formación de la programación del PLC. (Thelearningpit, 2012)

La pantalla principal (**Figura5**) del simulador contiene algunas partes tales como la barra de menú, la ventana donde se encuentra la simulación respectiva del proceso que deseamos automatizar y las instrucciones para realizar la programación ladder del proceso.

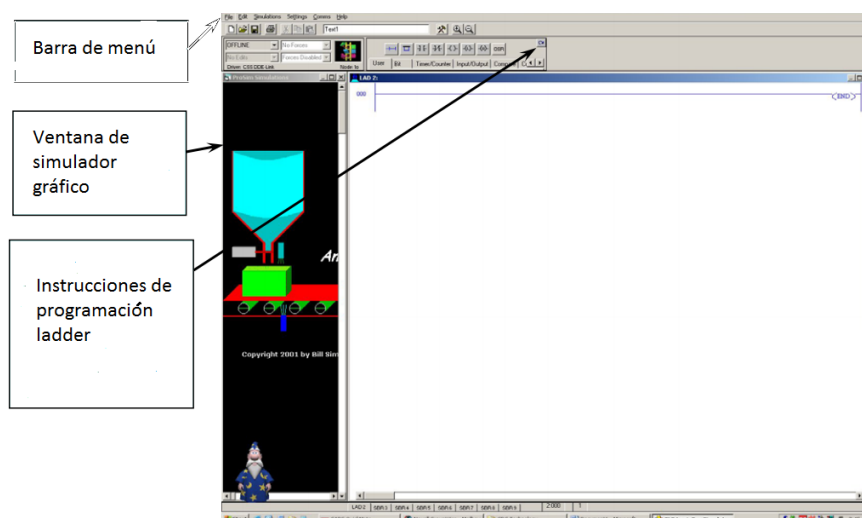


Figura 5: Componentes de la pantalla principal de LogixPro

Fuente: (Thelearningpit, 2012)

1.6. Software necesario para el Desarrollo del simulador de automatización

1.6.1. Labview

Es una herramienta de programación gráfica. Este programa estaba orientado hacia aplicaciones de control de equipos electrónicos usado en el desarrollo de sistemas de instrumentación, lo que comúnmente se conoce como instrumentación virtual. La extensión de los programas que son creados en Labview (**Figura 6**) se guardan en ficheros denominados VI (Virtual Instrumento). Labview posee dos ventanas principales, el panel frontal donde se encuentra los botones, pantallas, indicadores, controles numéricos, etc. La otra ventana se llama Diagrama de Bloques en esta se realiza toda la programación y suele tener un fondo blanco. (José Rafael Lajara Vizcaíno, 2011)

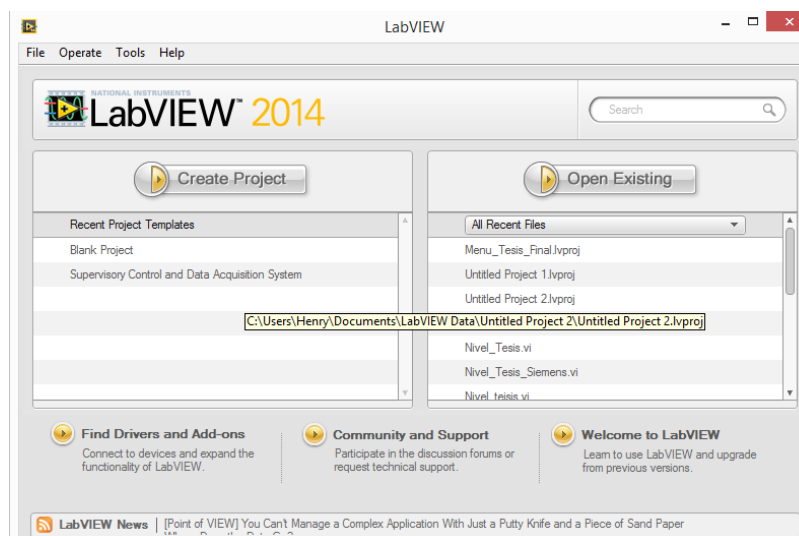


Figura 6: Software Labview

Fuente: (Labview , 2014)

El lenguaje que usa se llama lenguaje G, donde la G simboliza que es lenguaje Gráfico. Este lenguaje es de alto nivel, que cuenta con todas las estructuras, puede ser utilizado para desarrollar cualquier algoritmo. Los programas VI constan de una interfaz interactiva de usuario y un diagrama de flujo que hace las funciones de código fuente.

1.6.1.1. Principales Componentes de Labview

a. Instrumentos Virtuales (VI)

Todos los programas desarrollados en Labview se denominan instrumentos virtuales o VI, su apariencia y funcionamiento imita a los instrumentos físicos tales como osciloscopios y multímetros. Este software posee un número considerable de herramientas para adquisición, análisis, despliegue y almacenamiento de datos, incluido herramientas que ayudan a la resolución de su código de ejecución.

b. Panel Frontal de Labview

Cada VI consta de dos partes diferenciadas, una de estas es el Panel Frontal (**Figura 7**) en donde el usuario construye una interface mediante el uso de controles e indicadores. Los controles son texto, botones de acción, perrillas y otros dispositivos de entrada. Los indicadores son gráficas, Leds y otros que son utilizados como salidas.

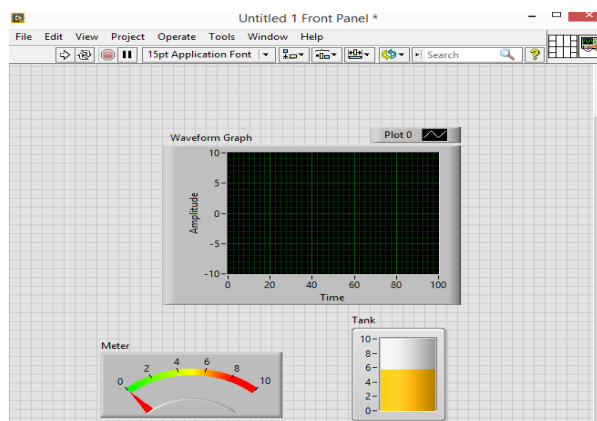


Figura 7: Panel Frontal Labview

Fuente: (Labview , 2014)

c. Diagrama de Bloques de Labview

El diagrama de bloques contiene el código fuente gráfico, así como también las funciones y estructuras incorporadas en las bibliotecas de Labview. Los objetos del panel frontal aparecen como terminales en el

diagrama de bloques es decir como elementos que sirven como entradas o salidas de datos. (Antonio Ruiz Canales, 2010)

En la siguiente (**Figura 8**) se muestra el Diagrama de Bloques de Labview:

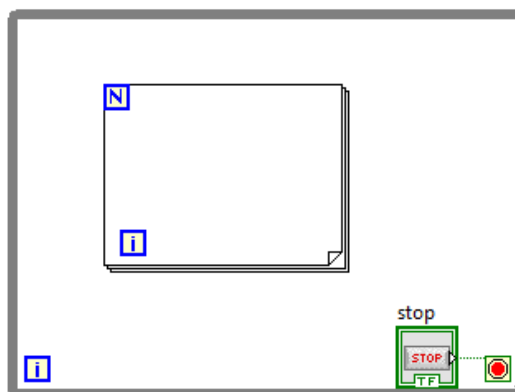


Figura 8: Diagrama de Bloques

Fuente: (Labview , 2014)

1.6.2. Módulo DSC de Labview

El Módulo DSC (Datalogging and Supervisory Control) (**Figura 9**) de Labview permite agregar beneficios importantes a la programación gráfica, gracias a este módulo es posible desarrollar aplicaciones SCADA (Supervisory Control and Data Acquisition) o registrar gran cantidad de datos de canales. Además, se puede registrar datos a bases de datos, administrar alarmas, crear interfaces humano-máquina (HMI) entre otras posibilidades.

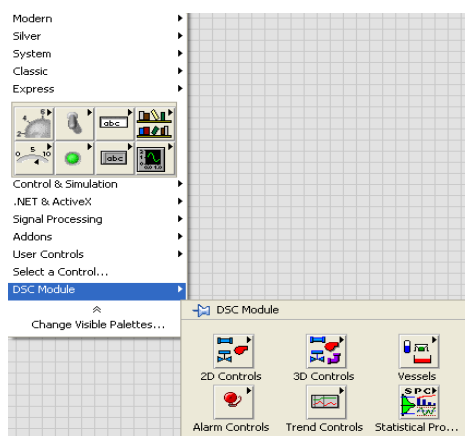


Figura 9: Módulo DSC Labview

Fuente: (National Instruments, 2014)

Las principales características del Módulo Labview DSC son las siguientes:

a. Permite Conectar Labview a Sistemas Industriales Existentes de Gran Cantidad de Canales

El Módulo Labview DSC es capaz de soportar protocolos industriales usados comúnmente, incluyendo OLE (Object Linking and Embedding) para control de procesos (OPC). Con esta flexibilidad, se puede integrar fácilmente Labview a sistemas SCADA/HMI existentes.

b. Registro de Datos a una Base de Datos Histórica, Establecer Alarmas y Administrar Eventos

Mediante el Módulo DSC de Labview, se puede registrar datos fácilmente a la base de datos históricos o registrar a su base de datos empresarial. Además, es posible usar estas mismas herramientas basadas en configuración para establecer el registro de alarmas y la notificación de datos. (National Instruments, 2014)

1.6.3. Módulo Control Design and Simulation de Labview

El módulo Control Design and Simulation (**Figura 10**) permite diseñar plantas y modelos de control usando función de transferencia, espacio de estado o cero-polos, también se puede analizar el comportamiento de modelos de lazo abierto, diseñar controladores de lazo cerrado, simular sistemas en y fuera de línea y realizar implementaciones físicas.

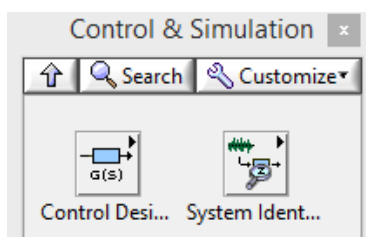


Figura 10: Toolkit Control Design and Simulation

Fuente: (National Instruments, 2014)

Otras características destacables que se pueden realizar al utilizar el módulo son las siguientes:

- Analizar el rendimiento del sistema con herramientas como respuesta de pasos, mapas de polo cero y gráficas de Bode.
- Desplegar sistemas dinámicos a hardware en tiempo real usando funciones integradas y el Módulo Labview Real-Time. (National Instruments, 2014)

1.7. Software de programación y simulación para PLC Allen Bradley

1.7.1. RSLogix Emulate 500

RSLogix Emulate 500 es un paquete de software que permite emular una o más estaciones de procesadores PLC-5 o SLC 500 respectivamente. Aquí se determina con cual PLC-5/SLC 500 se desea ejecutar el programa de escalera y RSLogix Emulate escanea la lógica escalera como un procesador real. Este software es utilizado para probar y depurar programas de lógica de escalera antes de la puesta en marcha y arranque. En la siguiente figura se muestra la pantalla principal de RSLogix Emulate 500 (**Figura 11**).

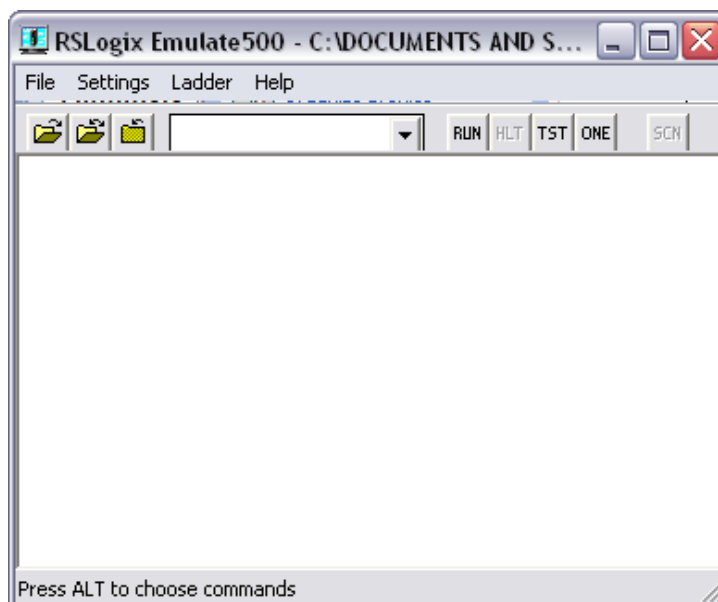


Figura 11: RSLogix Emulate 500

Fuente: (Rockwell Automation, 2004)

a. Características RSLogix Emulate 500

- Desde RSLogix Emulate no se conecta a los módulos de E/S, la emulación de E/S se maneja a través de archivos de depuración. Archivos de depuración son archivos especiales de lógica de escalera que se crea para simular las entradas. A continuación, los archivos se filtran automáticamente cuando se descarga a su procesador real.
- Cuando se prueba el programa, a menudo es útil detener el proceso cuando una determinada dirección sale del rango. RSLogix Emulate, puede detener el proceso cada vez que un renglón seleccionado se hace verdadero, congelando efectivamente el proceso en el instante en que ocurrió el error.
- RSLogix Emulate tiene muchos métodos para el escaneo de su lógica de escalera. Puede analizar su lógica de escalera continuamente, un programa de exploración en un tiempo o peldaño por peldaño. También puede seleccionar un bloque específico de peldaños de emular. (Rockwell Automation, 2004)

1.7.2. RSLogix 500

RSLogix 500 es un paquete de programación de lógica de escalera para los procesadores SLC 500 y Micrologix. RSLogix 500 es compatible con dichos procesadores cuyos programas son creados con los paquetes de programación de Rockwell Automation. (Rockwell Automation, 2015)

a. Características

RSLogix 500 presenta las siguientes características (**Tablas 2**):

Tabla 2

Características de RSLogix 500

Principales Características de RSLogix 500	
Incluye un editor de escalera.	Posee un asistente de dirección
Posee un verificador de proyecto para construir una lista de errores.	Los informes de tendencias y de histograma para el control.
Un monitor de datos personalizada	Posee bibliotecas para almacenar y recuperar partes de la lógica de escalera
Permite buscar y reemplazar las apariciones de una dirección.	Una comparación de la utilidad que le permite ver las diferencias de proyectos
La edición de arrastrar y soltar elementos de una tabla de datos de un archivo a otro.	Una interfaz de apuntar y hacer clic para acceder a todas las carpetas y archivos del proyecto.

Fuente: (Rockwell Automation, 2015)

b. Componentes Pantalla Principal

La pantalla principal del programa (**Figura 12**) consta de las siguientes partes:

1. Barra de menús: Las selecciones de menú.
 2. Barra de iconos: La barra de iconos muestra las funciones más comunes.
 3. Barra Online: Indica el modo de procesador y si las modificaciones o forzados están presentes en línea.
 4. Árbol del proyecto: Carpetas y los archivos del proyecto.
 5. Barra de estado: Indica información de estado y mensajes del sistema
 6. Barra de herramientas de instrucciones: Muestra nemotécnica de las operaciones de las categorías con pestañas.
 7. Vista de escalera: Aquí es donde se edita la lógica de escalera.
- (Rockwell Automation, 2015)

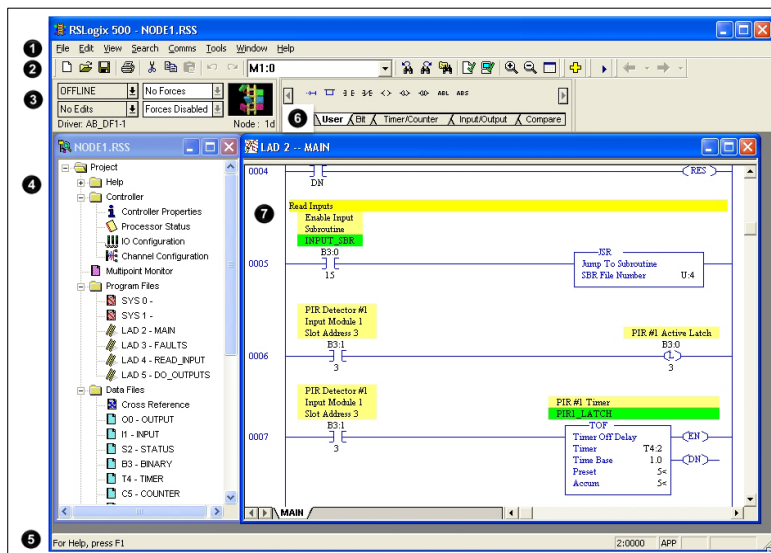


Figura 12: RSLogix 500

Fuente: (Rockwell Automation, 2015)

1.7.3. RSLogix Emulate 5000

RSLogix Emulate 5000 es un software que permite emular el comportamiento de los controladores Logix5000. También permite experimentar y depurar programas en un entorno seguro y controlado sin invertir en controladores físicos y módulos de E/S. Además, **(Figura 13)** permite poner a prueba sus aplicaciones HMI sin necesidad de usar un controlador real. (Rockwell Automation, 2010)

Mediante el emulador se puede programar y acceder tanto en el equipo donde está instalado y de forma remota a través RSLinx Classic.

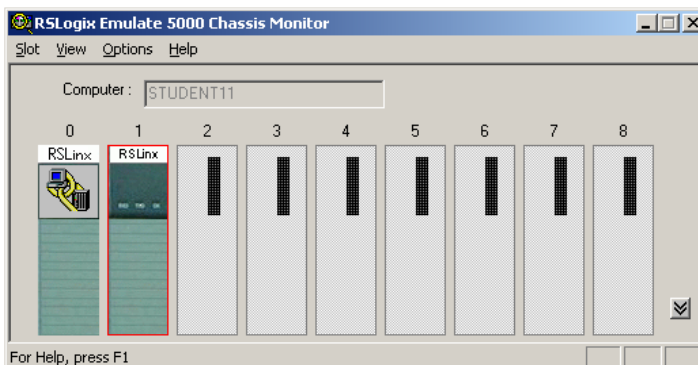


Figura 13: RSLogix Emulate 5000

Fuente: (Rockwell Automation, 2010)

a. Componentes de RSLogix Emulate 5000

Hay dos componentes de RSLogix Emular 5000:

- El Chasis Monitor es una aplicación de software que permite configurar emulación y módulos de E/S simuladas.
- Módulos de emulación son los servicios de Windows que simulan el comportamiento de controladores Logix5000 y ciertos módulos de E/S, son aplicaciones que se ejecutan sin las interfaces propias (**Figura 14**). El monitor de chasis sirve como una interfaz para la emulación de módulos, lo que le permite crear y configurar las instancias de los módulos. (Rockwell Automation, 2010)

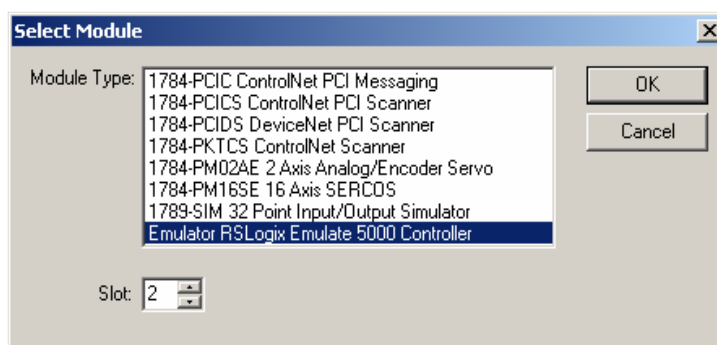


Figura 14: Módulos de Emulación

Fuente: (Rockwell Automation, 2010)

1.7.4. RSLogix 5000

El software RSLogix 5000 Enterprise Series (**Figura 15**) permite trabajar con las plataformas CompactLogix de Rockwell Automation. Es un programa que cumple con la normativa IEC 61131-3. Posee editores de lógica de escalera, diagramas de bloques de funciones y diagramas de función secuencial para el desarrollo de programas de aplicación. (Bastidas Roberto & Proaño Luis, 2010)

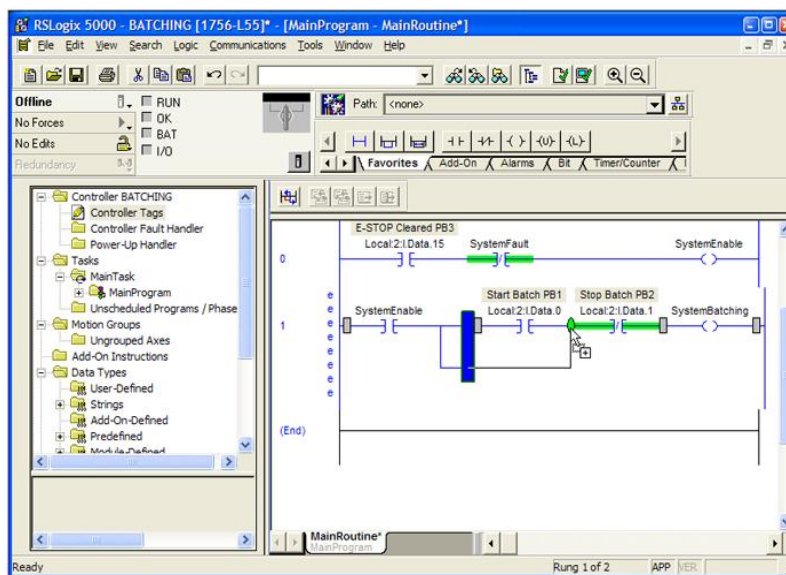


Figura 15: RSLogix 5000

Fuente: (Bastidas Roberto& Proaño Luis, 2010)

Con el software RSLogix 5000 Enterprise Series, sólo se necesita un paquete de software para programación de control secuencial, de procesos, variadores, de movimiento y de seguridad en la (Tabla 3) se muestra las principales características de RSLogix 5000.

Tabla 3

Características de RSLogix 5000

Características de RSLogix 5000	
Importación de rutinas, fases y programas.	Instrucciones de control de proceso avanzado.
Informe del organizador del controlador	Permite realizar actualizaciones de firmware de módulo
Ideal para aplicaciones de base directa, movimiento y seguridad.	Es compatible con (PAC) Logix.
Permite fragmentar la aplicación en programas más pequeños.	Permite realizar depuración y el mantenimiento de la aplicación.
Permite desarrollar la aplicación sin configurar la memoria.	Permite crear datos propios del usuario.

Fuente: (Bastidas Roberto& Proaño Luis, 2010)

1.8. Software de Programación para PLC Siemens

1.8.1. TIA PORTAL

TIA Portal (**Figura 16**) es un paquete de software avanzado que permite configurar de forma intuitiva y eficiente todos los procesos de planificación y producción. Contiene una generación de editores de programación más productivos que optimizan la calidad, la eficiencia de todo el proceso de producción. (Siemens AG, 2013)

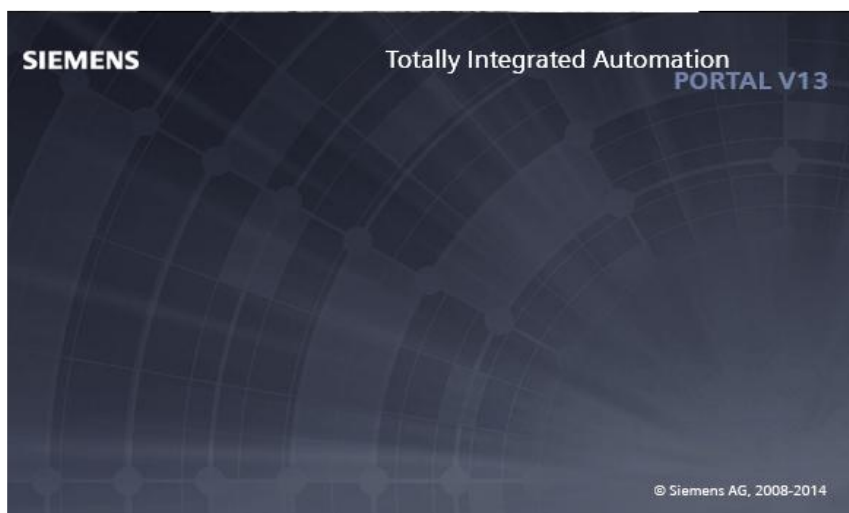


Figura 16: Software TIA PORTAL

Fuente: (Siemens AG, 2013)

1.8.1.1. Configuración de redes y dispositivos

Para diferenciar entre las tareas de conexión y de configuración de dispositivos, el editor posee tres vistas:

- Las conexiones entre dispositivos pueden crearse de modo gráfico en la Vista de redes.
- Los distintos dispositivos se parametrizan y configuran en la Vista de dispositivos.
- En la Vista de topología se muestra la interconexión real de los dispositivos PROFINET. (Siemens AG, 2013)

a. Vista de Red

La vista de red (**Figura 17**) permite configurar la comunicación en la instalación. Aquí se proyectan de forma gráfica e ilustrativa los enlaces de comunicación entre las distintas estaciones.

- Se puede observar todos los participantes en la red y los componentes de red.
- Proyección gráfica de las estaciones.
- Es posible utilizar varios controladores, periféricos, dispositivos HMI, estaciones SCADA, estaciones de PC y accionamientos en un solo proyecto. (Siemens AG, 2013)

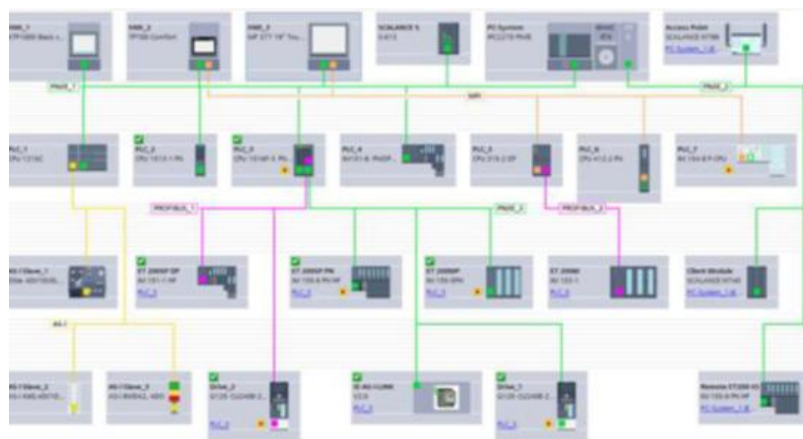


Figura 17: Vista de Red

Fuente: (Siemens AG, 2013)

b. Vista de Dispositivo

En la vista del dispositivo se realiza la configuración de racks, la declaración de direcciones etc. Todos los dispositivos están representados como se ve en la (**Figura 18**). (Siemens AG, 2013)



Figura 18: Vista de Dispositivo

Fuente: (Siemens AG, 2013)

c. Vista Topológica

Aquí se representa de forma gráfica a los controladores y los periféricos descentralizados que le están asignados (**Figura 19**). (Siemens AG, 2013)

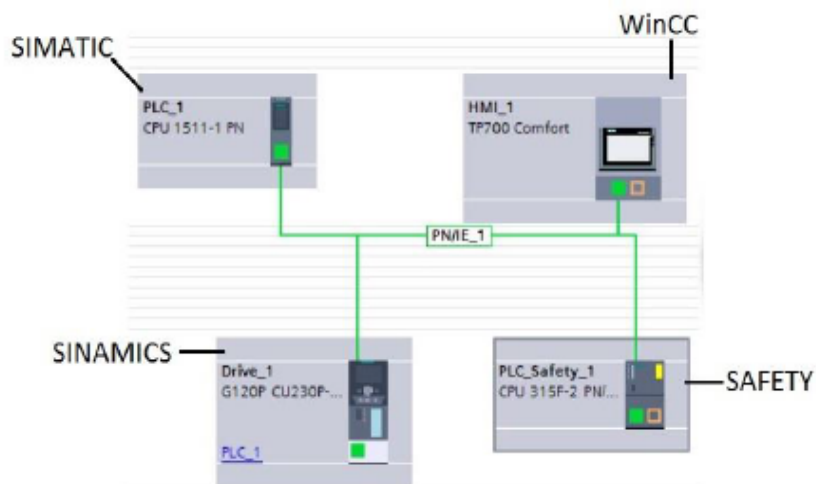


Figura 19: Vista Topológica

Fuente: (Siemens AG, 2013)

1.8.1.2. Lenguajes de Programación

SIMATIC STEP 7 contiene compiladores y editores de programación avanzados para los controladores SIMATIC S7.

- Texto estructurado (SCL), lenguaje ladder (KOP) y diagrama de funciones (FUP).
- Lista de instrucciones (AWL) y programación secuencial (GRAPH, SFC) (Siemens AG, 2013)

El lenguaje de programación que se utilizará para este proyecto será el lenguaje de escalera (KOP) el mismo que se describe a continuación:

a. KOP y FUP: Lenguajes gráficos de programación

Los editores de KOP (**Figura 20**) y FUP totalmente gráficos ofrecen una excelente claridad funcional y una rápida navegación.

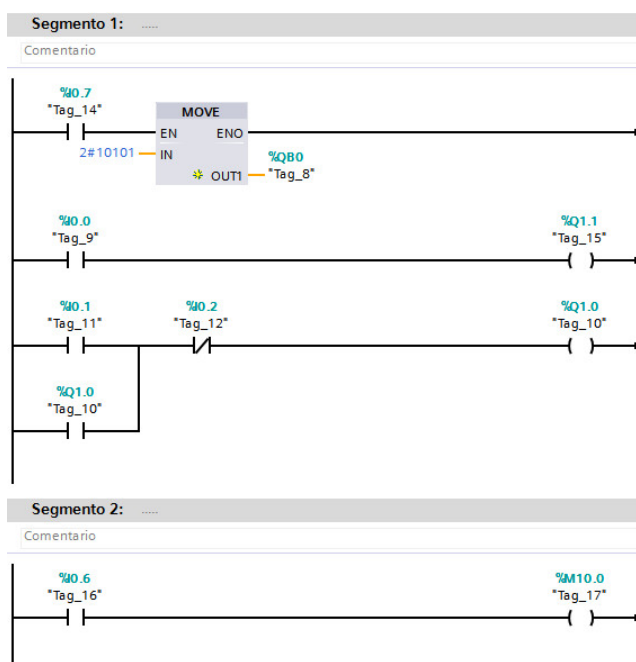


Figura 20: Lenguaje KOP

Fuente: (Siemens AG, 2013)

1.8.1.3. Modo Online

TIA PORTAL posee varias funciones online. Con un solo clic online: indicación del estado de funcionamiento, claro resumen del diagnóstico mediante esquema del proyecto y comparación online/offline a nivel de los componentes (**Figura 21**):

- Rápida transmisión de todos los cambios en el programa con una descarga continua.
- Comparación de proyectos offline/offline.
- Descarga en RUN.
- Subida de los datos de configuración del HW de los controladores existentes - HW Detect.
- Subida completa del software de proyecto a un PG vacío.
- Rápido servicio sin un proyecto existente (con HW Detect y SW Upload). (Siemens AG, 2013)

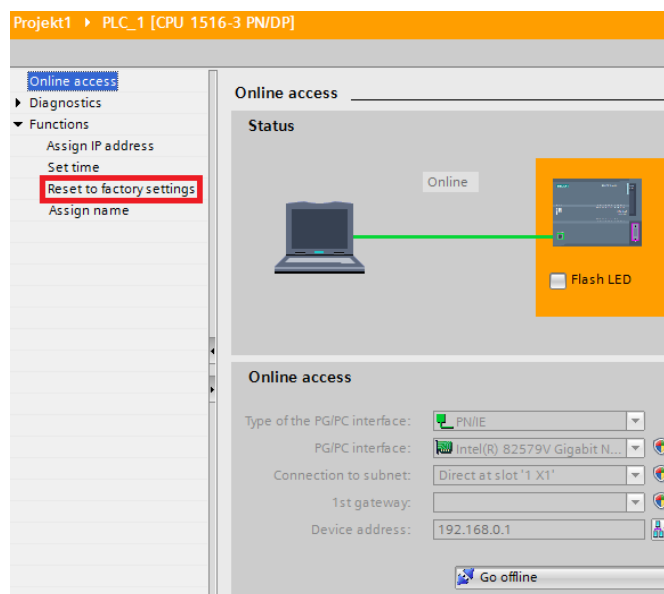


Figura 21: Modo Online

Fuente: (Siemens AG, 2013)

1.9. OPC

Es un paquete de software para el enlace eficaz de los componentes de automatización con los dispositivos de campo y el hardware de control. Es una

especificación técnica no propietaria que define un conjunto de interfaces estándar en la tecnología OLE/COM de Microsoft. Posibilita la interoperabilidad entre aplicaciones de automatización/control, dispositivos/sistemas de campo y aplicaciones de gestión oficina. (Moreno Valery, 2011)

1.9.1. Historia

- OPC: OLE for Process Control ha sido creado con la colaboración de suministradores líderes mundialmente en automatización, hardware y software en cooperación con Microsoft. La Fundación OPC que gestiona este estándar tiene 150 miembros en el mundo y la gestiona: Fisher-Rosemount, Rockwell Software, Opto 22, Intellution e Intuitive Technology. (Moreno Valery, 2011)

1.9.2. Beneficios de OPC

- Gran reducción de costos de todo tipo: desarrollo, mantenimiento, actualización, aprendizaje. Eliminando particularidades como lo hace OPC se garantiza reducir el costo del sistema de automatización completo.
- Con la introducción de los productos de automatización basados en OPC el usuario tiene libertad total de selección del sistema, teniendo en cuenta los mejores componentes de cada clase, sin tener que evaluar a los sistemas específicos de cada fabricante integralmente. Esto impulsará el desarrollo de otros sectores dentro del mercado de SCADA's.
- Independencia del Suministrador o liberación de compromiso propietario. Este es un beneficio sumamente importante, pues es uno de los mayores componentes de costos de los SCADA's. Sin OPC el cliente está a merced del suministrador para cualquier actualización de un componente del sistema, normalmente pagando por actualizaciones completas. Con OPC sólo el módulo de interés se actualiza y no el sistema entero.

- En el bajo nivel implica ahorro de tiempo y dinero: ahora no es necesario escoger dispositivos con drivers disponibles en el SCADA, sólo hay que buscar el OPC.
- Mayor mercado a través de conectividad e interoperabilidad aumentada.
- Centralización en actividades de valor añadido (HMI, SE, etc.). (Moreno Valery, 2011)

1.10. Servidores OPC

1.10.1. RSLinx Classic

RSLinx Classic (**Figura 22**) es una herramienta ideal para comunicaciones industriales. Permite que el controlador programable Allen-Bradley acceda a varias aplicaciones de Rockwell Software. Entre estas aplicaciones destacan RSLogix, RSNetWork, RSVIEW32. Además, utiliza técnicas avanzadas de optimización de datos. RSLinx Classic es un servidor compatible con OPC Data Access y un servidor DDE. (Rockwell Automation, 2008)

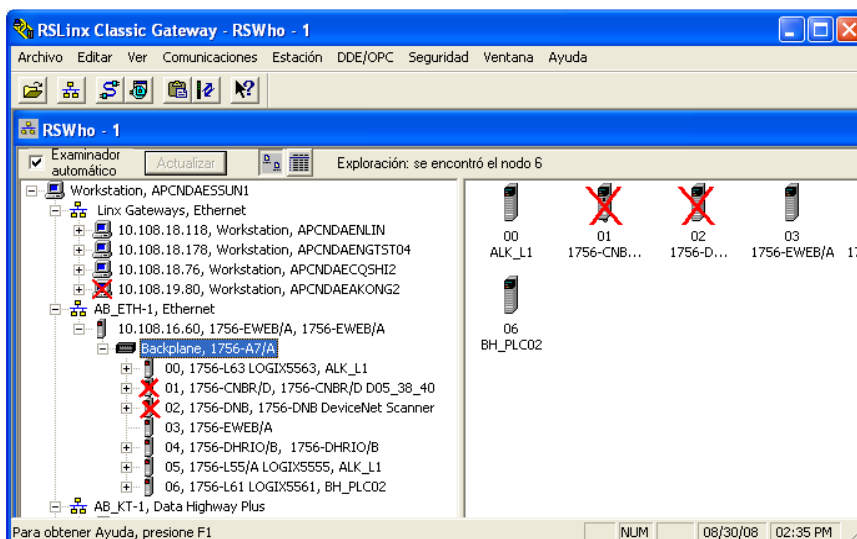


Figura 22: RSLinx Classic

Fuente: (Rockwell Automation, 2008)

RSLinx Classic permite conectar a redes TCP/IP haciendo que las comunicaciones basadas en RSLinx Classic, este software proporciona conectividad remota con:

- Mediante RSView32 se puede acceder a datos por medio de un RSLinx Classic.
- Aplicaciones Microsoft Office remotas que muestran datos de la planta como, por ejemplo, Excel.
- Una página Web que indica una serie de datos de la planta cuando el servidor Web y RSLinx Classic se encuentran en equipos distintos. (Rockwell Automation, 2008).

1.10.2. KEPServerEx

KEPServerEx (**Figura 23**) es un OPC desarrollado por la empresa Kepware Technologies, es la plataforma de comunicaciones líder en la industria que proporciona una única fuente de datos de automatización industrial para todas las aplicaciones. El diseño de la plataforma permite a los usuarios conectar, administrar, supervisar y controlar los dispositivos de automatización y aplicaciones de software diversa a través de una interfaz de usuario intuitiva. (Kepware Technologies, 2016)

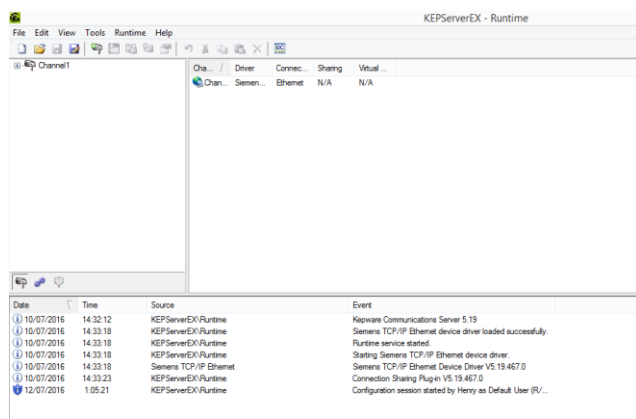


Figura 23: OPC KEPServerEx

Fuente: (Kepware Technologies, 2016)

1.11. PLC SIEMENS S7-1200

1.11.1. Introducción al PLC S7-1200

El PLC Siemens S7-1200 (**Figura 24**) brinda gran flexibilidad y capacidad de controlar una gran cantidad de dispositivos para las distintas tareas de

automatización. La CPU incorpora un microprocesador, una fuente de alimentación, así como circuitos de entrada y salida. (Siemens AG, 2009)



Figura 24: PLC Siemens S7-1200

Fuente: (Siemens AG, 2009)

1.11.2. Componentes del PLC Siemens S7-1200

La CPU incorpora un puerto PROFINET para la comunicación en una red PROFINET (**Figura 25**). Los módulos de comunicación están disponibles para la comunicación en redes RS485 o RS232. (Siemens AG, 2009)

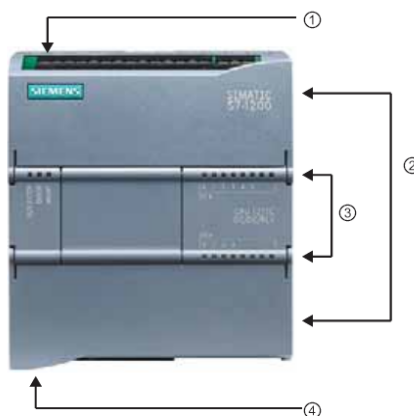


Figura 25: Componentes del PLC S7-1200

Fuente: (Siemens AG, 2009)

El S7-1200 posee las siguientes partes principales (**Tabla 4**):

Tabla 4**Partes del PLC Siemens S7-1200**

Ítem	Parte del PLC
1	Conector de corriente
2	Conectores para cable de alimentación
2	Espacio para Memory Card
3	LED's que indica estado de entradas y salidas
4	Conector PROFINET

Fuente: (Siemens AG, 2009)

1.11.3. Características del PLC Siemens S7-1200

- Montaje, programación y uso fáciles.
- De alta escala de integración, requiere poco espacio, potente.
- Adecuado para aplicaciones de automatización pequeñas y medias.
- Aplicable tanto para los controles más simples como también para tareas complejas de automatización.
- Aplicable aislado, interconectado en red o en configuraciones descentralizadas.
- El controlador es apto también para campos donde, por motivos económicos, no se aplicaban hasta ahora autómatas programables.
- Con excepcional capacidad de tiempo. (Siemens AG, 2013)

1.12. PLC Allen Bradley CompactLogix 1769-L32E**1.12.1. Introducción**

Los dispositivos CompactLogix 1769 (**Figura 26**) son controladores utilizados para desarrollar aplicaciones de control. Estos dispositivos poseen Ethernet/IP o ControlNet integrados y comunicaciones DeviceNet modulares. (Rockwell Automation, 2016)



Figura 26: PLC Allen Bradley CompactLogix

Fuente: (Rockwell Automation, 2016)

1.12.2. Características

- Incluye un puerto serial RS-232 incorporado en cada controlador 1769-L3x.
- Ofrece puertos de comunicación incorporados para redes ethernet/IP o ControlNet.
- Incluye un módulo de interface de comunicación 1769-SDN para control de E/S y configuración de dispositivos remotos en DeviceNet.
- Proporciona una conexión en puente transparente con control y recolección de datos a través de la misma red.
- Se integra con módulos Compact I/O 1769
- Ofrece opciones flexibles de memoria de usuario
- No requiere un chasis (Rockwell Automation, 2016)

1.12.3. Partes del PLC CompactLogix 1769-L32E

a. Puerto Ethernet

El PLC posee un puerto Ethernet para realizar la programación y monitoreo, para asignar una dirección IP al puerto Ethernet se utiliza la aplicación BOOTP-DHCP para que el controlador se pueda comunicar con la computadora a través de una RED Ethernet. (Navas Alex& Zambrano Jorge, 2015)

b. Puerto Serial RS232

El controlador posee un puerto serial RS232 macho de 9 pines denominado “Canal 0”, el cual es utilizado para el monitoreo y control de procesos. (Navas Alex& Zambrano Jorge, 2015)

c. Módulos de entrada/salida CompactLogix

Los Módulos de E/S Compact 1769 pueden ser usados como E/S locales y distribuidas para un controlador COMPACTLOGIX a continuación se muestra los diferentes módulos que posee este PLC.

