



**ESCUELA SUPERIOR POLITECNICA DE CHIMBORAZO**

**FACULTAD DE INFORMATICA Y ELECTRONICA**

**ESCUELA DE INGENIERIA EN SISTEMAS**

**“IMPLEMENTACION DE UN SISTEMA MECATRONICO PARA EL  
MEZCLADO DE LIQUIDOS, CASO PRÁCTICO LABORATORIO DE  
AUTOMATIZACIÓN INDUSTRIAL DE LA EIS”**

**TESIS DE GRADO**

**PREVIA LA OBTENCION DEL TITULO DE  
INGENIERO EN SISTEMAS INFORMATICOS**

**MARIANA MERCEDES MOROCHO MISAL**

**MONICA MERCEDES URQUIZO CRUZ**

**RIOBAMBA – ECUADOR**

**- 2010 -**

## **AGRADECIMIENTO**

En primer lugar agradezco al Director de la Tesis Ing. Marco Viteri, al Ing. Danny Velasco miembro de la Tesis por la paciencia, comprensión y por habernos guiado para la culminación de este tema, a mi mami, mi hermana y mi esposo quienes me apoyaron incondicionalmente en todo momento.

**Mariana Mercedes Morocho Misal**

## **AGRADECIMIENTO**

Agradecimientos a la Escuela Superior Politécnica de Chimborazo, por ser forjadora de mi educación, a todos los maestros que aportaron con sus conocimientos para que pueda culminar esta etapa de mi vida; en especial al Ing. Marco Viteri, miembro de la Tesis de Grado, por su tiempo y ayuda brindada.

A mis queridos padres por ser el sostén de mi vida y darme su apoyo incondicional así a mis hermanos por su apoyo desinteresado.

**Mónica Mercedes Urquizo Cruz**

## **DEDICATORIA**

Dedico el esfuerzo a mis padres Pascual y Juana quienes con nobleza, entusiasmo y mucho sacrificio hicieron posible la culminación de una etapa importante en mi vida, a mis dos hermanas María Humbelina y Susana quienes creyeron en mí en todo momento, a mi hija que es el motivo principal para llegar a la meta trazada, a mi esposo que me ha apoyado moralmente.

A mi padre que desde el cielo me guía y me ayuda siempre, quien con cariño, devoción y entusiasmo me brindó todo su sustento para culminar mi carrera, depositando toda su confianza, y aunque él este ausente siempre lo tendré en lo más profundo de mi corazón, ya que sin su apoyo constante, no hubiese alcanzado mi meta.

.

**Mariana Mercedes Morocho Misal**

## **DEDICATORIA**

Este trabajo está dedicado a mis padres en especial a mi madre Maria Cruz que ha estado en todo momento y lugar siempre apoyándome incondicionalmente sin importar a veces mi mala conducta. Gracias por su apoyo y su valioso aporte en mi vida.

Dedico este trabajo especialmente a mi hija Isabel que es y ha sido mi inspiración, la luz de mi vida y mi razón de ser, también a mi esposo Ivan por su comprensión y ayuda para la realización del presente trabajo.

**Mónica Mercedes Urquizo Cruz**

**FIRMAS RESPONSABLES Y NOTAS**

<b>NOMBRE</b>	<b>FIRMA</b>	<b>FECHA</b>
Ing. Iván Menes DECANO FACULTAD DE INFORMÁTICA Y ELECTRÓNICA	.....	.....
Ing. Raul Rosero DIRECTOR DE LA ESCUELA INGENIERÍA EN SISTEMAS	.....	.....
Ing. Marco Viteri DIRECTOR DE TESIS	.....	.....
Ing. Danny Velazco MIEMBRO DEL TRIBUNAL	.....	.....
Tlgo. Carlos Rodríguez DIRECTOR DEL CENTRO DE DOCUMENTACIÓN	.....	.....
NOTA DE LA TESIS	.....	

## **RESPONSABILIDAD DE LOS AUTORES**

Bajo las leyes de derecho de autor, la Documentación, el Software, junto con todos sus elementos, no pueden copiarse, reproducirse, registrarse o transmitirse por un sistema de recuperación de información, en ninguna forma parcial o total, ni por ningún medio, sea electrónico, mecánico, foto-químico, magnético o por fotocopia, sin permiso previo por escrito de los autores.

---

**Mariana Mercedes Morocho Misal**

---

**Mónica Mercedes Urquizo Cruz**

## INDICE DE ABREVIATURAS

<b>AC</b>	<b>Corriente Alterna</b>
<b>ASCII</b>	<b>American Estándar Code for Information Interchange</b>
<b>AR</b>	<b>Área auxiliar</b>
<b>BD</b>	<b>Base de Datos</b>
<b>BIT</b>	<b>Binary Digit</b>
<b>COM</b>	<b>Component Object Model</b>
<b>CPU</b>	<b>Central Process Unit</b>
<b>DBMS</b>	Database management system, Sistema de administración de bases de datos
<b>DC</b>	<b>Corriente Directa</b>
<b>DM</b>	<b>Área de datos</b>
<b>HR</b>	<b>Área de retención</b>
<b>IL</b>	<b>Lista de Instrucciones</b>
<b>I/O</b>	<b>Input/Output</b>
<b>IEEE</b>	<b>Instituto de ingenieros Eléctricos y Electrónicos</b>
<b>LabVIEW</b>	<b>Laboratory Virtual Instrument Engineering Workbench</b>
<b>LAN</b>	<b>Local Area Network</b>
<b>LR</b>	<b>Área de enlace</b>
<b>PC</b>	<b>Computador Personal</b>
<b>PLC</b>	<b>Programmable Logic Control</b>
<b>RAM</b>	<b>Random Access Memory</b>
<b>ROM.</b>	Memoria de solo lectura, no reprogramable.
<b>SCADA</b>	<b>Supervisory Control and Data Acquisition</b>



<b>SR</b>	<b>Área especial</b>
<b>TIM/CNT</b>	<b>Área de temporizadores y contadores</b>
<b>VI</b>	<b>Virtual Instrument</b>

## ÍNDICE GENERAL

**PORTADA**

**AGRADECIMIENTO**

**DEDICATORIA**

**ÍNDICE DE ABREVIATURAS**

**ÍNDICE GENERAL**

<b>CAPITULO I : MARCO REFERENCIAL .....</b>	<b>21</b>
<b>1.1. Introducción .....</b>	<b>21</b>
<b>1.2. Antecedentes .....</b>	<b>22</b>
<b>1.3. Lugar De Aplicación .....</b>	<b>23</b>
<b>1.4. Alcance .....</b>	<b>23</b>
<b>1.5. Justificación .....</b>	<b>24</b>
1.5.1. Justificación Teórica.....	24
1.5.2. Justificación Aplicativa .....	24
<b>1.6. Objetivos .....</b>	<b>25</b>
1.6.1. Objetivo General .....	25
1.6.2. Objetivos Específicos .....	25
<b>1.7. Hipótesis .....</b>	<b>26</b>
<b>CAPITULO II SISTEMAS DEL MODULO .....</b>	<b>27</b>
<b>2.1. Introducción .....</b>	<b>27</b>
<b>2.2. Sistema Eléctrico .....</b>	<b>27</b>
2.2.1 Sensores .....	28
2.2.1.1 Principios de Funcionamiento.....	28
2.2.1.2 Tipos .....	29
2.2.1.3 Aplicaciones.....	35
2.2.2 Conductores Eléctricos .....	36
2.2.2.1 Clases .....	36
2.2.2.2 Codificación.....	43

2.2.2.3 Aplicaciones.....	43
<b>2.3 Sistema Neumático .....</b>	<b>44</b>
2.3.1 Motores Neumáticos.....	44
2.3.1.1. Principios de Funcionamiento.....	45
2.3.1.2. Tipos .....	46
2.3.1.3. Aplicación .....	48
2.3.2. Válvulas.....	49
2.3.2.1. Principios de Funcionamiento.....	49
2.3.2.2. Tipos .....	52
2.3.2.3. Aplicaciones.....	57
<b>2.4. Sistema Mecánico .....</b>	<b>58</b>
2.4.1. ¿Estructura Metálica construida en aluminio .....	58
2.4.2. Características del aluminio .....	58
2.4.3. Ventajas del Aluminio .....	61
<b>2.5. Sistema Informático .....</b>	<b>61</b>
2.5.1. PLC.....	61
2.5.1.1. Definición .....	61
2.5.1.2. Marcas – Tipos.....	62
2.5.1.3. Estructura del PLC .....	63
2.5.1.4. Principio de funcionamiento del PLC.....	78
2.5.1.5. Clases de Programación .....	80
2.5.1.6. Aplicación .....	81
2.5.2. Diseño.....	82
2.5.3. Base de Datos .....	84
<b>CAPITULO III PARTE INVESTIGATIVA .....</b>	<b>87</b>
<b>3.1. Introducción .....</b>	<b>87</b>