- Módulos de E/S Digitales CompactLogix.
- Módulos de entrada digital AC.
- Módulos de salida digital AC.
- Módulos de entrada digital DC.
- Módulos de salida digital DC.
- Módulos de salida de contacto digital.
- Módulos de E/S Analógicos CompactLogix.
- Los módulos analógicos son de termopares o RTD. (Navas Alex& Zambrano Jorge, 2015)

El CompactLogix posee las siguientes partes como se muestra en la (Figura 27).

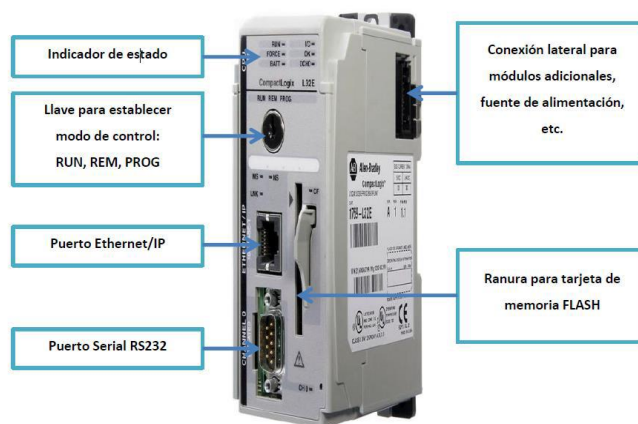


Figura 27: Partes de un PLC CompactLogix

Fuente: (Navas Alex& Zambrano Jorge, 2015)

1.13. PLC Allen Bradley Micrologix 1200

1.13.1. Descripción

Los dispositivos Micrologix 1200 (**Figura 28**) permiten desarrollar una amplia variedad de aplicaciones. Se encuentran disponibles en versiones de 24 y 40 puntos. Las características más importantes de estos dispositivos son flexibilidad de aplicación, la capacidad de expansión a menor inversión y un inventario reducido de componentes. (Rockwell Automation, 2016)



Figura 28: PLC Allen Bradley Micrologix

Fuente: (Rockwell Automation, 2016)

1.13.2. Características y ventajas clave

- Cuatro entradas de enclavamiento o captura de impulsos. Las entradas de enclavamiento permiten al controlador capturar y retener señales para el procesamiento de entradas.
- Contador de alta velocidad incorporado utiliza números enteros de 32 bits para rango extendido, cuenta con 8 modos de operación y acepta control directo de salidas independiente del escáner del programa.
- Función de interruptor de final de carrera programable. Esta función le permite configurar el contador de alta velocidad para funcionar como interruptor de final de carrera programable.
- Potenciómetros de ajuste: Dos potenciómetros de ajuste analógicos de 3/4 de vuelta con una salida digital permiten un ajuste rápido y fácil de temporizadores, contadores, puntos de ajuste y mucho más. (Rockwell Automation, 2016)

CAPITULO II

DISEÑO E IMPLEMENTACIÓN DEL SIMULADOR

2.1. Introducción

El desarrollo de este capítulo consta del diseño e implementación del simulador de procesos industriales, en el cual en primer lugar se desarrolla la interfaz principal del simulador, los HMI's de los menús secundarios, las interfaces de los subVI de los procesos (Silo, Garaje, Mezclador, Semáforo, Nivel, Flujo). En segundo lugar, se procede a realizar la programación de cada uno de los procesos a simular donde se describe etapa a etapa como fueron desarrollados. Así como también se muestra como se debe realizar la correcta configuración y programación de los controladores lógicos programables (PLC's) que son compatibles con este simulador como son el Siemens S7-1200, Micrologix y CompactLogix.

2.2. Requerimientos del Proyecto

Para desarrollar la programación del simulador de procesos industriales es necesario el uso del Software Labview el cual permite realizar programas de gran calidad debido a las prestaciones que ofrece este programa, aquí se va a crear todas las simulaciones que formaran parte del simulador de procesos industriales, para realizar los HMI de cada proceso es necesario el uso del módulo DSC de Labview el cual contiene imágenes ideales para este proyecto como son válvulas, semáforos, indicadores, silo, puertas, tuberías, tanques, bombas entre otros. Para el desarrollo de los procesos Nivel y Flujo es necesario que se encuentre instalado el módulo PID y Real-Time de Labview, para la programación de los PLC compatibles con el simulador se utiliza TIA PORTAL para el S7-1200, RSLogix 500 para el Micrologix, RSLogix 5000 para los CompactLogix, para realizar la conexión entre el simulador y cada uno del software de programación de PLC se utiliza el OPC KEPServerEx en el caso de Siemens y RSLinx para PLC de tecnología Allen Bradley.

2.3. Diseño e Implementación del Simulador de Procesos Industriales

La implementación del simulador se divide en dos etapas, en la primera se realiza el diseño del simulador, el cual se lo realiza completamente en el software Labview, mediante el uso del módulo DSC, se desarrolla todas las interfaces gráficas del simulador, cada proceso está compuesto por diferentes indicadores y controles que serán programados de tal manera que a la hora de realizar las pruebas de funcionamiento del simulador estas sean fáciles y entendible para el usuario. A continuación, se indica un esquema de cómo se desarrolla el simulador de Procesos Industriales (**Figura 29**).

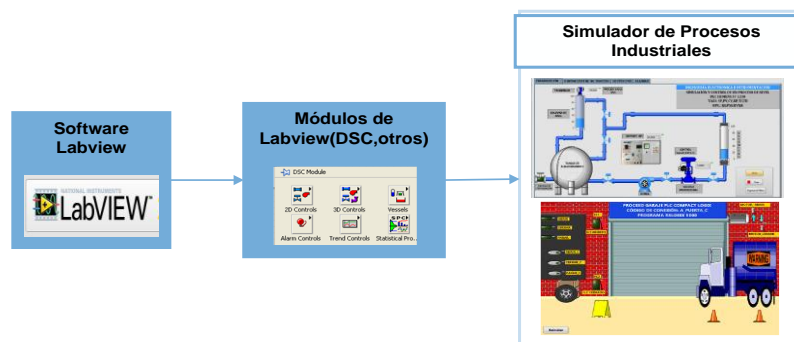


Figura 29: Diseño del Simulador de Procesos Industriales

En la segunda etapa se realiza la programación y configuración de los PLC que son compatibles con el simulador, además se configura los servidores OPC para que exista comunicación entre el simulador y el software que se utiliza para configurar cada uno de los controladores. A continuación, se indica un esquema de cómo se desarrolla esta etapa. (**Figura 30**)

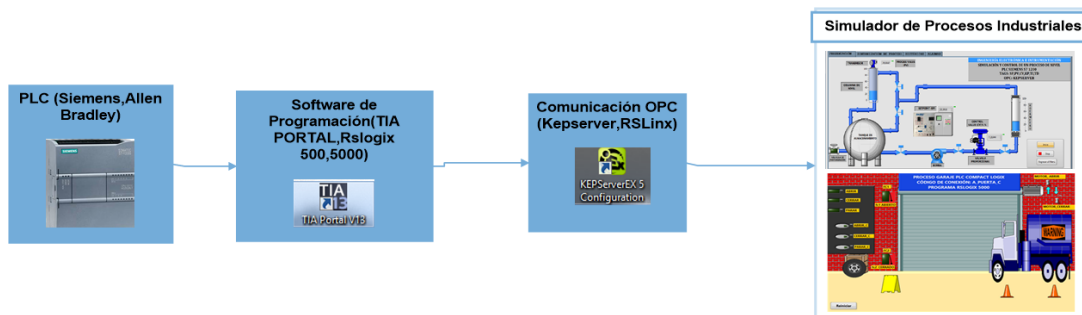


Figura 30: Implementación de Simulador de Procesos Industriales

2.4. Diseño del HMI Principal del Simulador

En el siguiente apartado se detalla todo el desarrollo del HMI del Simulador de Procesos Industriales, el mismo está conformado por un menú principal, tres submenús secundarios y cada uno de estos poseen 6 subVI respectivamente.

2.4.1. Diagrama de Flujo del HMI del Simulador

En la siguiente (**Figura 31**), se muestra el diagrama de flujo del HMI del simulador.

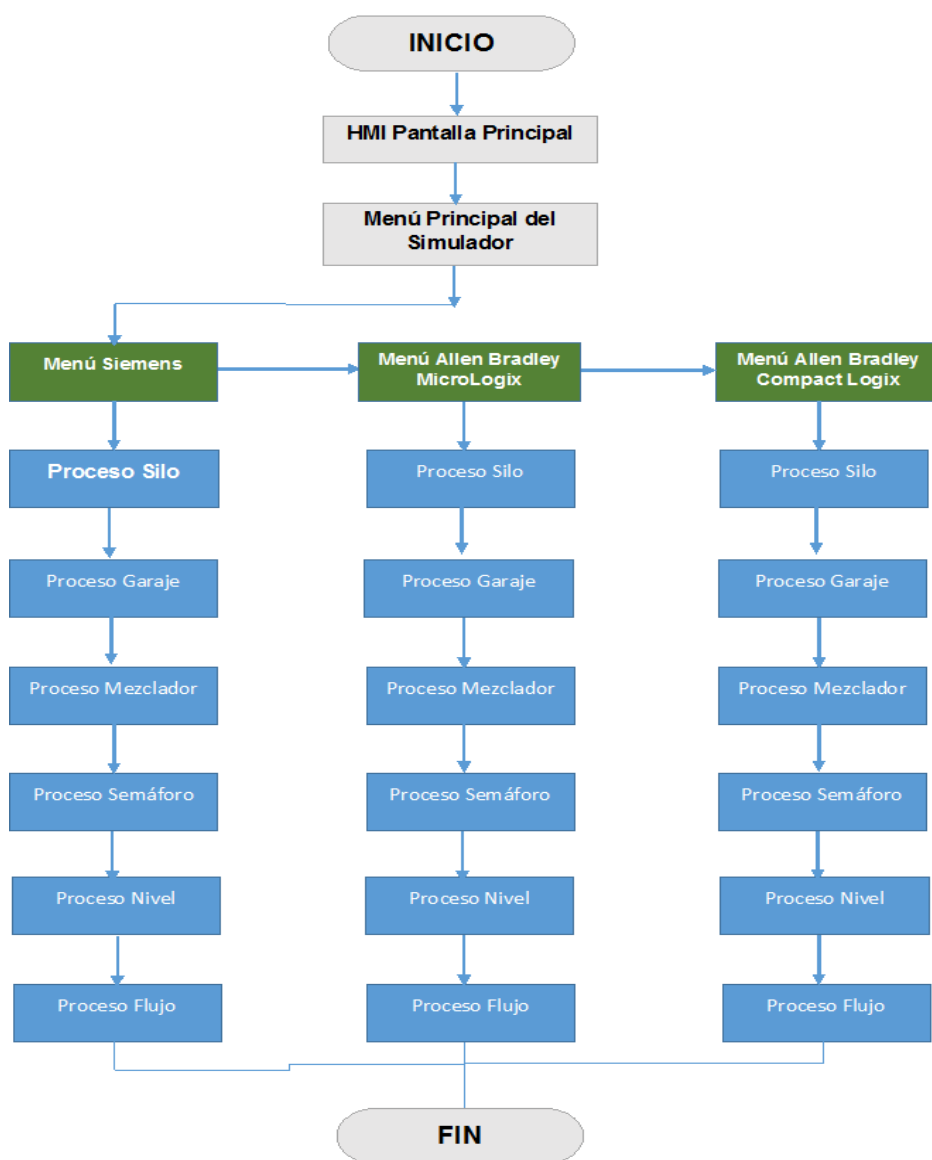


Figura 31: Diagrama de Flujo del HMI del Simulador

2.4.2. Pantalla Principal

El presente HMI (**Figura 32**) está compuesto por la presentación del proyecto en donde se observa el tema del proyecto, autores, director, fecha, varias imágenes y finalmente un botón llamado “INICIO SIMULADOR” que permite el acceso al menú principal del Simulador.



Figura 32: Pantalla Principal del Simulador

2.4.3. Menú Principal del Simulador

a. Panel Frontal

El diseño del menú principal comprende el uso de imágenes de los PLC que van a soportar el uso de este simulador como son (**Tabla 5**):

Tabla 5

Tecnologías de PLC compatibles con el simulador

Tecnología	Equipo
Allen Bradley	Micrologix
Allen Bradley	Compact Logix
Siemens	S7-1200

Este menú principal (**Figura 33**) presenta tres menús secundarios “PLC Micrologix”, “PLC CompactLogix” y finalmente “PLC Siemens 1200” los cuales poseen un botón de acceso a cada uno de ellos. Además, existe el botón “regresar” que permite volver a la pantalla principal del simulador.



Figura 33: Menú Principal del Simulador.

b. Diagrama de Bloques

El diagrama de bloques del menú principal del simulador presenta las siguientes etapas de programación. (**Figura 34**).

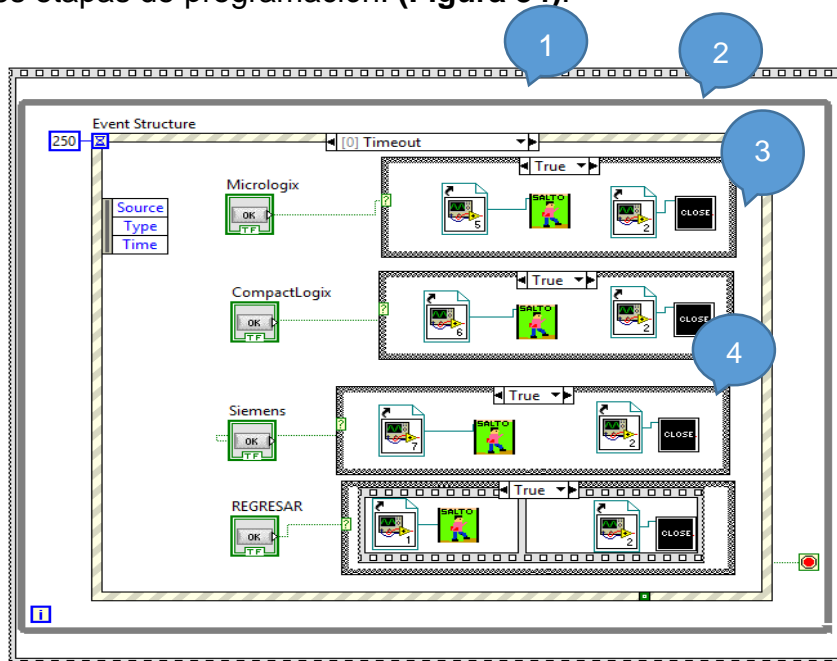


Figura 34: Diagrama de bloques del Menú Principal

c. Programación del Menú Principal

Cada etapa del diagrama de bloques cumple una función específica las mismas que se describen a continuación:

1. Función **“Flat Sequence Structure”** en donde se realiza toda la secuencia que se encuentra en su interior.
2. Función **“While Loop”** donde se desarrolla toda la programación.
3. Función **“Event Structure”** para realizar varios eventos. En el segundo evento de la función **“Event Structure” (Figura 35)** se programa una constante verdadera la cual desactiva el lazo While Loop.

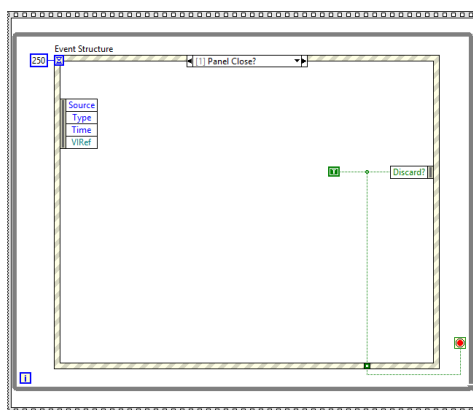


Figura 35: Segundo evento de la función "Event Structure".

En el tercer caso de **“Event Structure” (Figura 36)** se programa la acción de un botón STOP para detener toda acción del **“While Loop”** en cualquier instante de la ejecución del programa.

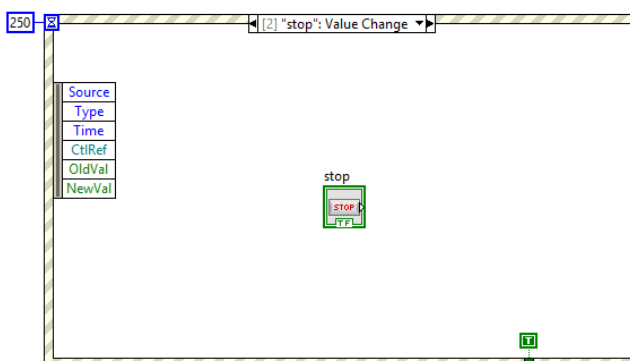


Figura 36: Tercer evento de la función "Event Structure".

4. Todos los botones del menú van conectados a su respectivo **“Case Structure”** el caso de seleccionar un botón se abrirá el SubVI descrito mediante el SubVI “salto.vi” y se cerrará el SubVI anterior mediante el SubVI “cerrar.vi” respectivamente cuando el botón este activado.

Nota: El desarrollo y explicación de los SubVI “salto.vi” y “cerrar.vi” se presenta en el apartado 2.11.1 y 2.11.2 respectivamente.

2.4.4. Menús secundarios del Simulador

El simulador posee tres menús secundarios cada uno de ellos perteneciente a un único tipo de PLC, en tecnología Allen Bradley se encuentra el Micrologix y CompactLogix y en Siemens el S7-1200.

2.4.4.1. Submenú Tecnología Siemens

El submenú SIEMENS permite el acceso a 6 subVI, cuatro simulaciones de procesos digitales y dos analógicos. Para ingresar se da clic en el botón “PLC Siemens 1200”, el cual está programado a través del subVI “salto.vi”, se abre el panel frontal del submenú Siemens y se cierra el Menú Principal mediante el subVI “cerrar.vi”. **(Figura 37)**

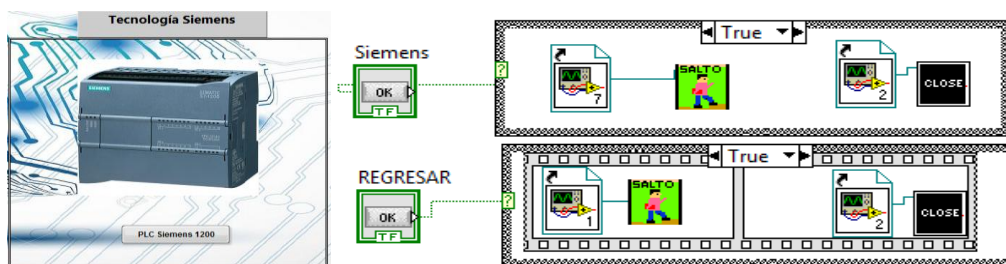


Figura 37: Acceso al Menú Siemens.

a. Panel Frontal

El diseño de la interfaz de este submenú Siemens esta formado de una imagen del PLC Siemens S7-1200, 6 imágenes que ofrecen una clara referencia de cada uno de los procesos que posee este simulador, además un botón de acceso a los subVI (Silo, Garaje, Semáforo, Mezclador, Nivel, Flujo) y un botón para regresar al Menu principal **(Figura 38)**.

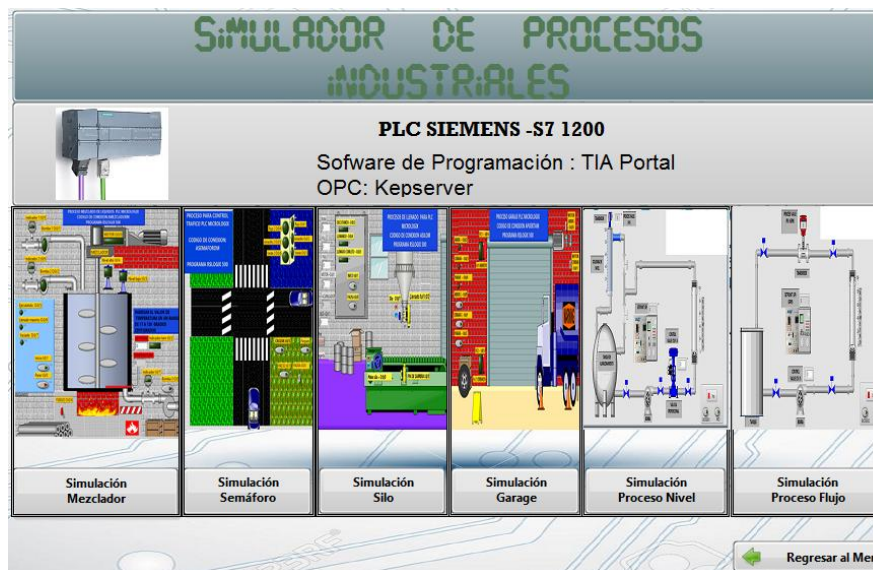


Figura 38: Panel Frontal Menú Siemens

b. Diagrama de Bloques

En el Diagrama de bloques del submenú Siemens (**Figura 39**), toda la programación se la realiza dentro de la función “While Loop” y dentro la función “Event Structure”.

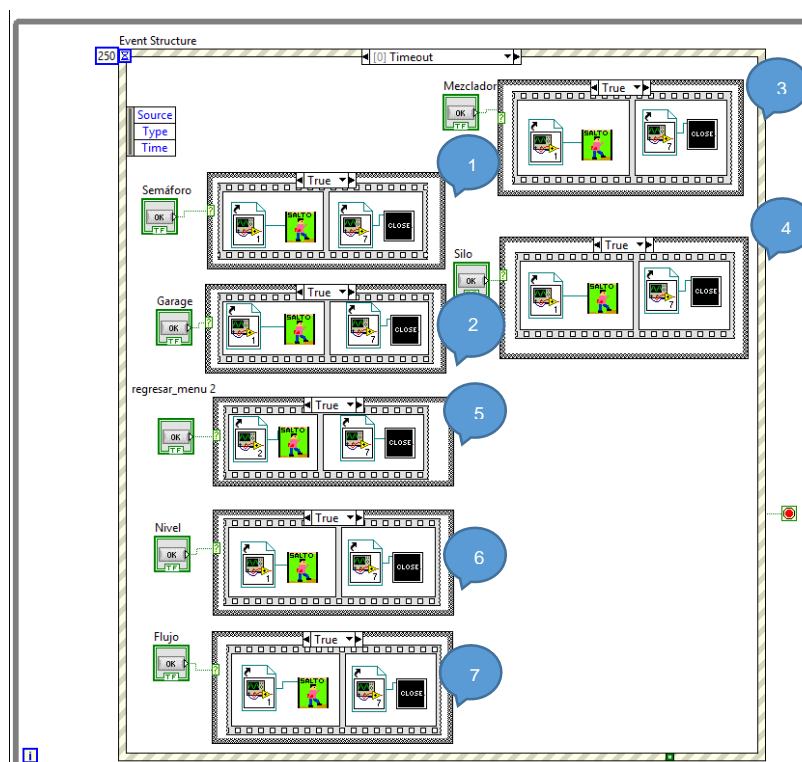


Figura 39: Diagrama de bloques del Menú Siemens.

c. Programación del menú Siemens

Dentro del “Flat Sequence Structure” se coloca todos los botones correspondientes a los seis subVI y cada botón va enlazado con su respectivo “Case Estructure” en donde, cuando la condición sea verdadera se abrirá la simulación del botón que fue seleccionado y a su vez se cierra el Menú Siemens. A continuación, se detalla la función de cada uno de los botones como se muestra en la **(Figura 39)**.

1. El botón “Semáforo” permite abrir la simulación Semáforo y cerrar el VI “Menú Siemens”.
2. El botón “Garaje” permite abrir la simulación Garaje y cerrar el VI “Menú Siemens”.
3. El botón “Mezclador” permite abrir la simulación Mezclador y cerrar el VI “Menú Siemens”.
4. El botón “Silo” permite abrir la simulación Silo y cerrar VI “Menú Siemens”.
5. El botón “regresar_menu2” permite regresar al menú principal y cerrar el VI “Menú Siemens”.
6. El botón “Nivel” permite abrir la simulación Nivel y cerrar el VI “Menú Siemens”.
7. El botón “Flujo” permite abrir la simulación Flujo y cerrar el VI “Menú Siemens”.

2.4.4.2. Submenú Allen Bradley Micrologix

Dentro del Menú principal se encuentra el submenú Allen Bradley Micrologix el cual permite el acceso a 6 subVI, cuatro simulaciones de procesos digitales y dos analógicos. Para ingresar se da clic en el botón “PLC Micrologix”, el cual está programado a través del subVI “salto.vi”, se abre el panel frontal del submenú Micrologix y se cierra el Menú Principal mediante el subVI “cerrar.vi” **(Figura 40)**.

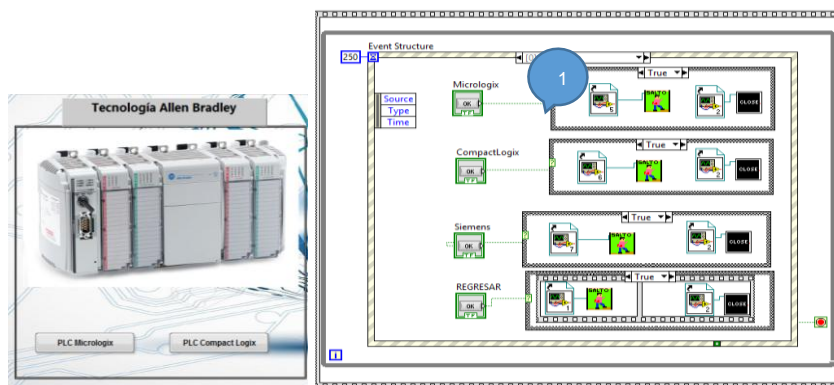


Figura 40: Acceso al Menú Allen Bradley Micrologix

a. Panel Frontal

El diseño de la interfaz de este submenú MicroLogix está formado de una imagen del PLC Allen Bradley MicroLogix, 6 imágenes que ofrecen una clara referencia de cada uno de los procesos que posee este simulador, además un botón de acceso a cada subVI (Silo, Garaje, Semáforo, Mezclador, Nivel, Flujo) y un botón para regresar al Menu principal (**Figura 41**).

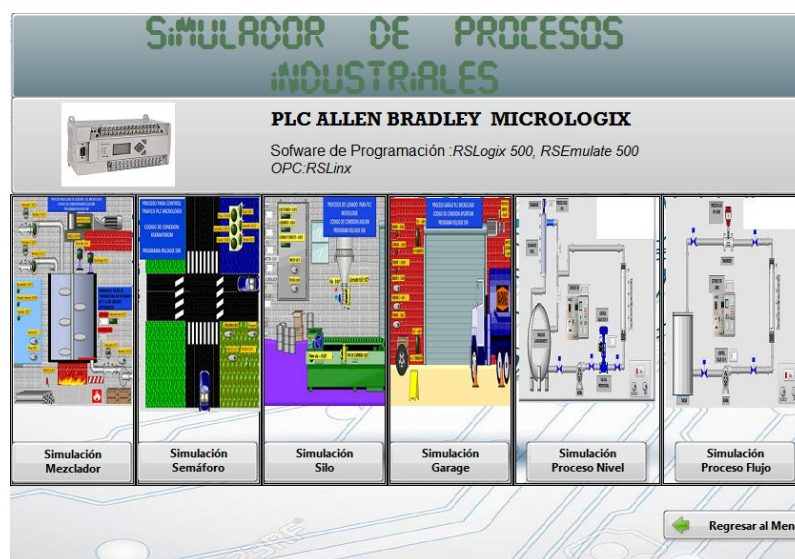


Figura 41: Panel Frontal Menú de PLC Micrologix

La programación de los botones de los seis subVI del submenú PLC Allen Bradley MicroLogix es la misma que se realizó anteriormente para el submenú Siemens.

2.4.4.3. Submenú Allen Bradley CompactLogix

En el Menú principal se encuentra el submenú Allen Bradley CompactLogix el cual permite el acceso a 6 subVI, cuatro simulaciones de procesos digitales y dos analógicos. Para ingresar se da clic en el botón “PLC Compact Logix”, el cual está programado a través del subVI “salto.vi”, se abre el panel frontal del submenú CompactLogix y se cierra el Menú Principal mediante el subVI “cerrar.vi” (**Figura 42**)

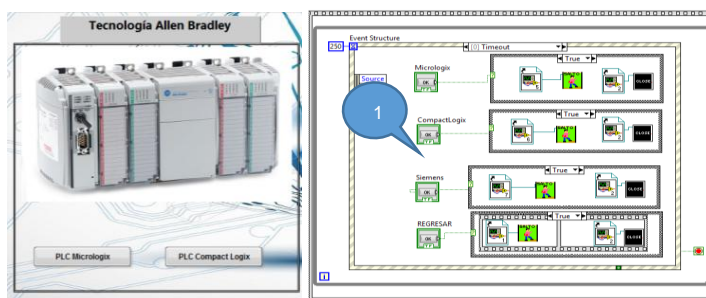


Figura 42: Acceso al Menú Allen Bradley CompactLogix

a. Panel Frontal

El diseño de la interfaz de este submenú CompactLogix está formado de una imagen del PLC Allen Bradley CompactLogix, 6 imágenes que ofrecen una referencia de cada uno de los procesos que posee este simulador, incluye un botón de acceso a cada subVI (Silo, Garaje, Semáforo, Mezclador, Nivel, Flujo) e incluye un botón para regresar al Menu principal (**Figura 43**).

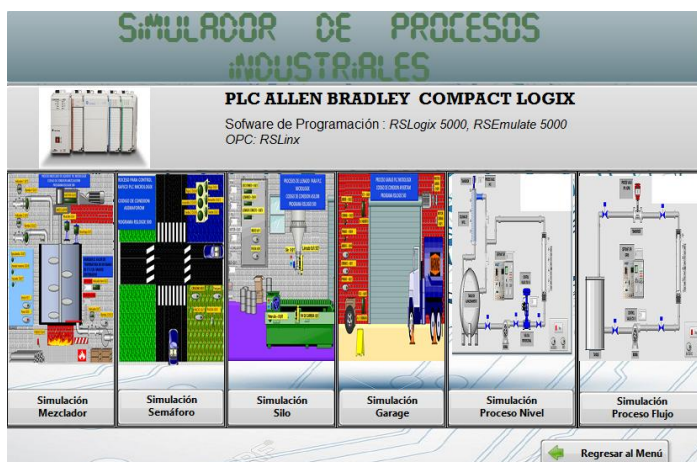


Figura 43: Panel Frontal Menú de PLC CompactLogix

La programación de los botones de los seis subVI del submenú PLC Allen Bradley CompactLogix es la misma que se realizó anteriormente para el submenú Siemens y Micrologix.

2.5. Desarrollo de la simulación del Proceso Silo

A continuación, se realiza la programación e implementación de todos los procesos (subVI's) que contiene este simulador como son: Silo, Mezclador, Semáforo, Garaje, Nivel, Flujo. En primer lugar, se detalla todo lo referente al desarrollo de la simulación del proceso Silo.

2.5.1. Diagrama de flujo del Proceso

La (Figura 44), muestra el diagrama de flujo del proceso Silo, en donde se detalla las condiciones necesarias para la programación de esta simulación.

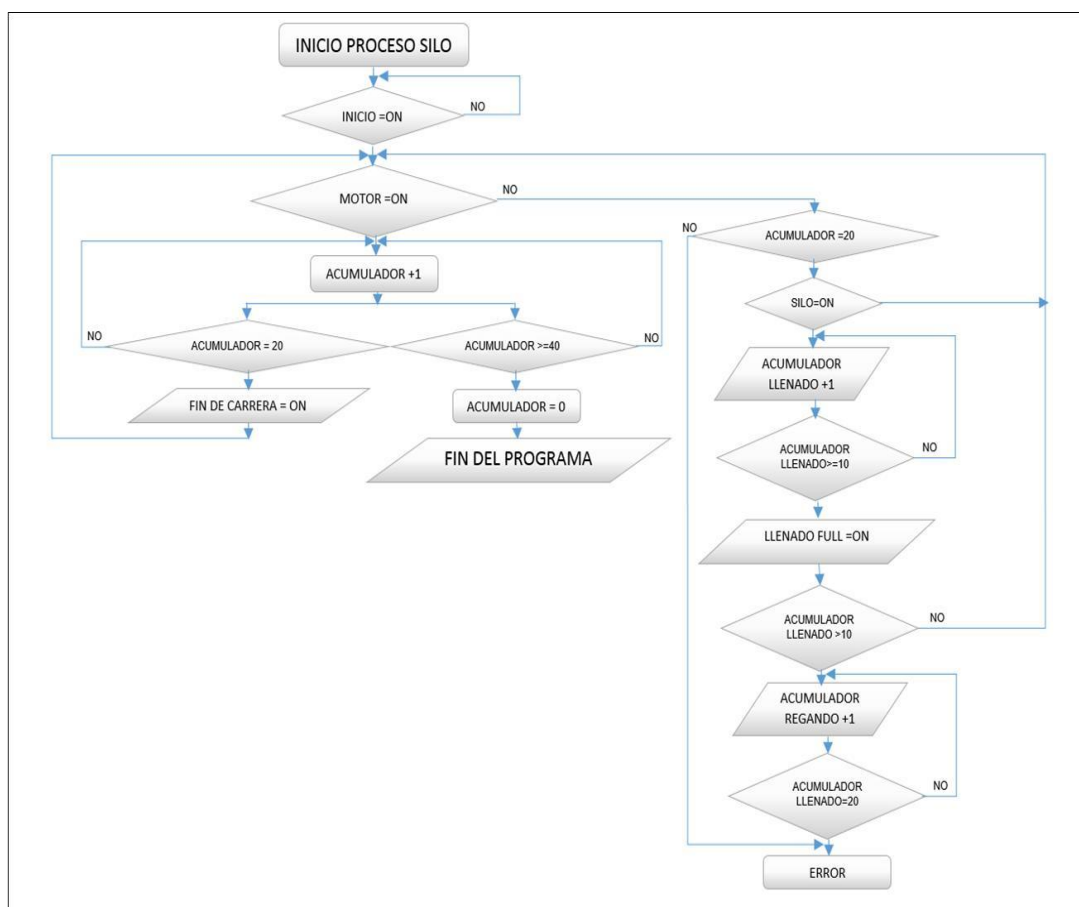


Figura 44: Diagrama de Flujo Programación Silo

2.5.2. Descripción del Proceso

El desarrollo de esta simulación tiene como finalidad controlar los distintos actuadores y sensores que intervienen en el proceso de llenado de recipientes, el proceso consta de una banda transportadora de tanques y en una determinada distancia se encuentra un silo de llenado, además sensores de proximidad y llenado, un panel de control y un motor que da movimiento a la banda transportadora.

2.5.3. Panel Frontal del Proceso Silo

El diseño del HMI del Proceso Silo consta de varias imágenes extraídas del módulo DSC de Labview tales como motor, silo, banda transportadora, un tanque, indicadores led, un panel de control donde se encuentran los interruptores para iniciar y parar el proceso, además posee un botón denominado reiniciar el cual permite regresar al menú anterior. Todos estos elementos van distribuidos de manera que formen la apariencia de un proceso de llenado de líquidos (**Figura 45**).

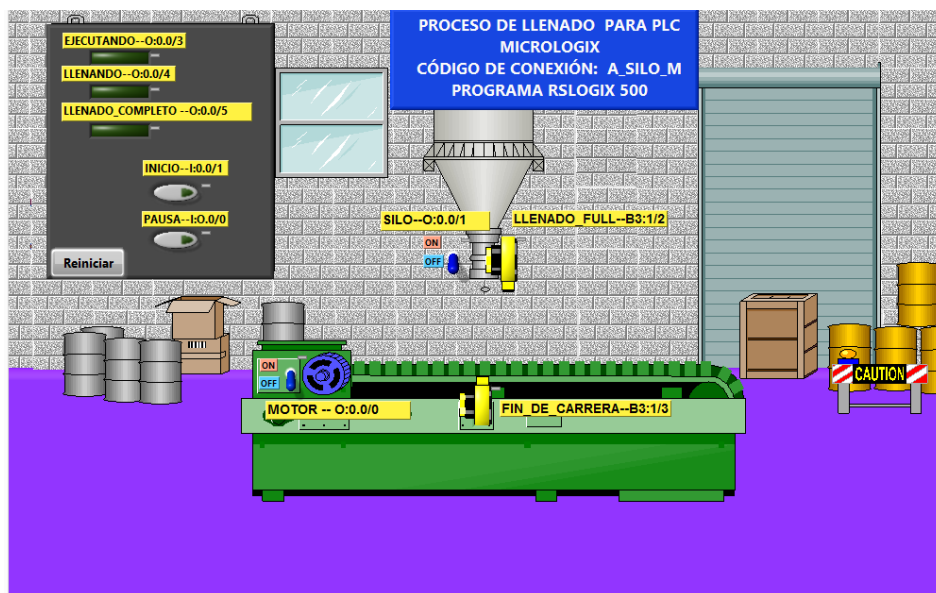


Figura 45: Panel frontal del Proceso Silo.

El panel frontal del Proceso Silo posee indicadores (Salidas) y controladores (Entradas) que se describen a continuación en la (**Tabla 6**).

Tabla 6

Indicadores y Controles del Panel Frontal Silo

Entradas	Salidas
Botón Pausa	Indicador Ejecutando
Botón Reiniciar	Indicador Llenando
Botón Inicio	Indicador Llenado_Completo
Sensor de Proximidad	Motor
Sensor de Llenado_full	Silo

2.5.4. Programación del Proceso Silo

En el diagrama de bloques general se realizará toda la programación del proceso silo, dentro del “Flat Sequence Structure” el desarrollo de este programa se dividirá en cuatro secuencias, cada una de ellas cumple una función específica que se describe a continuación (**Figura 46**).

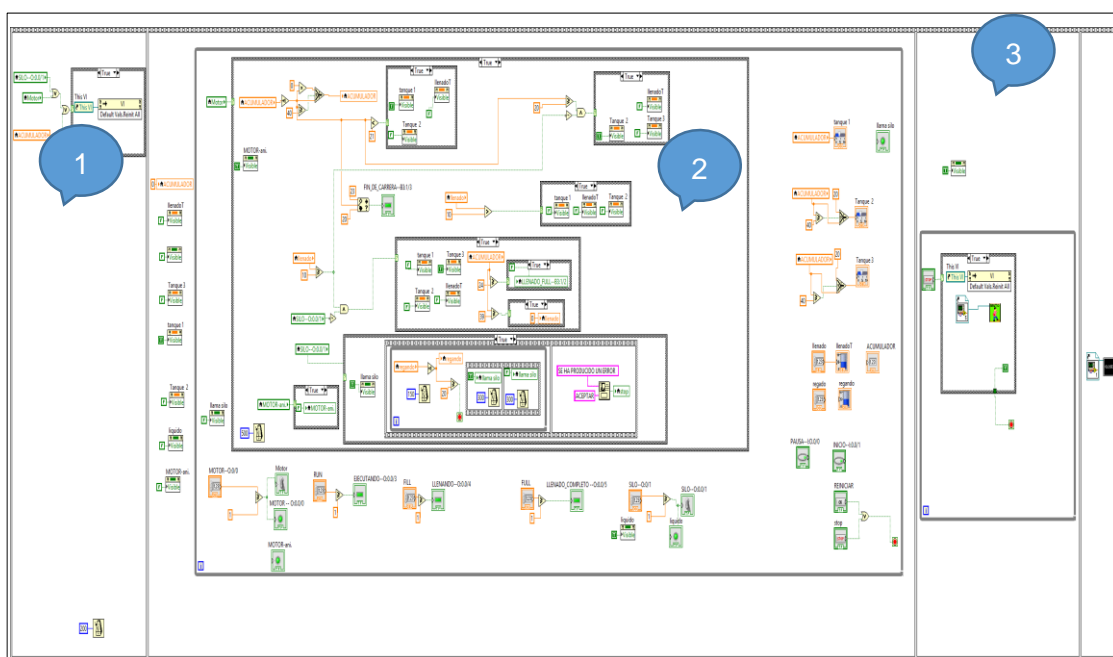


Figura 46: Diagrama de bloques General del Proceso Silo

1. La primera secuencia permite reiniciar los valores predeterminados de todo el archivo VI.
2. En la segunda secuencia se desarrolla la simulación del proceso silo.

3. La penúltima secuencia permite Desplegar el mensaje SHUTDOWN para reiniciar el archivo VI y regresar al menú anterior.
4. La ultima secuencia sirve para cerrar el archivo VI actual.

2.5.4.1. Etapas de programación de la simulación del Proceso Silo

Dentro de la segunda secuencia se va a detallar paso a paso todas las etapas de programación implementadas para el desarrollo de la simulación del Silo, las mismas que se describen a continuación. **(Figura 47).**

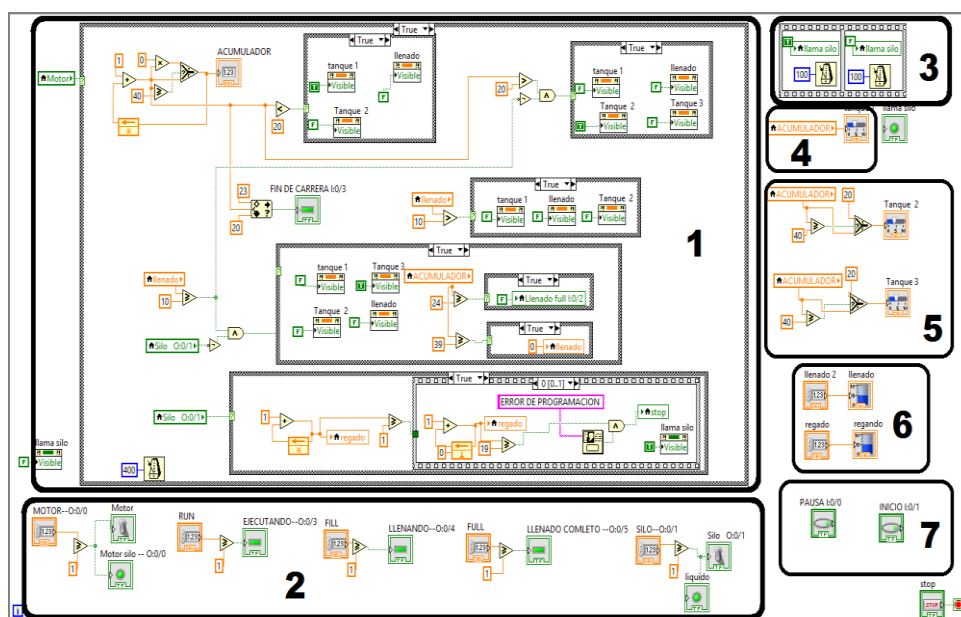


Figura 47: Diagrama de bloques Simulación del Proceso Silo.

Descripción de las etapas de programación de la simulación del Proceso Silo

1. Aquí se programa el motor que se encuentra en la banda transportadora cuando está activado o desactivado.
2. Se compara que los controles numéricos sean “mayor o igual” a uno y dicho resultado se envía a controles booleanos, cada uno de estos controles booleanos son indicadores del Proceso Silo.
3. En este punto se ejecuta esta acción cuando existe un error en el proceso, el mismo se activa conjuntamente con un mensaje de aviso y se detiene el proceso.
4. El acumulador pasa los valores a un slide “tanque1” simulando así el movimiento del primer tanque.

5. Se realiza la comparación de los valores del acumulador para el retorno del tanque cuando llega al límite del recorrido, regresando a la posición inicial. El tanque que parte de 20 unidades al llegar a 40 unidades se reinicia a 20 unidades, permaneciendo ocultos mediante la función “visible”.
6. Desarrollo del Llenado del líquido en el tanque “tank” de 10 unidades mediante la activación del silo. Si sobrepasa el contador las diez unidades el tanque “tank” causa un error.
7. Configuración de los pulsadores INICIO y PAUSA del proceso.

A continuación, se detalla cada una de las etapas de programación que se describieron anteriormente

a. Programación del Motor del Proceso Silo

En la primera etapa se realiza la programación del motor que se encuentra en la banda transportadora del Proceso Silo, además el funcionamiento de todas las etapas de la simulación del proceso cuando el motor se encuentra activado o desactivado según sea el caso. Lo primero que se realiza es escoger una imagen de un motor que cuente con un interruptor para su activación, esto se lo hace mediante Image Navigator del módulo DSC de Labview. En el motor se realiza una conversión de control numérico a control booleano para simular la acción de activar y desactivar el interruptor lo cual indica el encendido o apagado del motor respectivamente, además al motor se marca de color rojo para indicar su funcionamiento. **(Figura 48)**.

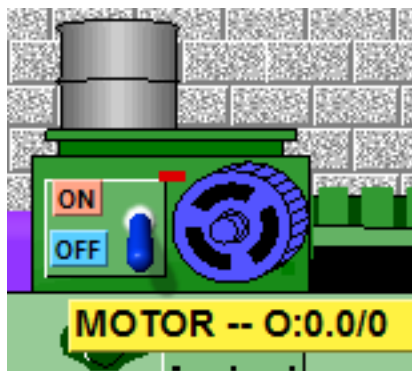


Figura 48: Control de motor en proceso silo.

- **Programación de la primera etapa cuando el motor está activado**

En la primera etapa el control del motor está enlazado a la función “Case Structure”, en donde cuando la condición sea verdadera, es decir cuando el motor se encuentre activado cumple las siguientes funciones que se describen a continuación (**Figura 49**).

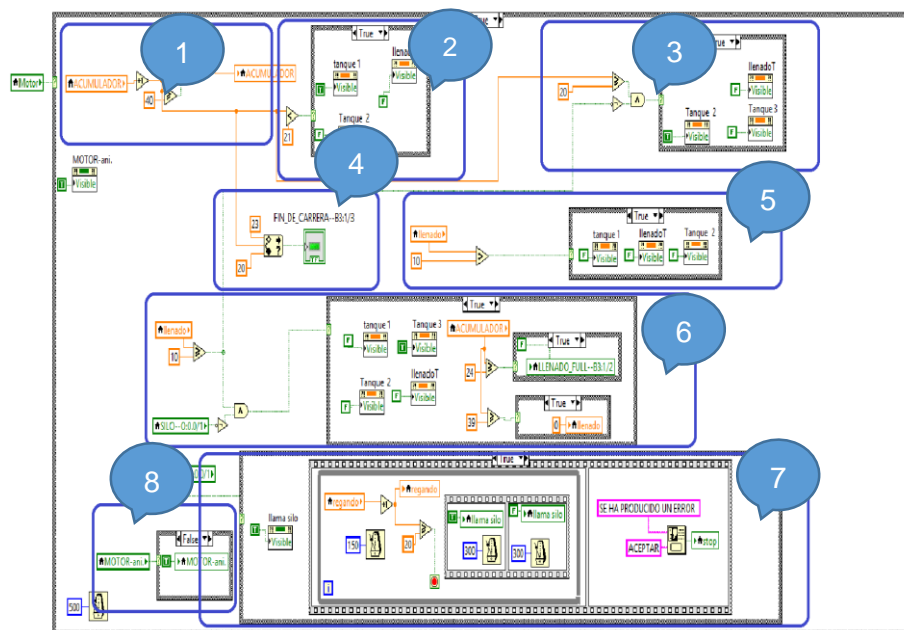


Figura 49: Función “Case Structure” cuando el motor está activado.

Descripción de las funciones que cumple el motor cuando se encuentra activado

1. Se incrementa la variable “ACUMULADOR” desde 0 a 40 y retornara cero simulando el movimiento del tanque por toda la banda transportadora.
2. Compara si el valor del acumulador es menor a 21 para activar que slide “tanque 1” sea visible y los slide restantes se mantengan ocultos.
3. Si el valor del acumulador es mayor a 20 y el tanque este vacío se activa el slide “tanque 2” los otros se desactivan.
4. Entra en funcionamiento el sensor de proximidad en un rango de 20 a 23 unidades donde se indica el instante que el tanque está arriba del sensor Fin de carrera.

5. Compara si el llenado del tanque es igual a 10 como máximo y si el control silo esta desactivado, de ser verdadero se desactivan todos los slides.
6. Compara si el tanque está lleno y el silo esta desactivado, se activa el slide “tanque 3” el cual continuará el recorrido hasta la unidad 40. Además, se Desactiva el sensor de “llenado full” en la unidad 24 y el reinicio del tanque en la unidad 39.
7. Compara si el silo está activado y el motor está en funcionamiento, un segundo acumulador “regando” se incrementa simulando el regado de contenido del tanque, con un mensaje de ERROR se detiene la simulación.
8. Simula el movimiento de un motor mediante un control booleano creado mediante el módulo DSC de Labview. Usando una función “Case Structure” que se configurará en el estado falso del indicador una constante verdadera “true” y en el estado verdadero “true” una constante falsa “false”.

- **Programación de la primera etapa cuando el motor está desactivado**

En la primera etapa el control del motor está enlazado a la función “Case Structure”, cuando la condición sea falsa es decir cuando el motor se encuentre desactivado cumple las siguientes funciones que se describen a continuación (**Figura 50**).

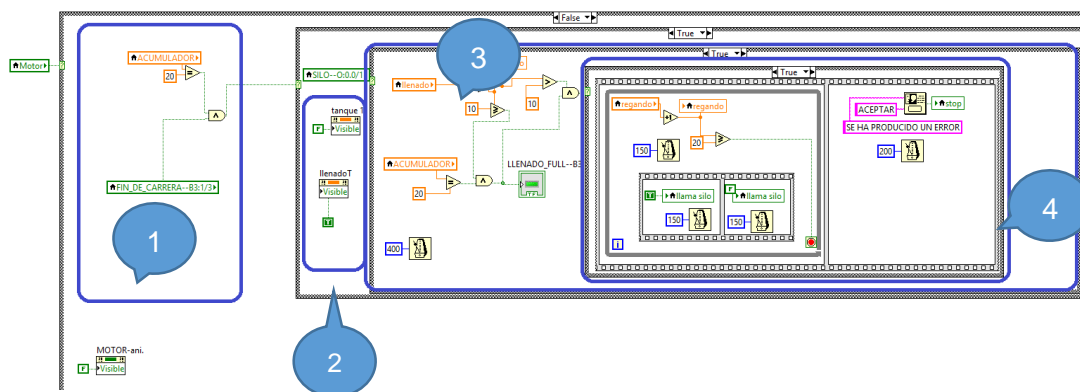


Figura 50: Función “Case Structure” cuando el motor esta desactivado.

Descripción de las funciones que cumple el motor cuando se encuentra desactivado

1. Compara si el valor del acumulador es igual a 20 unidades y el sensor de proximidad “Fin de carrera” se encuentra activado.
2. En caso de ser verdadero la primera etapa se activa el tanque de llenado “tank” y se oculta el slide “tanque 1”.
3. Si el silo está activado incrementa el acumulador “llenado”. Además, compara si “llenado” es igual o mayor a 10 unidades y “Acumulador” igual a 20 unidades, si es verdadero se enciende el indicador “llenado full”.
4. Compara que el acumulador “llenado” sea mayor que 10 si es así se activa un mensaje de error con una simulación de fuego y el incremento del acumulador “regando”.

b. Configuración de los indicadores del proceso Silo

En la segunda etapa se configura los indicadores que conformarán el panel frontal del Proceso Silo, los indicadores led se activarán cuando el proceso este ejecutándose, durante el llenado del tanque y cuando esté completamente lleno el tanque. **(Figura 51)**.

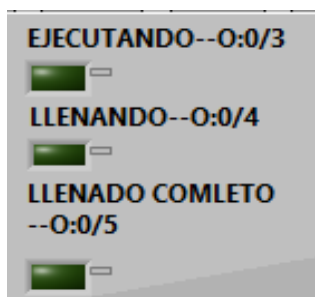


Figura 51: Indicadores led de la acción del proceso

Para lo cual cada uno de los indicadores numéricos deben ser convertidos en indicadores booleanos, ya que los indicadores son led's **(Figura 52)**.

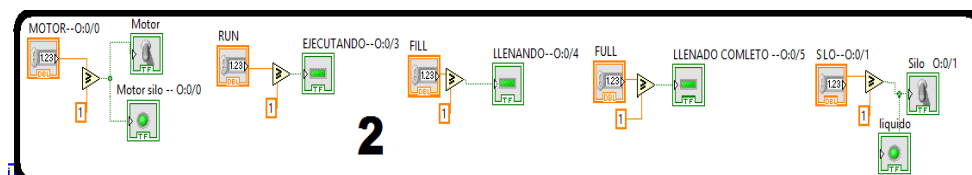


Figura 52: Transformación de un indicador numérico a booleano

c. Programación de Errores en el proceso silo.

Los errores que posee la simulación del proceso Silo están configurados en dos secuencias, en la primera secuencia se incrementa el acumulador “regando” a 20 unidades conjuntamente con una simulación de fuego “llama silo” la misma que es creada mediante el módulo DSC y en la segunda secuencia se coloca un mensaje de error “SE HA PRODUCIDO UN ERROR” mediante la función “One Button Dialog” (Figura 53) y con botón llamado SHUTDOWN para reiniciar la simulación del proceso.

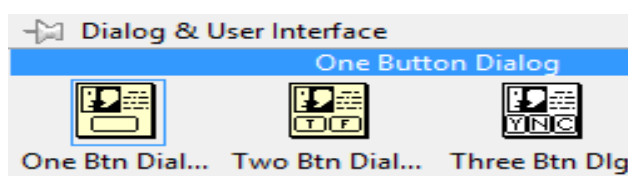


Figura 53: Localización de la función “One Button Dialog”

La activación de los errores se produce cuando se ejecutan las acciones que se describen a continuación (Figura 54).

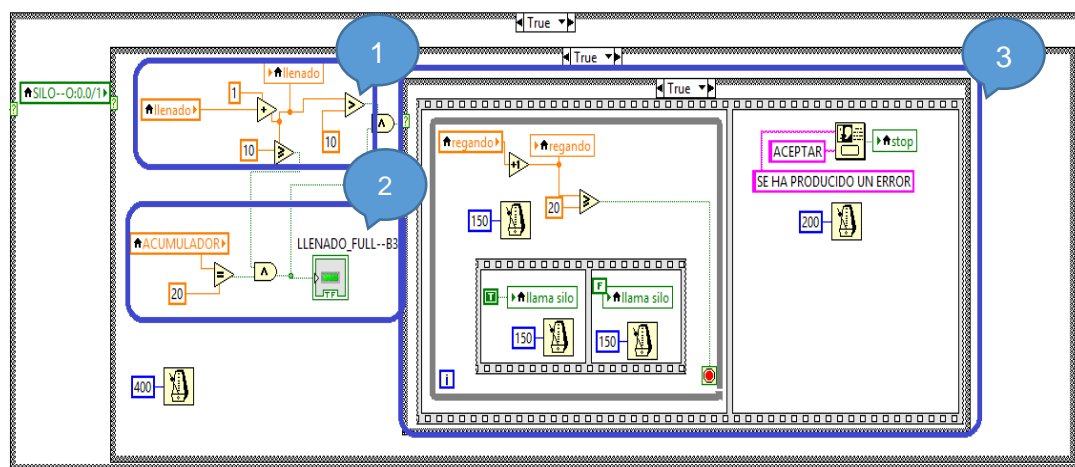


Figura 54: Activación de errores en Simulación Silo

1. Si se sobrepasa el acumulador sobre las 10 unidades indica que el contenido se está regando y activa un mensaje de error.
2. Compara si el “Acumulador” es igual a 20 unidades para activar el sensor de llenado full.
3. Activa un error cuando el acumulador “llenando” sea mayor a 10 unidades y llenar un tanque “regando” acompañado de una simulación de candela “llama silo”.

d. Desarrollo de la Animación del movimiento del tanque del Proceso Silo

Las etapas 4,5,6 son las encargadas de la animación del tanque que se traslada a través de la banda transportadora del Proceso Silo. Para realizar esta animación es necesario el uso del módulo DSC de Labview, el cual proporciona una gran variedad de imágenes relacionadas con equipos, instrumentos pertenecientes a la industria, en este caso utilizaremos la imagen de un tanque. Para crear el movimiento del tanque se utiliza tres slides “Horizontal Pointer Slide”, cada slide va enlazado a un tanque. El primer slide posee una longitud de 40 unidades cuyo valor es el total de recorrido del primer tanque (**Figura 55**).

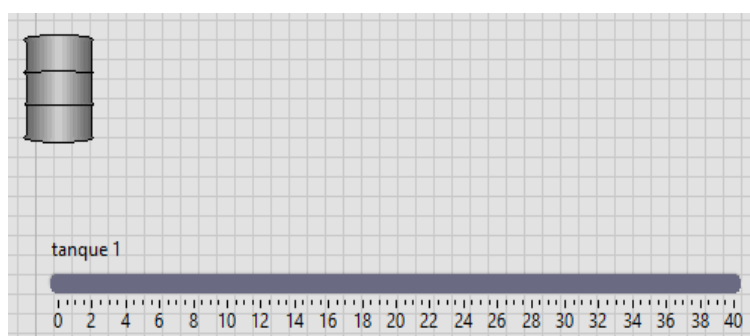


Figura 55: Slide de 0 a 40 unidades tanque 1.

El segundo “Horizontal Pointer Slide” va entre 20 y 40 unidades cuyo recorrido va desde la mitad hasta el final, se utiliza para el traslado de un tanque vacío cuando el silo no es activado (**Figura 56**).

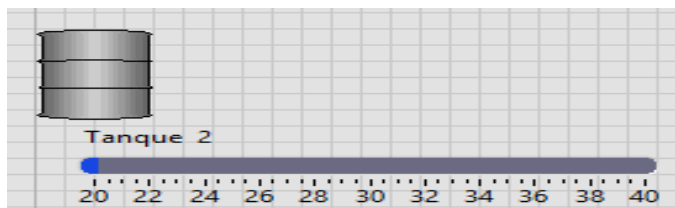


Figura 56: Slide de 20 a 40 unidades tanque 2.

El tercer “Horizontal Pointer Slide” de 20 a 40 unidades sirve para el traslado de un tanque lleno cuando el silo está activado **(Figura 57)**.

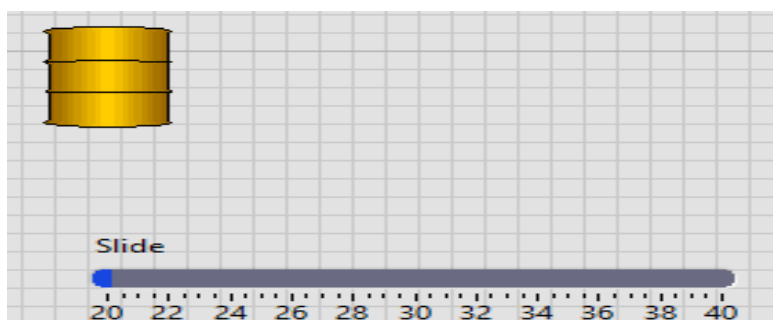


Figura 57: Slide de 20 a 40 unidades tanque3 lleno.

Para el llenado del tanque se usa un control numérico “tank” este se une a un tanque de 0 a 10 unidades por medio del módulo DSC **(Figura 58)**.

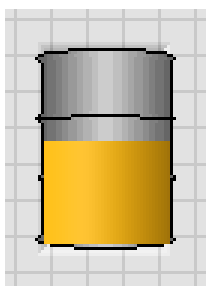


Figura 58: Tanque de llenado de 0 a 10 unidades

e. Configuración de los controles con las entradas y salidas del PLC

Mediante la función “Data Binding” se realiza el enlace entre el servidor OPC, el simulador y el software de programación del PLC, lo cual permite activar o desactivar el motor, indicadores led, el silo y la interacción de los botones inicio, pausa y sensores.

2.6. Desarrollo de la simulación del Proceso Mezclador

A continuación, se realizará la programación e implementación de la simulación del proceso Mezclador.

2.6.1. Diagrama de Flujo del Proceso

En la siguiente (**Figura 59**), se muestra el diagrama de flujo del proceso Mezclador, en donde se indica las condiciones necesarias para la programación de esta simulación.

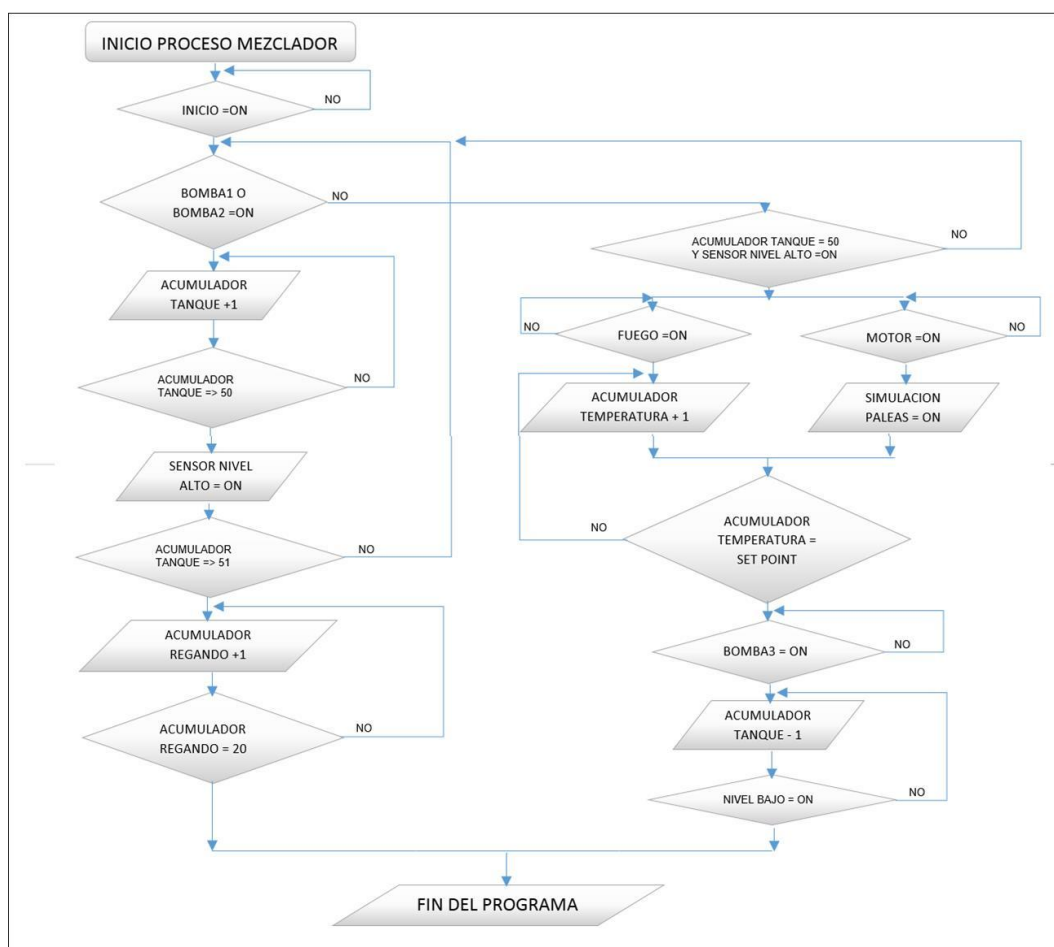


Figura 59: Diagrama de Flujo Programación Mezclador

2.6.2. Descripción del Proceso

El proceso está desarrollado con la finalidad de controlar los distintos actuadores y sensores, el mismo que está compuesto de dos válvulas con su respectivo indicador de flujo, un tanque de llenado con sensores de nivel bajo

y alto, un motor que acciona un mezclador, una hornilla que será usada para calentar el contenido del tanque, un sensor de temperatura y una bomba de escape con indicador de flujo usada para el vaciado del tanque.

2.6.3. Panel Frontal del Proceso Mezclador

El diseño del HMI del Proceso Mezclador consta de varias imágenes extraídas del módulo DSC de Labview tales como tuberías, motor, válvulas, bombas, un tanque donde se va a mezclar los líquidos, indicadores led, un panel de control donde se encuentran los interruptores para iniciar y parar el proceso, además posee un botón denominado reiniciar el cual permite regresar al menú anterior. Todos estos elementos van distribuidos de manera que formen la apariencia de un proceso de mezclado de líquidos (**Figura 60**).

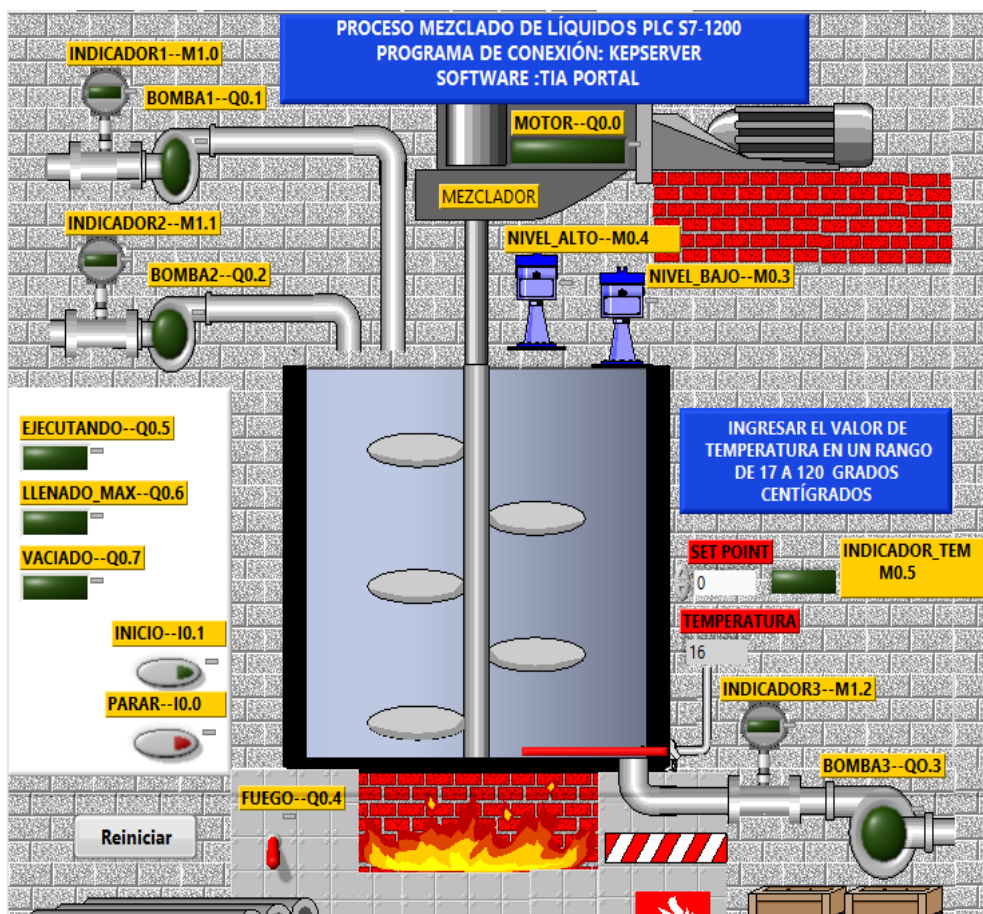


Figura 60: Panel frontal del Proceso Mezclador

El panel frontal del Proceso Mezclador posee indicadores (Salidas) y controles (Entradas) que se describen a continuación en la siguiente (**tabla 7**).

Tabla 7

Indicadores y Controles del Panel Frontal Mezclador

Entradas	Salidas
Pulsador Pausa	Indicador Ejecutando
Pulsador Inicio	Indicador Llenado Máximo
Indicador de las bombas 1,2,3	Indicador Vaciado
Indicador de Nivel Alto	Bombas de Líquido 1,2,3
Indicador de Nivel Bajo	Fuego
Indicador de Temperatura	Motor

2.6.4. Programación del Proceso Mezclador

En el diagrama de bloques se desarrollará toda la programación del proceso Mezclador, dentro del “Flat Sequence Structure” el desarrollo de este programa se dividirá en cuatro secuencias, cada una de ellas cumple una función específica que se describe a continuación (**Figura 61**).

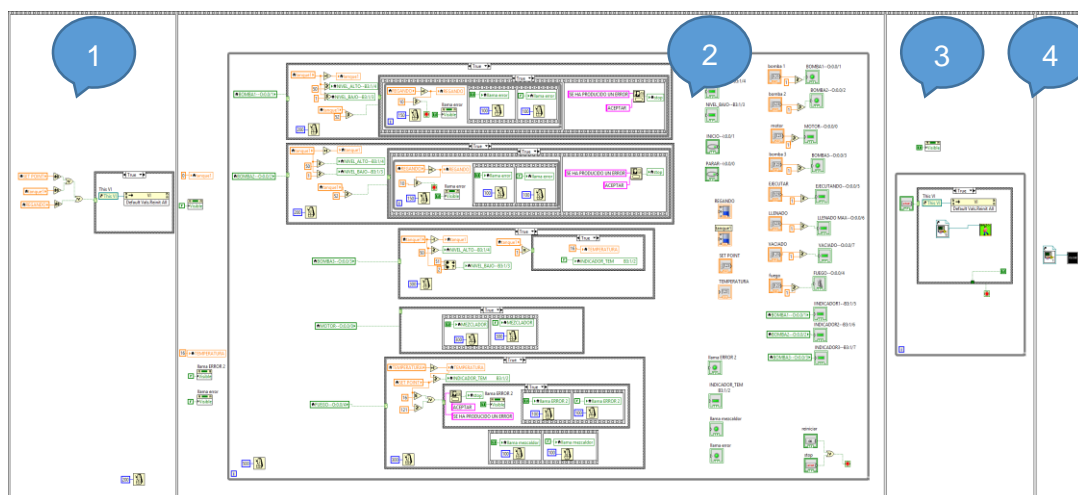


Figura 61: Diagrama de bloques General del proceso Mezclador.

1. La primera secuencia permite reiniciar todo el archivo VI a valores predeterminados.
2. En la segunda secuencia se encuentra la programación principal del proceso mezclador, todas las funciones, controles, indicadores, etc. elementos que son necesarios en el funcionamiento de la simulación.

3. La tercera secuencia contiene un botón denominado “shutdown”, cuando ocurre un error activa la función “Case Structure” la cual permite reiniciar el archivo VI dejándolo en valores predeterminados y abre el archivo VI menú anterior.
4. La última secuencia permite cerrar el archivo VI actual mediante el subVI cerrar.

2.6.4.1. Etapas de programación de la simulación del Proceso Mezclador

En la segunda secuencia se va desarrollar toda la simulación del Proceso Mezclador, aquí se detalla paso a paso las funciones de cada una de las etapas de programación implementadas para el desarrollo de la simulación, las mismas que se describen a continuación (**Figura 62**).

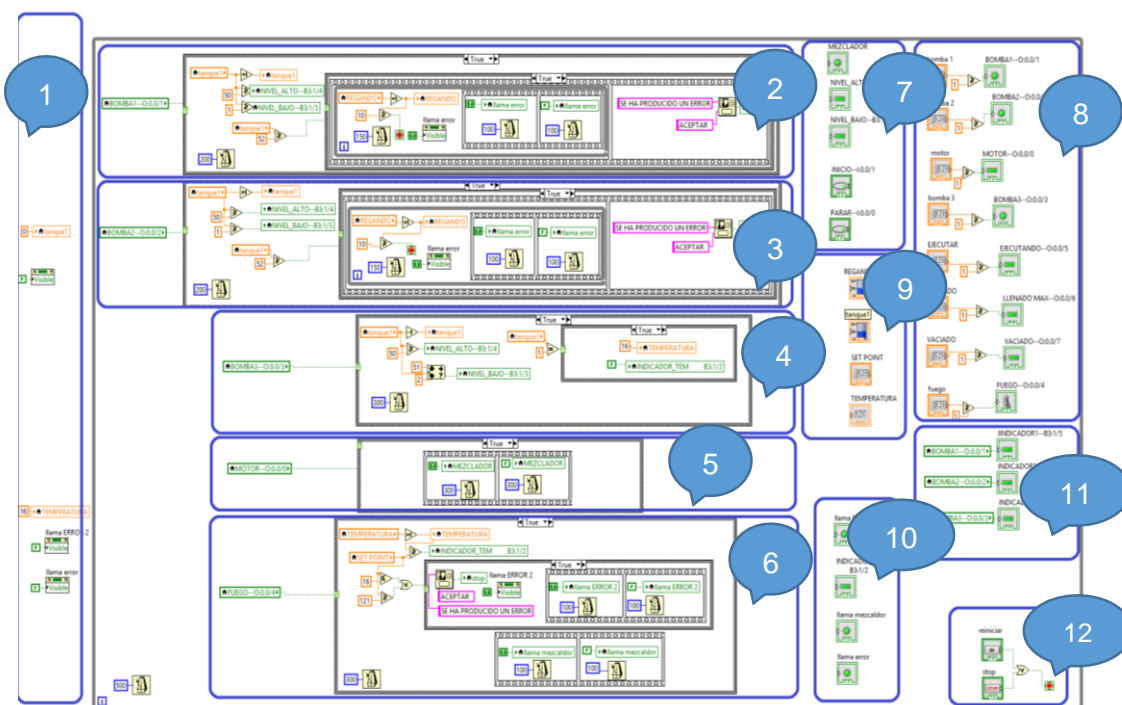


Figura 62: Diagrama de bloques Simulación del Proceso Mezclador.

Descripción de las funciones de todas las etapas de programación de la simulación del Proceso Mezclador.

1. Se inicia el acumulador tanque1 en cero y en 16 el control temperatura, además se conecta una constante falsa "false" a las funciones visibles de botón shutdown, llama del sensor, llama de regado.
2. Se activa la función "Case Structure" cuando el control bomba1 esté activado, ahí se incrementa el acumulador tanque1 y a su vez se activa el indicador de nivel bajo cuando sea igual o mayor a 1 unidad y de nivel alto cuando sea igual o mayor a 50 unidades. También se compara si el acumulador tanque1 es mayor a 52 unidades el cual activa la función "Case Structure", dentro de esta se encuentra la función "Flat Sequence Structure" con dos secuencias. En la primera secuencia se incrementa el acumulador regando a 10 unidades, el cual activa el error del tanque y la simulación del fuego, en la segunda secuencia se coloca un mensaje de error mediante la función "One Button Dialog".
3. En la tercera etapa se activa la función "Case Structure" cuando el control bomba2 esté activado, dentro del case se incrementa el acumulador tanque1 y a su vez activa el indicador de nivel bajo cuando sea igual o mayor a 1 unidad y el indicador de nivel alto cuando sea igual o mayor a 50 unidades. Además, se compara si el acumulador tanque1 es mayor a 52 unidades para activar la función "Case Structure" dentro de esta se encuentra la función "Flat Sequence Structure" con dos secuencias. En la primera secuencia se incrementa el acumulador "regando" a 10 unidades, el cual activa el fuego del error del tanque y la simulación del fuego, en la segunda secuencia se coloca un mensaje de error mediante la función "One Button Dialog".
4. Disminuye el acumulador tanque1 de 50 unidades a 0 unidades y compara si acumulador es mayor o igual a 50 unidades para encender el indicador nivel alto y mantener activado de nivel bajo de 51 unidades a 2 unidades, si el acumulador tanque1 es igual a 1 restablece el valor de la temperatura a 16 y pone en falso el indicador temperatura.
5. Se activa la función "Case Structure" mediante el control motor, la función "Flat Sequence" posee dos secuencias en la primera se conecta una variable verdadera "true" a la función mezclador y en la segunda secuencia una constante falsa "false" a la función mezclador.

6. La sexta etapa activa el indicador fuego mediante la función “Case Structure”, en el caso verdadero “true” incrementa el acumulador temperatura acompañado de la simulación de fuego en la hornilla, además se compara si el valor del acumulador temperatura es mayor o igual al setpoint aquí se activa el indicador “indicador_tem”, si el setpoint es mayor o igual a 121 o menor o igual a 16 se activa la función de error la cual contiene un mensaje de error y una simulación de error.
7. Configuración de los indicadores de entrada con sus respectivas etiquetas de acuerdo al tipo de PLC que se esté usando.
8. En la octava etapa se realiza la configuración los indicadores numéricos de salida con sus respectivas etiquetas y también se compara si es mayor o igual a uno para activar el indicador booleano.
9. En esta etapa se encuentra los indicadores numéricos tipo tanque, control numérico setpoint y acumulador numérico temperatura.
10. Aquí se colocan los controles booleanos indicadores de fuego e indicador de temperatura mediante el uso del módulo DSC de Labview.
11. En esta etapa se conecta los controles de activación de las bombas a sus respectivos indicadores.
12. Se compara mediante una función “OR” la activación del control reinicio o el stop general deteniendo la simulación y saliendo de la función “while loop”.

2.7. Desarrollo de la simulación del Proceso Semáforo

A continuación, se realiza la programación e implementación de la simulación del proceso Semáforo.

2.7.1. Diagrama de Flujo del Proceso

La siguiente (**Figura 63**), muestra el diagrama de flujo del proceso Semáforo, en donde se indica las condiciones necesarias para la programación de esta simulación.

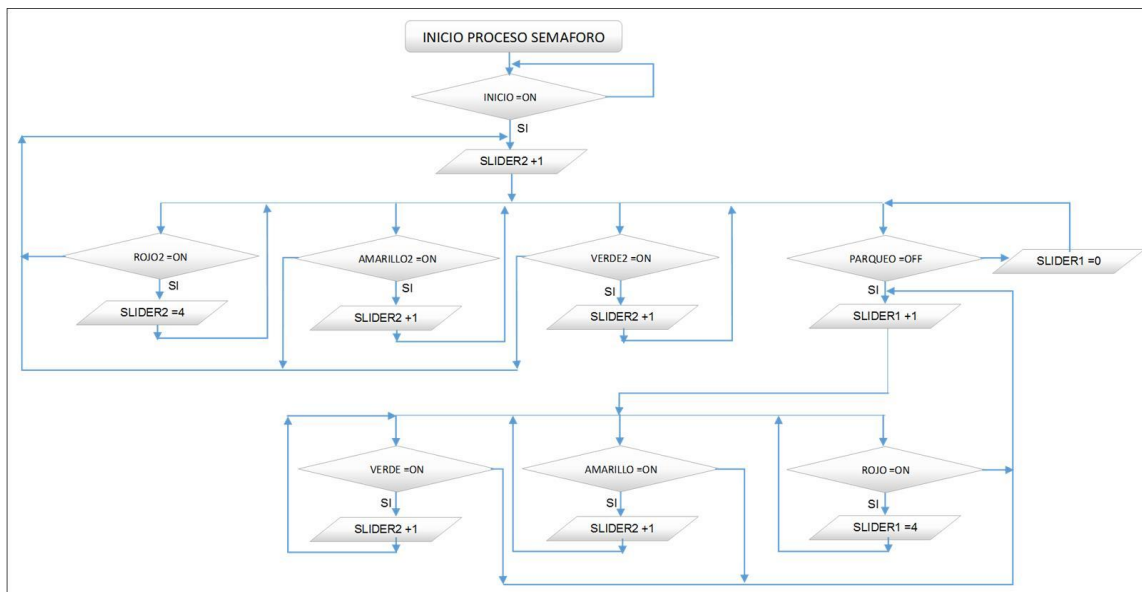


Figura 63: Diagrama de Flujo Programación Semáforo

2.7.2. Descripción del Proceso

El siguiente proceso está desarrollado con el fin de controlar los distintos actuadores y sensores, consta de un semáforo de dos vías, un pulsador de pase de emergencia (Cruce de peatones) y un control de parqueo solamente para una vía.

2.7.3. Panel Frontal del Proceso Semáforo

El diseño del HMI del Proceso Semáforo consta de varias imágenes extraídas del módulo DSC de Labview tales como carros, carriles de una autopista, texturas de césped, indicadores led, un panel de control donde se encuentran los interruptores para iniciar el proceso, cruce de peatones y parar el proceso, además posee un botón denominado reiniciar el cual permite regresar al menú anterior. Todos estos elementos van distribuidos de manera que formen la apariencia de un proceso para el control de Tráfico en dos vías (Figura 64).

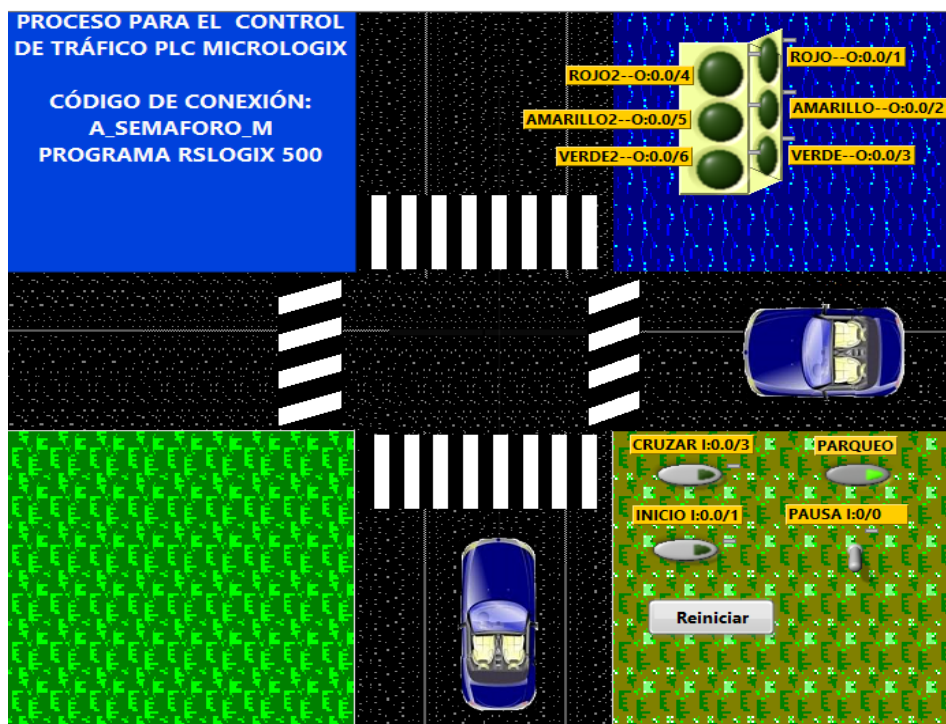


Figura 64: Panel frontal de proyecto semáforo.