<b>3.2. Análisis del estándar IEC 611 31-3.....</b>	<b>88</b>
3.2.1 La Norma IEC-61131 se refiere a: .....	88
3.2.2. La finalidad de esta Norma IEC-61131 es: .....	89
3.2.3 Lenguajes de Programación .....	96
<b>3.3. Control y monitoreo .....</b>	<b>109</b>
3.3.1 Protocolos de comunicación.....	109
3.3.2. protocolo Modbus.....	113
<b>3.4. Lenguajes de Programación.....</b>	<b>118</b>
3.4.1. Basic / Visual Basic .....	118
3.4.2. Java .....	120
3.4.3. Pascal / Delphi .....	121
3.4.4. C#.....	123
3.4.5. LabVIEW.....	124
<b>3.5. Elección del lenguaje de programación.....</b>	<b>127</b>
<b>3.6. Análisis de las metodologías existentes para sistemas mecatrónicos.....</b>	<b>128</b>
3.6.1. Según La Universidad De Sevilla.....	129
3.6.1.1. Fase de Especificación de los requisitos software. ....	129
3.6.1.2. Fase de Diseño del Sistema de Control.....	130
3.6.1.3. Fase de Diseño de la Arquitectura Software.....	135
3.1.6.4. Fase de Diseño de los Módulos o Unidades de Programa.....	138
3.6.1.5. Fase de Codificación (programación).....	142
3.6.2. Según José Guadalupe Castro Lugo, Juan José Padilla Ybarra, Eduardo Romero A .....	144
3.6.2.1. Descripción Del Sistema.....	144
3.6.2.2. Diagrama De Flujo .....	145
3.6.2.3. Descripción De Los Equipos Del Sistema.....	145
3.6.2.4. Requerimientos Del Cliente.....	145

3.6.2.5. Selección Del Autómata Programable .....	146
3.6.2.6. Programación del plc. ....	147
<b>3.7. Descripción de la solución planteada en la investigación .....</b>	<b>148</b>
<b>3.8. Propuesta metodológica para la implementación de proyectos mecatrónicos.</b> .....	<b>149</b>
<b>CAPITULO IV DESARROLLO DE UN SISTEMA MECATRÓNICO PARA EL MEZCLADO DE LÍQUIDOS .....</b>	<b>155</b>
<b>4.1. Introducción .....</b>	<b>155</b>
4.1.1. Componentes Del Modulo De La Mezcladora Automática .....	156
4.1.2. Descripción de los dispositivos utilizados para el ensamblaje del mezclador .....	157
4.1.2.1. Sensor sonda .....	157
4.1.2.2. Sensor Flotador Camsco PD-76AB .....	158
4.1.2.3. Bomba eléctrica Flojet modelo: P46XD094 .....	159
4.1.2.4. PLC Telemecanique TWDLCAE40DRF .....	160
4.1.2.5. Válvulas 4V120-06 .....	160
4.1.2.6. Válvula AV210 .....	161
4.1.2.7. Sócalos 3UG05-02-2AF00-A7B10000052296.....	162
4.1.2.8. Relé controlador de nivel Siemens RXM4AB1BD .....	162
4.1.2.9. Relé controlador de nivel Siemens RXM4AB2BD .....	163
4.1.2.10. Concentrador de señales: ABE7 H16R31 .....	163
4.1.2.11. Pretul .....	165
4.1.2.12. Compresor .....	166
4.1.2.13. Mezclador .....	167
4.1.2.14. Break para el control del voltaje de 110 voltios .....	168
<b>4.2. Planificación del Proyecto .....</b>	<b>168</b>
4.2.1. Descripción del Sistema .....	168

4.2.2. Especificación de Requerimientos .....	169
4.2.3. Planificación Inicial.....	169
4.2.4. Plan de Iteraciones (Descripción de cada Iteración) .....	169
<b>4.3. Diseño .....</b>	<b>171</b>
4.3.1. Diseño Mecatrónico.....	171
4.3.1.1. Diseño Mecánico .....	171
4.3.1.2. Diseño Eléctrico.....	174
4.3.1.3. Grafcet.....	179
4.3.2. Diseño Software .....	180
4.3.2.1. Diagrama de casos de uso .....	180
4.3.2.2. Diagramas de Secuencia .....	180
4.3.2.3. Diagramas de Actividades .....	182
4.3.2.4. Diagrama de Clases.....	183
4.3.2.5. Diagrama de Componentes .....	183
4.3.2.6. Diccionario de Datos.....	184
4.3.2.7. Modelo Lógico de la Base de Datos .....	184
<b>4.4. Implementación .....</b>	<b>184</b>
4.4.1. Procesos del Mezclado .....	184
4.4.1.1. Llenado de tanques .....	185
4.4.1.2. Bombeo de líquido(Revisar no entiendo) .....	186
4.4.1.3. Mezclado de líquido.....	186
4.4.1.4. Detección de nivel.....	187
4.4.1.5. Sistema de llenado .....	188
4.4.1.6. Modulo de control (Eléctrico, Neumático e Informático). .....	188
4.4.2. Estructura de Aluminio.....	189
4.4.3. Implementación Mecatrónica .....	190

4.4.3.1. Sistema Mecánico .....	190
4.4.3.2. Sistema Neumático .....	191
4.4.3.3. Sistema Eléctrico .....	192
4.4.3.4. Programación del PLC.....	193
4.4.3.5. Control del PLC .....	193
4.4.4. Implementación Software.....	194
4.4.4.1. Modelo Físico de la Base de Datos.....	194
4.4.4.2. Prototipos de Interfaces de Usuario .....	195
<b>4.5. Pruebas del Sistema .....</b>	<b>196</b>
4.5.1. Pruebas Hardware.....	196
4.5.2. Pruebas Software .....	197
<b>4.6. Implantación.....</b>	<b>197</b>
4.6.1. Manual de Configuración .....	197
4.6.1.1. Requerimientos.....	197
4.6.1.2. Configuración del equipo que albergará el Panel de Control.....	198
4.6.2. Manual de Usuario.....	199
<b>4.7. Análisis De Datos Y Comprobación De Hipótesis.....</b>	<b>199</b>
CONCLUSIONES .....	204
RECOMENDACIONES.....	206
RESUMEN .....	207
SUMARY .....	208
GLOSARIO .....	209
BIBLIOGRAFIA .....	211

## INDICE DE FIGURAS

Figura. II.1. Clasificación de sensores .....	29
Figura. II.2. Tipos de sensores de nivel .....	30
Figura. II.3. Sensor flotador .....	32
Figura. II.4. Flotante a cable .....	33
Figura. II.5. Flotante con cable .....	34
Figura. II.6. Flotante con columna magnética .....	34
Figura. II.7. Flotante lateral magnético.....	34
Figura. II.8. Sensor sonda .....	35
Figura. II.9. Partes del conductor eléctrico .....	37
Figura. II.10. Conductores eléctricos de acuerdo a su aislación o número de hebras ...	38
Figura. II.11. Cables para su distribución y poder.....	39
Figura. II.12. Cable armado .....	39
Figura. II.13. Conductores de control e instrumentación .....	40
Figura. II.14. Conductor cordones .....	40
Figura. II.15. Conductores portátiles .....	41
Figura. II.16. Cables submarinos .....	41
Figura. II.17. Cables navales .....	41
Figura. II.18. Funcionamiento del motor neumático 1 .....	45
Figura. II.19. Funcionamiento del motor neumático 2 .....	45
Figura. II.20. Motor de aletas .....	46
Figura. II.21. Motor axial.....	48
Figura. II.22. Válvula 3/2 inicialmente abierta accionada por rodillo.....	50
Figura. II.23. Regulador de caudal unidireccional.....	51
Figura. II.24. Regulación de la velocidad del vástago en su salida o entrada .....	51
Figura. II.25. Regulación de la velocidad del vástago en su entrada y salida .....	52
Figura. II.26. Válvula distribuidora 2/2 .....	53
Figura. II.27. Válvula 2/2 conectada a un cilindro por medio de tubos.....	54
Figura. II.28. Válvula distribuidora 4/2 .....	54
Figura. II.29. Funcionamiento de la válvula distribuidora 4/2 .....	54



Figura. II.30. Las 3 posiciones de la válvula 4/3 que esta conectada a un cilindro de doble efecto.....	55
Figura. II.31. Válvulas reguladoras de caudal .....	55
Figura. II.32. Funcionamiento de las válvulas reguladoras de caudal .....	56
Figura. II.33. Válvula reguladora de presión .....	56
Figura. II.34Válvula reg. uladora de presión .....	57
Figura. II.35. Válvulas de secuencia.....	57
Figura. II.36. Estructura de aluminio del sistema de mezclado de líquidos .....	58
Figura. II.37. Funcionamiento PLC .....	62
Figura. II.38. Estructura del PLC compacta .....	64
Figura. II.39. Ejecucion normal (ciclica) del PLC.....	79
Figura. II.40. Ejecución periódica del PLC .....	80
Figura. II.41. Tipos de Lenguajes de Programación.....	81
Figura. II.42. Panel frontal de LabVIEW .....	83
Figura. II.43. Diagrama de bloques de LabVIEW .....	84
Figura. II.44. Ejemplo de una Base de datos .....	85
Figura. III.1. El estándar IEC-61131 .....	90
Figura. III.2. Detalle Elementos comunes y lenguajes de programación .....	91
Figura. III.3. Lenguajes de Programación .....	97
Figura. III.4. Nomenclatura de IL Y ST .....	101
Figura. III.5. GRAFCET.....	104
Figura. III.6. GRAFCET Salto condicional de etapa.....	104
Figura. III.7. GRAFCET Direccinamiento condicional .....	105
Figura. III.8. Beneficios del estándar para los usuarios.....	106
Figura. III.9. Ejemplo de una conexión remota .....	110
Figura. III.10. Funciones del protocolo modbus.....	115
Figura. III.11. Ejemplo de comunicación modbus.....	116
Figura. III.12. Protocolo Modbus .....	117
Figura. III.13. Apariencia de LabVIEW .....	126
Figura. III.14. Etapas de la metodología Según La Universidad De Sevilla.....	129
Figura. III.15. Estructuras de control.....	133
Figura. III.16. Lenguaje de programación CFS .....	137