El panel frontal del proceso Semáforo posee indicadores (Salidas) y sensores (Entradas) que se describen a continuación en la siguiente **(Tabla 8)**.

Tabla 8

Indicadores y Controles del Panel Frontal Semáforo

Entradas	Salidas
Botón Pausa	Indicadores Rojo y Rojo2
Botón Inicio	Indicadores Amarillo y Amarillo2
Botón Cruzar	Indicadores Verde y Verde2
Botón Reiniciar	Bombas de líquido 1,2,3

2.7.4. Programación del Proceso Semáforo

En el siguiente diagrama de bloques **(Figura 65)** se desarrollará toda la programación del proceso Semáforo, el desarrollo de este programa se dividirá en cuatro secuencias, cada una de ellas cumple una función específica que se describe a continuación.

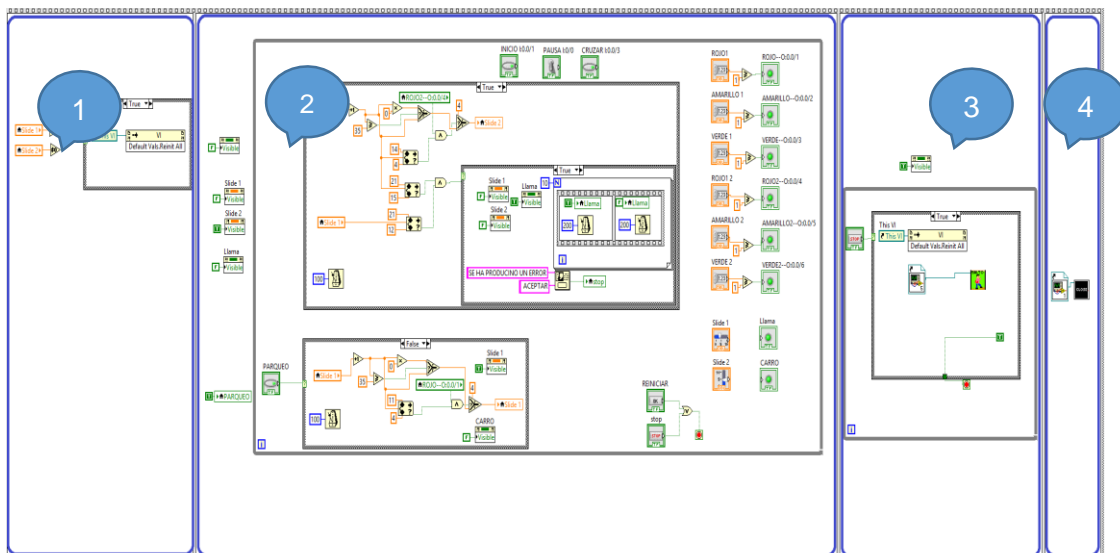


Figura 65: Diagrama de Bloques General del Proceso semáforo.

1. La primera secuencia permite comparar que los acumuladores slide1 y slide2 sean diferentes de cero para activar la función “Case Structure” la cual contiene la función “Invoke Node” configurada para reiniciar el archivo VI a valores predeterminados.
2. En la segunda secuencia se encuentra la programación principal del proceso Semáforo, todas las funciones, controles, indicadores, etc. elementos que son necesarios en el funcionamiento de la simulación.
3. La tercera secuencia contiene un botón denominado “shutdown”, cuando ocurre un error activa la función “Case Structure” la cual permite reiniciar el archivo VI dejándolo en valores predeterminados y abre el archivo VI menú anterior.
4. La ultima secuencia permite cerrar el archivo actual mediante el sub-VI cerrar.

2.7.4.1. Etapas de programación de la simulación del Proceso Semáforo

Dentro de la segunda secuencia se va a detallar paso a paso todas las etapas de programación implementadas para el desarrollo de la simulación del proceso Semáforo, a continuación, se describen cada una de ellas. (Figura 66).

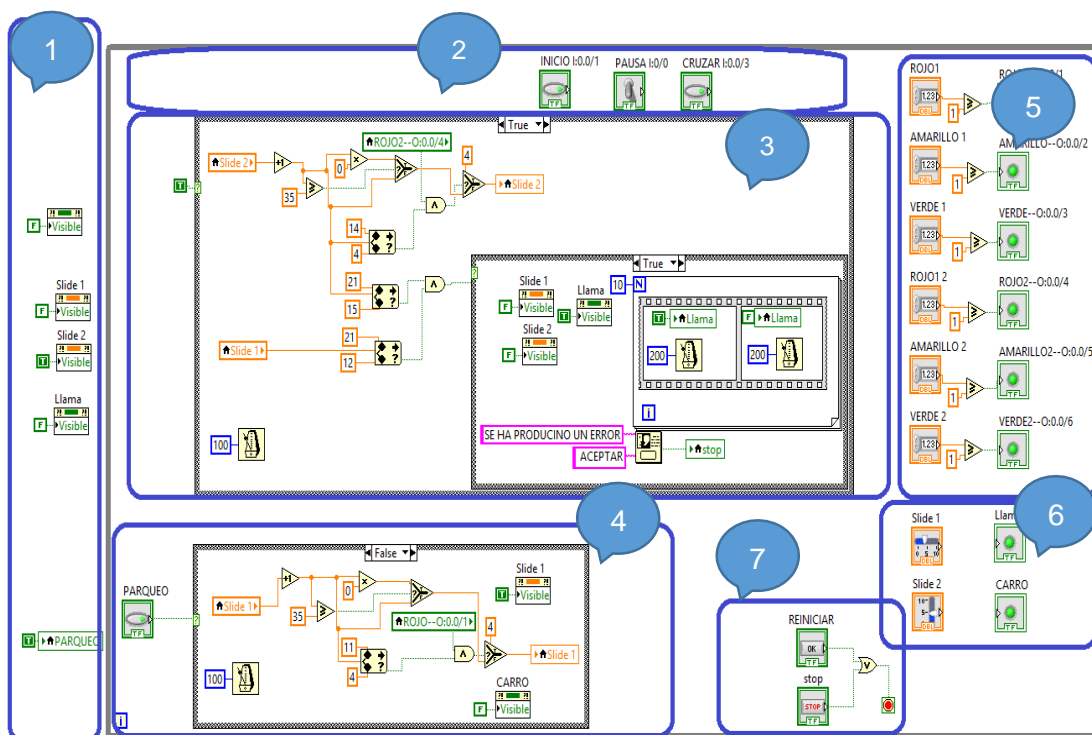


Figura 66: Diagrama de bloques Simulación del Proceso Semáforo.

Descripción de las funciones de todas las etapas de programación de la simulación del Proceso Semáforo.

1. En la primera etapa se conecta una constante “false” a la propiedad visible del shutdown, slide1, llama y una constante verdadera “true” en la propiedad visible de slide2 y en el botón parqueo.
2. Se configura el “Data Binding” de los controles inicio, pausa y cruzar.
3. En la tercera etapa se conecta una constante verdadera (T) a la función “Case Structure” la cual contiene el incremento del acumulador Slide2, este acumulador se compara si es mayor o igual a 35 unidades para multiplicar por 0 unidades con lo que se realiza el reinicio del movimiento del vehículo vertical, si está en el rango de 4 a 14 unidades y el indicador rojo2 está activado, se carga el acumulador con 4 unidades simulando el paro del vehículo antes del paso peatonal, si está en el rango de 15 a 21 unidades y el Slide1 de 12 a 21 unidades se activa la función “Case Structure” y dentro de esta se conecta una variable falsa a la propiedad visible de Slide1, Slide2, una variable

verdadera se conecta a llama, un mensaje de error y la función “For Loop” con la simulación fuego.

4. En esta etapa se activa la función “Case Structure” mediante el botón Paro en su caso “True” se carga cero al Slide1, también se conecta una variable verdadera a las propiedades visible carro y una constante falsa a la propiedad visible Slide1. En el caso “false” se incrementa el Slide1 y en caso de que sea mayor o igual a 35 unidades se cargar con 0 el Slide1, caso contrario se compara si el Slide1 se encuentra en el rango de 4 a 11 unidades y el control indicador rojo este activado para cargar con 4 unidades al Slide1.
5. Se realiza la conversión de todos los indicadores numéricos (rojo, rojo1, amarillo, amarillo1, verde, verde1) a indicadores booleanos.
6. En esta etapa se encuentra los controles Slide1, slide2, carro y llama los cuales se activan mediante variables locales en distintas etapas del proceso.
7. Se conecta los controles stop y reiniciar al paro general de la función “while Loop” general.

2.8. Desarrollo de la simulación un Proceso Garaje

A continuación, se realiza la programación e implementación de la simulación del proceso Garaje.

2.8.1. Diagrama de Flujo del Proceso

En la siguiente (**Figura 67**) se observa el diagrama de flujo del Proceso Garaje, en donde se indica las condiciones necesarias para la programación de esta simulación.

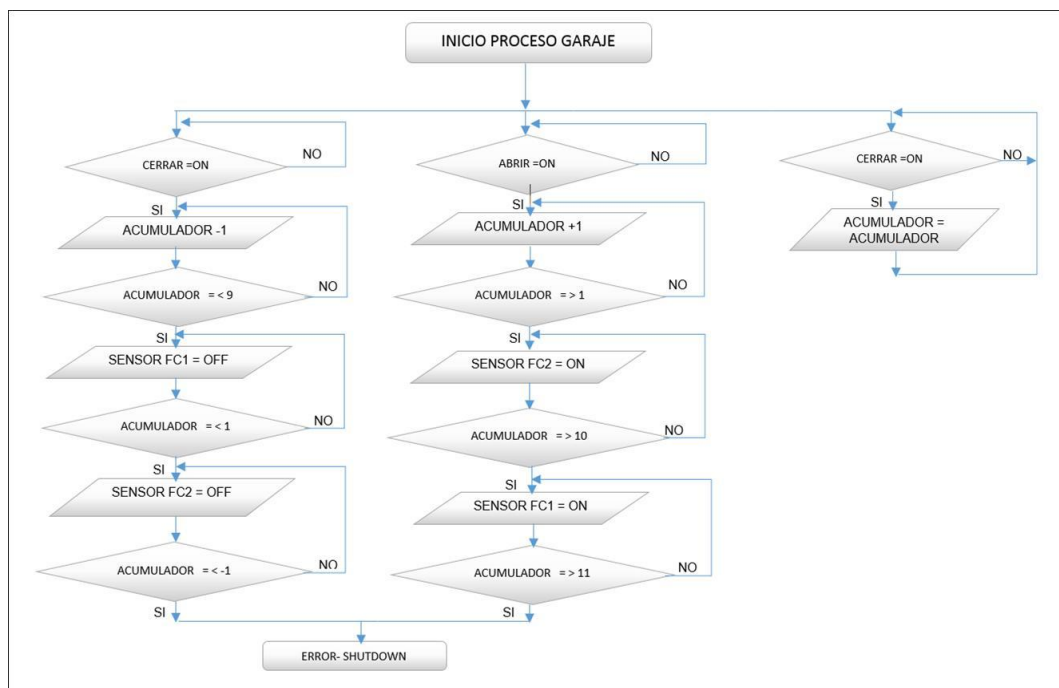


Figura 67: Diagrama de Flujo Programación Garaje

2.8.2. Descripción del Proceso

El proceso consta de una puerta de garaje accionada con dos motores para abrir y cerrar respectivamente, para su control dos sensores fin de carrera para controlar las acciones de los motores en los extremos de la puerta, además existe un interruptor para el proceso, ya sea cuando la puerta se esté abriendo o cuando se esté cerrando.

2.8.3. Panel Frontal del Proceso Garaje

El diseño del HMI del Proceso Garaje consta de varias imágenes extraídas del módulo DSC de Labview tales como carros, puerta, motor, texturas de ladrillos, indicadores led, un panel de control donde se encuentran los interruptores para abrir la puerta, cerrar y parar el proceso, además posee un botón denominado reiniciar el cual permite regresar al menú anterior. Todos estos elementos van distribuidos de manera que formen la apariencia de un proceso para el control de una puerta de un Garaje (**Figura 68**).

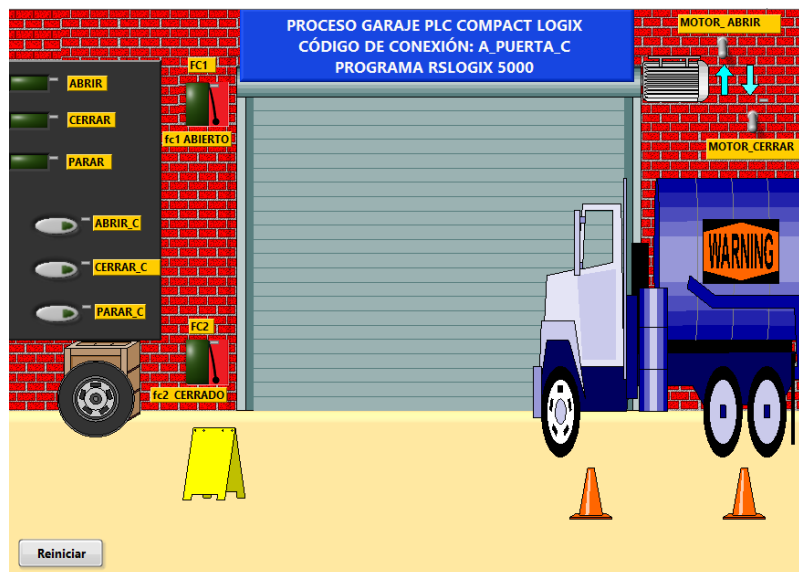


Figura 68: Panel frontal del Proceso Garaje.

El panel frontal del proceso Garaje posee indicadores (Salidas) y controladores(Entradas) que se describen a continuación en la siguiente (Tabla 9).

Tabla 9

Indicadores y Controles del Panel Frontal Garaje

Entradas	Salidas
Botón Pausa	Indicador Abrir
Botón Abrir	Indicador Cerrar
Botón Cerrar	Indicador Parar
Sensor fin de carrera Puerta Abierta	Motor
Sensor fin de carrera Puerta Cerrada	

2.8.4. Programación del Proceso Garaje

En el siguiente diagrama de bloques (**Figura 69**) se realizará toda la programación del proceso Garaje, el desarrollo de este programa se dividirá en cuatro secuencias dentro de la función “Flat Sequence Structure”, cada una de ellas cumple una función específica las cuales se describen a continuación.

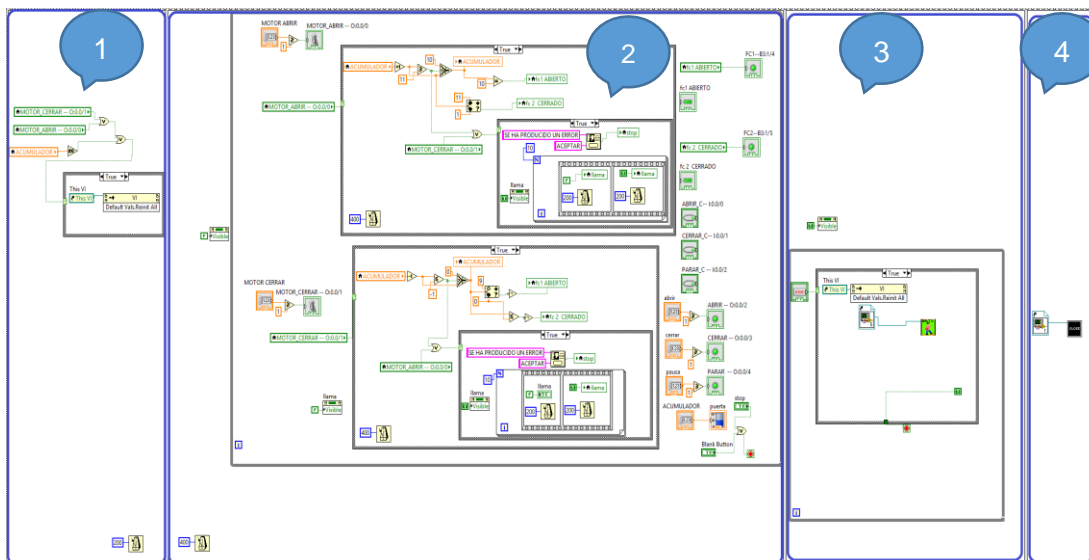


Figura 69: Diagrama de bloques completo del Proceso Garaje.

1. En la primera secuencia se compara si acumulador es diferente de cero o el indicador motor abrir o motor cerrar este activado para activar la función “Case Structure” para reiniciar el archivo VI actual a los valores predeterminados.
2. En la segunda secuencia se encuentra la programación de la simulación del proceso Garaje, en la cual se da movimiento a la puerta y la activación de los sensores dependiendo la posición de la puerta
3. En la penúltima secuencia se activa la propiedad visible del botón shutdown, además se conecta el botón shutdown a la función “Case Structure” la cual contiene la función “Invoke Node” configurada para reiniciar al archivo VI a valores predeterminados y abrir el menú anterior mediante el sub-VI salto.
4. La última secuencia permita cerrar el archivo actual mediante el sub-VI cerrar.

2.8.4.1. Etapas de programación de la simulación del Proceso Garaje

Dentro de la segunda secuencia se desarrolla la simulación del proceso Garaje, todas las etapas de programación implementadas paso a paso se describe a continuación (**Figura 70**).

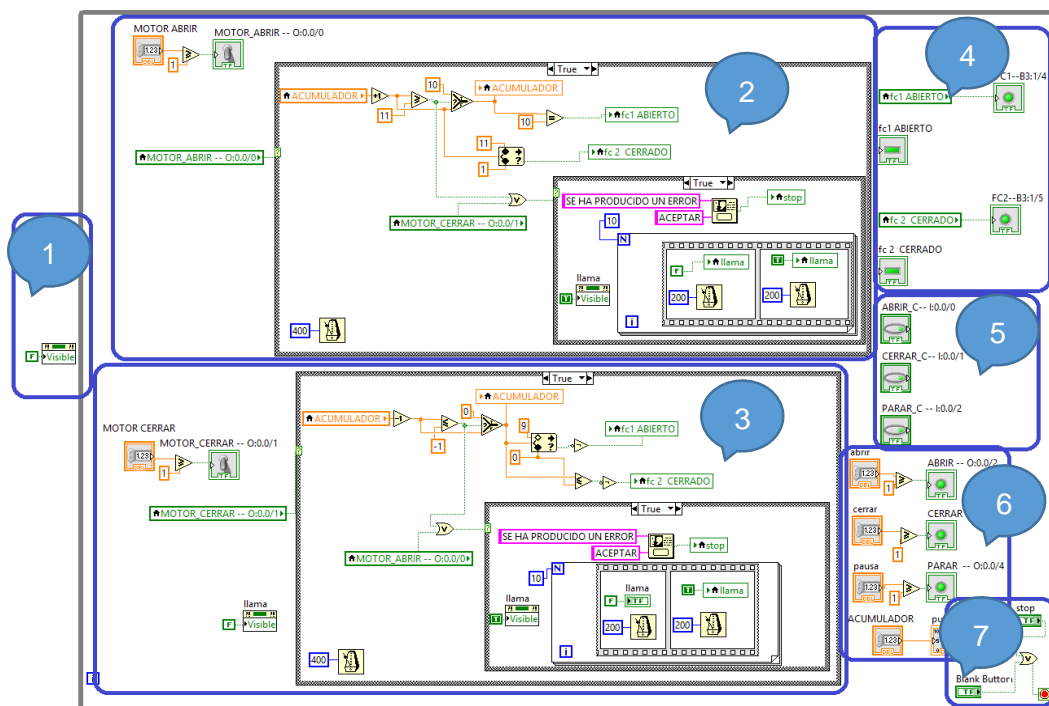


Figura 70: Diagrama de bloques Simulación del Proceso Garaje.

Descripción de las funciones de todas las etapas de programación de la simulación del Proceso Garaje

1. En la primera etapa se conecta una constante "false" a la propiedad visible para ocultar el botón shutdown.
2. Se compara si el control numérico motor_abrir es mayor o igual a 1 unidad para activar el indicador booleano, mediante una variable local se activa la función "Case Structure" en la cual se incrementa el acumulador el mismo que está conectado a la puerta y compara si es mayor o igual a 11 unidades para cargar con 10 unidades y si es igual a 10 unidades se activa el control fc1 abierto que simula un sensor. En un rango de 1 a 11 unidades se activa el control fc2 cerrado que simula un sensor. Si el acumulador es mayor o igual a 11 unidades o el control motor_cerrar está activado, se activa la función "Case Structure" donde se encuentra una simulación de fuego en una función "For Loop" y un mensaje de error, en esta función "case structure" se activa cuando la puerta llega al límite de abrir y el motor está activado o cuando está encendido el motor_abrir y a la vez se activa el motor _cerrar.

3. En la tercera etapa se compara si el control numérico motor_cerrar es mayor o igual a 1 unidad para activar el indicador booleano, mediante una variable local a la función “Case Structure” en la cual el acumulador disminuye su valor, este se encuentra conectado a la puerta y se compara si es menor o igual a -1 unidad para cargar con 0 unidades y si es igual a 0 unidades se activa el control fc2 cerrado que simula un sensor. En un rango de 0 a 9 unidades se activa el control fc1 abierto que simula otro sensor. Si el acumulador es menor o igual a -1 unidades o el control motor_abrir está activado, se procede a activarse la función “Case Structure” donde se encuentra una simulación de fuego en una función “For Loop” y un mensaje de error, la función “case structure” se activa cuando la puerta llega al límite de cerrar y el motor está activado o cuando está encendido el motor cerrar y a la vez se activa el motor abrir.
4. Aquí se configura los indicadores que indican los estados del sensor fin de carrera cuando la puerta del garaje se abra o se cierre.
5. En esta etapa se encuentran los indicadores de abrir, cerrar y parar, los cuales están configurados mediante la función “Data Binding” para la conexión entre el OPC, el simulador y el software de programación del PLC.
6. En la sexta etapa se compara si el control numérico es mayor o igual de uno para activar un control booleano. Conectar el control acumulador al tanque para apreciar el movimiento de la puerta según el valor del acumulador.
7. En la última etapa se compara la activación del botón stop o el botón reiniciar para activar el paro de la función “While Loop” general.

2.9. Desarrollo Simulación de un Proceso de Nivel

A continuación, se realiza la programación e implementación de la simulación del proceso Nivel. En primer lugar, se realiza el diagrama de flujo del proceso nivel (**Figura 71**).

2.9.1. Diagrama de Flujo del Proceso

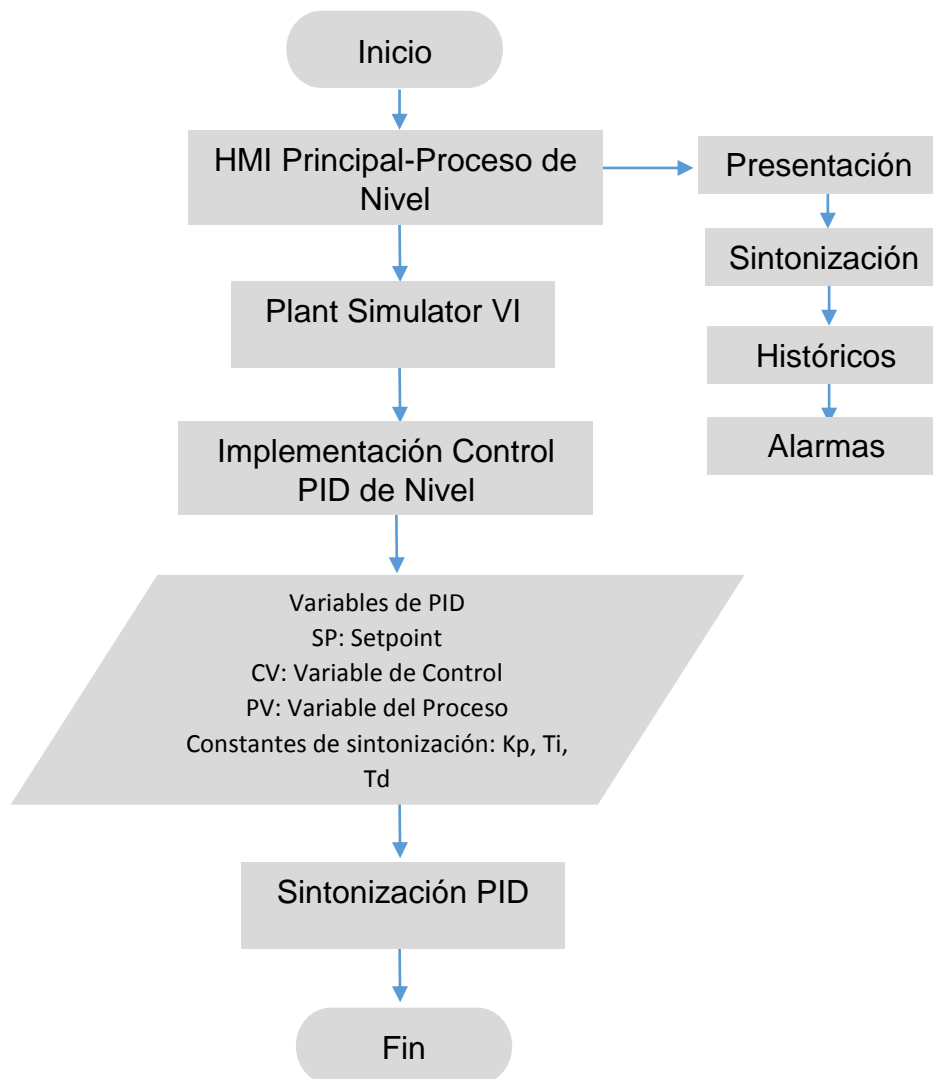


Figura 71: Diagrama de Flujo del Proceso Nivel

2.9.2. Descripción del Proceso

El siguiente proceso consta del control de nivel de un tanque de almacenamiento de agua, donde las unidades de medición de la variable están establecidas en cm. El rango de entrada es de 0 a 100 cm, el setpoint también posee un rango de 0 a 100 cm.

2.9.3. Panel Frontal del Proceso Nivel

El diseño del HMI del Proceso Nivel consta de un Tab control en donde se divide en 4 pestañas, cada una de ellas brindan información oportuna del proceso Nivel. El menú del proceso Nivel tiene las siguientes ventanas:

- Presentación del Proceso
- Sintonización de Proceso
- Históricos
- Alarmas

Cada una de estas ventanas se describen a continuación:

a. Presentación del Proceso Nivel

En la pestaña presentación se realiza la animación del proceso nivel, mediante el módulo DSC se extrae imágenes tales como tuberías, válvulas, bomba, tanque de almacenamiento, rotámetro, válvula proporcional, transmisor, indicador de tanque, incluye además los indicadores de los valores del setpoint (cm), Process Value (cm), Control Value (%). También existe un botón para regresar al menú anterior, un botón para iniciar la simulación y otro para detener el proceso (**Figura 72**).

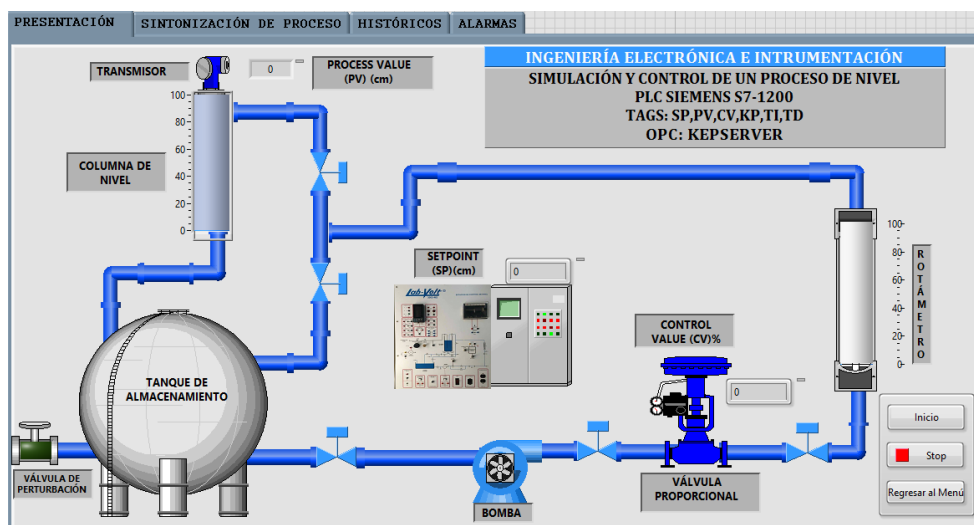


Figura 72: Presentación del Proceso Nivel

d. Alarmas

En la ventana Alarmas se encuentran las alarmas del proceso en forma de indicadores, además se visualizan los históricos de estas alarmas donde se muestra la fecha y la hora en que se activaron cada una de ellas. Las alarmas que se configuraron para este proceso son: High-High, High, Low, Low-Low (**Figura 75**).

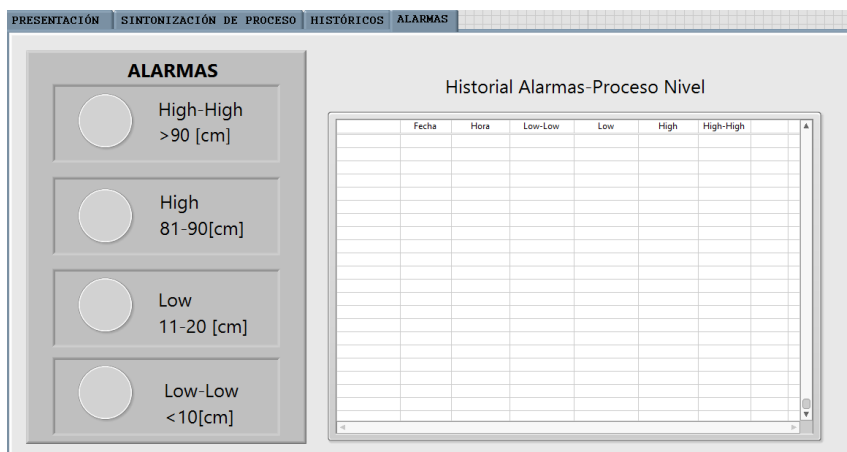


Figura 75: Ventana Alarmas

El panel frontal del proceso Nivel posee variables que se configuran mediante un servidor OPC las cuales se describen a continuación en la siguiente (**Tabla 10**).

Tabla 10

Variables del Panel Frontal del Proceso Nivel

Variable	Descripción
SP	Variable Setpoint
CV	Variable Control Value
PV	Variable Process Value
KP	Constante Proporcional
Ti	Tiempo Integral
Td	Tiempo Derivativo

2.9.4. Etapas de programación de la simulación del Proceso Nivel

El desarrollo de la simulación Nivel consta de varias etapas muy específicas que se describen a continuación (**Figura 76**):

1. En primer lugar se encuentra la animación de la interfaz gráfica de un proceso de nivel, contiene todos los elementos que componen un sistema de control como es un transmisor, un actuador, elemento de control final, un indicador de nivel y las variables del proceso.
2. La siguiente etapa es la simulación de la planta de proceso para cual se recurre a un subVi denominado PlantaSimulator.vi que es facilitado por National Instruments para el desarrollo de aprendizaje de control de procesos, este subVi simula la respuesta física de cualquier planta industrial.
3. En la tercera etapa se encuentran las variables del proceso las cuales mediante algún software de programación y un OPC serán configuradas para realizar un control PID.
4. En la penúltima etapa describe las alarmas que adquiere este proceso.
5. Finalmente la etapa final es el registro de los datos históricos que se guardarán durante la puesta en marcha de la simulación.

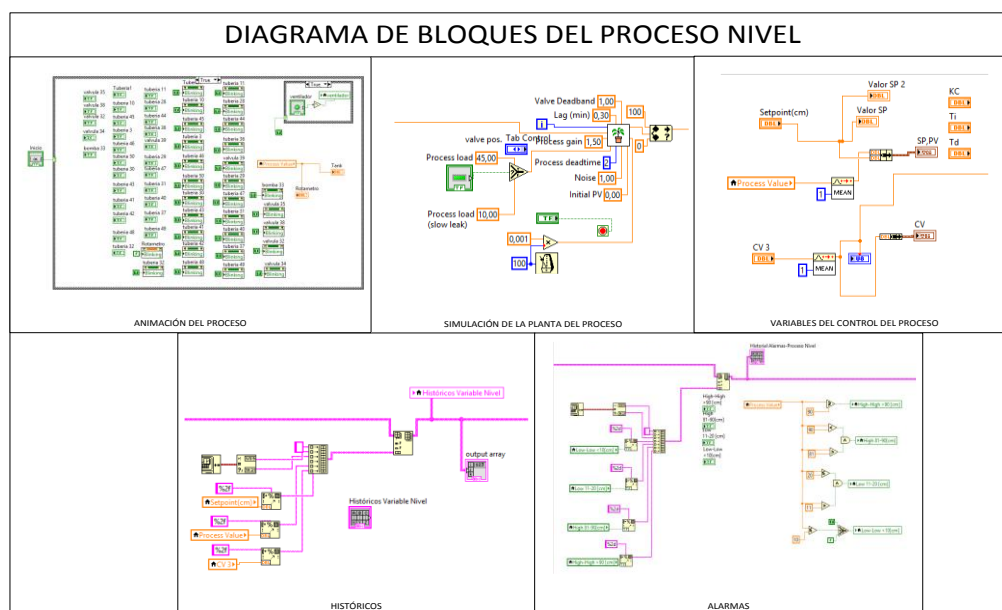


Figura 76: Diagrama de Bloque de Simulación Proceso Nivel

- **Descripción de las funciones de todas las etapas de programación de la simulación del Proceso Nivel**

El estructura total del programa se encuentra dentro de un lazo while. Aquí se dan todos los procesos de la simulación. El programa se irá ejecutando hasta que el usuario detenga el proceso mediante el botón Stop, dentro del diagrama de bloques se realiza la implementación de la planta de nivel. Las etapas de la simulación Nivel se describen a continuación:

a. Animación de la Interfaz Principal

En la primera etapa se realiza la animación de la interfaz principal se lo realiza mediante el Toolkit DSC de Labview el cual posee un sinnúmero de equipos e instrumentos de control de procesos para la simulación de un proceso de nivel se escogio válvulas, tuberías, rotámetro, tanque, luego en el diagrama de bloques se encuentran todas las variables booleanas de los elementos que se crearon en la pantalla principal, para cada una de estas variables se crea un property node denominado Blinking (**Figura 77**) el cual permite que la variable parpadee de un color cuando se le asigna una variable TRUE y de otro color cuando sea FALSE.

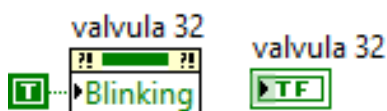


Figura 77: Función Blinking

b. Simulación de la Planta de Nivel

Para el desarrollo de la simulación de la planta de nivel se recurre al subVI "Plant Simulator.vi", el cual es proporcionado por National Instruments y se encuentra dentro de los ejemplos predeterminados que existen en labview, este subVI agrupa una serie de acciones como (**Figura 78**):

1. Deadband Simulator VI: Proporciona márgenes mínimos de cambio de salida de la planta

2. Lead-Lag VI: Este VI contiene un sistema de primer orden con función de transferencia estándar.
3. Shift Register Delay VI: Es un arreglo que permite dar retroalimentación al sistema.

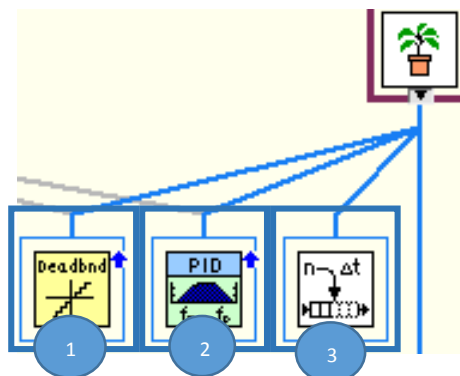


Figura 78: Elementos del SubVI Plant Simulator

- **Configuración subVI Plant Simulator**

Para este caso configuramos a la planta de la siguiente manera. Le introducimos un nivel inicial del tanque de 0. Un ruido de 1, que puede ser provocados por imperfecciones en la válvula de entrada, tubería entre otros. En el process deadtime le asignamos una constante de 2 que quiere decir que cuando la función while haya realizado la segunda iteración la planta entrará en funcionamiento, en process gain cuyo valor es 1.5 se encarga de multiplicar a la ganancia del sistema.

La variable iteración del subvi debe estar conectado a la función while del programa principal. El time Lag es la constante de tiempo en minutos del polo del sistema, y por último el deadband se coloca un valor de 1, este parámetro representa los márgenes mínimos de cambio de la salida de la planta. La salida no cambiará hasta que la entrada varíe el tanto por ciento indicado por deadband de su valor inicial (**Figura 79**).

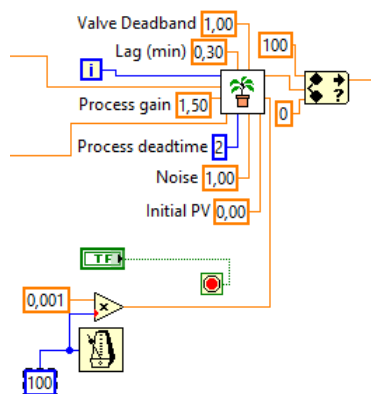


Figura 79: Configuración de subVi Plant Simulator

c. Tendencias del Control PID de Nivel

En esta etapa se crean todas las variables necesarias para el desarrollo del control pid, el cual va ser puesto a prueba a la planta de nivel. Tales como K_p , T_i , T_d , CV , PV , SP , la variable CV va conectado a la entrada de la planta para realimentar al sistema y así crear un lazo cerrado del proceso. A estas variables mediante Data Binding se les asocia las tags que se crea en el servidor OPC. Aquí también se crea un Waveform Chart para realizar la gráfica de las tendencia del proceso.

d. Registro de Históricos

En la cuarta etapa se crea un arreglo mediante Build Array con algunas variables como son fecha y hora, los valores de SP , CV , PV y se asocia a una tabla .

e. Registro de Alarmas

Primeramente se realiza la limitación de cada uno de los niveles de alarma, tanto para alarma Low, Low-Low, High, High-High y se asocia a cada nivel un indicador led con un color diferente. Luego para el registro de alarmas se crea un arreglo mediante Build Array que contiene fecha y hora, valores de encendido de las diferentes alarmas y finalmente se crea una tabla para organizar el arreglo anteriormente mencionado.

2.10. Desarrollo Simulación de un Proceso de Flujo

A continuación, se realiza la programación e implementación de la simulación del proceso Flujo. Como primer paso se realiza el diagrama de flujo del Proceso (**Figura 80**).

2.10.1. Diagrama de Flujo del Proceso

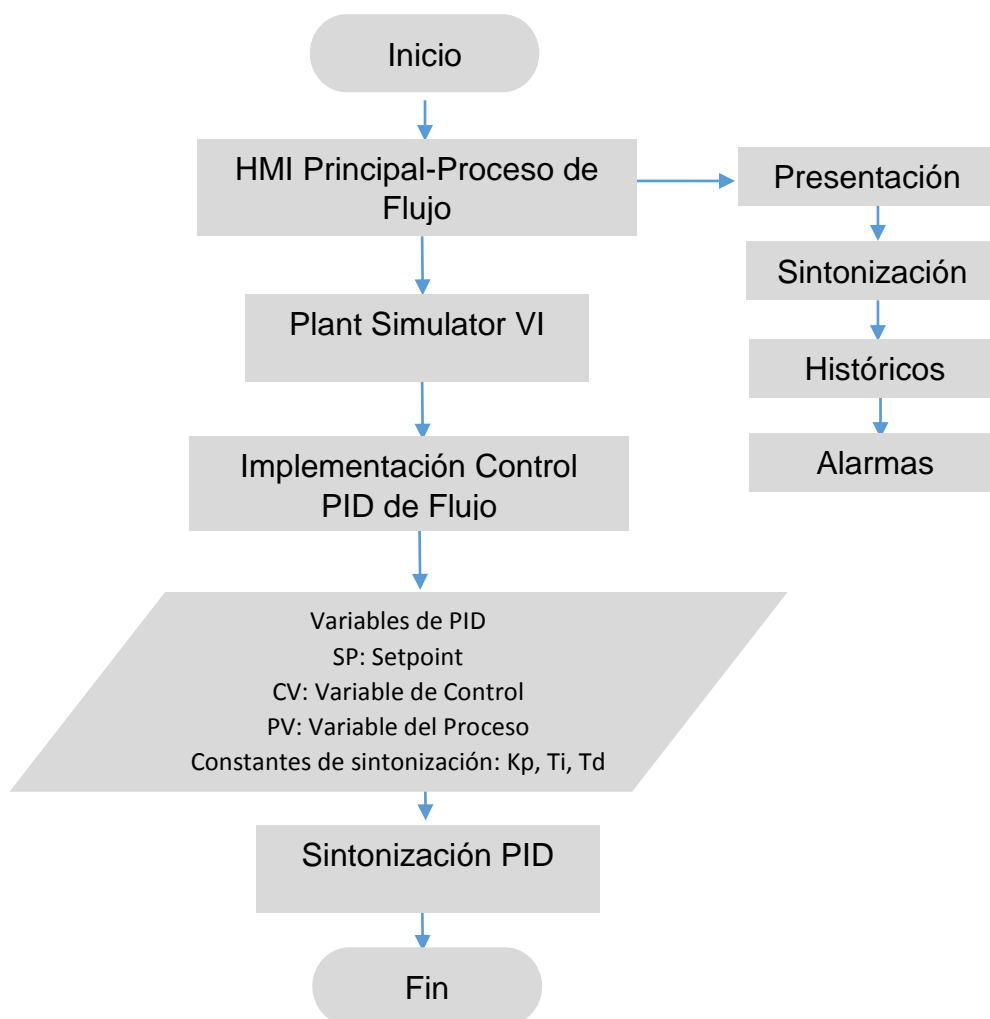


Figura 80: Diagrama de Flujo del Proceso Flujo

2.10.2. Descripción del Proceso

El siguiente proceso consta del control de flujo de agua hacia un tanque de almacenamiento, donde las unidades de medición de la variable están establecidas en GPM. El rango de entrada es de 0 a 80 GPM, el setpoint también posee un rango de 0 a 80 GPM.

2.10.3. Panel Frontal del Proceso Flujo

El diseño del HMI del Proceso Flujo está compuesto de un Tab control en donde se divide en 4 pestañas. El menú del proceso Flujo tiene las siguientes ventanas:

- Presentación del Proceso
- Sintonización de Proceso
- Históricos
- Alarmas

a. Presentación del Proceso Flujo

En la ventana presentación se desarrolla la animación del proceso flujo, mediante el módulo DSC se extrae imágenes tales como tuberías, válvulas, bomba, tanque, rotámetro, transmisor, indicador de tanque, incluye además los indicadores de los valores del setpoint (GPM), Process Value (GPM), Control Value (%). También existe un botón para regresar al menú anterior, un botón para iniciar la simulación y otro para detener el proceso (**Figura 81**).

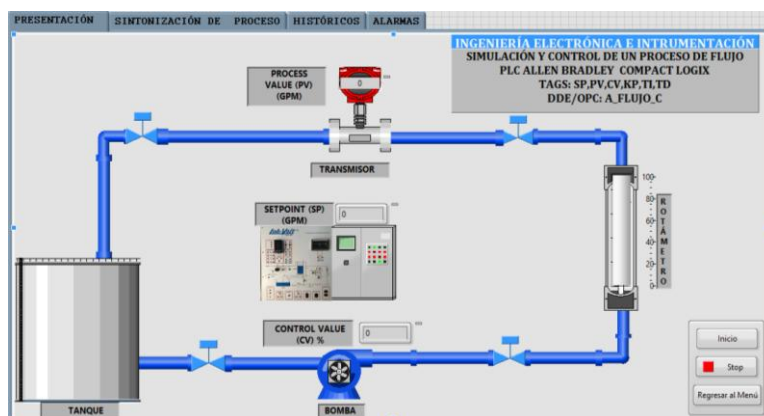


Figura 81: Presentación del Proceso Flujo

b. Sintonización de Proceso

En esta pestaña se realiza la sintonización del controlador PID, contiene los valores del proceso (SP, PV, CV), las constantes de sintonización las cuales son la constante proporcional (K_p), el tiempo integral (T_i) y el tiempo derivativo (T_d), además está la gráfica de tendencias del proceso. (**Figura 82**)

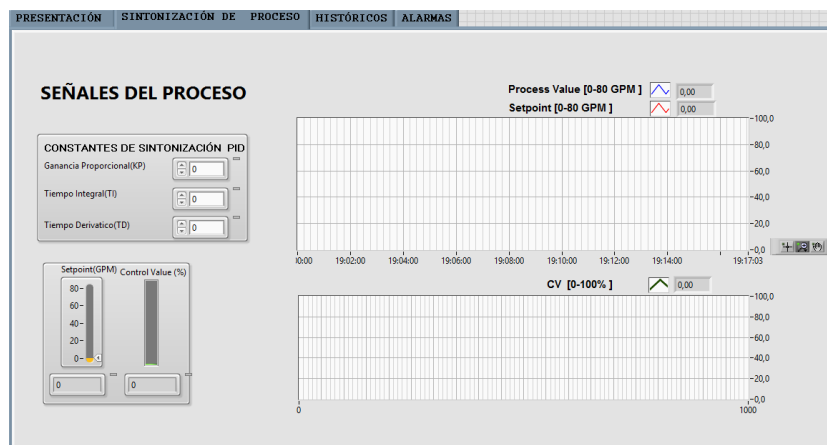


Figura 82: Ventana Sintonización de Proceso Flujo

c. Históricos

La ventana Históricos presenta una tabla de histórico general de todas las variables que interviene en el proceso, además la Hora y Fecha (**Figura 83**).

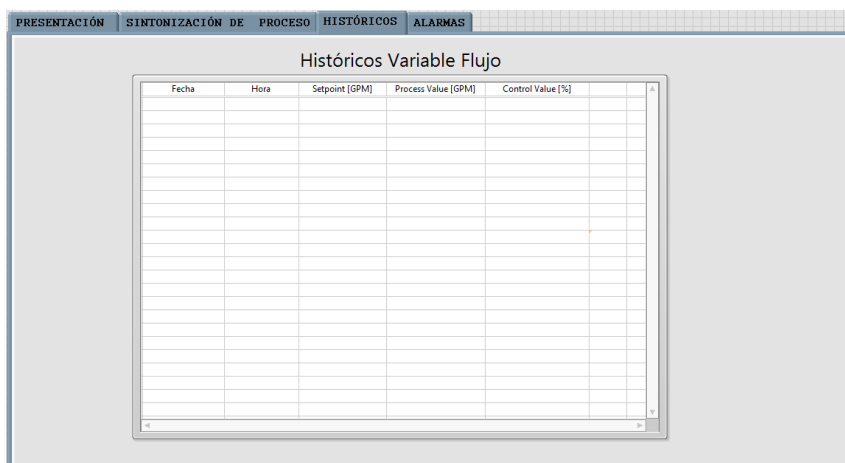


Figura 83: Ventana Históricos de Proceso Flujo

d. Alarmas

En la ventana Alarmas en encuentran las alarmas del proceso en formas de indicadores, además se visualizan los históricos de estas alarmas donde se muestra la fecha y la hora en que se activaron cada una de ellas (**Figura 84**).

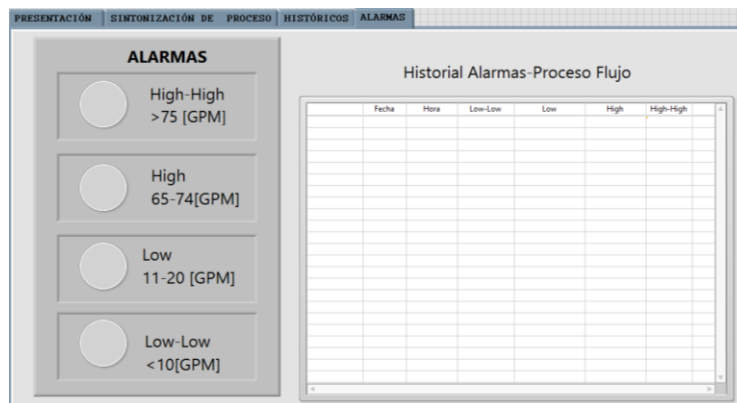


Figura 84: Ventana Alarmas de Proceso Flujo

El panel frontal del proceso Flujo posee variables que se configuran mediante un servidor OPC las cuales se describen a continuación en la siguiente **(Tabla 11)**.

Tabla 11

Variables del Panel Frontal del Proceso Flujo

Variable	Descripción
SP	Variable Setpoint
CV	Variable Control Value
PV	Variable Process Value
KP	Constante Proporcional
Ti	Tiempo Integral
Td	Tiempo Derivativo

2.10.4. Etapas de programación de la simulación del Proceso Flujo

El desarrollo de la simulación Flujo consta de varias etapas muy específicas que se describen a continuación **(Figura 85)**:

1. En la primera etapa se encuentra la animación de la interfaz gráfica de un proceso de flujo.
2. La siguiente etapa es la simulación de la planta de proceso flujo para cual se recurre a un subVi denominado PlantaSimulator.vi .

3. En la tercera etapa se encuentran las variables del proceso las cuales mediante algún software de programación y un OPC serán configuradas para realizar un control PID.
4. En la penúltima etapa describe las alarmas que adquiere este proceso.
5. La etapa final es el registro de los datos históricos que se guardarán durante la puesta en marcha de la simulación.

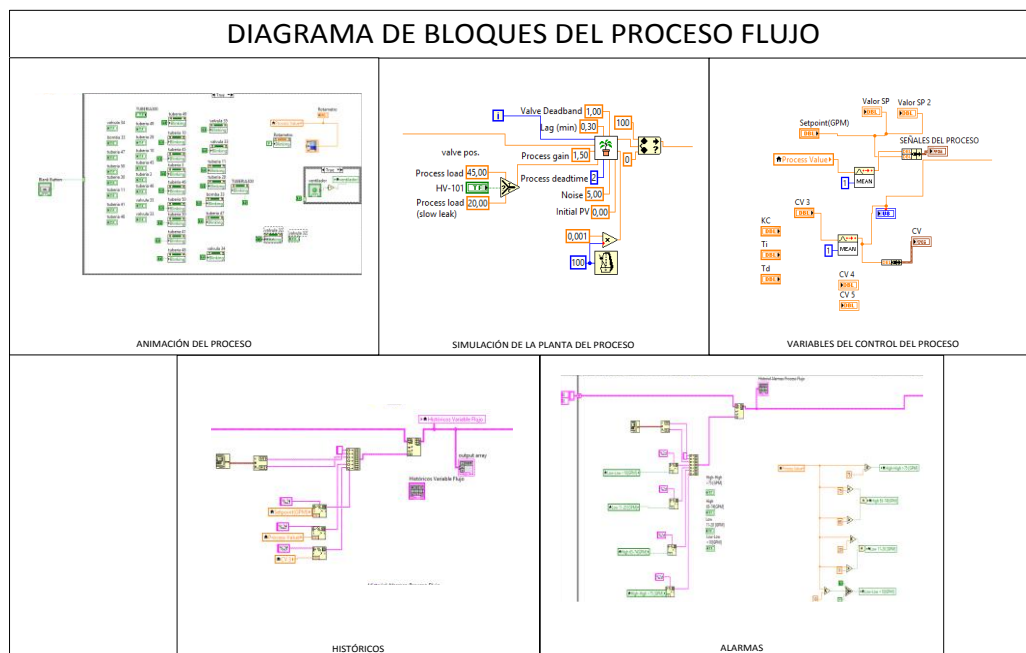


Figura 85: Diagrama de Bloques de Simulación Proceso Flujo

- **Descripción de las funciones de todas las etapas de programación de la simulación del Proceso Flujo**

Dentro del diagrama de bloques se realiza la implementación de la planta de Flujo. Las etapas de la simulación Flujo se describen a continuación:

a. Animación de la Interfaz Principal

En la primera etapa se realiza la animación de la interfaz principal se lo realiza mediante el Toolkit DSC de Labview para el proceso de flujo se escogió válvulas, tuberías, rotámetro, tanque.

b. Simulación de la Planta de Flujo

Para el desarrollo de la simulación de la planta de Flujo se recurre al subVI “Plant Simulator.vi”.

- **Configuración subVI Plant Simulator**

Para este caso configuramos a la planta de flujo de la siguiente manera. Le introducimos un valor inicial del tanque de 0. Un ruido de 5, que puede ser provocados por perturbaciones del sistema. En el process deadtime le asignamos una constante de 2, en process gain cuyo valor es 1.5 se encarga de multiplicar a la ganancia del sistema. La variable iteracion del subvi debe estar conectado a la funcion while del programa principal. El time Lag es la constante de tiempo en minutos del polo del sistema, y por ultimo el deadband se coloca un valor de 1, este parametro representa los márgenes mínimos de cambio de la salida de la planta (**Figura 86**).

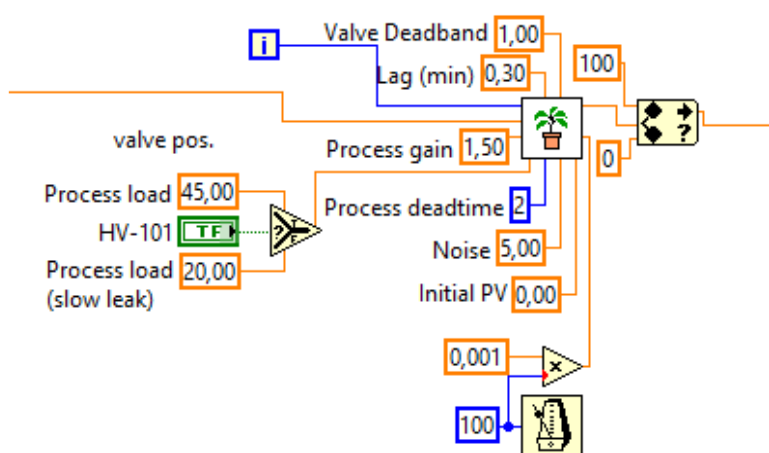


Figura 86: Configuración de subVi Plant Simulator

c. Tendencias del Control PID de Flujo

En esta etapa se crean todas las variables necesarias para el desarrollo del control pid de la planta de Flujo. Tales como K_p , T_i , T_d , C_V , P_V , S_P , la variable C_V va conectado a la entrada de la planta para realimentar al sistema y así crear un lazo cerrado del proceso.

d. Registro de Históricos

En esta etapa se crea un arreglo mediante Build Array con algunas variables como son fecha y hora, los valores de SP, CV, PV y se asocia a una tabla.

e. Registro de Alarmas

Primeramente se realiza la limitacion de cada uno de los niveles de alarma, tanto para alarma Low, Low-Low, High, High-High y se asocia a cada nivel un indicador led con un color diferente. Luego para el registro de alarmas se crea un arreglo mediante Build Array que contiene fecha y hora, valores de encendido de las diferentes alarmas y finalmente se crea una tabla para organizar el arreglo anteriormente mencionado.

2.11. Desarrollo de los SubVI Salto y Cerrar

2.11.1. Desarrollo del sub-VI Salto

Se representa con un icono sub-VI de salto y la función es abrir con un archivo VI que sea requerido por un menú o función en especial (**Figura 87**).



Figura 87: Apariencia del icono del Sub-VI salto

En el siguiente Diagrama de bloques del archivo VI salto (**Figura 88**) se explica las etapas del subvi Salto.

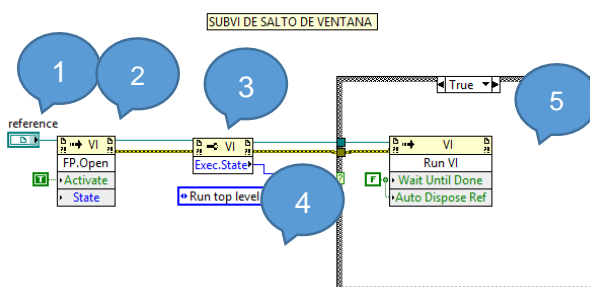


Figura 88: Diagrama de bloques de Sub-VI salto

El diagrama de bloque del sub-VI salto se encuentra constituido por las siguientes funciones.

1. La Función “Reference” encargada de recibir al Sub-VI conectado.
2. Función “Invoke Node” para abrir el Sub-VI recibido, configurar de la siguiente manera.
3. Función “Property Node” para ejecutar el archivo VI.
4. La Función “Not Equal?” sirve para comparar si el archivo VI no se encuentra en ejecución.
5. Función “Invoke Node” contenida en la función “Case Structure” para ejecutar el archivo VI.

2.11.2. Desarrollo del subVI Cerrar

Se representa con un icono sub-VI de cerrar archivo VI (**Figura 89**).

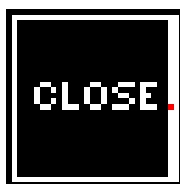


Figura 89: Apariencia del icono del Sub-VI Cerrar

El diagrama de bloque del sub-VI salto se encuentra constituido por la siguiente función (**Figura 90**). Esta función permite cerrar el archivo Vi del cual es referenciado mediante la instrucción FP. Close.

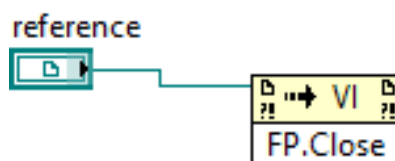


Figura 90: Función Cerrar VI

2.12. Configuración y Programación de un PLC SIEMENS S7-1200

Para configurar un S7-1200 es necesario utilizar el paquete de software TIA PORTAL (Total Integrated Automation), el uso de este software es sencillo ya que es muy intuitivo para el usuario, además es una gran solución para desarrollar proyectos de automatización de cualquier índole.

2.12.1. Creación de un nuevo proyecto en TIA PORTAL

En TIA PORTAL un nuevo proyecto guardara toda la información de programación, configuración, variables, datos dispositivos que estén asociados al PLC, etc. Siempre que se desarrolle un nuevo proyecto en el TIA PORTAL es necesario cumplir con los siguientes pasos que se detallan a continuación:

1. Abrir el paquete de software TIA PORTAL.
2. Aparece una ventana (**Figura 91**) con varias opciones, para crear un nuevo proyecto seleccionar “Crear Proyecto”, luego colocar un nombre para el nuevo proyecto y la dirección donde desee que se guarde el mismo, para finalizar dar clic en “Crear”.

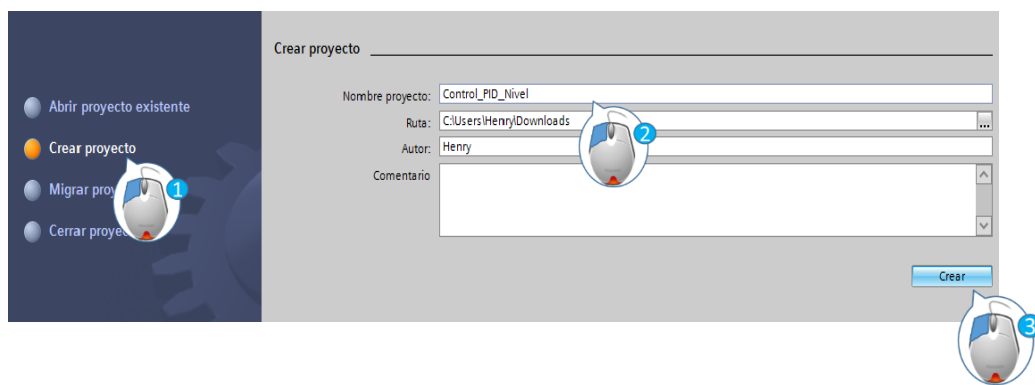


Figura 91: Ventana para crear un nuevo proyecto

3. A continuación, se va configurar el dispositivo con el cual se va a trabajar, en este caso un PLC Siemens S7-1200.

2.12.2. Agregar y Configurar un nuevo dispositivo

Para agregar un nuevo dispositivo es importante seguir los siguientes pasos:

1. Primeramente, seleccionar la opción “Agregar Dispositivo” (**Figura 92**).

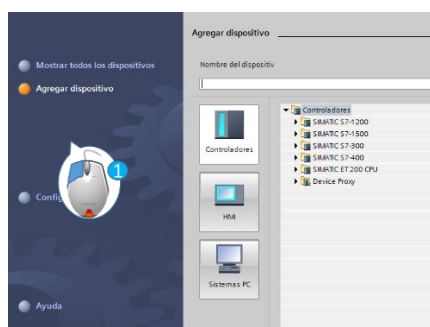


Figura 92: Agregar un nuevo dispositivo

2. Seleccionar el controlador con el cual se va a trabajar, para este caso es el PLC Siemens S7-1200, cuyo CPU es 1214C AC/DC/Rly, finalmente dar clic en “Agregar” (**Figura 93**).

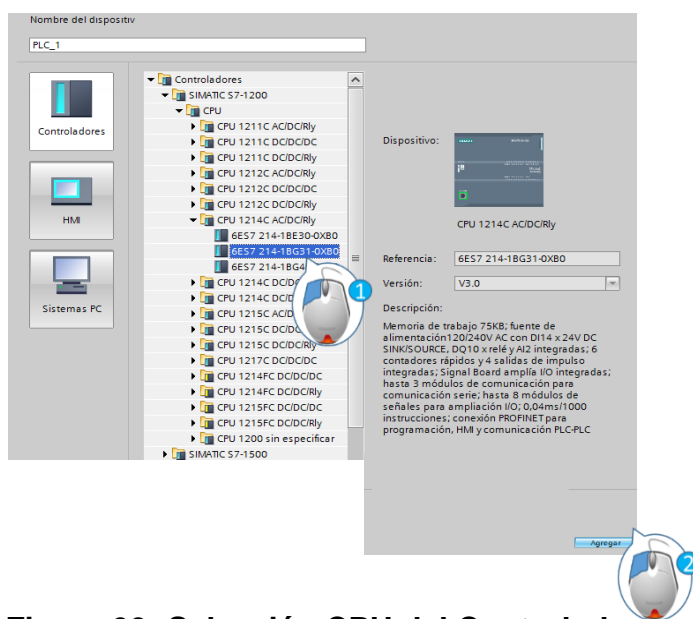


Figura 93: Selección CPU del Controlador

2.12.3. Configuración de la interfaz Profinet del controlador S7-1200

1. Abrir las “Propiedades” del controlador para configurar la interfaz Profinet del mismo.
2. Seleccionar la opción “Direcciones Ethernet”, en “Protocolo IP” ingresar la dirección IP que poseerá el controlador (**Figura 94**).

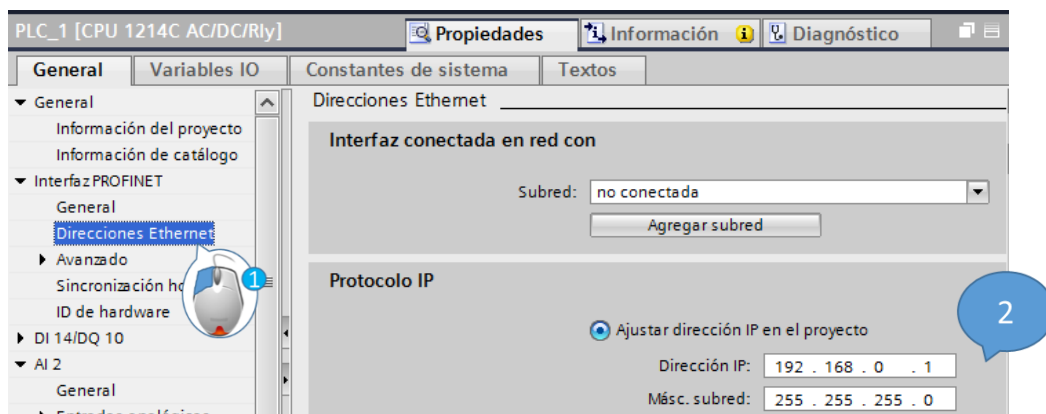


Figura 94: Asignación de dirección IP

2.12.4. Cargar un programa en la CPU del PLC S7-1200

Una vez terminada toda la programación se procede a compilar y cargar el programa. A continuación, se describe el proceso para cargar el programa.

1. Primero, “Compilar” el programa para verificar que no exista errores.
2. En la Barra de Menú del TIA PORTAL seleccionar la opción “Online”, luego “Cargar en dispositivo”.
3. Aparecerá una ventana (**Figura 95**) de configuración para detectar al PLC donde se va cargar el programa, luego de seleccionar correctamente el PLC presionar la opción “Cargar”.

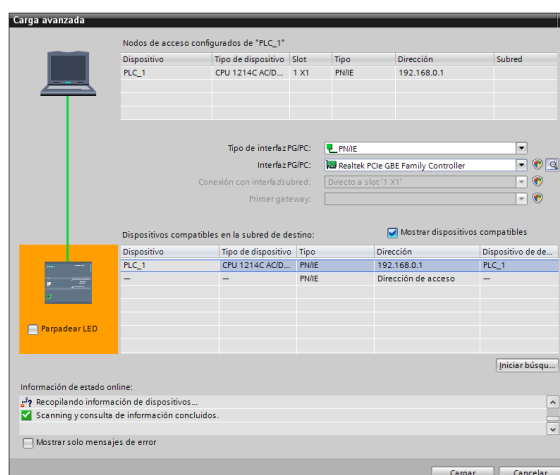


Figura 95: Configuración Avanzada para cargar un programa en el PLC

4. Finalmente “Establecer conexión online” para verificar el correcto funcionamiento del programa.

2.13. Configuración y Programación del PLC Allen Bradley Micrologix 1200

Para configurar un PLC Allen Bradley Micrologix 1200 se utiliza el paquete de software RSLogix 500, así como también RSLinx Classic los cuales permiten el correcto desarrollo de proyectos de automatización. Los pasos que se debe seguir para configurar correctamente un PLC Allen Bradley Micrologix 1200 son los siguientes:

2.13.1. Configurar RSLinx para PLC Micrologix 1200

1. Abrir “RSLinx Classic”, en el menú de opciones seleccionar “Communications”, luego “Configure Drivers”, en la ventana que aparece (**Figura 96**) seleccionar RS-232 DF1, dar clic en Add New. Aparece una ventana en la cual se debe colocar un nombre para el driver de comunicación. Por último, clic en OK.

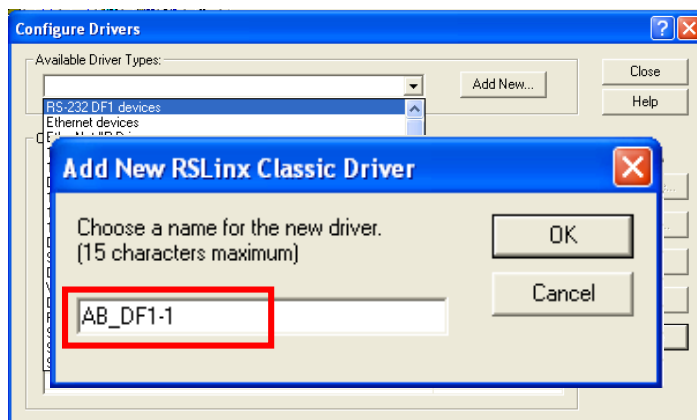


Figura 96: Configuración RSLinx para PLC Micrologix 1200

2. A continuación, aparece otra ventana (**Figura 97**) aquí seleccionar el puerto COM donde se encuentra conectado el PLC, luego dar clic en “Auto-Configure”, finalmente clic en OK.

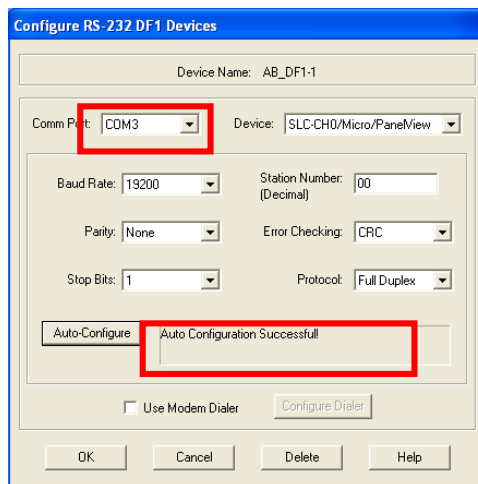


Figura 97: Configuración de comunicación entre RSLinx y PLC

2.13.2. Configurar RSLogix 500 para PLC Micrologix 1200

Para desarrollar la programación del PLC Micrologix 1200 es necesario el paquete RSLogix 500, el procedimiento se describe a continuación:

1. Primero Ejecutar el RSLogix 500, seguido a esto en el menú de opciones escoger Nuevo, en la ventana que aparece seleccionar el Tipo de controlador, como se muestra en la figura es el “Micrologix 1200 Series C (1 or 2 comm Ports)” (**Figura 98**), verificar que el driver de comunicación sea el que se cree anteriormente mediante RSLinx en este caso es AB_DF1-1, finalmente dar clic en OK.

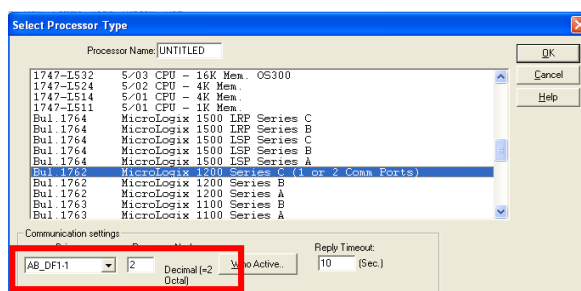


Figura 98: Seleccionar el Tipo de Controlador en RSLogix 500

2. Luego en el árbol de proyecto seleccionar “Controller Properties”, luego ir a la opción “Controller Communications”, ahí escoger el driver AB_DF1-1 que anteriormente fue creado en RSLinx como se muestra a continuación (**Figura 99**).

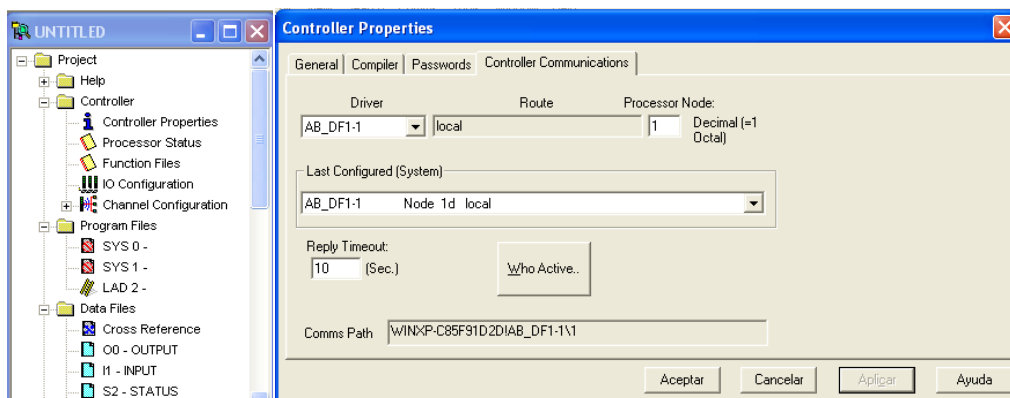


Figura 99: Propiedades del Controlador

2.15. Configuración y Programación del PLC Allen Bradley Micrologix Emulate

Para configurar un PLC Allen Bradley Micrologix Emulate se utiliza el software RSLogix 500, RSEmulate 500, además RSLinx Classic estos programas permiten el desarrollo de proyectos de automatización. Los pasos que se debe seguir para configurar correctamente un PLC Allen Bradley Micrologix Emulate son los siguientes:

2.15.1. Configurar RSLinx para PLC Micrologix Emulate

1. Abrir "RSLinx Classic", en el menú de opciones seleccionar "Communications", luego "Configure Drivers", en la ventana que aparece seleccionar "SLC 500(DH485) Emulate Driver", dar clic en Add New. Aparece una ventana en cual se debe colocar un nombre para el driver de comunicación. Por último, clic en **OK. (Figura 100)**

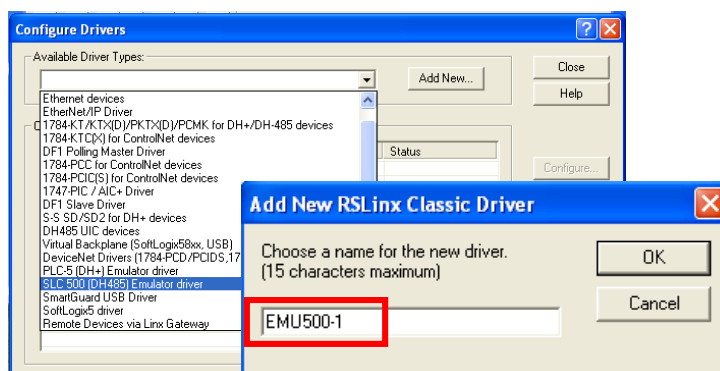


Figura 100: Configuración RSLinx para Micrologix Emulate

2. Luego aparece una ventana (**Figura 101**) para Configurar el driver del emulador aquí se debe seleccionar el número de estación y colocar un nombre para la misma.

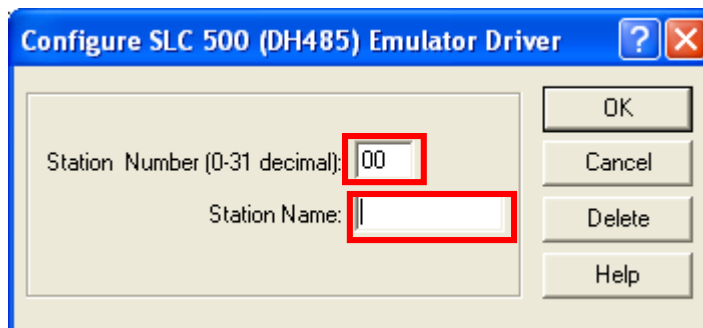


Figura 101: Configuración driver del emulador SLC 500

2.15.2. Configurar RSLogix 500 para PLC Micrologix Emulate

Al igual que en un PLC Micrologix Físico Para desarrollar la programación del PLC Micrologix Emulate se la realiza a través del paquete RSLogix 500, el procedimiento se describe a continuación:

1. Abrir el RSLogix 500, luego en el menú de opciones escoger Nuevo, en la ventana (**Figura 102**) que aparece seleccionar el Tipo de controlador, como se muestra en la figura es el “Micrologix 1200 Series C (1 or 2 comm Ports)”, verificar que el driver de comunicación sea el que se cree anteriormente mediante RSLinx en este caso es EMU500-1, finalmente dar clic en OK.

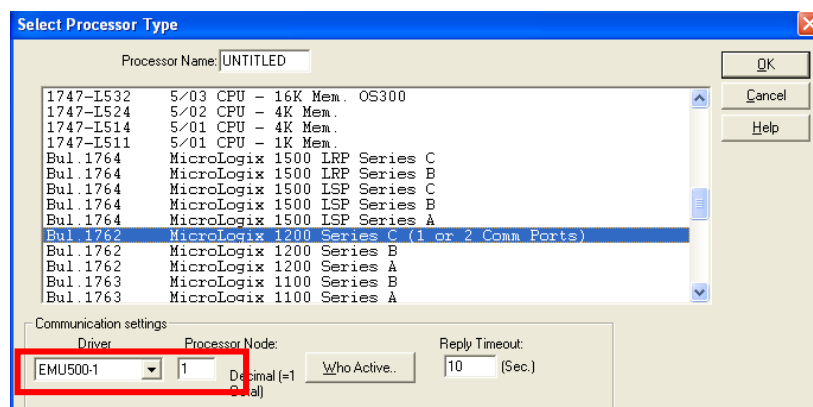


Figura 102: Seleccionar el Tipo de Controlador en RSLogix 500

2. En el árbol de proyecto seleccionar “Controller Properties”, luego ir a la opción “Controller Communications”, ahí escoger el driver EMU500-1 que anteriormente fue creado en RSLinx (**Figura 103**).

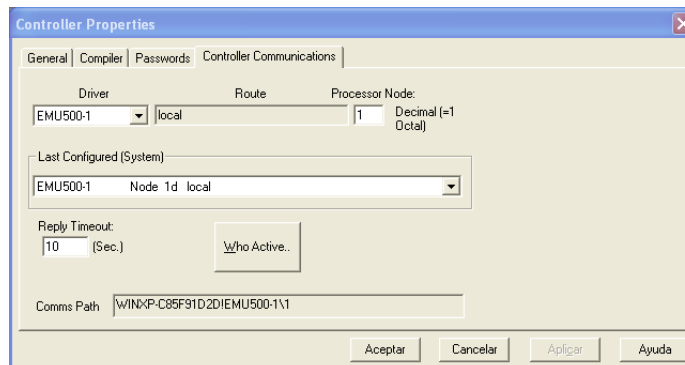


Figura 103: Configuración del Controlador en RSLogix

2.15.3. Configurar RSLogix Emulate 500

Para emular un PLC Micrologix se utiliza el paquete RSLogix Emulate 500 en donde se puede emular el PLC a continuación, se describe el procedimiento para configurar el RS Emulate.

1. Ejecutar RSLogix Emulate 500, luego buscar la opción Abrir, ahí se debe seleccionar el archivo o programa que fue desarrollado en RSLogix 500 (**Figura 104**).



Figura 104: Abrir proyecto en RSLogix Emulate 500

En la siguiente ventana se debe colocar el número de estación para establecer conexión de este programa con RSLogix 500 y el RSLinx. Se

observa que el archivo se encuentra configurado de manera correcta, finalmente seleccionamos RUN para poner en marcha el emulador (**Figura 105**).

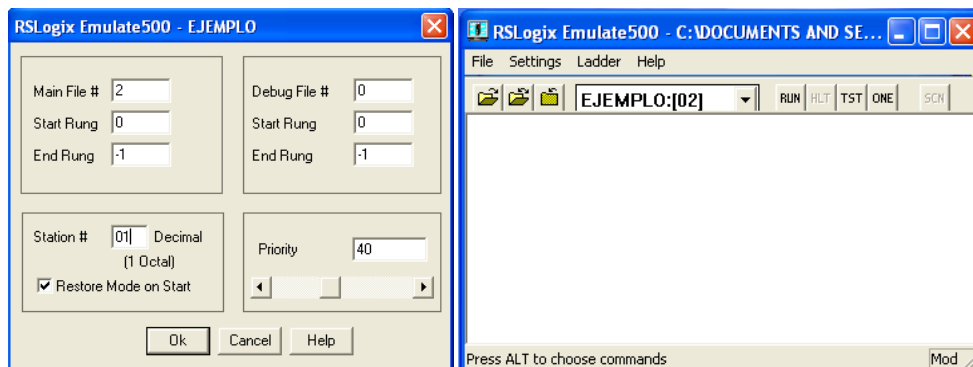


Figura 105: Configuración RSEmulate 500

2.15.4. Cargar programa en RSLogix 500

Luego de finalizar la programación se procede a compilar y cargar el programa. El proceso para cargar el programa en RSLogix 500 es el mismo tanto para el PLC Micrologix 1200 como para el Micrologix Emulate este proceso se describe a continuación.

1. Primeramente, se debe compilar el programa para verificar que no exista algún error. Luego en la barra de Menú seleccionar Comms, escoger la opción “System Comms” en la ventana que aparece seleccionar el PLC Micrologix 1200 ya sea físico como emulado, finalmente dar clic en “Download” (**Figura 106**).

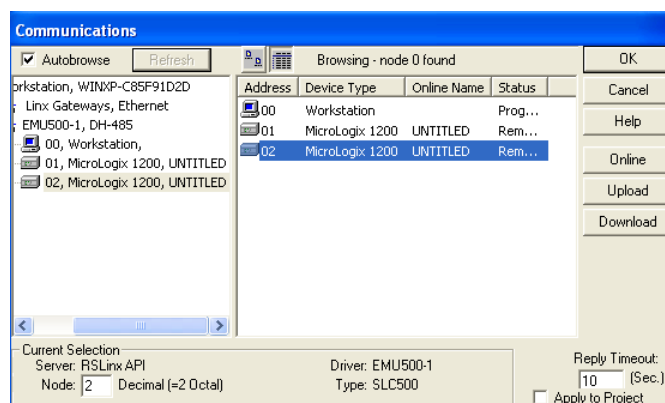


Figura 106: Cargar un programa en RSLogix

2. Luego escoger la opción Run para ejecutar el programa, aparecerá una advertencia dar clic en Si para ir a el modo online.

2.16. Configuración y Programación del PLC Allen Bradley Compact Logix 1769-L32E

Para configurar un PLC Allen Bradley CompactLogix 1769-L32E se utiliza el paquete de software RSLogix 5000, así como también RSLinx Classic los cuales de manera conjunta permitirán el correcto desarrollo de proyectos de automatización, como es el caso del simulador de procesos industriales. Los pasos que se debe seguir para configurar correctamente un PLC Allen Bradley CompactLogix son los siguientes:

2.16.1. Configurar RSLinx Classic para PLC Compact Logix 1769-L32E

1. Primero Abrir “RSLinx Classic”, en el menú de opciones seleccionar “Communications”, luego Configure Drivers, en la ventana (**Figura 107**) que aparece seleccionar Ethernet/IP Driver, por ultimo dar clic en Add New.