Figura. III.17. PC – entorno de programación 1 (a) .....	141
Figura. III.18. PC – entorno de programación 1 (b) .....	142
Figura. IV.1. Sensor sonda.....	157
Figura. IV.2. Sensor Flotador .....	158
Figura. IV.3. Características técnicas Sensor Flotador Camsco PD-76AB .....	159
Figura. IV.4. Bomba eléctrica Flojet .....	159
Figura. IV.5. Válvula 4V120-06 .....	160
Figura. IV.6. Válvula AV210 .....	161
Figura. IV.7. Relé RXM4AB2BD .....	163
Figura. IV.8. Concentrador de señales: ABE7 H16R31 .....	164
Figura. IV.9. Compresor .....	166
Figura. IV.10. Breaker .....	168
Figura. IV.11. Plan de Iteración 1 .....	170
Figura. IV.12. Plan de Iteración 2.....	170
Figura. IV.13. Vista frontal del Sistema de mezclado de líquidos .....	171
Figura. IV.14. Vista posterior del sistema de mezclado de líquidos .....	172
Figura. IV.15. Vista Superior del Sistema de mezclado de líquidos .....	172
Figura. IV.16. Vista lateral izquierda del Sistema de mezclado de Líquidos.....	173
Figura. IV.17. Vista lateral derecha del sistema de mezclado de líquido.....	173
Figura. IV.18. Diseño del Sistema Eléctrico del Mezclado de Líquidos.....	176
Figura. IV.19. Diseño del sistema de Control del Sistema de Mezclado de Líquidos 1 .....	177
Figura. IV.20. Diseño del sistema de Control del Sistema de Mezclado de Líquidos 2 .....	178
Figura. IV.21. GRAFCET del sistema de mezclado de líquidos.....	179
Figura. IV.22. Casos de usos del sistema de mezclado de líquidos.....	180
Figura. IV.23. Diagrama de secuencia sistema de mezclado de líquidos.....	181
Figura. IV.24. Diagrama de actividades sistema de mezclado de líquidos .....	182
Figura. IV.25. Diagrama de componentes sistema de mezclado de líquidos .....	183
Figura. IV.26. Modelo lógico de BD del sistema de mezclado de líquidos .....	184
Figura. IV.27. Modulo de control del sistema de mezclado de líquidos .....	188
Figura. IV.28. Estructura de Aluminio .....	190

Figura. IV.29. Implementación mecánica.....	191
Figura. IV.30. Manguereado de la Bomba eléctrica .....	191
Figura. IV.31. Manguereado de las válvulas .....	192
Figura. IV.32. Tubería plástica .....	192
Figura. IV.33. Conexión de Entradas y Salidas al PLC.....	192
Figura. IV.34. Conexión de entradas y salidas a las borneras .....	193
Figura. IV.35. Modelo físico del Sistema de Mezclado de líquidos.....	194
Figura. IV.36. Menú principal del Sistema de Mezclado de líquido .....	195
Figura. IV.37. Monitorización del Sistema de Mezclado de líquido.....	195
Figura. IV.38. Extracción de los datos de la base de datos del Sistema de Mezclado de líquido .....	196
Figura. IV.39. Carpeta Almacen_I copiada a la unidad C.....	198
Figura. IV.40. Representación Barras.....	203
Figura. IV.41. Representación Pastel.....	203

## ÍNDICE DE TABLAS

Tabla II.1. Principio de funcionamiento de los sensores .....	28
Tabla II.2. Sensores de nivel de líquidos .....	31
Tabla II.3. Características del aluminio y cobre .....	36
Tabla III.1. Instrucciones .....	102
Tabla IV.1. Especificaciones técnicas del PLC Telemecanique TWDLCAE40DRF .	160
Tabla IV.2- Características específicas de la válvula 4V120-06 .....	161
Tabla IV.3. Características específicas del Relé controlador de nivel Siemens RXM4AB1BD .....	162
Tabla IV.4. Características específicas del Relé controlador de nivel Siemens RXM4AB2BD .....	163
Tabla IV.5. Características Técnicas Concentrador de señales: ABE7 H16R31.....	164
Tabla IV.6. Características Técnicas complementarias Concentrador de señales: ABE7 H16R31 .....	165
Tabla IV.7. Características técnicas Breaker .....	168
Tabla IV.8. Asignación de las entradas del sistema de mezclado .....	175
Tabla IV.9. Asignación de las salidas del sistema de mezclado.....	175
Tabla IV.10. Asignación de las memorias del sistema de mezclado.....	175
Tabla IV.11. Análisis Cualitativo de Juicio de Expertos.....	201
Tabla IV.12. Tabla Parámetros de calificación del análisis cuantitativo.....	201
Tabla IV.13. Tabla Análisis Cuantitativo del Juicio de Expertos.....	201
Tabla IV.14. Resultados de la encuesta .....	203

# CAPITULO I : MARCO REFERENCIAL

## 1.1. Introducción

Hoy en día la preferencia actual de las empresas es incorporar automatización industrial a sus procesos, para la optimización de recursos y llegar a ser más competitivos, es por esta razón por la que nos hemos planteado el desarrollo de un sistema de mezclado de líquidos, el mismo que será desarrollado en el Laboratorio de Automatización Industrial de la EIS, en el cual se combina la Mecatrónica con la Informática en busca de brindar una alternativa optima para las empresas. Para lo cual se definirá las metas que se quiere cumplir con el proyecto, de acuerdo a una adecuada planificación del mismo.

## 1.2. Antecedentes

Debido a que muchas de las empresas industriales necesitan mejorar sus procesos de control de manera eficiente, rápida y precisa; tales como aquellas que se dedican a elaborar productos químicos, farmacéuticos y cosméticos, entre otras, ya que es importante que realicen un mezclado de líquidos de forma homogénea para obtener un producto de calidad para los consumidores.

Las distintas empresas que utilizan el control eléctrico, pueden simplificar la línea de producción pero el inconveniente aparece en que ocupan mucho espacio físico y para poder realizar el recorrido del sistema crearon un proceso complejo.

A medida del avance tecnológico en la informática fue fusionando sus aplicaciones con el uso mecánico en las empresas. Hoy por hoy todas las empresas han modificado los procesos antiguos por los nuevos autómatas o PLCs la misma que brinda varios beneficios como la reducción del espacio físico y la complejidad de la implementación.

La automatización industrial brinda múltiples beneficios en el campo ocupacional y frente a la necesidad de fortalecer el aprendizaje, se proyecta implementar un sistema de mezclado de líquidos de manera automática, este módulo permitirá establecer un espacio práctico para los estudiantes de la EIS, pues permitirá entender, aprender y aplicar en la vida profesional cuando cada uno de ellos acudan a dichas empresas industriales para emprender la profesión.

Debido a que este módulo, mezclado de líquidos es el complemento del módulo de la envasadora automática la misma que ya está siendo ejecutada.

### **1.3. Lugar De Aplicación**

La presente investigación formará parte del sistema de la envasadora, el mismo que ya se encuentra desarrollado y funcionando en el laboratorio de Automatización Industrial de la Escuela de Ingeniería en Sistemas de la ESPOCH.

### **1.4. Alcance**

El Módulo de mezclado de líquidos automático representa a un sistema de producción en el perfil de mezclado de líquidos livianos el mismo que es controlado automáticamente utilizando un PLC y sus interfaces.

Este módulo está compuesto por subsistemas los cuales son:

- ✓ Bombeo de liquido
- ✓ Llenado de tanques
- ✓ Mezclado de liquido
- ✓ Detección de nivel
- ✓ Sistema de llenado
- ✓ Modulo de control (Eléctrico, Neumático e Informático).

El conjunto de los subsistemas en su totalidad forma el módulo de la mezcladora de líquidos, el mismo que funciona de la siguiente manera: el bombeo de líquidos proporciona el líquido para poder hacer efectivo el llenado de los envases o

tanques, consiguiendo al llenado de los tanques se procede al mezclado de líquidos, las mismas que es controlada por los sensores para la detección de nivel del líquido obteniendo una mezcla homogénea.

Este sistema proporciona el líquido para poder hacer efectivo el llenado de los envases en las tolvas de almacenaje de líquido y en conexión con la tolva de distribución, llega el líquido a la tolva principal y procede a la eliminación de gases.

Todo el control de las estaciones o subsistemas se manejan mediante sensores inductivos, ópticos e interruptores los cuales dan una señal al PLC, el mismo que hará funcionar al módulo de envasadora.

## **1.5. Justificación**

### **1.5.1. Justificación Teórica**

En la actualidad con el avance de la nueva tecnología nos permite realizar operaciones mecánicas-informáticas, eléctricas y de control, logrando una automatización de los procesos industriales integrando un lenguaje de programación que proporcione el diseño del módulo sin la complejidad de las herramientas de desarrollo tradicionales.

### **1.5.2. Justificación Aplicativa**

Los estudiantes a futuro contarán con un módulo didáctico de una mezcladora automática, este aprendizaje teórico-práctico permitirá relacionar su uso en



las empresas industriales lo cual influirá en el mejoramiento de la producción y el desarrollo.

## **1.6. Objetivos**

### **1.6.1. Objetivo General**

Implementar una Mezcladora Automática utilizando las herramientas tecnológicas actuales que permitan agrupar la parte informática y electromecánica para beneficiarnos con el uso en el aprendizaje didáctico en el Laboratorio de Automatización.

### **1.6.2. Objetivos Específicos**

- Analizar el Estándar IEC 611 31-3 para poder seleccionar el lenguaje de programación adecuado a la complejidad del módulo.
- Programar el proceso de control del módulo bajo el lenguaje de programación seleccionado y diseñar la interfaz de usuario para controlar el proceso de mezclado de líquidos.
- Diseñar e implementar una base de datos que permita registrar datos para la inspección del proceso.
- Construir el módulo final en el laboratorio de Automatización industrial para obtener el mezclado de líquidos.

## 1.7. Hipótesis

La implementación de un sistema mecatrónico para el mezclado de líquidos, caso práctico laboratorio de automatización de la EIS permitirá conseguir el prototipo básico con un diseño sencillo y sólido, mediante la programación de los de los procesos de control del módulo

## CAPITULO II SISTEMAS DEL MODULO

### 2.1. Introducción

En este capítulo se recopila toda la información necesaria para la realización de nuestra investigación, aquí se estudia todo lo que tiene que ver con Automatización, Mecatrónica, elementos neumáticos, Sensórica, Relés, PLCs, la misma que es ampliamente utilizada en el sistema de mezclado de líquidos para el control de procesos industriales, todo esto nos servirá para una mejor comprensión de los temas que se mencionan anteriormente, para dar la solución más óptima posible, durante el desarrollo del Sistema de Mezclado de Líquidos.

### 2.2. Sistema Eléctrico

Un sistema eléctrico es un conjunto de elementos dinámicamente relacionados, que permiten generar, conducir y recibir corriente eléctrica. Dependiendo de

cómo estén dispuestos los elementos dentro del o los circuitos, las fallas o daños causados serán variables.

### 2.2.1 Sensores

Un sensor o captador, como prefiera llamársele, no es más que un dispositivo diseñado para recibir información de una magnitud del exterior y transformarla en otra magnitud, normalmente eléctrica, que seamos capaces de cuantificar y manipular.

#### 2.2.1.1 Principios de Funcionamiento

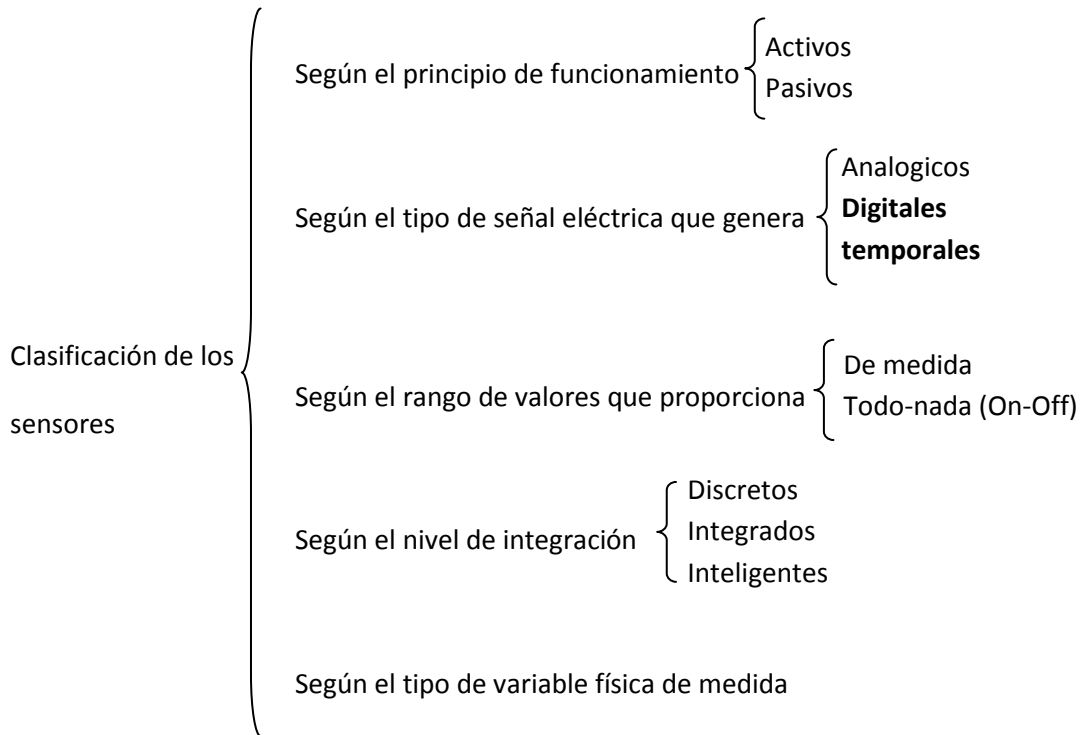
El principio de funcionamiento depende de las variables físicas medibles

		Variable física medida										
		Posición	Desplazamiento	Velocidad	Aceleración	Tamaño	Nivel	Presión	Fuerza	Proximidad	Temperatura	Radiación luminosa
Principio de funcionamiento	Microrruptores	X				X						
	Finales carrera	X										
	Extensiómetros	X	X	X	X			X	X			
	Termorresistivos										X	
	Magnetorresistivos	X	X	X								
	Capacitivos	X	X		X		X	X	X	X		
	Inductivos	X	X	X	X			X	X	X		
	Optoelectrónicos	X	X	X						X		
	Piezoeléctricos		X	X	X			X	X			
	Fotovoltaicos											X
	Ultrasónicos	X					X					

Tabla II.1. Principio de funcionamiento de los sensores

### 2.2.1.2 Tipos

Esta es la clasificación general de los sensores más adelante se analizaran solamente los sensores de nivel



**Figura. II.1. Clasificación de sensores**

#### **Sensores de nivel**

Para el control de nivel de los líquidos existen una gran variedad de sensores entre los cuales están: tipo flotador, conductivo, capacitivo, ultrasónico, radiación, resistivo, peso etc

**La clasificación de los sensores de nivel el cual se utiliza para realizar el modulo**

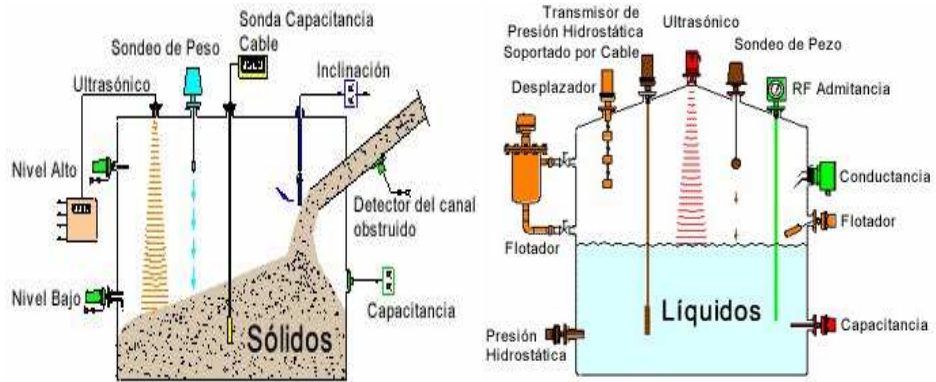


Figura. II.2. Tipos de sensores de nivel

- Desplazamiento (flotador)
- Sonda
- Presión diferencial
- Burbujeo
- Radioactivo
- Capacitivo
- Ultrasonidos
- Conductividad
- Radar
- Servoposicionador

Instrumento	Campo de medida	Precisión % escala	Presión máxima Bar	Temperatura Máxima C	Desventajas	Ventajas
<b>Sonda</b>	Limite	0.5 mm	Atm	60	Manual, sin olas. Tanques abiertos	Barato Preciso
<b>Flotador</b>	0-10 m	$\pm 1-2\%$	400	250	Posible agarrotamiento	Simple, independiente, naturaleza líquida
<b>Manométrico</b>	Alt. Tanque	$\pm 1\%$	Atm	60	Tanques Abiertos, Fluidos	Barato

					Limpios	
<b>Membrana</b>	0-25 m	$\pm 1\%$	*	60	Tanques Abiertos,	Barato
<b>Burbujeo</b>	Alt. Tanque	$\pm 1\%$	400	200	Mantenimiento, contaminación líquidos	Barato y Versátil
<b>Presión diferencial</b>	0.3 m	$\pm 0,15\%$ $-\pm 0,5\%$	150	200	Posible agarrotamiento	Interfaz líquido
<b>Desplazamiento</b>	0-25 m	$\pm 0,5\%$	100	170	Expuesto a corrosión	Fácil limpieza, robusto, interfaces
<b>Conductivo</b>	Ilimitado	-	80	200	Líquido conductor	Versátil
<b>Capacitivo</b>	0,6 m	$\pm 1\%$	80-250	200-400	Recubrimiento electrodo	Resistencia, corrosión
<b>Ultrasónico</b>	0,30 m	$\pm 1\%$	400	200	Sensible a la densidad	Todo tipo de tanques y líquidos
<b>Radar</b>	0-30 m	$\pm 2,5\%$			Sensible a la constante dieléctrica	Todo tipo de tanques y líquidos con espuma
<b>Radiación</b>	0-2,5 m	$\pm 0,5-2\%$	-	150	Fuente radioactiva	Todo tipo de tanques y sin contacto líquido
<b>Laser</b>	0-2 m	$\pm 0,5-2\%$	-	1500	Laser	Todo tipo de tanques y sin contacto líquido