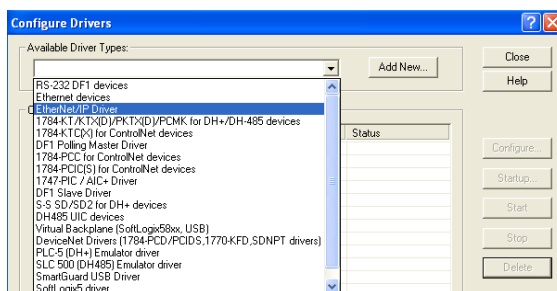


Figura 107: Configuración RSLinx Classic

2. A continuación, aparece una ventana en cual se debe colocar un nombre para el driver de comunicación, se puede dejar el nombre por defecto. Por último, clic en OK.
3. En la ventana (**Figura 108**) que aparece seleccionar Browser Remote Subnet, en IP Address colocar la dirección IP que posee el PLC y su máscara. Finalmente, clic en Aceptar.

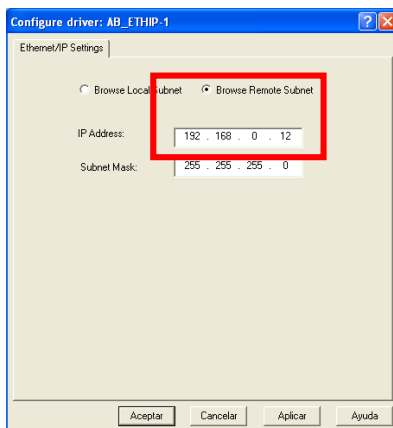


Figura 108: Configuración del nuevo driver instalado

2.16.2. Configurar RSLogix 5000 para PLC CompactLogix 1769-L32E

La programación del PLC CompactLogix se realiza a través del paquete RSLogix 5000, es necesario seguir los siguientes pasos:

1. Abrir el RSLogix 5000, en el menú de opciones escoger Nuevo, en la ventana (**Figura 109**) que aparece seleccionar el Tipo de controlador, como se muestra en la figura es el 1769-L32E, además colocar un nombre y por último clic en OK.

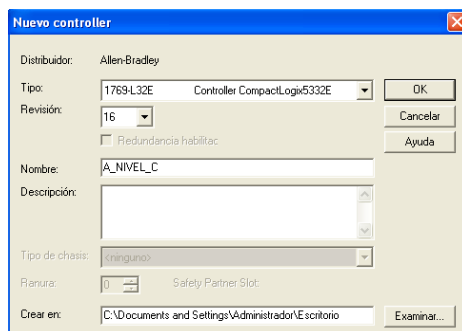


Figura 109: Configuración nuevo proyecto RSLogix 5000

2.17. Configuración y Programación del PLC Allen Bradley CompactLogix Emulate

Para configurar un PLC Allen Bradley Compact Emulate se utiliza el paquete de software RSLogix 5000, RSEmulate 5000, además el RSLinx Classic los cuales permiten el desarrollo de proyectos de automatización, en este caso del simulador de procesos industriales. Los pasos los siguientes:

2.17.1. Configurar RSEmulate 5000

Para emular un PLC CompactLogix es necesario utilizar el paquete RSEmulate 5000 en donde se puede configurar y emular el procesador del PLC, además simular entradas y salidas digitales.

a. Crear módulo de procesador del PLC

1. Abrir el RSEmulate 5000, en la ventana **(Figura 110)** que aparece se van a crear y configurar módulos, el programa tiene la capacidad de abarcar 16 módulos o slots, el Módulo 0 está establecido por default para la comunicación entre el RSLinx con el RSLogix 5000.

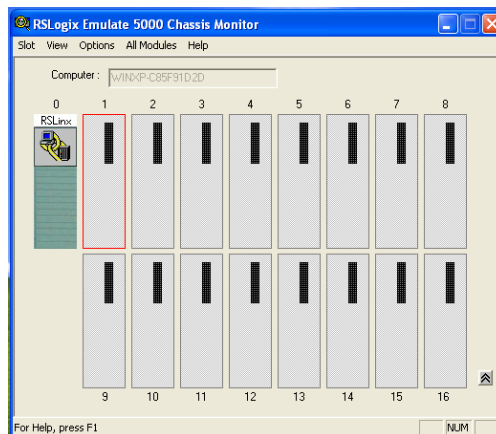


Figura 110: RSLogix Emulate 5000

2. Para crear un nuevo módulo es necesario dar clic en Create en el slot 1, se muestra una ventana **(Figura 111)** en la que se debe escoger el tipo de modulo a crear en este caso el Emulador RSLogix Emulate 5000 Controller.

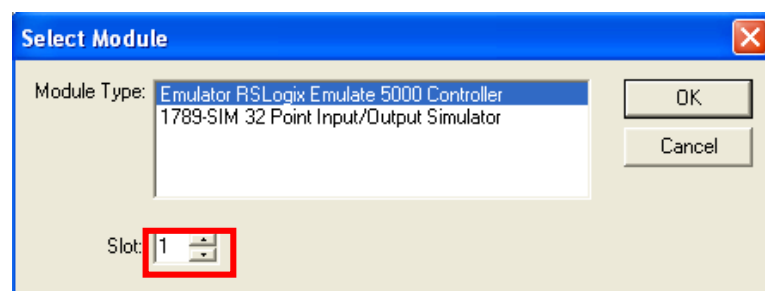


Figura 111: Añadir un nuevo módulo

b. Crear módulo de entradas y salidas

El proceso de configuración de módulo de entradas y salidas en RSEmulate se describe a continuación. En el slot número 2 se debe dar clic en Create, después de esto escoger el tipo de módulo en este caso el 1789-SIM 32 Point Input/Output Simulator para simular entradas/salidas. Luego aparece una ventana donde se debe colocar el nombre para el módulo (**Figura 112**).

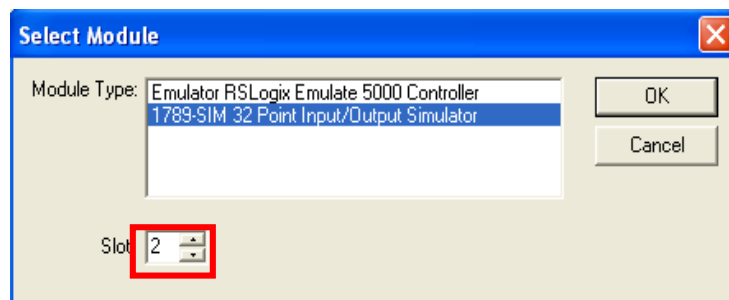


Figura 112: Módulo de entradas y salidas

2.17.2. Configurar RSLinx Classic para PLC Compact Logix Emulate

Abrir “RSLinx Classic”, en el menú de opciones seleccionar “Communications”, luego Configure Drivers, en la ventana (**Figura 113**) que aparece seleccionar “Virtual Backplane”, dar clic en Add New. Luego, aparece una ventana en cual se debe colocar un nombre para el driver de comunicación, se puede dejar el nombre por defecto. Por último, clic en OK.

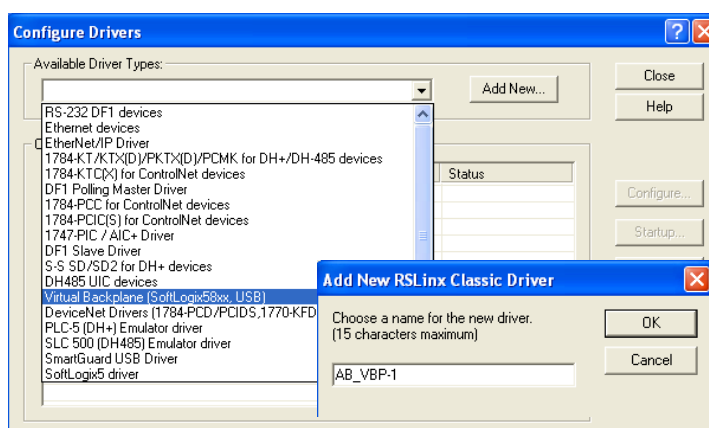


Figura 113: Configuración RSLinx

2.17.3. Configurar RSLogix 5000 para PLC CompactLogix Emulate

Al igual que en un PLC Físico, la programación del mismo se la realiza a través del paquete RSLogix 5000, para lo cual es necesario seguir los siguientes pasos:

1. Abrir el RSLogix 5000, en el menú de opciones escoger Nuevo, en la ventana que aparece seleccionar el Tipo de controlador, como se muestra en la **(Figura 114)** es el tipo Emulador Controller RSLogix Emulate 5000, luego colocar un nombre además se debe verificar que la ranura sea la misma que se configuro anteriormente y por ultimo clic en OK.

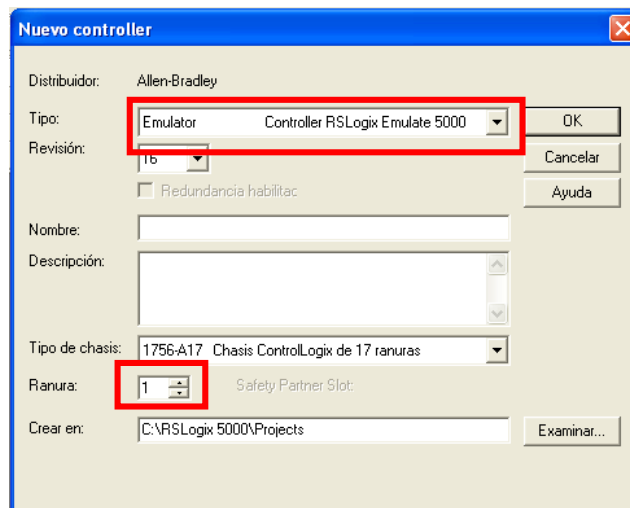


Figura 114: Configuración de un nuevo proyecto en RSLogix 5000

a. Configurar Módulo de Entradas y Salidas en RSLogix 5000

El siguiente procedimiento es para realizar la configuración del módulo de entradas y salidas.

1. Primeramente, en la ventana “Organizador del controlador” se debe seleccionar la opción “Configuración de I/O”, luego seleccionar “Nuevo module”, a continuación, en la opción Otros escoger “1756-Module” cuya descripción es Módulo 1756 Genérico, **(Figura 115)**.

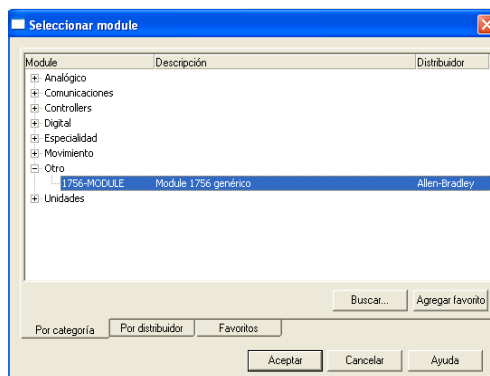


Figura 115: Crear Nuevo Módulo De Entradas y Salidas 1756-MODULE

2. A continuación, en la ventana que aparece se debe ingresar un nombre al módulo, colocar el número de ranura en el que se encuentra el modulo, y en los parámetros de conexión colocar la siguiente configuración como se muestra en la figura (**Figura 116**) y clic en Aceptar. Luego en Conexión colocar un tiempo para configurar la velocidad de conexión.

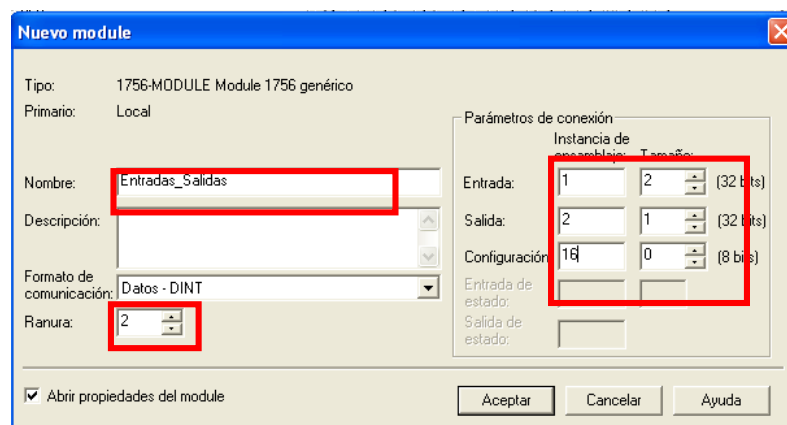


Figura 116: Parámetros de Conexión

2.17.4. Cargar programa en RSLogix 5000

Después de realizar la programación se procede a compilar y cargar el programa. El proceso para cargar el programa es el mismo tanto para el PLC Compact Logix 1769-L32E como para el Compact Logix Emulate este proceso se describe a continuación.

Lo primero que se realiza es compilar el programa para verificar que no exista ningún error. Luego en la barra de Menú seleccionar Comunicaciones, escoger la opción “Elemento Activo” en la ventana que aparece seleccionar CompactLogix Processor, finalmente dar clic en Descargar. **(Figura 117).**

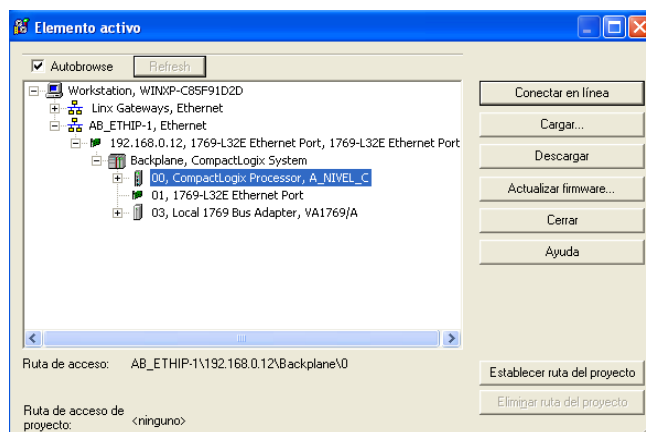


Figura 117: Cargar el Programa en RSLogix 5000

2.18. Programación de los controladores PID

2.18.1. Programación de un control PID para un PLC Siemens S7 1200

La implementación del control PID en TIA PORTAL es la misma tanto para el proceso Nivel como para el proceso Flujo y el proceso es el siguiente. Dentro del árbol de Proyecto en la opción Bloques de programa seleccionar “Agregar un nuevo bloque”. Aquí se escoge el “Cyclic Interrupt”, el cual permite que el programa sea cíclico **(Figura 118).**

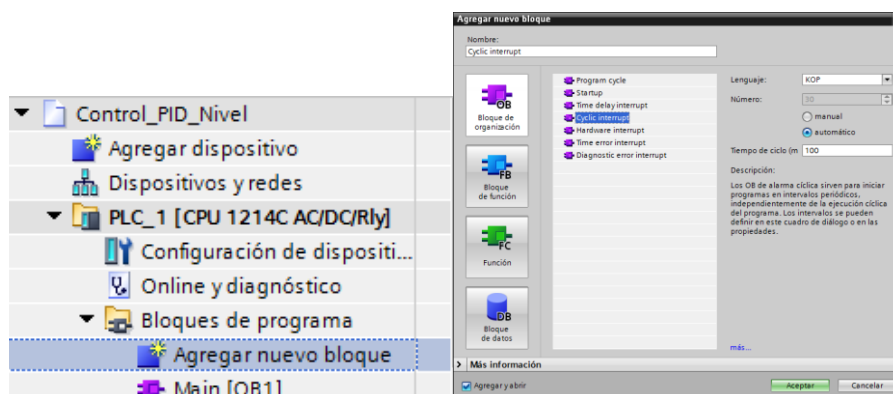


Figura 118: Añadir un nuevo bloque

a. Creación de Variables para el Bloque PID_Compact

Es importante definir todas las variables que serán utilizadas en el desarrollo del control PID, hay que considerar las direcciones tanto de entrada y salida del proceso a controlar, además setpoint, constantes de sintonización. Los pasos para crear variables dentro del TIA PORTAL son los siguientes:

1. En el árbol de proyecto seleccionar la opción “Variables PLC” (**Figura 119**), luego escoger “Tablas de variables estándar”.

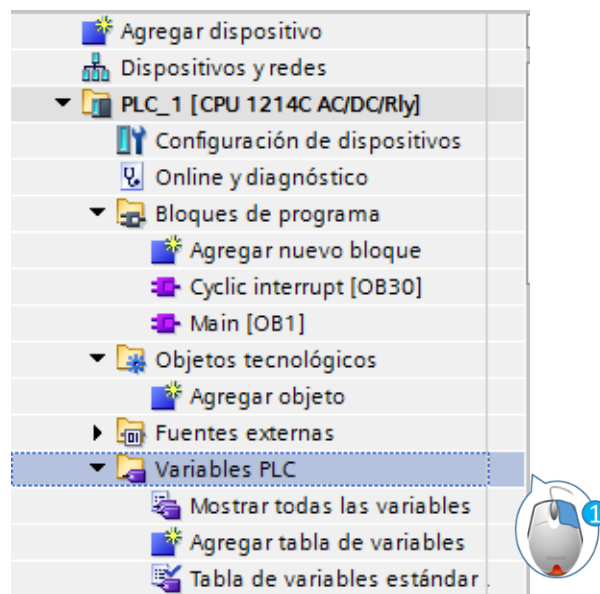


Figura 119: Crear variables para ser utilizadas por el PLC

2. Colocar un nombre, tipo de dato, dirección para cada una de las variables necesarias para este proyecto (**Figura 120**).

Variables							
Tabla de variables estándar							
	Nombre	Tipo de datos	Dirección	Rema...	Visibl...	Acces...	Comentario
1	PV	Real	%MD20	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
2	SP	Real	%MD40	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
3	CV	Real	%MD60	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
4	<Agregar>			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

Figura 120: Definición de Variables del Control PID

b. Creación de objeto tecnológico

1. En el árbol del proyecto seleccionar “Objetos Tecnológicos” y “Agregar Objeto” (**Figura 121**).

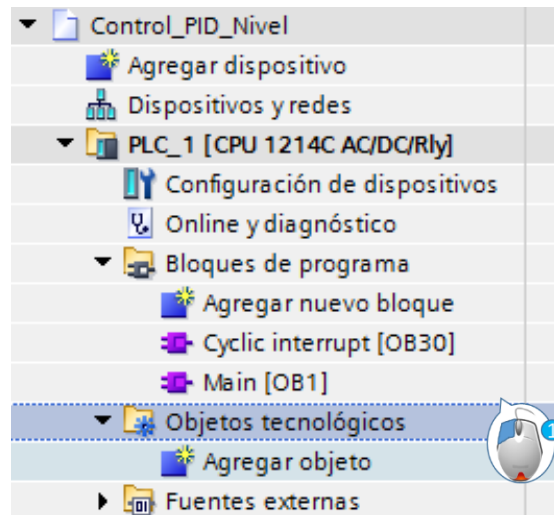


Figura 121: Agregar un nuevo bloque tecnológico

2. En la ventana que aparece seleccionar el tipo de objeto, en este caso el “PID_Compact”. Para finalizar clic en “Aceptar” (**Figura 122**).

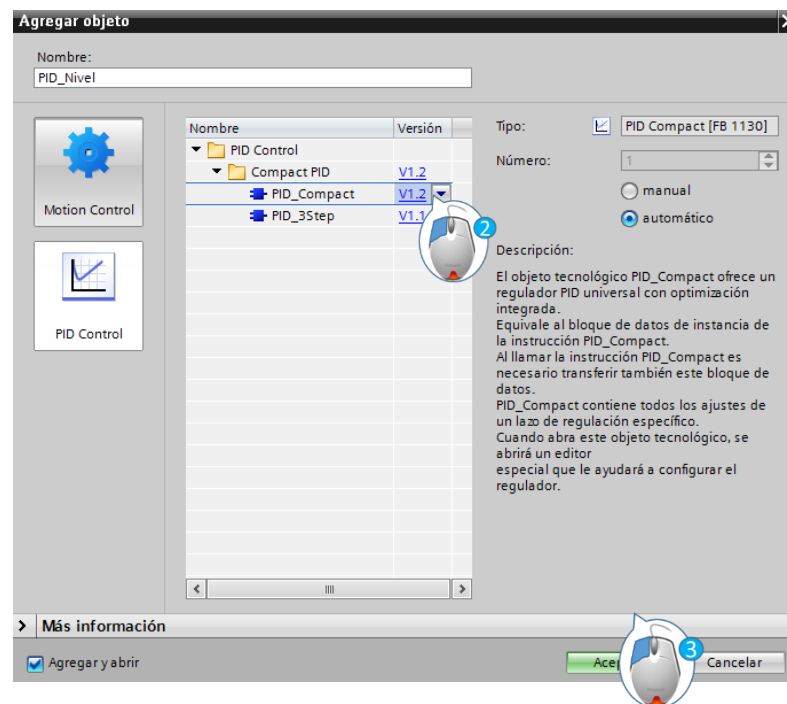


Figura 122: Objeto Tecnológico PID

- Colocar un nombre para el bloque PID Compact, finalmente ubicar el bloque tecnologico en la barra de trabajo y en cada una de las entradas asignar las variables pertinentes (**Figura 123**).

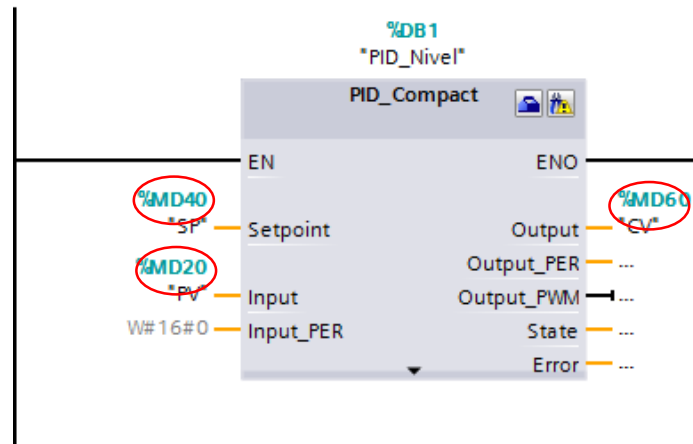


Figura 123: Asignación de Variables al Bloque PID

c. Configuración de los parámetros del PID en TIA PORTAL

Para configurar los parámetros del bloque PID es importante realizar los siguientes pasos:

- Como primer paso Abrir la ventana de configuración (**Figura 124**).

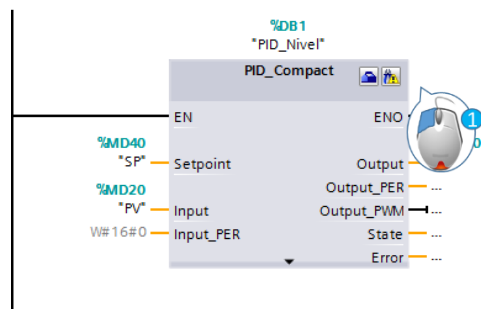


Figura 124: Abrir la ventana de configuración del bloque PID

- Se Abre una ventana (**Figura 125**) con varias opciones, en “Ajustes básicos” seleccionar “Tipo de Regulación” para este caso es General en porcentaje.

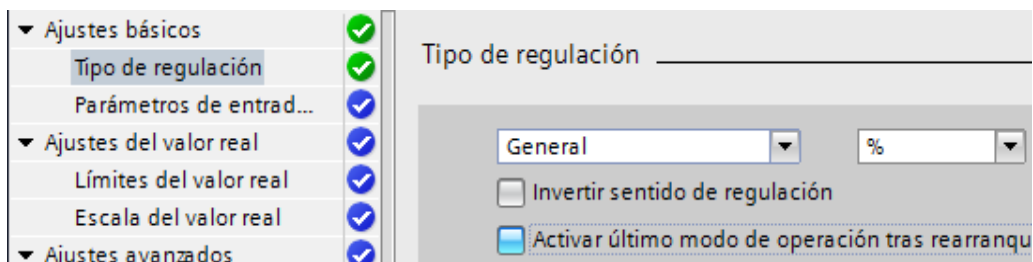


Figura 125: Tipo de Regulación

3. En “Ajustes Básicos”, la opción “Parámetros de Entrada/Salida” (**Figura 126**) seleccionar “Input” y “Output” respectivamente.

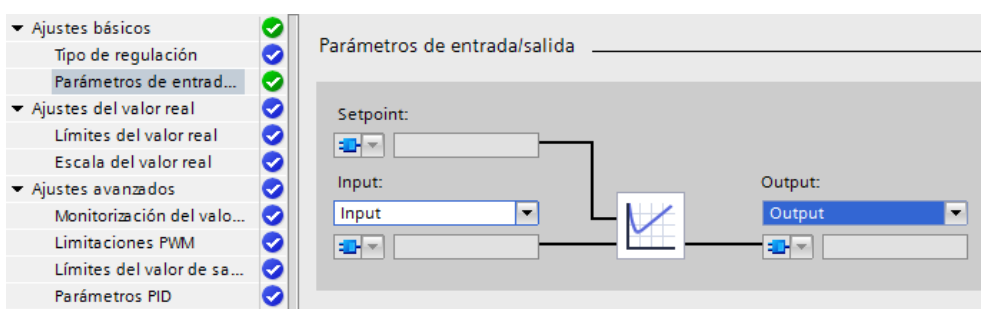


Figura 126: Parámetros de Entrada/Salida

4. En “Ajustes del valor real”, seleccionar “Límites del Valor Real de Entrada” (**Figura 127**).

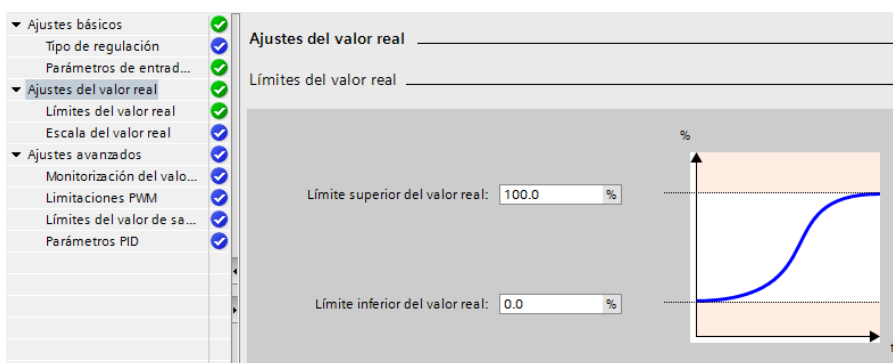


Figura 127: Escala del valor de entrada

5. En “Ajustes Avanzado”, colocar los límites del valor de salida de 0 a 100% para la automática regulación del proceso. (**Figura 128**).

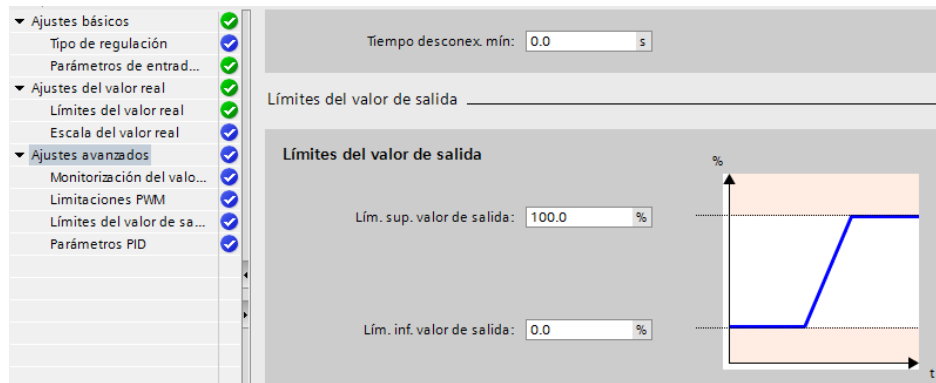


Figura 128: Escalar Variable de Salida

6. En “Parámetros PID” (**Figura 129**) se encuentra todos lo relacionado a la sintonización del proceso, como es la ganancia proporcional, tiempo de integración, tiempo derivativo, tiempo de muestreo del PID, entre los más importantes. Si se desea sintonizar el proceso de manera manual, es necesario Activar entrada manual.

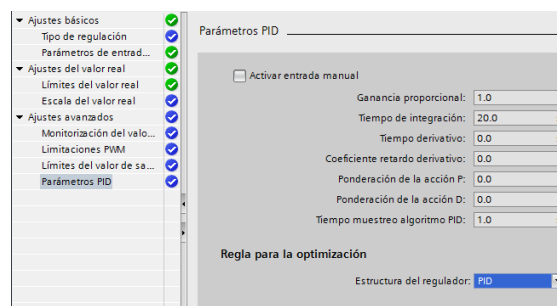


Figura 129: Parámetros PID

2.19. Programación de un control PID de Nivel para un PLC Allen Bradley CompactLogix.

Para realizar un control Proporcional Integral Derivativo en el RSLogix 5000 se debe cumplir los siguientes pasos:

a. Agregar Bloque PID

1. En el programa principal en instrucciones “especiales” seleccionar la instrucción PID.
2. Luego asignar un nombre al bloque PID, para lo cual se debe seleccionar Nuevo Tag. Ahí se debe colocar el nombre del bloque por ejemplo “PID_Nivel”, verificar que el Data Type sea PID (**Figura 130**).

Figura 130: Asignación nombre instrucción PID

b. Creación de variables en RSLogix 5000

Para crear Tags en RSLogix 5000 es necesario cumplir con los siguientes pasos que se describen a continuación:

1. En el menu principal seleccionar la carpeta Tasks, luego clic izquierdo sobre Tag de Programa y escoger la opcion Nuevo Tag. Aparece una nueva ventana (**Figura 131**), colocar el nombre de la tag o variable, una descripción de la misma, tipo en este caso Alias el mismo será direccionado hacia la instrucción PID anteriormente creada. La tag PV posee un direccionamiento PID_Nivel.PV

Figura 131: Creación de Variables para control PID

2. Repetir el paso anterior para crear todas las variables necesarias para desarrollar el control PID (**Figura 132**).

Nombre	Valor	Máscara de fuerza	Estilo	Data Type	Descripción
[-] PID_Nivel	{...}	{...}		PID	
PV	0.0		Float	REAL	Variable de Proceso
CV	0.0		Float	REAL	Variable de Control
SP	0.0		Float	REAL	Variable Setpoint
KP	0.0		Float	REAL	Constante Proporcional
KD	0.0		Float	REAL	Constante Derivativa
KI	0.0		Float	REAL	Constante Integral

Figura 132: Variables de PID

Nota: Los tags de las variables Setpoint, Process Value, Control Value, Constante Proporcional, derivativa e Integral están direccionadas hacia los valores del bloque PID anteriormente creado como se muestra en la siguiente (**Tabla 12**).

Tabla 12

Tags del Bloque PID

Parámetro PID	Direccionamiento	Descripción
CV	PID_Nivel.OUT	Variable de Control
PV	PID_Nivel.PV	Variable del Proceso
SP	PID_Nivel.SP	Valor de Consigna
KP	PID_Nivel.KP	Constante Proporcional
KI	PID_Nivel.KI	Constante Integral
KD	PID_Nivel.KD	Constante Derivativa

c. Configuración de Parámetros PID en RSLogix 5000

Para configurar los parámetros PID se debe realizar los siguientes pasos:

1. Luego de haber creado las variables, se observa en la siguiente imagen (**Figura 133**) que el bloque PID tiene varios campos de asignación como la variable de proceso "PID_Nivel.PV", la variable de control "PID_NIVEL.OUT", así como también el Punto de Ajuste, entre otros. Una vez asignado variables se proceden a realizar la configuración del

PID, en el siguiente proceso se muestra como configurar el bloque PID de manera correcta. Dar clic en las opciones del Bloque PID.

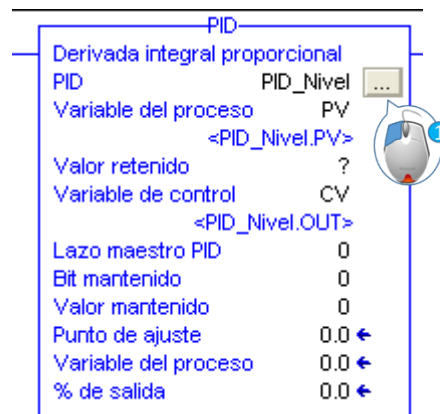


Figura 133: Bloque PID

2. A continuación, aparece una ventana (**Figura 134**) seleccionar la pestaña “Configuración” ahí configurar el método para obtener el error en la opción “Acción de Control”, además colocar el “Tiempo de actualización de lazo”, el cual indica el tiempo que se demorara el controlador en enviar una acción de control hacia el actuador. Finalmente colocar los límites del valor de CV.

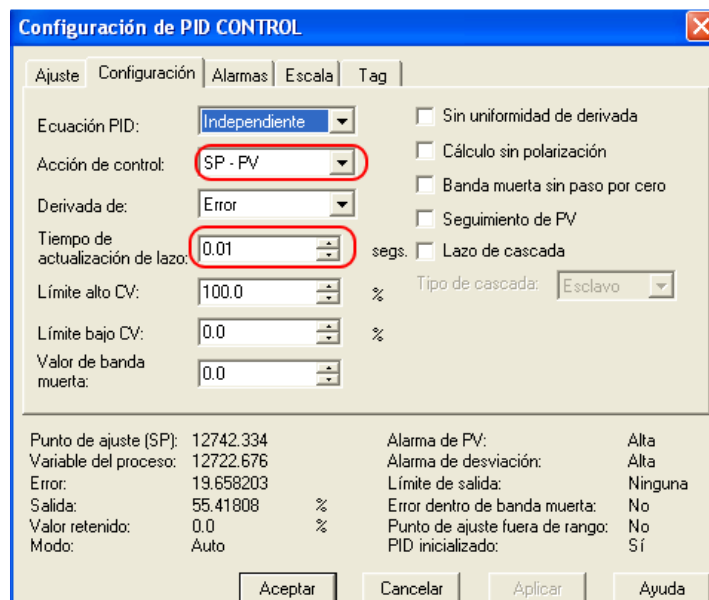


Figura 134: Configuración Bloque PID

3. Luego en la pestaña Escala colocar los valores máximos y mínimos de la variable de control y la variable de proceso como se muestra en la (Figura 135).

Figura 135: Escala Bloque PID

4. Por ultimo en la pestaña Ajuste se encuentran las constantes KP, KI, KD, además el valor de Setpoint. Con estos valores se procede a sintonizar el proceso el cual se desea controlar (Figura 136).

Figura 136: Ajuste de Bloque PID

CAPÍTULO III

RESULTADOS Y PRUEBAS EXPERIMENTALES

3.1. Introducción

En el presente capítulo se detalla de forma clara cómo funciona el simulador de procesos industriales desarrollado en el software Labview. Se realiza la prueba de funcionamiento de cada uno de los procesos simulados mediante el uso de los PLC's Siemens y Allen Bradley, además se realiza la conexión del OPC con el Simulador, además se explica los alcances y limitaciones que presenta este proyecto.

3.2. Pasos para el ingreso al Simulador de Procesos Industriales

El usuario lo primero que debe realizar es ingresar al Programa Principal, luego dar clic en "Inicio Simulador" a continuación, aparece una ventana donde se encuentran los menús secundarios del Simulador tales como PLC Siemens, PLC CompactLogix, PLC Micrologix. Cada menú secundario posee 6 subVI's que corresponden a cada proceso simulado. **(Figura 137)**



Figura 137: Ingreso al Simulador de Procesos

3.3. Conexión y enlaces del OPC con el Simulador de Procesos Industriales

Es importante señalar que para realizar las pruebas del Simulador de Procesos Industriales es necesario establecer la conexión entre el software de Programación de cada PLC y el Simulador para lo cual se recurre al uso

de un servidor OPC, el enlace es distinto según las marcas del PLC en el caso de ALLEN BRADLEY se usa el OPC RSLinx y en SIEMENS se usa el OPC KEPServerEx.

3.3.1. Configuración del servidor OPC para PLC SIEMENS S7-1200

Para establecer la comunicación entre el PLC Siemens S7-1200 y el simulador desarrollado en el software Labview, se utiliza el KEPServerEx, la forma de configurar correctamente el servidor OPC es primero crear el canal, luego Dispositivo y por último las TAG's.

a. Configuración de Canal

Para crear el canal se debe seleccionar “New Channel”, luego se ingresa un nombre para el canal para este caso es “Channel 1”, en la siguiente ventana se debe seleccionar “Siemens TCP/IP Ethernet”, posterior a esto aparece otra ventana en donde se elige la tarjeta de red a la cual se encuentra conectado el PLC Siemens S7-1200, finalmente dar clic en Finalizar (**Figura 138**).

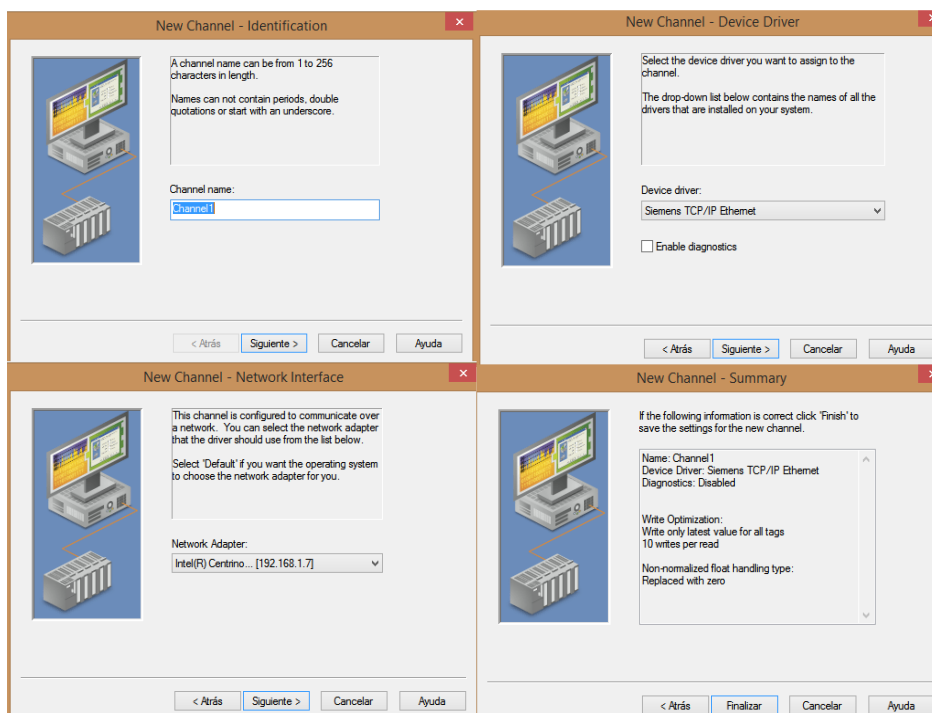


Figura 138: Configuración Canal

b. Configuración de Dispositivo

Para la creación de un dispositivo seleccionar “Add a new device”, colocar un nombre al nuevo dispositivo en este caso “Device1”, posterior a esto seleccionar el PLC en este caso el S7-1200, en la siguiente ventana colocar la dirección IP del dispositivo, luego escoger el puerto 102 para la configuración TCP/IP, finalmente dar clic en Finalizar (**Figura 139**).

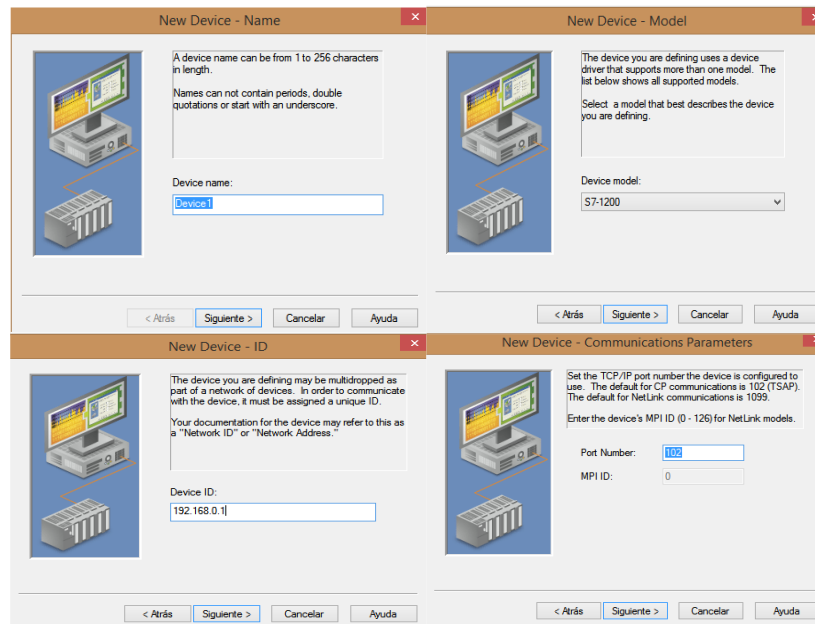


Figura 139: Configuración Dispositivo

c. Configuración de Tags

Para crear las TAG's se debe seleccionar la opción “New Tag”, aparece una nueva venta de las propiedades de la tag, aquí se debe colocar el Nombre de la variable, la dirección que desea leer o escribir dicha tag, además una corta descripción y el tipo de dato que desea asignar a cada variable. Este proceso se repite para todas las variables en cada uno de los procesos del simulador que utilice el PLC Siemens S7-1200 (**Figura 140**).