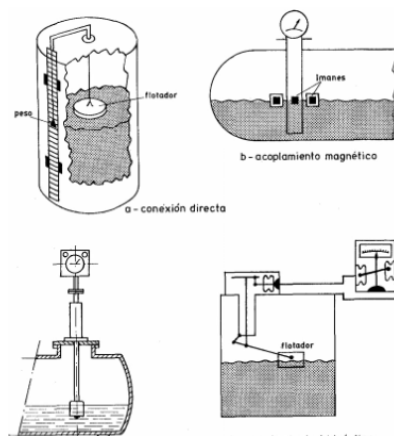
**Tabla II.2. Sensores de nivel de líquidos**

Los más importantes para nuestro estudio es el flotador y sonda

- **Flotador**

El sensor flotador consiste en un flotador situado en el seno de líquido y conectado al exterior del tanque indicando directamente el nivel. La conexión puede ser directa, magnética o hidráulica.

El flotador conectado directamente (Fig. a) está unido por un cable que se desliza en un juego de poleas a un índice exterior que señala sobre una escala graduada. El modelo más antiguo y más utilizado en tanques de gran capacidad. Tiene que el inconveniente de que las parte móviles están expuestas al fluido y puede romperse y el tanque no puede estar sometido a presión. Además el flotador debe mantenerse limpio



**Figura. II.3. Sensor flotador**

El flotador acoplado magnéticamente se desliza exteriormente a lo largo del tubo guía sellado, situado verticalmente en el interior del tanque. Dentro del tubo una pieza magnética sigue al flotador en su movimiento y mediante un cable y un juego de poleas arrastra el índice de un instrumento en la parte superior del tanque. El instrumento puede además ser transmisor neumático o eléctrico. Una variante de la conexión magnética consta de un tubo conteniendo un flotador, dotado de un imán que orienta una serie de cintas magnéticas dispuestas en el exterior y lo largo del tubo. A medida que el nivel sube o baja, las cintas giran y como tiene colores distintos en el anverso y reverso, visualiza

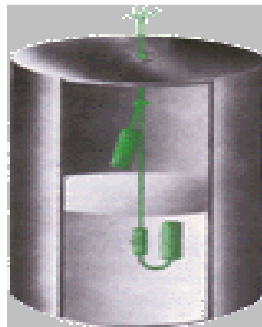


directamente el nivel del tanque el sensor puede tener interruptores de alarma y transmisor incorporado.

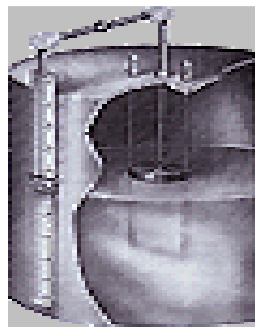
En tanques pequeños, el flotador puede adaptarse para actuar magnéticamente sobre un transmisor neumático o eléctrico dispuesto en el exterior del tanque permitiendo así un control de nivel

El flotador acoplado hidráulicamente actúa en su movimiento sobre un fuelle de tal modo que varía la presión del circuito hidráulico y señala a distancia en el receptor el nivel correspondiente. Permite distancias de transmisión de hasta 75 metros y puede emplearse en tanques cerrados.

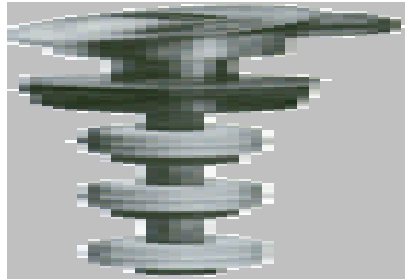
Hay que señalar que estos sensores, el flotador puede tener formas muy variadas y estar formado por materiales muy diversos según sea el tipo de fluido



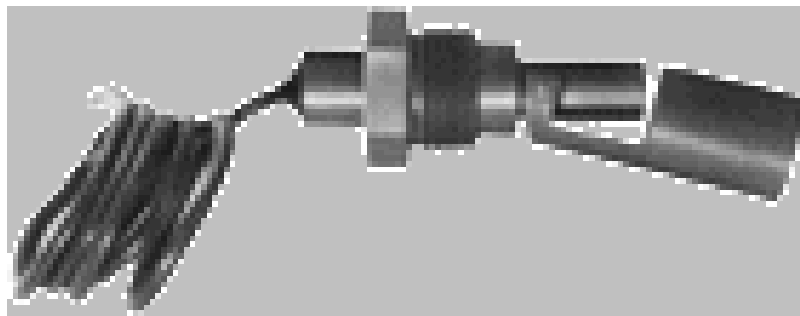
**Figura. II.4. Flotante a cable**



**Figura. II.5. Flotante con cable**



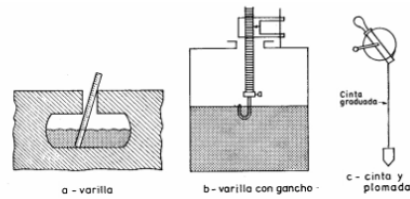
**Figura. II.6. Flotante con columna magnética**



**Figura. II.7. Flotante lateral magnético**

- **Sonda**

El sensor sonda consiste en una varilla o regla graduada a la longitud conveniente para introducirla dentro del tanque. La determinación del nivel se efectúa por lectura directa de la longitud mojada por el líquido. En momento de la lectura el tanque debe estar abierto a presión atmosférica. Otro medido consiste en una varilla graduada, con un gancho que se sumerge en el seno del líquido. La distancia desde esta superficie hasta la parte superior del tanque representa indirectamente el nivel. Otro sistema parecido es el medidor de cinta graduada y plomada que se emplea cuando es difícil que la regla graduada tenga acceso al fondo del tanque.



**Figura. II.8. Sensor sonda**

### 2.2.1.3 Aplicaciones

- Elementos de nivel de líquido integrados a la electrónica del cliente.
- Elemento de nivel de líquido con sistemas electrónicos de interrupción de estado sólido.
- Elemento de nivel de líquido con sistema electrónico multipunto de estado sólido.
- Elemento de nivel de líquido dotado de transmisor con sistema electrónico de estado sólido.

Detección de líquidos como:

- Adecuados para el agua, aceite, combustibles y la mayoría de líquidos industriales
- Corrosivos
- Fluidos a alta temperatura
- Mats en suspensión
- interruptores de nivel en líquidos conductores
- Viscosos (asfaltos)
- Movimiento (barcos)

### 2.2.2 Conductores Eléctricos

Son cuerpos capaces de conducir o transmitir la electricidad, está formado primeramente por el conductor propiamente tal, usualmente de cobre, este puede ser alambre, es decir, una sola hebra o un cable formado por varias hebras o alambres retorcidos entre sí.

#### 2.2.2.1 Clases

Conductores más utilizados:

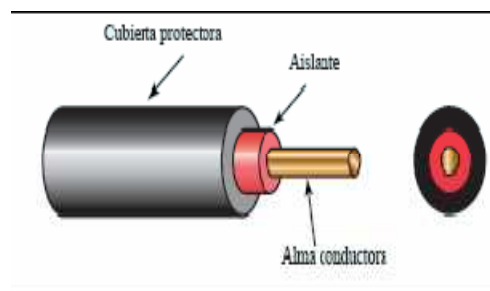
- Cobre
- Aluminio

<b>Características</b>		<b>Cobre</b>	<b>Aluminio</b>
Resistencia eléctrica		1	1.56
Resistencia mecánica		1	0.45
Para igual volumen:	Relación de pesos	1	0.30
Para igual conductancia:	Relación de áreas	1	1.64
	Relación de diámetros	1	1.27
	Relación de pesos	1	0.49
Para igual diámetro:	Relación de resistencias	1	1.61
	Capacidad de corriente	1	0.78

**Tabla II.3. Características del aluminio y cobre**

### Partes de un conductor eléctrico

- El alma, o elemento conductor
- Aislante
- Las cubiertas protectoras



**Figura. II.9. Partes del conductor eléctrico**

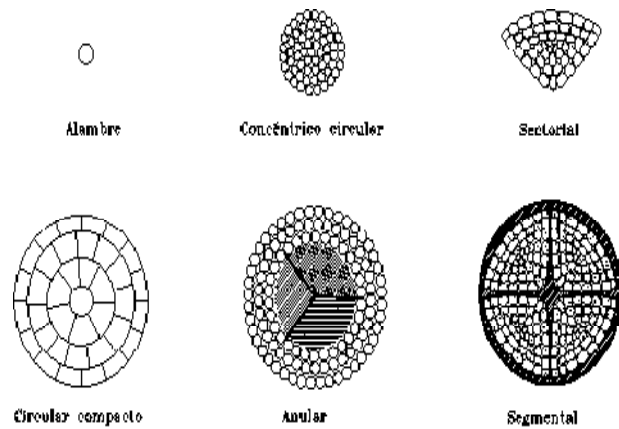
### Clasificación de los conductores eléctricos de acuerdo a su aislación o número de hebras

La parte más importante de un sistema de alimentación eléctrica está constituida por conductores.