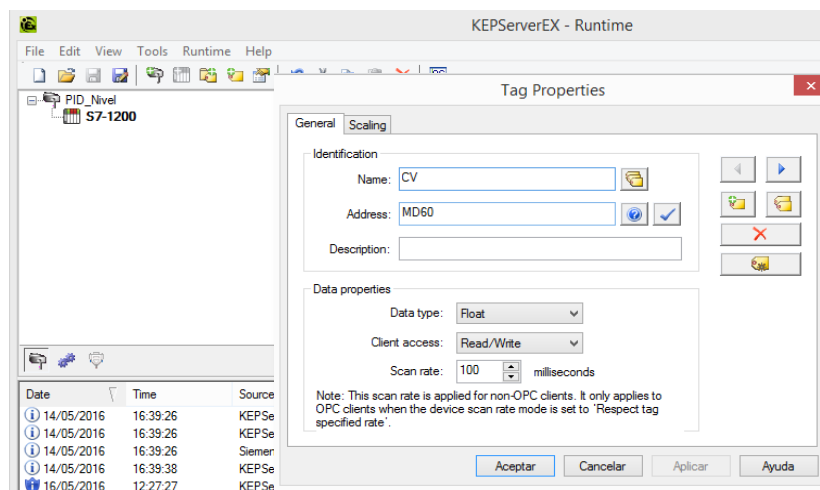


Figura 140: Configuración Tag

3.3.2. Configuración del servidor OPC para PLC Allen Bradley

Para establecer la comunicación entre PLC Allen Bradley ya sea el CompactLogix como el Micrologix y el simulador se utiliza el RSLinx, la forma de configurar correctamente el servidor OPC es la siguiente:

1. Primero Abrir RSLinx, en la barra de Menú seleccionar “DDE/OPC”, luego escoger “Topic Configuration” como se muestra en la figura. Luego aparece otra ventana en la cual se va realizar la configuración DDE/OPC ahí en Topic List se debe escribir el nombre del proyecto. **(Figura 141)**

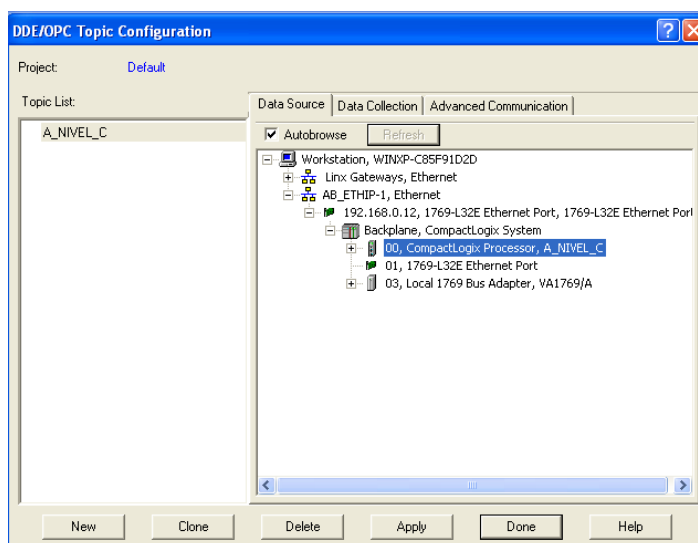


Figura 141: Configuración OPC

Dependiendo del proceso con el cual se esté trabajando, en el Topic List se debe colocar los siguientes nombres para que funcione el enlace entre el OPC y cada proceso del simulador. Cuando se trabaja con PLC Allen Bradley Micrologix se coloca los siguientes nombres al proyecto. **(Tabla 13)**

Tabla 13

Nombres de código de conexión de los procesos al OPC.

MICRO LOGIX	
Proceso	Nombre
Silo	A_SILO_M
Puerta	A_PUERTA_M
Mezclador	A_MEZCLADOR_M
Semáforo	A_SEMAFORO_M
Nivel	A_NIVEL_M
Flujo	A_FLUJO_M

Cuando se trabaja con PLC Allen Bradley CompactLogix se coloca los siguientes nombres al proyecto. **(Tabla 14)**

Tabla 14

Nombres de código de conexión de los procesos al OPC.

COMPACTLOGIX	
Proceso	Nombre
Silo	A_SILO_C
Puerta	A_PUERTA_C
Mezclador	A_MEZCLADOR_C
Semáforo	A_SEMAFORO_C
Nivel	A_NIVEL_C
Flujo	A_FLUJO_C

3.4. Pruebas Experimentales del funcionamiento y Análisis de Resultados de la simulación del Proceso Silo

3.4.1. Descripción del funcionamiento del proceso

El proceso presenta el siguiente funcionamiento, es necesario encender el motor para que la banda transportadora comience a moverse con el tanque, en el momento que el sensor de proximidad se activa el motor se detiene y se activa el silo para el llenado, una vez activado el sensor de llenado full se activa indicado que está lleno se apaga el silo y se activa el motor para que continúe el movimiento del tanque (**Figura 142**).

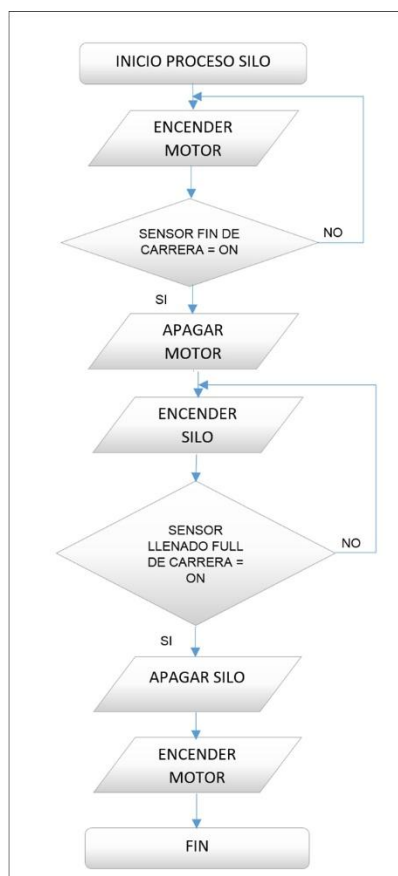


Figura 142: Diagrama de Funcionamiento Silo

3.4.2. Pruebas de funcionamiento de la Simulación Silo

La simulación del Proceso Silo es sometida a diferentes pruebas para verificar el correcto funcionamiento de la misma, para este caso utilizaremos la simulación Silo para el PLC CompactLogix. Las pruebas efectuadas a la simulación del proceso Silo mediante el PLC CompactLogix son las mismas

para el PLC Siemens S7-1200 y el Micrologix. El desarrollo de la programación ladder se realiza en el software de programación RSLogix 5000 y como OPC se utiliza el RSLinx. La diferencia entre cada uno de las simulaciones para cada tipo de PLC radica en las direcciones que poseen cada una de las variables del Proceso Silo como se indica en la **(Tabla 15)**.

Tabla 15

Variables del Proceso Silo

Variables del Proceso Silo				
NOMBRE	DIRECCION PARA CADA PLC			DESCRIPCIÓN
	Micrologix	CompactLogix	S7-1200	
EJECUTANDO	O:0.0/3	EJECUTANDO	Q0.3	Indicador led del simulador en funcionamiento
LLENADO	O:0.0/4	LLENADO	Q0.4	Indicador led del llenado del tanque
LLENADO_COMPLETO	O:0.0/5	LLENADO_COMPLETO	Q0.5	Indicador led del llenado completo del tanque
MOTOR	O:0.0/0	MOTOR	Q0.0	Realiza el movimiento de la banda transportadora
SILO	O:0.0/1	SILO	Q0.1	Realiza el llenado del tanque
FIN_DE_CARRERA	B3:1/3	FIN_DE_CARRERA	M0.2	Indica cuando el tanque se encuentre bajo el silo
LLENADO_FULL	B3:1/3	LLENADO_FULL	M0.3	Indica el llenado máximo del tanque
INICIO	I:0.0/1	INICIO	I0.1	Activa el proceso
PAUSA	I:0.0/2	PAUSA	I0.0	Detiene el proceso

Luego, se realiza las pruebas al proceso Silo mediante la programación ladder del proceso. En la **(Figura 143)** se observa las diferentes etapas de programación de la Simulación, las cuales se detallan a continuación.

- a. En la primera etapa se muestra el panel frontal del proceso silo, donde 'se programa el proceso para movilizar el tanque mediante la banda transportadora. A su vez se activa el indicador Ejecutando.
- b. Luego se observa la programación de la acción de llenar el tanque mediante el silo cuando el sensor 1 (Fin de Carrera) este activado.

- c. La tercera etapa indica que el tanque se encuentra completamente lleno mediante el sensor 2 (Sensor de Llenado).
- d. Finalmente, se realiza la movilización del tanque lleno a través de la banda transportadora.

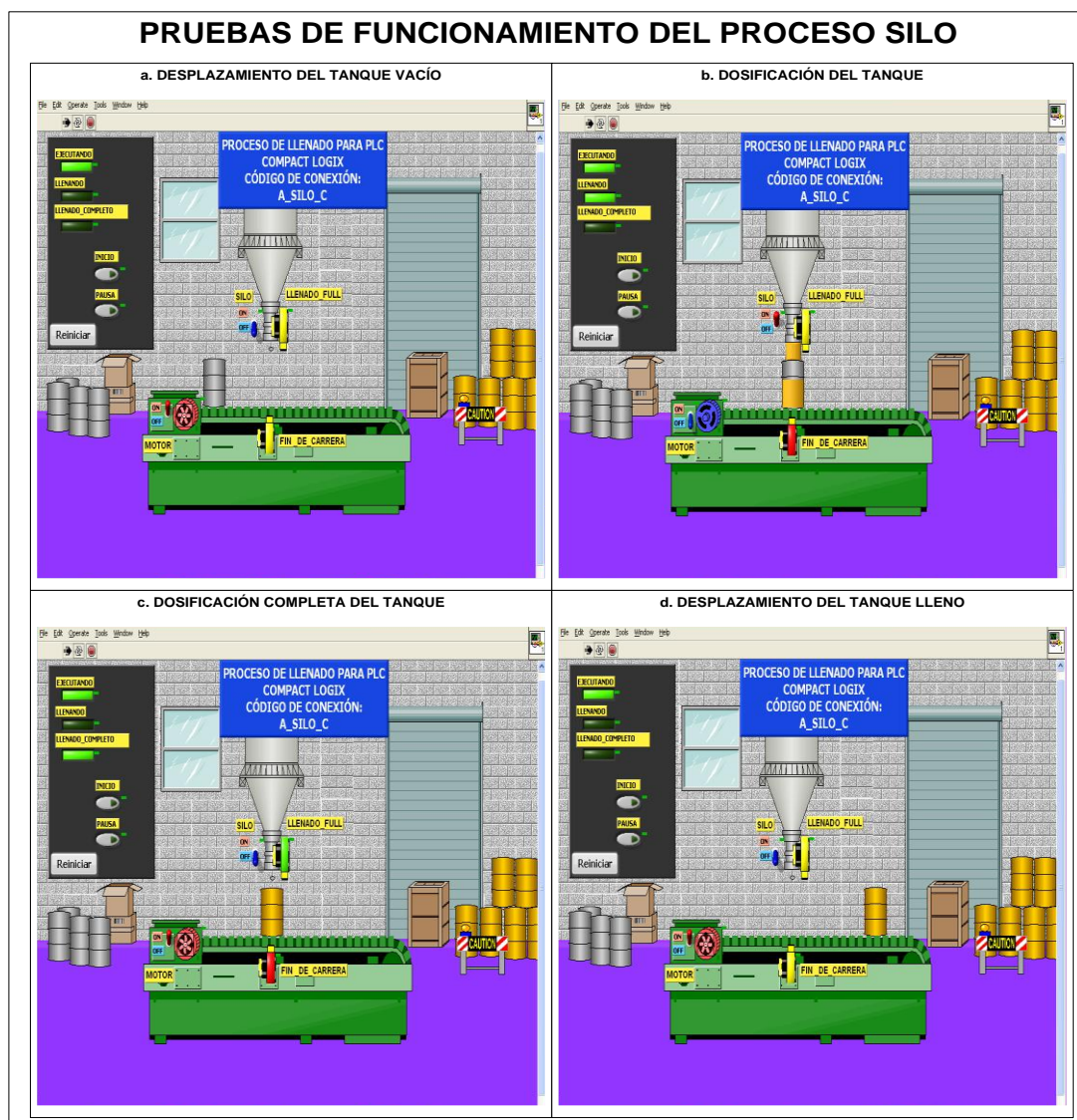


Figura 143: Pruebas de Funcionamiento del Proceso Silo

En la (Figura 144) se encuentra cada una de las programaciones ladder del Proceso Silo para cada uno de los PLC's compatibles con esta simulación como son: Micrologix, CompactLogix y Siemens S7-1200.

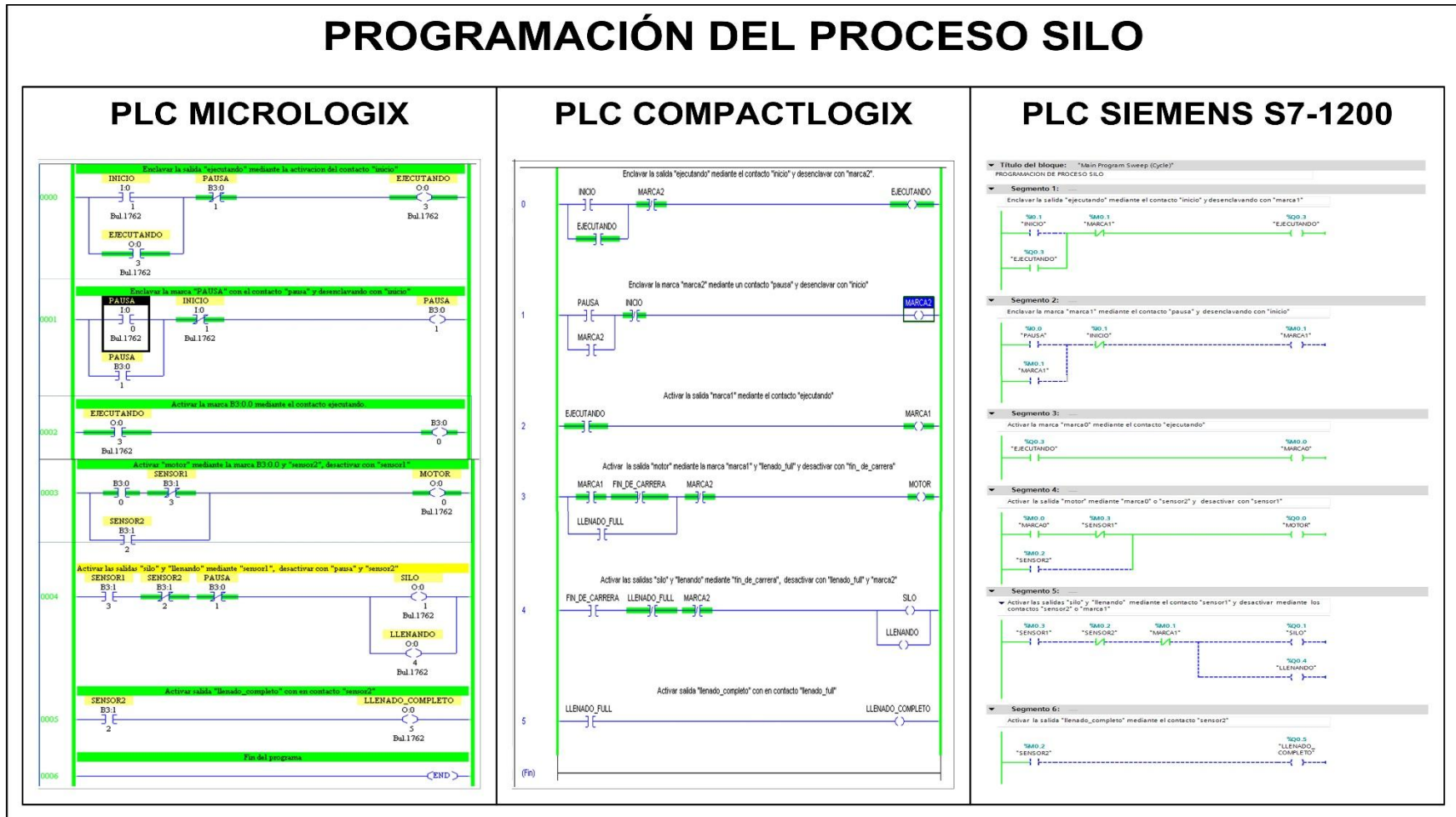


Figura 144: Programación Ladder Proceso Silo

3.5. Pruebas Experimentales del funcionamiento y Análisis de Resultados de la simulación del Proceso Semáforo

3.5.1. Descripción del funcionamiento del proceso

El proyecto semáforo funciona de una forma estándar como cualquier semáforo, lo primero que se ejecuta es el encendido de Rojo y Verde2 por un tiempo de 10s. Trascurrido este tiempo cambia a Rojo y Amarillo2 por un tiempo de 5s. Luego cambia a Verde y Rojo2 por un tiempo de 10s. Trascurrido el tiempo cambia a Amarillo y Rojo2 por un tiempo de 5s. Trascurrido un tiempo regresa a la posición inicial y continua sucesivamente. (Figura 145)

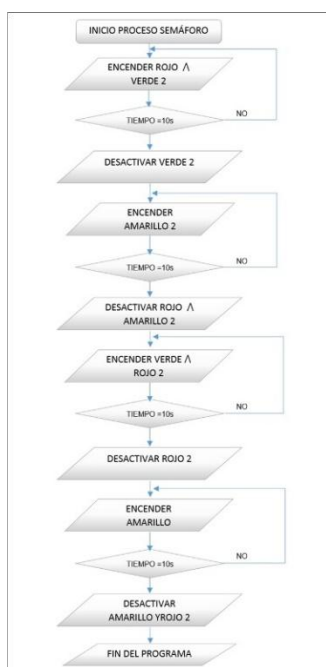


Figura 145: Diagrama de Funcionamiento Semáforo

3.5.2. Pruebas de funcionamiento de la Simulación Semáforo

Para verificar el correcto funcionamiento de la simulación del Proceso Semáforo se realiza diferentes pruebas, para este caso utilizaremos la simulación Semáforo para el PLC CompactLogix. El desarrollo de la programación ladder se realiza en el software de programación RSLogix 5000 y como OPC se utiliza el RSLinx. Las pruebas efectuadas a la simulación mediante el PLC CompactLogix son las mismas para el PLC Siemens S7-1200 y el Micrologix. La diferencia entre cada uno de las simulaciones para

cada tipo de PLC radica en las direcciones que poseen cada una de las variables del Proceso Semáforo como se indica en la **(Tabla 16)**.

Tabla 16

Variables del Proceso Semáforo

Variables del Proceso Semáforo				
NOMBRE	DIRECCIÓN PARA CADA PLC			DESCRIPCIÓN
	CompactLogix	Micrologix	S7-1200	
ROJO	ROJO	O:0.0/1	Q0.1	Indicador led de encendido la luz roja
AMARILLO	AMARILLO	O:0.0/2	Q0.2	Indicador led de encendido la luz amarillo
VERDE	VERDE	O:0.0/3	Q0.3	Indicador led de encendido la luz verde
ROJO2	ROJO2	O:0.0/4	Q0.4	Indicador led de encendido la luz roja2
AMARILLO2	AMARILLO2	O:0.0/5	Q0.5	Indicador led de encendido la luz amarillo2
VERDE2	VERDE2	O:0.0/6	Q0.6	Indicador led de encendido la luz verde2
CRUZAR	CRUZAR	I:0.0/3	I0.3	Activa las dos luces rojas
PARQUEO	-----	-----	-----	Parquea el vehículo
INICIO	INICIO	I:0.0/1	I0.1	Activa el proceso
PAUSA	PAUSA	I:0.0/0	I0.0	Detiene el proceso

En la **(Figura 146)** se observa las diferentes etapas de programación de la Simulación, las cuales se detallan a continuación.

- a. Activación del color verde en el carril vertical y el color rojo en el carril horizontal.
- b. Activación del color amarillo en el carril vertical y color rojo en el carril horizontal.
- c. Activación del color rojo en el carril vertical y color verde en el carril horizontal.
- d. Activación del color rojo en el carril vertical y el color amarillo en el carril horizontal.
- e. Activación del cruce de peatones con los colores rojo en los dos carriles.
- f. Activación del botón Parqueo en el carril horizontal

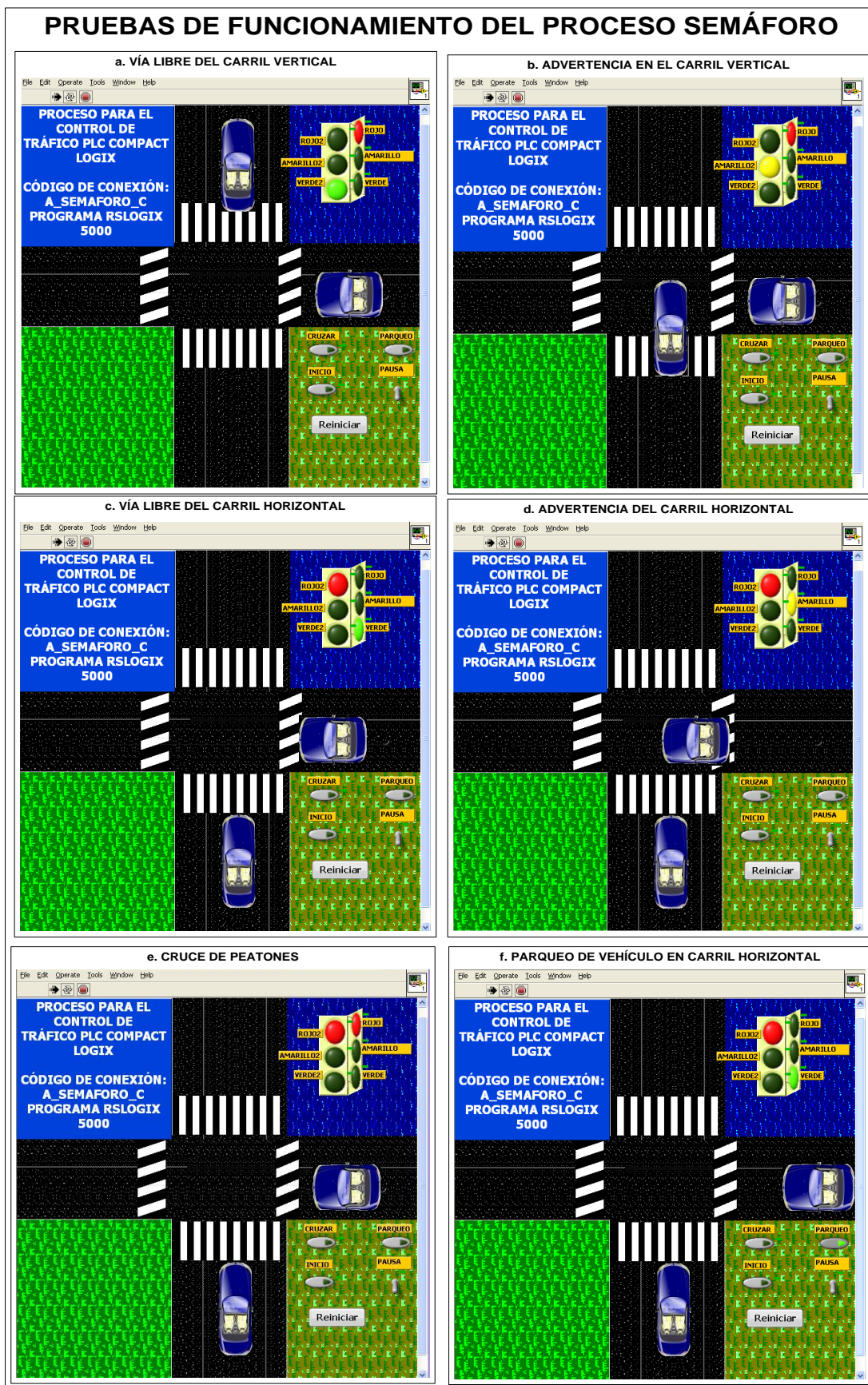


Figura 146: Pruebas de Funcionamiento del Proceso Semáforo

En la (Figura 147) se encuentra cada una de las programaciones ladder del Proceso Semáforo para cada uno de los PLC's compatibles con esta simulación como son: Micrologix, CompactLogix y Siemens S7-1200.

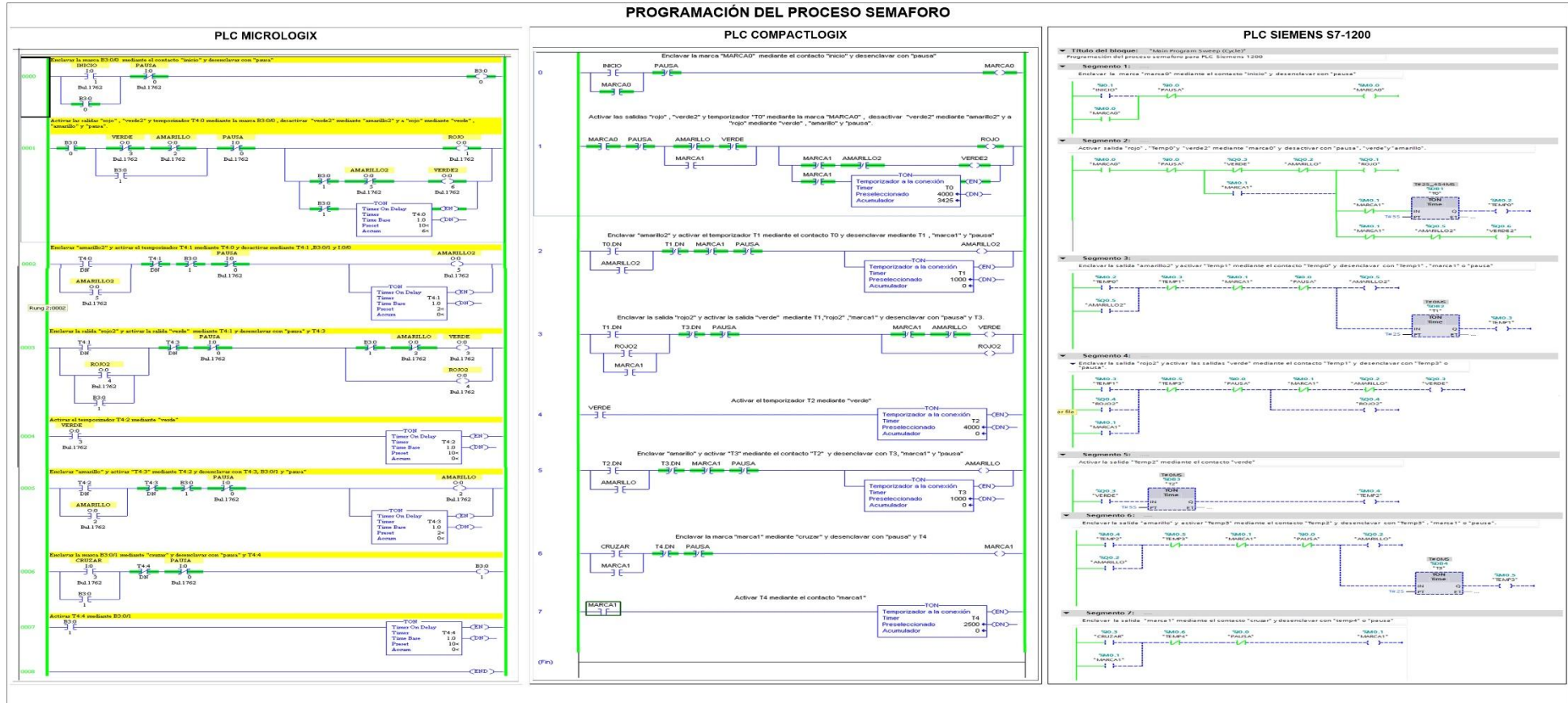


Figura 147: Programación Ladder del Proceso Semáforo

3.6. Pruebas Experimentales del funcionamiento y Análisis de Resultados de la simulación del Proceso Garaje

3.6.1. Descripción del funcionamiento del proceso

El funcionamiento del proceso es controlado por el usuario quien acciona mediante un pulsador abrir, el cual acciona el motor para abrir la puerta, al igual que sucede con el pulsador cerrar el cual acciona otro motor para cerrar la puerta, los motores no pueden funcionar los dos al mismo tiempo y para controlar los motores se debe presionar el pulsador parar el cual detiene el motor tanto de abrir como cerrar para dar paso al otro motor (**Figura 148**).

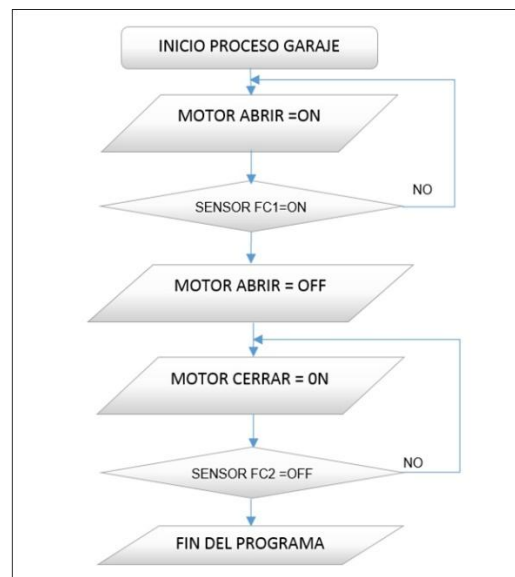


Figura 148: Diagrama de Funcionamiento Garaje

3.6.2. Pruebas de Funcionamiento de la simulación Garaje

Para verificar el correcto funcionamiento de la simulación del Proceso Garaje se realiza diferentes pruebas, para este caso utilizaremos la simulación Garaje para el PLC CompactLogix. El desarrollo de la programación ladder se realiza en el software de programación RSLogix 5000 y como OPC se utiliza el RSLinx. Las pruebas efectuadas a la simulación mediante el PLC CompactLogix son las mismas para el PLC Siemens S7-1200 y el Micrologix. La diferencia entre cada uno de las simulaciones para cada tipo de PLC radica en las direcciones que poseen cada una de las variables del Proceso Garaje como se indica en la (**Tabla 17**).

Tabla 17

Variables del Proceso Garaje

Variables del Proceso Garaje				
Nombre	Direcciones para cada PLC			Descripción
	CompactLogix	Micrologix	S7-1200	
ABRIR	ABRIR	O:0.0/2	Q0.3	Indica la activación del motor abrir
CERRAR	CERRAR	O:0.0/3	Q0.4	Indica la activación del motor cerrar
PARAR	PARAR	O:0.0/4	Q0.5	Indica el paro de los motores abrir y cerrar
MOTOR_ABRIR	MOTOR_ABRIR	O:0.0/0	Q0.1	Realiza la acción de abrir la puerta
MOTOR_CERRAR	MOTOR_CERRAR	O:0.0/1	Q0.2	Realiza la acción de cerrar la puerta
FC1	FC1	B3:1/4	M0.4	Indicador que la puerta está abierta al máximo
FC2	FC2	B3:1/5	M0.5	Indica que la puerta está cerrada totalmente
ABRIR_C	ABRIR_C	I:0.0/0	I0.0	Realiza la activación del motor abrir
CERRAR_C	CERRAR_C	I:0.0/1	I0.1	Realiza la activación del motor cerrar
PAUSA_C	PAUSA_C	I:0.0/2	I0.3	Realiza el paro de los motores abrir y cerrar

Luego, se realiza las pruebas al proceso Garaje mediante la programación ladder del proceso. En la **(Figura 149)** se observa las diferentes etapas de programación de la Simulación, las cuales se detallan a continuación.

- a. Se muestra el panel frontal del proceso donde se realiza la apertura de la puerta, además se activa el indicador Abrir y.
- b. Se observa la apertura total de la puerta del Proceso Garaje.
- c. Indica el panel frontal del proceso garaje en donde se realiza el cierre de la puerta y la activación del indicador cerrar.
- d. Se observa el cierre total de la puerta.



Figura 149: Pruebas de Funcionamiento del Proceso Garaje

En la (Figura 150) se encuentra cada una de las programaciones ladder del Proceso Garaje para cada uno de los PLC's compatibles con esta simulación como son: Micrologix, CompactLogix y Siemens S7-1200.

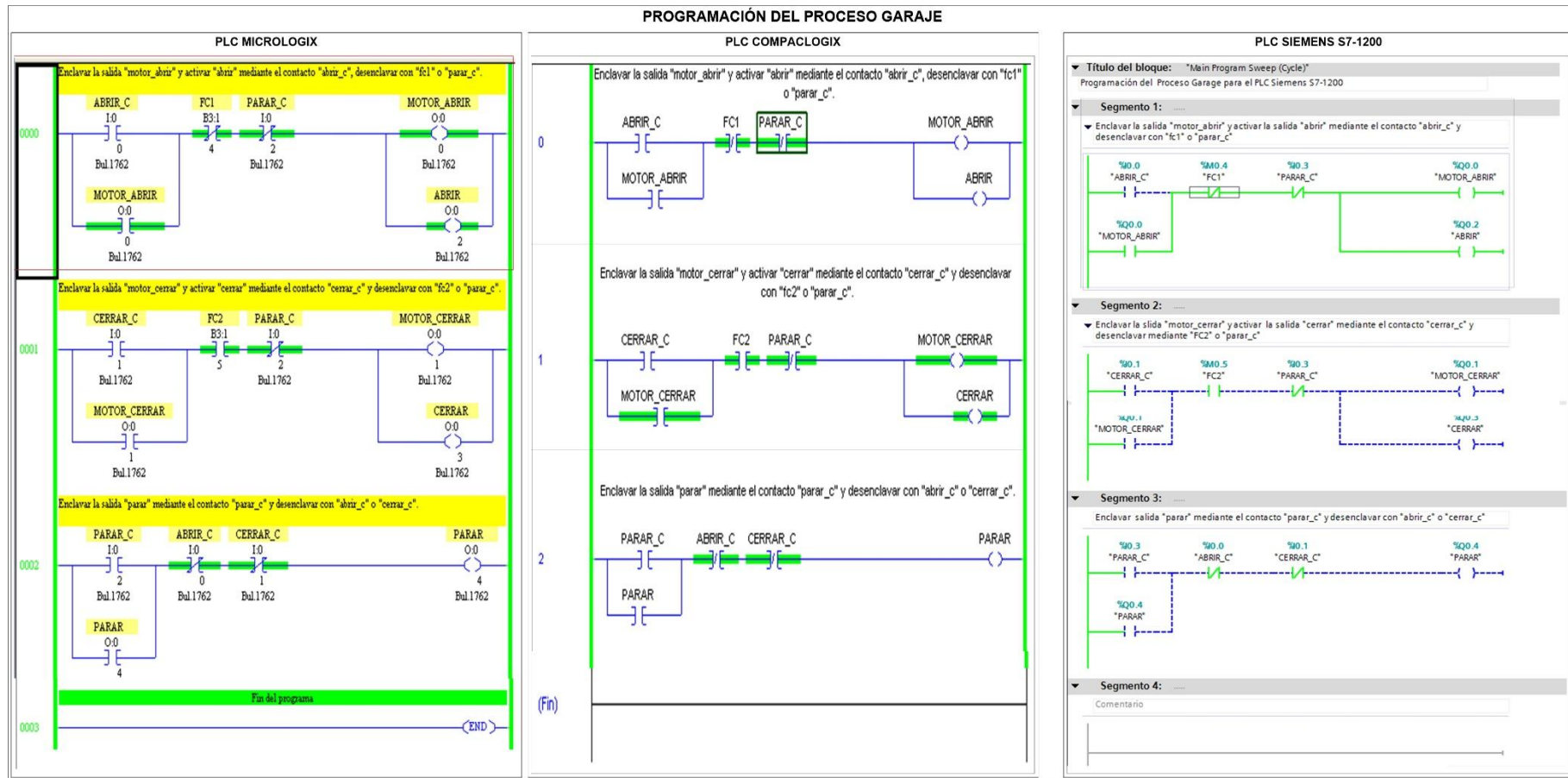


Figura 150: Programación Ladder del Proceso Garaje

3.7. Pruebas Experimentales del funcionamiento y Análisis de Resultados de la simulación del Proceso Mezclador

3.7.1. Descripción del funcionamiento del proceso

El proyecto Mezclador realiza el llenado de un tanque mediante dos bombas las cuales son usadas de forma independiente y controladas mediante dos sensores de nivel, el contenido del Tanque es mezclado y calentado hasta una temperatura establecida por el operador, el tanque es vaciado por la activación de una tercera bomba. **(Figura 151)**

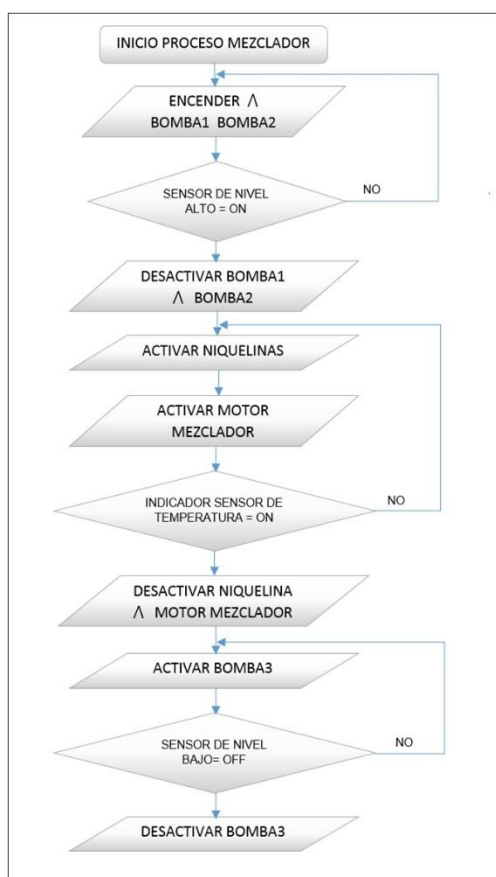


Figura 151: Diagrama de Funcionamiento del Proceso Mezclador

3.7.2. Pruebas de Funcionamiento de la simulación Mezclador

Para verificar el correcto funcionamiento de la simulación del Proceso Mezclador se realiza diferentes pruebas, para este caso utilizaremos la simulación Mezclador para el PLC CompactLogix. El desarrollo de la

programación ladder se realiza en el software de programación RSLogix 5000 y como OPC se utiliza el RSLinx. Las pruebas efectuadas a la simulación mediante el PLC CompactLogix son las mismas para el S7-1200 y el Micrologix. La diferencia entre cada uno de las simulaciones para cada tipo de PLC radica en las direcciones que poseen cada una de las variables del Proceso Mezclador como se indica en la **(Tabla 18)**.

Tabla 18

Variables del Proceso Mezclador

Variables del Proceso Mezclador				
Nombre	Direcciones para cada PLC			Descripción
	S7-1200	CompactLogix	Micrologix	
EJECUTANDO	Q0.5	EJECUTANDO	O:0.0/5	Indicador led del simulador en funcionamiento
LLENADO_MAX	Q0.6	LLENADO_MAX	O:0.0/6	Indicador led del llenado máximo del tanque
VACIADO	Q0.7	VACIADO	O:0.0/7	Indicador led del vaciado del tanque
MOTOR	Q0.0	MOTOR	O:0.0/0	Realiza el movimiento de las paletas para mezclar
FUEGO	Q0.4	FUEGO	O:0.0/4	Realiza el calentamiento del tanque
BOMBA1	Q0.1	BOMBA1	O:0.0/1	Llena el tanque mediante la bomba 1
BOMBA2	Q0.2	BOMBA2	O:0.0/2	Llena el tanque mediante la bomba 2
BOMBA3	Q0.3	BOMBA3	O:0.0/3	Llena el tanque mediante la bomba 3
INDICADOR 1	M1.0	INDICADOR 1	B3:1/5	Indica la activación de la bomba 1
INDICADOR2	M1.1	INDICADOR 2	B3:1/6	Indica la activación de la bomba 2
INDICADOR3	M1.2	INDICADOR 3	B3:1/7	Indica la activación de la bomba 3
NIVEL_ALTO	M0.4	NIVEL_ALTO	B3:1/4	Indica el llenado máximo del tanque
NIVEL_BAJO	M0.3	NIVEL_BAJO	B3:1/3	Indica el vaciado total del tanque
INDICADOR_TEM	M0.5	INDICADOR_TEM	B3:1/2	Indica que la temperatura llego al valor del set point
INICIO	I0.1	INICIO	I:0.0/1	Activa el proceso
PAUSA	I0.0	PAUSA	I:0.0/0	Detiene el proceso

Luego, se realiza las pruebas al proceso Mezclador mediante la programación ladder del proceso. En la **(Figura 152)** se detallan estas pruebas

- Se realiza llenado del tanque mediante la bomba1 y se activa el indicador ejecutando.
- Se ejecuta las acciones de calentar y mezclar el líquido del tanque.
- Se realiza el vaciado del tanque mediante la bomba3.
- Se observa el tanque completamente vacío.