Al proyectar un sistema, ya sea de poder; de control o de información, deben respetarse ciertos parámetros imprescindibles para la especificación de la cablería.

- Voltaje del sistema, tipo (CC o CA), fases y neutro, sistema de potencia, punto central aterramiento.
- Corriente o potencia a suministrar.
- Temperatura de servicio, temperatura ambiente y resistividad térmica de alrededores.
- Tipo de instalación, dimensiones (profundidad, radios de curvatura, distancia entre vanos, etc.).
- Sobrecargas o cargas intermitentes.

- Tipo de aislación.
- Cubierta protectora.



**Figura. II.10. Conductores eléctricos de acuerdo a su aislación o número de hebras**

Todos estos parámetros están íntimamente ligados al tipo de aislación y a las diferencias constructivas de los conductores eléctricos, lo que permite determinar de acuerdo a estos antecedentes la clase de uso que se les dará.

De acuerdo a éstos, podemos clasificar los conductores eléctricos según su aislación, construcción y número de hebras en monoconductores y multiconductores.

Tomando en cuenta su tipo, uso, medio ambiente y consumos que servirán, los conductores eléctricos se clasifican en la siguiente forma:

#### **Conductores para distribución y poder:**

- Alambres y cables (Nº de hebras: 7 a 61).
- Tensiones de servicio: 0,6 a 35 kV (MT) y 46 a 65 kV (AT).
- Uso: Instalaciones de fuerza y alumbrado (aéreas, subterráneas e interiores).
- Tendido fijo.



**Figura. II.11. Cables para su distribución y poder**

**Cables armados:**

- Cable (N0 de hebras: 7 a 37).
- Tensión de servicio: 600 a 35 000 volts.
- Uso: Instalaciones en minas subterráneas para piques y galerías (ductos, bandejas, aéreas y subterráneas)
- Tendido fijo



**Figura. II.12. Cable armado**

**Conductores para control e instrumentación:**

- Cable (N0de hebras: 2 a 27).
- Tensión de servicio: 600 volts.
- Uso: Operación e interconexión en zonas de hornos y altas temperaturas.  
(ductos, bandejas, aérea o directamente bajo tierra).
- Tendido fijo.



**Figura. II.13. Conductores de control e instrumentación**

**Cordones:**

- Cables (Nº de hebras: 26 a 104).
- Tensión de servicio: 300 volts.
- Tendido portátil.



**Figura. II.14. Conductor cordones**

**Cables portátiles:**

- Cables (Nº de hebras: 266 a 2 107).
- Tensión de servicio: 1 000 a 5 000 volts
- Uso: en soldadoras eléctricas, locomotoras y máquinas de tracción de minas subterráneas. Grúas, palas y perforadoras de uso minero.
- Resistente a: intemperie, agentes químicos, a la llama y grandes sollicitaciones mecánicas como arrastres, cortes e impactos.
- Tendido portátil.





**Figura. II.15. Conductores portátiles**

**Cables submarinos:**

- Cables (Nº de hebras: 7 a 37).
- Tensión de servicio: 5 y 15 kV.
- Uso: en zonas bajo agua o totalmente sumergidos, con protección mecánica que los hacen resistentes a corrientes y fondos marinos.
- Tendido fijo.



**Figura. II.16. Cables submarinos**

**Cables navales:**

- Cables (Nº de hebras: 3 a 37).
- Tensión de servicio: 750 volts.
- Uso: diseñados para ser instalados en barcos en circuitos de poder, distribución y alumbrado.
- Tendido fijo.



**Figura. II.17. Cables navales**

Dentro de la gama de alambres y cables que se fabrican en el país, existen otros tipos, destinados a diferentes usos industriales, como los cables telefónicos, los alambres magnéticos esmaltados para uso en la industria electrónica y en el embobinado de partidas y motores de tracción, los cables para conexiones automotrices a baterías y motores de arranque, los cables para parlantes y el alambre para timbres.

### **Clasificación de los conductores eléctricos de acuerdo a sus condiciones de empleo**

Para tendidos eléctricos de alta y baja tensión, existen en nuestro país diversos tipos de conductores de cobre, desnudos y aislados, diseñados para responder a distintas necesidades de conducción y a las características del medio en que la instalación prestará sus servicios.

La selección de un conductor se hará considerando que debe asegurarse una suficiente capacidad de transporte de corriente, una adecuada capacidad de soportar corrientes de cortocircuito, una adecuada resistencia mecánica y un comportamiento apropiado a las condiciones ambientales en que operará.

#### **Conductores de cobre desnudos**

Estos son alambres o cables y son utilizados para:

- Líneas aéreas de redes urbanas y suburbanas.
- Tendidos aéreos de alta tensión a la intemperie.
- Líneas aéreas de contacto para ferrocarriles y trolley-buses.

#### **Alambres y cables de cobre con aislación**

Estos son utilizados en:

- Líneas aéreas de distribución y poder, empalmes, etc.
- Instalaciones interiores de fuerza motriz y alumbrado, ubicadas en ambientes de distintas naturaleza y con diferentes tipos de canalización.
- Tendidos aéreos en faenas mineras (tronadura, grúas, perforadoras, etc.).
- Tendidos directamente bajo tierra, bandejas o ductos.
- Minas subterráneas para piques y galerías.
- Control y comando de circuitos eléctricos (subestaciones, industriales, etc.).
- Tendidos eléctricos en zonas de hornos y altas temperaturas.
- Tendidos eléctricos bajo el agua (cable submarino) y en barcos (conductores navales).

#### **2.2.2.2 Codificación**

Codificación de cables eléctricos

Los conductores de una canalización eléctrica se identifica según el siguiente código de colores:

Conductor de la fase 1 azul

Conductor de la fase 2 negro

Conductor de la fase 3 rojo

Conductor de neutro y tierra de servicio blanco

Conductor de protección verde o verde/amarillo

#### **2.2.2.3 Aplicaciones**

Para derivaciones a cajas para aparatos, tableros de mando, sistemas de alarma, timbres y portalámparas.

Cableado interno de equipos y tableros.

## 2.3 Sistema Neumático

Sistemas neumáticos utilizan unos mecanismos que convierten la energía del aire comprimido en trabajo mecánico por medio de un movimiento lineal de vaivén, o de motores.

### 2.3.1 Motores Neumáticos

Estos elementos transforman la energía neumática en un movimiento de giro mecánico. Funcionan igual que los cilindros de giro pero el ángulo de giro no está limitado. Por eso es de los cilindros neumáticos más usados.

Estos cilindros por método de la presión introducida podemos obtener un movimiento rotativo

#### **Características**

- Regulación sin escalones de la velocidad de rotación y del par motor
- Gran selección de velocidades de rotación
- Pequeñas dimensiones (y reducido peso)
- Gran fiabilidad, seguros contra sobrecarga
- Insensibilidad al polvo, agua, calor y frío
- Ausencia de peligro de explosión
- Reducido mantenimiento
- Sentido de rotación fácilmente reversible

### 2.3.1.1. Principios de Funcionamiento

**Funcionamiento del motor de aletas:** Estos motores se constituyen en el principio de la inversión del compresor rotativo.

Un rotor dotado de ranuras gira en una cámara cilíndrica. En las ranuras se deslizan aletas, que son empujadas contra la pared interior del cilindro por el efecto de la fuerza centrífuga.

Normalmente estos motores tienen de 3 a 10 aletas, estas forman cámaras en el interior del motor. El aire entra en la cámara más pequeña y se dilata a medida que el volumen de la cámara aumenta

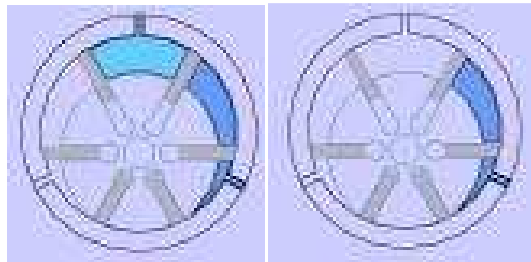


Figura. II.18. Funcionamiento del motor neumático 1

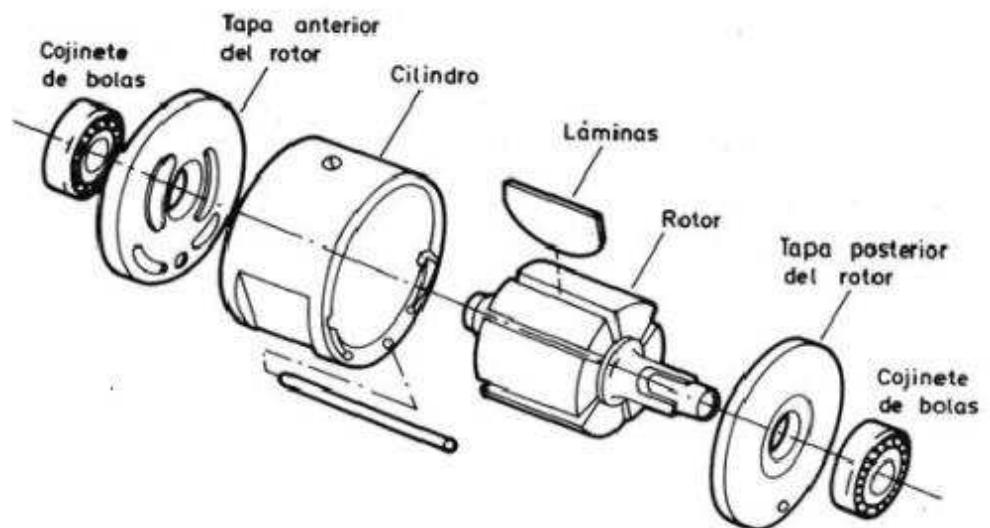


Figura. II.19. Funcionamiento del motor neumático 2

### 2.3.1.2. Tipos

De este tipo de actuador podemos encontrar:

- Motores de émbolo
- Motores de aletas
- Motores de engranajes
- Turbomotores

#### **Ventajas:**

- Construcción sencilla ( peso ligero )
- Arranque y paro muy rápido
- Insensibilidad al polvo, agua, calor y frío
- La velocidad varia entre 3.000 y 8.500 rpm .
- Alta aceleración y baja inercia.