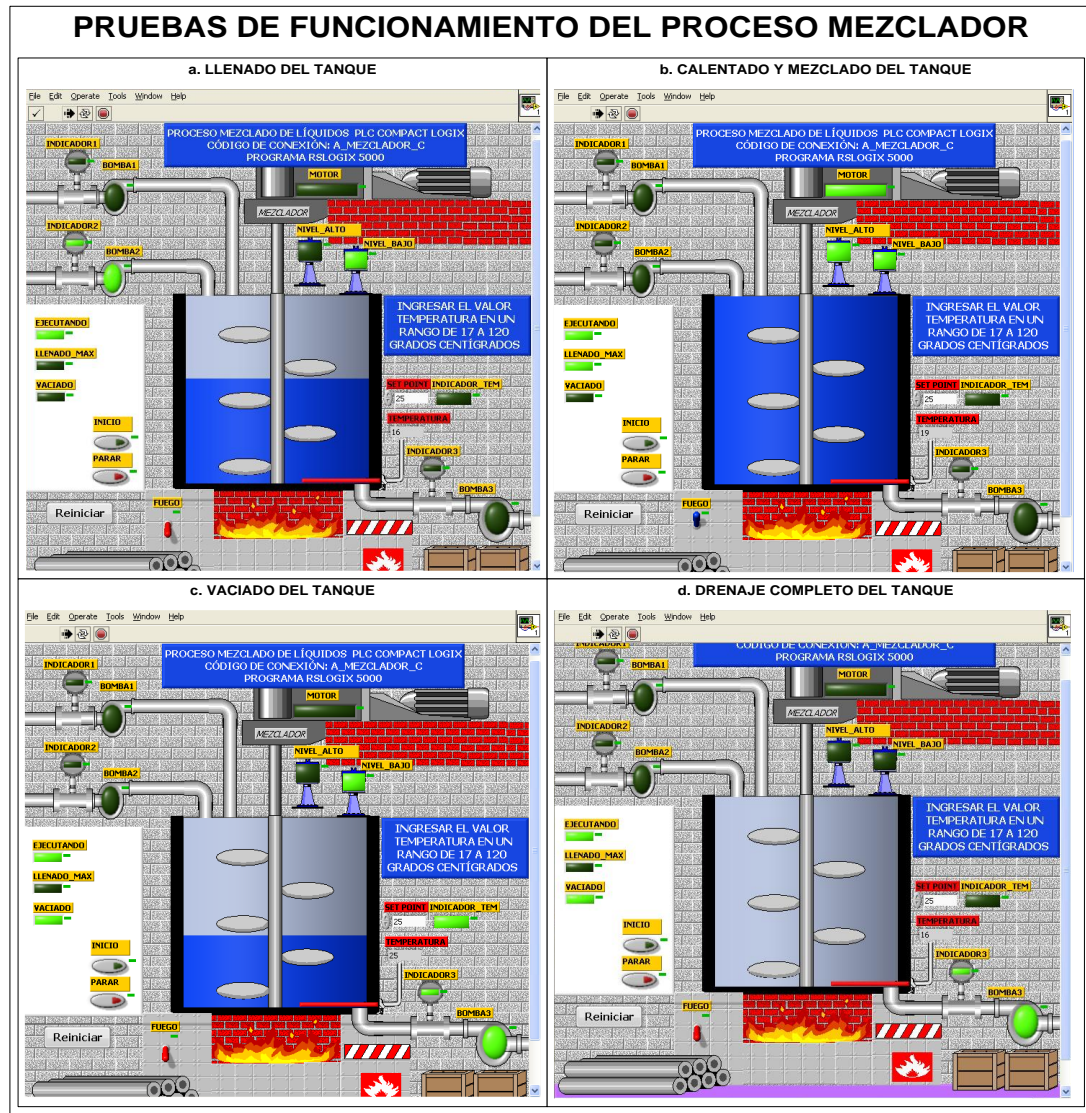


Figura 152: Pruebas de Funcionamiento del Proceso Mezclador

En la (Figura 153) se encuentra cada una de las programaciones ladder del Proceso Mezclador para cada uno de los PLC's compatibles con esta simulación como son: Micrologix, CompactLogix

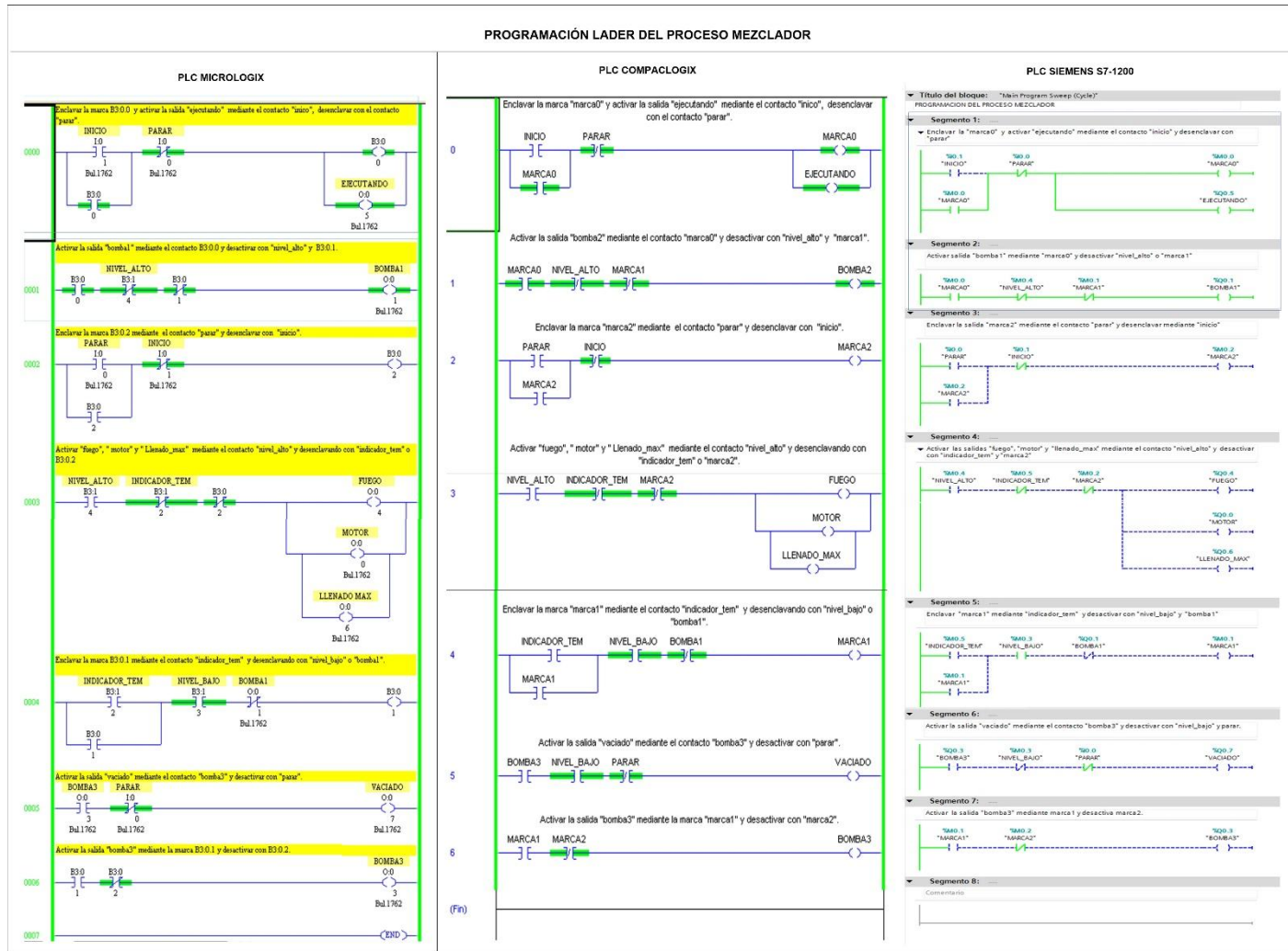


Figura 153: Programación Ladder del Proceso Mezclador

3.8. Pruebas Experimentales del funcionamiento de la simulación del Proceso Nivel

3.8.1. Descripción del funcionamiento del proceso

La simulación del proceso Nivel consta de un menú en la primera pestaña se encuentra la presentación del Proceso, aquí se observa la animación del proceso, además los valores de SP, PV, CV respectivamente, en la segunda pestaña se realiza la sintonización del proceso, aquí que coloca las constantes de sintonización del proceso incluye también las tendencias del proceso Nivel, en la tercera pestaña el usuario puede observar los históricos del Proceso, y finalmente en la última pestaña se encuentra las alarmas las cuales se activarán de acuerdo a los valores que fueron programados dentro de este programa. El proceso nivel posee un rango de 0 a 100 cm.

3.8.2. Diagrama Ladder

La programación del controlador se realizó mediante el uso del bloque PID Compact el cual se encuentra en las librerías del software TIA PORTAL, además las constantes de sintonización Kp, Ti, Td se encuentran direccionadas hacia el bloque PID mediante la instrucción MOVE. **(Figura 154)**

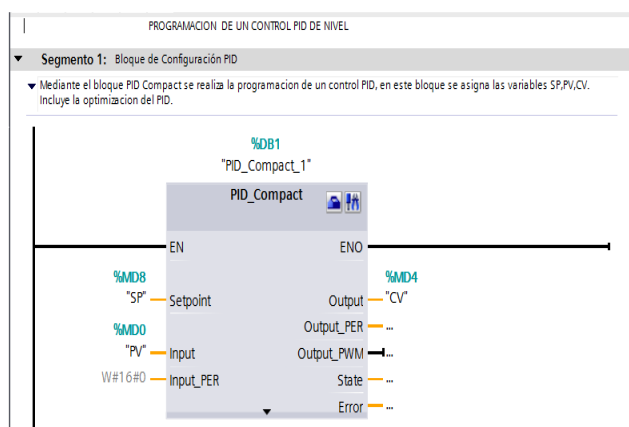


Figura 154: Diagrama Ladder de Control PID de Nivel

3.8.3. Sintonización del Proceso Nivel

Para la sintonización del lazo de control PID implementado en la simulación de este proceso se utilizó el método de tanteo. El cual consiste en la aplicación de los siguientes pasos. Lo primero que se debe realizar es colocar un valor de K_p (ganancia proporcional) hasta que la respuesta del sistema presente una oscilación de forma estable, seguido a esto se ingresa un valor de T_d (Tiempo Derivativo) para lograr reducir el sobre impulso de la respuesta finalmente se coloca el valor de T_i (Tiempo Integral) cuyo objetivo es eliminar el error en estado estable. Después de haber aplicado el método de tanteo se obtuvo las constantes que se pueden observar en la **(Tabla 19)** las cuales permiten realizar la correcta sintonización del control PID de la simulación del Proceso Nivel.

Tabla 19

Constantes de sintonización de la simulación Nivel

PROCESO NIVEL		
Contantes	Valor	Dimensión
Constante Proporcional (K_p)	10.5	[adim]
Tiempo Integral (T_i)	10	[adim]
Tiempo Derivativo (T_d)	0.0002	[adim]

Luego de colocar el valor de las constantes de sintonización se realiza la prueba del proceso nivel mediante el cambio de valor de la variable Setpoint (SP) y se observa como la variable de proceso (PV) sigue el valor de consigna además se observa la gráfica de la Variable de control (CV) **(Figura 155)**.

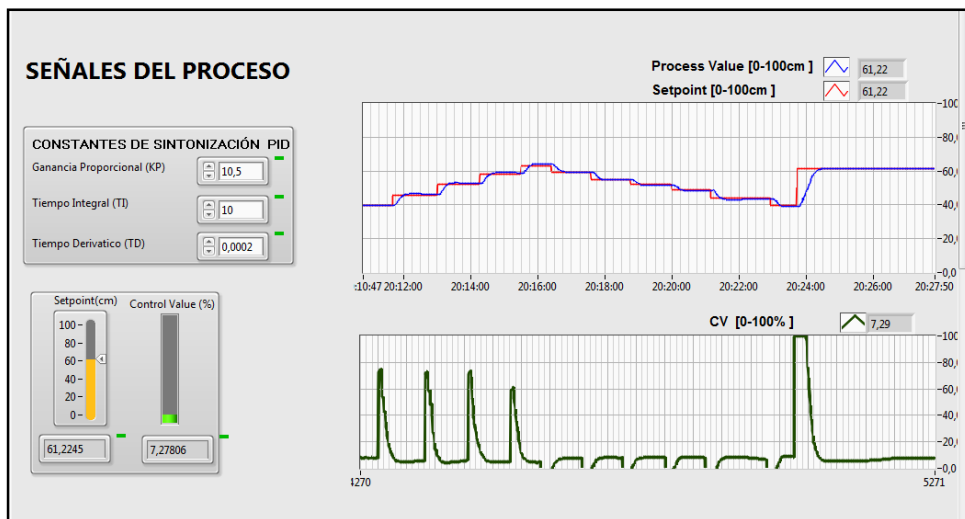


Figura 155: Resultados del controlador PID

3.8.4. Pruebas de funcionamiento de la simulación Nivel

Para verificar el correcto funcionamiento de la simulación del Proceso Nivel se realiza diferentes pruebas, para este caso se utiliza la simulación Nivel para el PLC Siemens S7-1200. El desarrollo de la programación ladder se realiza en el software de programación TIA PORTAL y como OPC se utiliza el KEPServerEx. Las pruebas efectuadas a la simulación mediante el S7-1200 son las mismas para el CompactLogix y el Micrologix. La diferencia entre cada uno de las simulaciones para cada tipo de PLC se indica en la **(Tabla 20)**.

Tabla 20

Variables del Proceso Nivel

NOMBRE	Variables del Proceso Nivel			DESCRIPCIÓN
	DIRECCIÓN PARA CADA PLC			
	Micrologix	CompactLogix	S7-1200	
PV	F8:1	PV	MD0	Variable del Proceso
SP	F8:2	SP	MD8	Variable de Setpoint
CV	N7:2	CV	MD4	Variable de Control
KP	F8:3	KP	MD12	Constante Proporcional
TI	F8:4	TI	MD16	Tiempo Integral
TD	F8:5	TD	MD20	Tiempo Derivativo

Luego, se realiza las pruebas al proceso Nivel mediante la programación ladder del proceso. En la **(Figura 156)** se muestra el panel frontal del proceso Nivel en este caso la pestaña presentación en funcionamiento, en donde se ve la animación del proceso y la programación ladder del controlador PID en el TIA PORTAL.

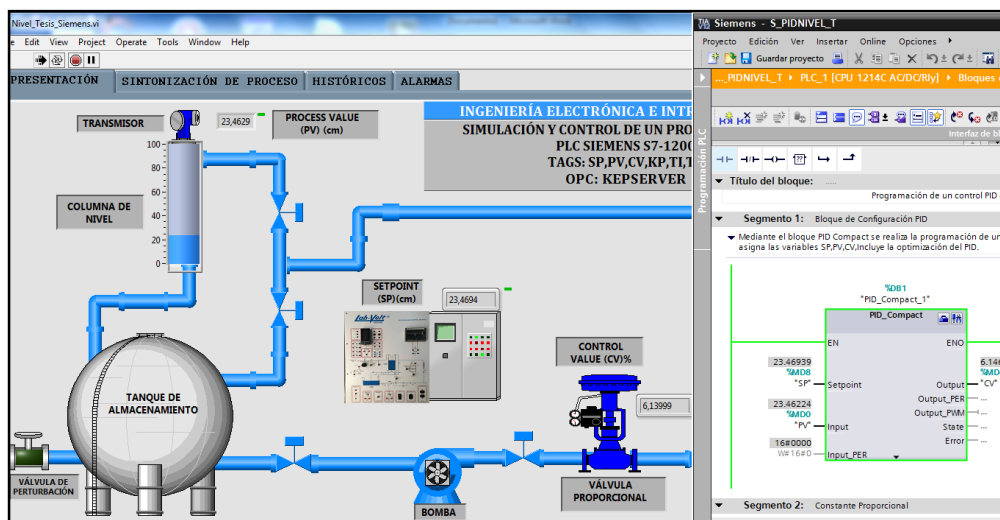


Figura 156: Presentación del Proceso Nivel-PLC Siemens

En la siguiente **(Figura 157)** se muestra el panel frontal del proceso Nivel en este caso la pestaña Sintonización de Proceso en funcionamiento, en donde se realiza la sintonía del controlador PID y la programación ladder del controlador PID.

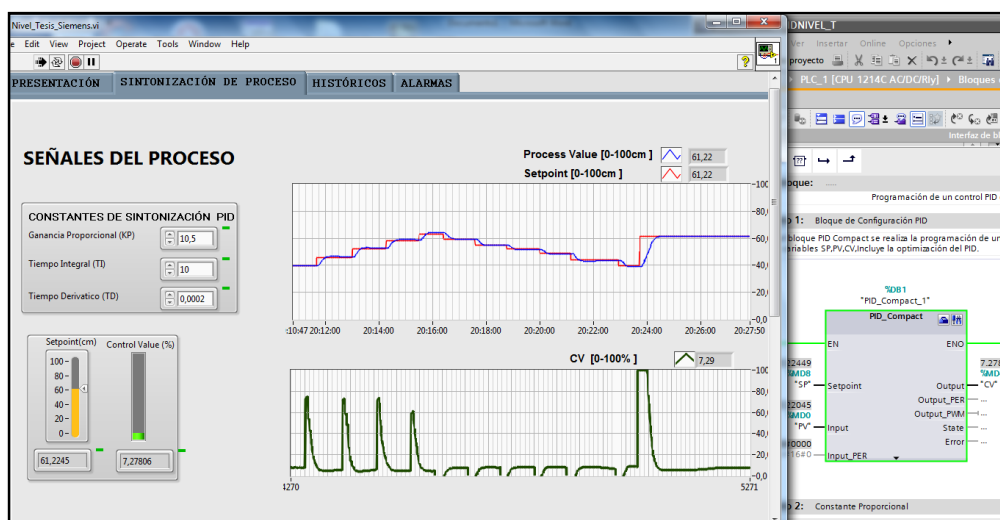


Figura 157: Sintonización de Proceso Nivel-PLC Siemens

La (Figura 158) se muestra el panel frontal del proceso Nivel en este caso la pestaña Históricos en funcionamiento, en donde presenta todos los cambios de las variables del Proceso Nivel y la programación ladder del controlador PID.

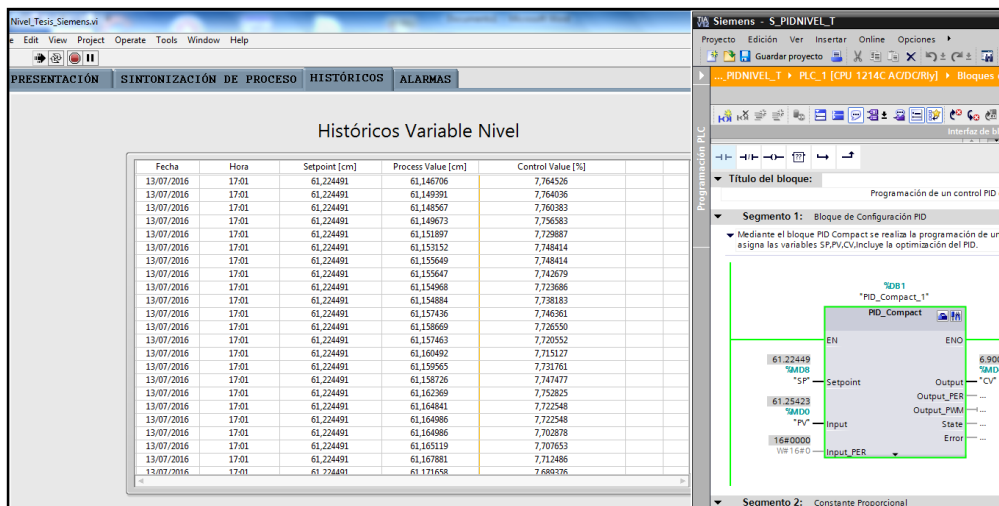


Figura 158: Históricos del Proceso Nivel-PLC Siemens

En la siguiente (Figura 159) se muestra el panel frontal del proceso Nivel en este caso la pestaña Alarmas en funcionamiento, en donde se puede observar la activación de las alarmas que fueron configuradas para este proceso y la programación ladder del controlador PID.

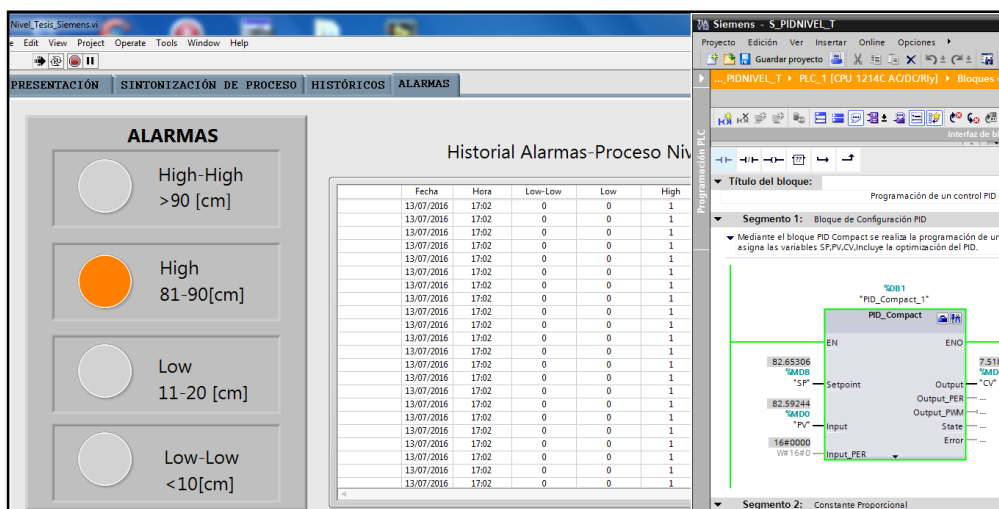


Figura 159: Alarmas del Proceso Nivel-PLC Siemens

3.9. Pruebas Experimentales del funcionamiento de la simulación del Proceso Flujo

3.9.1. Descripción del funcionamiento del proceso

La simulación del proceso Flujo consta de un menú en la primera pestaña se encuentra la presentación del Proceso, aquí se observa la animación del proceso, en la segunda pestaña se realiza la sintonización del proceso y se observa las tendencias del proceso Flujo, en la tercera pestaña el usuario puede observar los históricos del Proceso, y finalmente en la última pestaña se encuentra las alarmas. El proceso Flujo posee un rango de 0 a 80 GPM.

3.9.2. Diagrama Ladder

La programación del controlador se realizó mediante el uso del bloque PID Compact el cual se encuentra en las librerías del software TIA PORTAL, además las constantes de sintonización K_p , T_i , T_d se encuentran direccionadas hacia el bloque PID mediante la instrucción MOVE. (**Figura 160**)

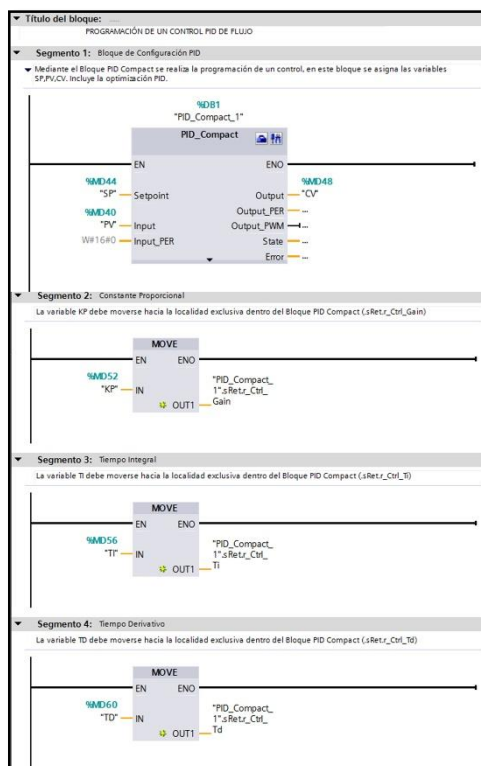


Figura 160: Programación Ladder del Proceso Flujo

3.9.3. Sintonización del Proceso Flujo

Al igual que el proceso Nivel para la sintonización del lazo de control PID implementado en la simulación de este proceso Flujo se utilizó el método de tanteo. El cual consiste en la aplicación de los siguientes pasos. Lo primero que se debe realizar es colocar un valor de K_p (ganancia proporcional) hasta que la respuesta del sistema presente una oscilación de forma estable, seguido a esto se ingresa un valor de T_d (Tiempo Derivativo) para lograr reducir el sobre impulso de la respuesta finalmente se coloca el valor de T_i (Tiempo Integral) cuyo objetivo es eliminar el error en estado estable. Después de haber aplicado el método de tanteo se obtuvo las constantes que se pueden observar en la **(Tabla 21)** las cuales permiten realizar la correcta sintonización del control PID de la simulación del Proceso Flujo.

Tabla 21

Constantes de sintonización de la simulación Flujo

PROCESO FLUJO		
Contantes	Valor	Dimensión
Constante Proporcional (K_p)	10	[adim]
Tiempo Integral (T_i)	20	[adim]
Tiempo Derivativo (T_d)	0.008	[adim]

Luego de colocar el valor de las constantes de sintonización se realiza la prueba del proceso Flujo mediante el cambio de valor de la variable Setpoint (SP) y se observa como la variable de proceso (PV) sigue el valor de consigna además se observa la gráfica de la Variable de control (CV) **(Figura 161)**.

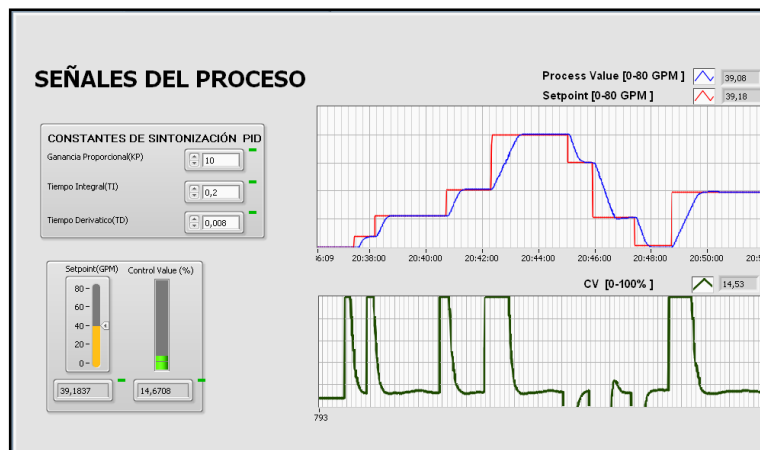


Figura 161: Resultados PID Flujo

Para verificar el correcto funcionamiento de la simulación del Proceso Flujo se realiza diferentes pruebas, para este caso se utiliza la simulación Flujo para el PLC CompactLogix. El desarrollo de la programación ladder se realiza en el software de programación RSLogix 5000 y como OPC se utiliza el RSLinx. Las pruebas efectuadas a la simulación mediante el CompactLogix son las mismas para el S7-1200 y el Micrologix. La diferencia entre cada uno de las simulaciones para cada tipo de PLC radica en las direcciones que poseen cada una de las variables del Proceso Mezclador como se indica en la **(Tabla 22)**.

Tabla 22

Variables del Proceso Flujo

Variables del Proceso Flujo				
NOMBRE	DIRECCIÓN PARA CADA PLC			DESCRIPCIÓN
	Micrologix	CompactLogix	S7-1200	
	x	x		
PV	F8:1	PV	MD40	Variable del Proceso
SP	F8:2	SP	MD44	Variable de Setpoint
CV	N7:2	CV	MD48	Variable de Control
KP	F8:3	KP	MD52	Constante Proporcional
TI	F8:4	TI	MD56	Tiempo Integral
TD	F8:5	TD	MD60	Tiempo Derivativo

Luego, se realiza las pruebas al proceso Flujo mediante la programación ladder del proceso. En la **(Figura 162)** se muestra el panel frontal del proceso Flujo en este caso la pestaña presentación en funcionamiento, en donde se ve la animación del proceso y la programación ladder del controlador PID en el RSLogix 5000.

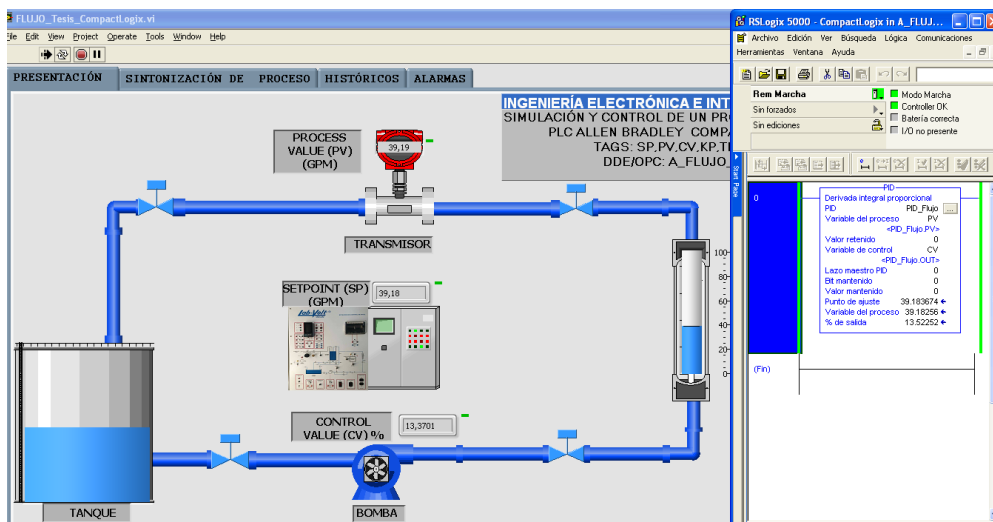


Figura 162: Presentación del Proceso Flujo-Compactlogix

En la siguiente **(Figura 163)** se muestra la pestaña Sintonización de Proceso en funcionamiento, en donde se realiza la sintonía del controlador PID y la programación ladder del Proceso Flujo.

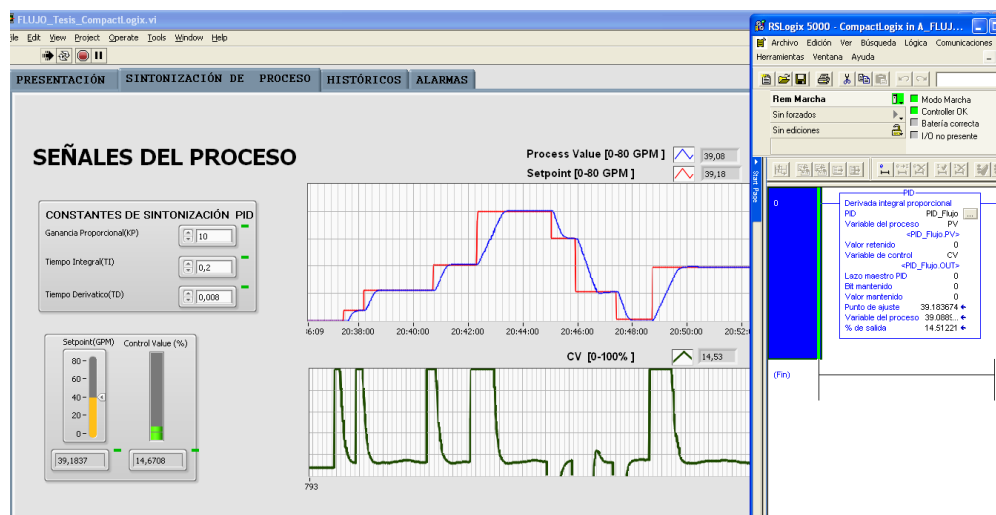


Figura 163: Sintonización del Proceso Flujo-Compactlogix

3.10. Alcances y Limitaciones

3.10.1. Alcances

En el diseño e implementación del simulador de procesos industriales se determinaron los siguientes alcances:

- El simulador brinda un entorno amigable e intuitivo al usuario, facilitando el aprendizaje del funcionamiento de procesos de automatización.
- Permite desarrollar la destreza de programación en automatización de procesos y controles para diferentes PLC usados en la industria.
- Comprensión de la utilización de sensores actuadores e indicadores en un proceso de automatización.
- El simulador de Procesos Industriales es compatible con los PLC Siemens S7-1200, CompactLogix, Micrologix.
- Permite el monitoreo y control de procesos de Nivel y Flujo.
- El simulador posee un HMI muy intuitivo para el usuario.
- El simulador es compatible con los sistemas operativos XP, Windows 7, Windows 8.

3.10.2. Limitaciones

De igual manera se pudo observar limitaciones del simulador de Procesos Industriales, mismas que se detallan a continuación:

- La programación del simulador no puede ser modificada, debido a que es un archivo ejecutable.
- El usuario no puede asignar cualquier nombre de enlace OPC porque este ya se encuentra previamente configurado.
- El usuario no puede modificar el nombre de las tag's debido a que se encuentran configuradas previamente.
- Este simulador es compatible solamente con los OPC KEPServerEx y RSLinx siendo los más utilizados.

CAPÍTULO IV

CONCLUSIONES Y RECOMENDACIONES

4.1. Conclusiones

Luego de culminar el proyecto se obtienen las siguientes conclusiones:

4.1.1. Simulador

- Se ha diseñado e implementado un simulador de procesos que permite desarrollar destrezas a la hora de realizar la programación del PLC Siemens y Allen Bradley los cuales se encuentran en el laboratorio de Control de Procesos y Redes Industriales de la Universidad de las Fuerzas Armadas ESPE Extensión Latacunga.
- Se ha diseñado e implementado la simulación de procesos de control industrial que facilitan el aprendizaje de automatización de los futuros profesionales sin hacer el uso de procesos físicos.
- Se ha implementado las simulaciones Silo, Garaje, Mezclador, Semáforo, Nivel, Flujo que permiten la familiarización con sensores y actuadores relacionados en automatización industrial.
- Para la puesta en marcha de los procesos digitales en el PLC Allen Bradley, estos pueden funcionar con el PLC emulado y físico, optimizando el tiempo de los equipos en el laboratorio.
- Los procesos diseñados tienen una complejidad baja, media y avanzada por las opciones de programación y las formas de funcionamiento.
- Los procesos digitales y analógicos son desarrollados en el software Labview facilitando la creación de simulaciones con la ayuda del módulo DSC y Control and Design Simulation.
- Para realizar la programación de los PLC Siemens S7-1200 se utiliza el software de programación TIA PORTAL.
- Mediante los procesos Flujo y Nivel es posible realizar evaluaciones de controles PID, además se realiza la sintonización de los procesos simulados anteriormente mencionados.

4.2. Recomendaciones

- Tener en cuenta la conexión de las tag's al PLC considerando que hay que escribir correctamente los nombres establecidos en cada proceso de acuerdo al tipo de PLC.
- Para reiniciar la simulación también se debe reiniciar la programación del PLC para evitar conflictos de funcionamiento.
- Es necesario establecer correctamente la configuración del servidor OPC con el simulador para evitar fallos.
- Es necesario tener conocimientos básicos de Programación de PLC's para evitar mal manejo del simulador.
- Es importante aclarar que para detener la simulación de un Proceso Digital no se debe utilizar el botón stop que se encuentra en la barra de menú de Labview.

BIBLIOGRAFÍA

Ruiz Canales, A., Molina Martínez, J. (2010). *Automatización y telecontrol de sistemas de riego*. España. Marcombo.

Lajara Vizcaíno, J., Pelegrí, J. (2011). *Labview : entorno gráfico de programación*. España. Marcombo.

LINKOGRAFÍA

Bastidas Roberto & Proaño Luis, B. (Septiembre de 2010). *DISEÑO Y CONSTRUCCIÓN DE MÓDULOS DIDÁCTICOS PARA EL LABORATORIO DE CONTROLADORES LÓGICOS PROGRAMABLES PARA LA ESCUELA POLITÉCNICA DEL EJÉRCITO SEDE LATACUNGA*. Recuperado el 05 de Marzo de 2016, de repositorio.espe.edu.ec/bitstream/21000/8104/1/T-ESPEL-0754.pdf

Calvo Guadaño, G. (Julio de 2013). *SIMULACIÓN DE LA AUTOMATIZACIÓN DE PROCESOS CON UNITY PRO*. Recuperado el 18 de Mayo de 2016, de http://e-archivo.uc3m.es/bitstream/handle/10016/19868/TFG_Guillermo_Calvo_Guadano.pdf?sequence=1

DYNSIM. (2015). Recuperado el 22 de Abril de 2016, de http://iom.invensys.com/LA/Pages/SimSci_DynSimSuite_DYNSIM.aspx

Gutiérrez, J. (2008). *La simulación como Instrumento de Aprendizaje. Evaluación de Herramientas y estrategias de aplicación en el aula*. Recuperado el 21 de Abril de 2016, de http://fp.atxuri.net/escenarios/Simulacion_como_Instrumento_de_Aprendizaje.pdf

Kepware Technologies. (2016). *KeptserverEx*. Recuperado el 09 de Marzo de 2016, de <https://www.kepware.com/products/keptserverex/>

Labview . (2014). *National Instruments*. Recuperado el 15 de Mayo de 2016, de <http://www.ni.com/labview/why/esa/>

Moreno Valery. (Agosto de 2011). *Sistemas SCADAS*. Recuperado el 09 de 03 de 2016, de <http://ariadne.cti.espol.edu.ec/>

- National Instruments, N. (2014). *Módulo LabVIEW Datalogging and Supervisory Control (DSC)*. Recuperado el 30 de Abril de 2016, de <http://www.ni.com/labview/labviewdsc/esa/>
- National Instruments, N. (2014). *Módulo NI LabVIEW Control Design and Simulation para Windows*. Recuperado el 30 de Abril de 2016, de <http://sine.ni.com/nips/cds/view/p/lang/es/nid/209850>
- Navas Alex& Zambrano Jorge. (2015). *Diseño e implementación de una red industrial inalámbrica wirelesshart para el monitoreo de las variables temperatura, presión, caudal y el monitoreo y el control de la variable nivel del Laboratorio de redes industriales y control de procesos de la Unive*. Recuperado el 15 de Marzo de 2016, de <http://repositorio.espe.edu.ec/handle/21000/10066>
- ProModel. (2014). *Promodel México*. Recuperado el 22 de Abril de 2016, de <http://www.promodel.com.mx/promodel.php>
- Rockwell Automation. (2004). *RSLogix Emulate 500*. Recuperado el 02 de 05 de 2016, de http://www.software.rockwell.com/download/logic/rslogix-emulate/emulate_td.pdf
- Rockwell Automation. (2008). Recuperado el 05 de Marzo de 2016, de RSLINX CLASSIC: CÓMO OBTENER RESULTADOS CON RSLINX CLASSIC:
http://literature.rockwellautomation.com/idc/groups/literature/documents/gr/lnx-gr001_-es-e.pdf
- Rockwell Automation. (2010). *RSLogix Emulate 5000*. Recuperado el 03 de Marzo de 2016, de <https://rockwellautomation.custhelp.com/ci/fattach/get/96143/>
- Rockwell Automation. (2015). *RSLogix 500 Getting Results Guide*. Recuperado el 02 de Mayo de 2016, de http://literature.rockwellautomation.com/idc/groups/literature/documents/gr/lg500-gr002_-en-e.pdf
- Rockwell Automation. (2016). *Controladores CompactLogix L32x y L3X*. Recuperado el 14 de Marzo de 2016, de <http://ab.rockwellautomation.com/es/Programmable-Controllers/CompactLogix-1769-Controllers#overview>

- Rockwell Automation. (2016). *Sistemas de controlador lógico programable MicroLogix 1200*. Recuperado el 16 de Marzo de 2016, de <http://ab.rockwellautomation.com/es/Programmable-Controllers/MicroLogix-1200#documentation>
- Salas Ramón, A. P. (1995). *La simulación como método de enseñanza y aprendizaje*. Recuperado el 21 de Abril de 2016, de http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S0864-21411995000100002
- SIEMENS. (2016). *Siemens-Plan Simulation*. Recuperado el 22 de Abril de 2016, de https://www.plm.automation.siemens.com/es_sa/products/tecnomatix/manufacturing-simulation/material-flow/plant-simulation.shtml
- Siemens AG. (Noviembre de 2009). Recuperado el 11 de Marzo de 2016, de <https://w5.siemens.com/spain/web/es/industry/automatizacion/simatic/Documents/S71200-MANUAL%20DEL%20SISTEMA.PDF>
- Siemens AG. (2013). *Simatic S7-1200*. Recuperado el 12 de Marzo de 2016, de http://www.coevagi.com/Docs/Si_S7-1200.pdf
- Siemens AG. (2013). *Simatic Step 7 en el TIA PORTAL*. Recuperado el 06 de Marzo de 2016, de https://www.automation.siemens.com/salesmaterial-as/brochure/es/brochure_simatic-step7_tia-portal_es.pdf
- Thelearningpit. (2012). *LogixPro500*. Recuperado el 24 de Abril de 2016, de <http://thelearningpit.com/lp/logixpro.html>

ANEXOS



**DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA
CARRERA DE ELECTRÓNICA E INSTRUMENTACIÓN**

CERTIFICACIÓN

Se certifica que el presente trabajo fue desarrollado por los señores:

Oscar Fabricio Bayas Sánchez y Henry Paúl Cocha Apupalo

En la ciudad de Latacunga, a los 18 días del mes de agosto del 2016.

**Ing. Edwin Patricio Pruna Panchi
DIRECTOR DEL PROYECTO**

Aprobado por:

**Ing. Franklin Silva Monteros
DIRECTOR DE CARRERA**

**Dr. Rodrigo Vaca
SECRETARIO ACADÉMICO**