#### **Motor de aletas:**



**Figura. II.20. Motor de aletas**

Este motor por medio de aletas transforma la energía neumática en energía rotativa (también podría por medio de piñones)

Este es el motor neumático usado más frecuentemente.

#### **Ventajas:**

- Son mas ligeros y baratos que los motores de pistones
- La velocidad se controla fácilmente y oscila entre 3000 y 25000 r.p.m.

- La relación peso/potencia es muy alta

### **Motores de émbolo**

Este tipo se subdivide además en motores de émbolo axial y de émbolo radial. Por medio de cilindros de movimiento alternativo, el aire comprimido acciona, a través de una biela, el cigüeñal del motor. Se necesitan varios cilindros al objeto de asegurar un funcionamiento libre de sacudidas. La potencia de los motores depende de la presión de entrada, del número de émbolos y de la superficie y velocidad de éstos.

El funcionamiento del motor de émbolos axiales es idéntico al de émbolos radiales. En cinco cilindros dispuestos axialmente, la fuerza se transforma por medio de un plato oscilante en un movimiento rotativo. Dos cilindros reciben cada vez aire comprimido simultáneamente al objeto de equilibrar el par y obtener un funcionamiento tranquilo.

Estos motores de aire comprimido se ofrecen para giro a derechas y giro a izquierdas.

La velocidad máxima es de unas 5000 min<sup>-1</sup>, y la potencia a presión normal, varía entre 1,5 y 19 kW (2-25 CV).

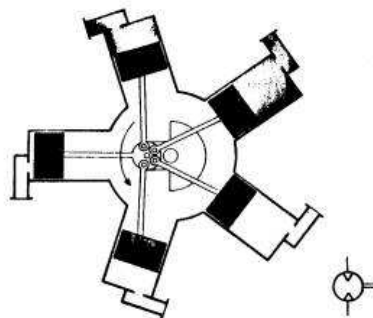
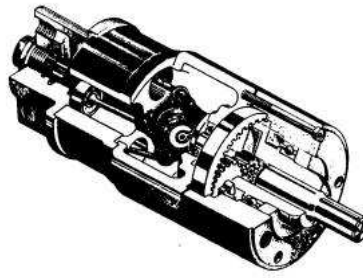


Figura: Motor radial



**Figura. II.21. Motor axial**

### **Motor de engranajes**

En este tipo de motor, el par de rotación es engendrado por la presión que ejerce el aire sobre los flancos de los dientes de piñones engranados. Uno de los piñones es solidario con el eje del motor.

Estos motores de engranaje sirven de máquinas propulsoras de gran potencia 44 kW (60 CV).

El sentido de rotación de estos motores, equipados con dentado recto o helicoidal, es reversible.

### **Turbomotores**

Pueden utilizarse únicamente para potencias pequeñas, pero su velocidad es muy alta (tornos neumáticos del dentista de hasta 500.000 rpm ) . Su principio de funcionamiento es inverso al de los turbocompresores.

#### **2.3.1.3. Aplicación**

Se emplean para trabajos a baja velocidad con grandes cargas, por ejemplo: estos motores, generalmente, están diseñados para ser usados en máquinas de perforación.

Numerosas aplicaciones en

- Grúas y cabrestantes.



- Perforadoras, cargadoras de minas.
- Reductores.
- Cintas, transportadores.
- Máquinas de inyección o proyección de mortero
- Refinerías, industria química, astilleros, etc.

Maquinaria de minas y obras públicas, siderurgia, industria química y petroquímica.

### 2.3.2. Válvulas

Una válvula neumática es un elemento de regulación y control de la presión y el caudal del aire a presión. Este aire es recibido directamente después de su generación o sino desde un dispositivo de almacenamiento. Las válvulas dirigen, distribuyen o pueden bloquear el paso del aire para accionar los elementos de trabajo (los actuadores).

**La función básica de las** válvulas es modificar alguna propiedad del flujo que pasa a través de ellas.

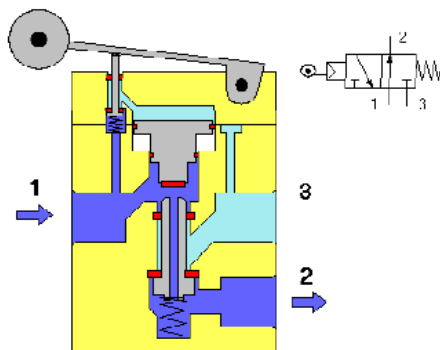
**Su función más simple** consiste en desviar cierto flujo de aire hacia un punto donde se exija una determinada presión

#### 2.3.2.1. Principios de Funcionamiento.

##### **Válvulas distribuidoras**

La válvula posee en su interior un pequeño conducto con una válvula auxiliar que conecta presión (1) con la cámara del émbolo que acciona la válvula. Cuando se acciona el rodillo, se abre la válvula auxiliar de servopilotaje, el aire comprimido circula hacia la cámara superior del émbolo que al desplazarlo modifica la posición de la válvula principal 3/2.

La inversión se realiza en dos fases para evitar el solape (Figura 4-9). En primer lugar se cierra el conducto de A(2) hacia R(3), y luego se abre el P(1) hacia A(2). La válvula se reposiciona por muelle al soltar el rodillo. Se cierra el paso de la tubería de presión hacia la cámara del émbolo y se purga de aire. El muelle hace regresar el émbolo de mando de la válvula principal a su posición inicial.



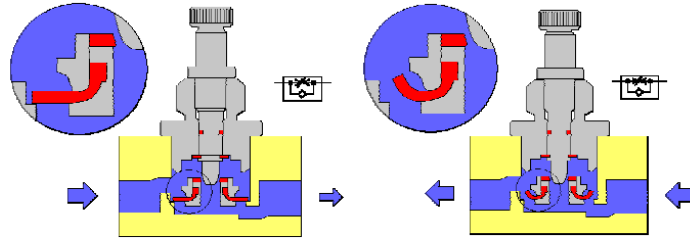
**Figura. II.22. Válvula 3/2 inicialmente abierta accionada por rodillo**

Este tipo de válvula puede emplearse opcionalmente como válvula normalmente abierta o normalmente cerrada. Para ello sólo hay que permutar los empalmes P y R e invertir el cabezal de accionamiento 180°

### **Válvulas reguladoras de caudal**

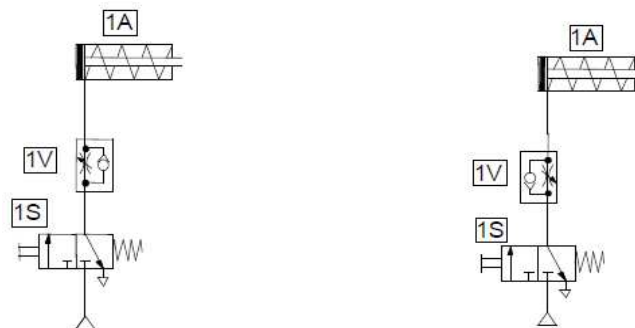
La válvula cierra el paso del aire en un sentido y el aire ha de circular forzosamente por la sección estrangulada. En el sentido contrario, el aire circula

libremente a través de la válvula abierta (figura 4-25). Las válvulas y de estrangulación deben montarse lo más cerca posible de los cilindros.



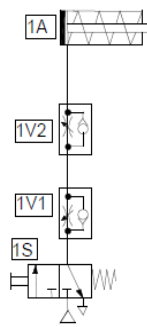
**Figura. II.23. Regulador de caudal unidireccional**

Se utilizan para aminorar y regular la velocidad del vástago de un cilindro, de simple o doble efecto. Según como se disponga la válvula antirretorno se consigue regular la velocidad del vástago en uno u otro sentido. En la Figura 4-26 se muestran esquemas correspondientes a la regulación de un cilindro de simple efecto donde se controla la velocidad del vástago en su salida y entrada respectivamente



**Figura. II.24. Regulación de la velocidad del vástago en su salida o entrada**

Si se desea ajustar y aminorar la velocidad del vástago de un cilindro de simple efecto cuando se desplace en los dos sentidos se han de instalar dos válvulas restrictoras colocadas en sentido inverso



**Figura. II.25. Regulación de la velocidad del vástago en su entrada y salida**

Para regular y aminorar la velocidad del vástago de un cilindro de doble efecto en su salida o entrada se dispone la válvula unidireccional con estrangulamiento en la alimentación y/o en el escape. Se ubicará en un solo lugar si solo se desea regular la velocidad en un único sentido y se instalará en los dos cuando se desee un regulación doble.

### **Válvulas reguladoras de presión**

Tiene la misión de mantener constante la presión en su salida independientemente de la presión que exista a la entrada. Tienen como finalidad fundamental obtener una presión invariable en los elementos de trabajo independientemente de las fluctuaciones de la presión que normalmente se producen en la red de distribución. La presión de entrada mínima debe ser siempre, obviamente, superior a la exigida a la salida.

#### **2.3.2.2. Tipos**

Existen tres tipos de válvulas dependiendo de lo que queremos hacer:

- Distribuidoras

- Reguladoras de caudal
- Reguladoras de presión

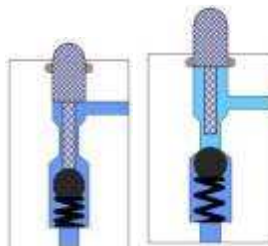
### Válvulas Distribuidoras

Estas válvulas inician, paran y dirigen el fluido a través de las diferentes conducciones de la instalación para hacer posible el control de los actuadores (cilindros o motores).

A continuación explicaremos las válvulas direccionales mas usuales:

#### 1-Válvula distribuidora 2/2:

Esta válvula al igual que la unidireccional es de asiento, es decir que abren y cierran el paso por medio de conos, discos, placas y bolas, evitando cualquier fuga. Estas válvulas son de concepción muy simple, pequeña y económica. Son ideales para gobernar cilindros de simple efecto.



**Figura. II.26. Válvula distribuidora 2/2**

Un muelle mantiene apretada la bola contra el la apertura, obstruyendo el paso del aire (posición de cerrado). Cuando presionamos sobre la bola, entonces el muelle se encoje provocando el paso del aire (posición de obertura).

Estas válvulas son distribuidoras 2/2, porque tienen dos posiciones (abierta y cerrada) y dos orificios (la entrada y la salida del aire)



Figura. II.27. Válvula 2/2 conectada a un cilindro por medio de tubos

### 2-Válvula distribuidora 4/2:

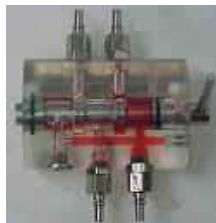


Figura. II.28. Válvula

distribuidora 4/2

Esta válvula permite que pase el aire por los dos sentidos ya que tiene dos entradas de aire y dos salidas. Dependiendo de la posición (cerrada o abierta) de la válvula algunas vías cambian de sentido del aire. Por eso se llama válvula 4/2 (4 vías, 2 posiciones). La válvula 4/2 es muy utilizada para gobernar cilindros de doble efecto:

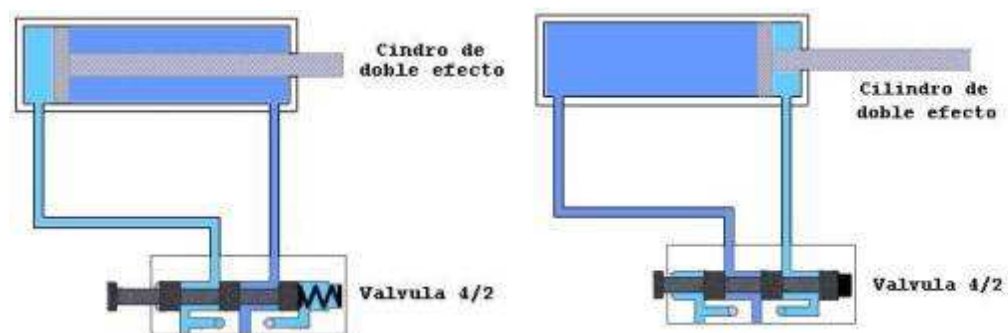
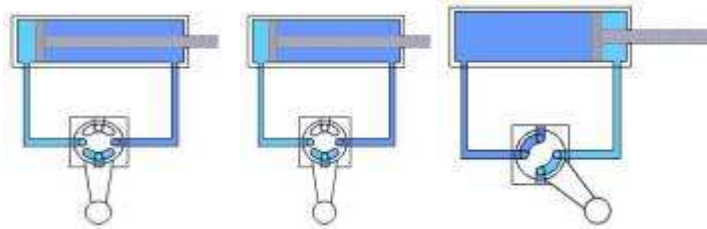


Figura. II.29. Funcionamiento de la válvula distribuidora 4/2

### 3-Válvula distribuidora 4/3:

La válvula 4/3 funciona casi igual que la 4/2 pero con la ventaja que tiene una pausa entre las posiciones del cilindro. Esta se acciona manualmente ( o por pedal) , dando mayor posibilidad de gobernar el cilindro: podemos conectar el cilindro cuando queremos. Estas válvulas se usan para gobernar tanto cilindro de doble efecto como motores neumáticos.



**Figura. II.30. Las 3 posiciones de la válvula 4/3 que esta conectada a un cilindro de doble efecto.**

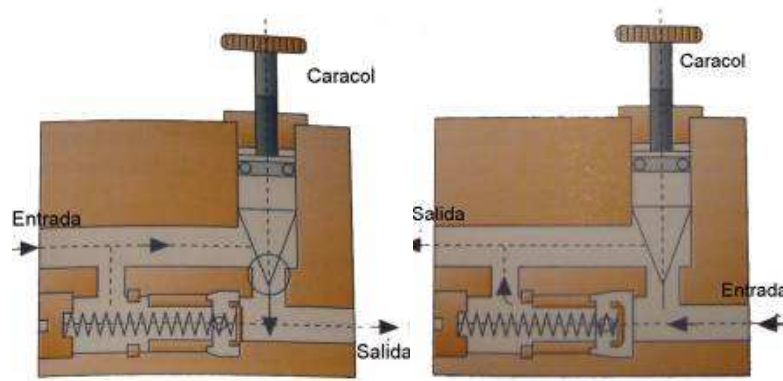
### Válvulas Regulatoras De Caudal



**Figura. II.31. Válvulas**

**regulatoras de caudal**

Cuando se genera mucho aire a presión y este va a mucha velocidad y queremos reducir el caudal para que funcione bien el cilindro, para eso usaremos una válvula reguladora de caudal. Esta funciona de tal forma que cuando enroscamos el “caracol” el caudal disminuye ya que hace frenar el aire a presión. Normalmente se acopla un antiretorno, para que el fluido solamente vaya estrictamente en un sentido, evitando así grandes problemas.



**Figura. II.32. Funcionamiento de las válvulas reguladoras de caudal**

### **Válvulas Reguladoras De Presión ::**



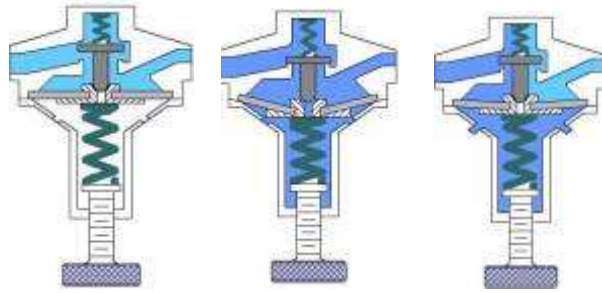
**Figura. II.33. Válvula reguladora de presión**

Estas válvulas se encargan de regular la presión que recibe para enviar la presión ideal obtenida al actuador (sea cilindro o motor). Notablemente hay diferentes tipos: válvula reguladora de presión, válvula limitadora de presión, válvula de secuencia.

#### **1-Válvula reguladora de presión**

Es una válvula con dos vías: el aire entra por la entrada y si la presión es mas elevada de la que queremos entonces la fuerza del muelle hace reducirla, luego el aire comprimido sale hacia el actuador.

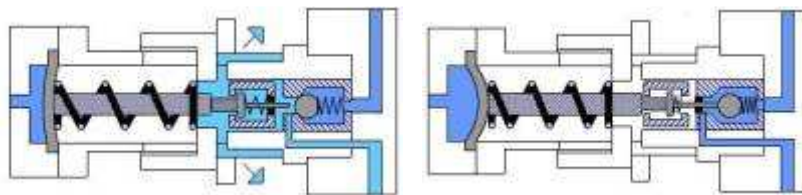




**Figura. II.34** Válvula reg. uladora de presión

## 2-Válvula de secuencia:

Esta válvula cuando alcanza cierta presión, que esta ajustada por un muelle, entrega una señal neumática como salida. Se usa para controlar presiones distintas para luego ejecutar unas u otras funciones (es decir que esta válvula simplemente calcula la presión y envía una señal )



**Figura. II.35.** Válvulas de secuencia

### 2.3.2.3. Aplicaciones

- Existen válvulas de carácter especial como las de descarga rápida, las válvulas de caudal variable y las válvulas de pilotaje interno
- Existen también válvulas de carácter normal como las de secuencia, y las válvulas de seguridad
- Una válvula normal podría estar clasificada en otras categorías dependiendo de la función
- Para aplicaciones industriales severas

## 2.4. Sistema Mecánico

Un sistema mecánico es un conjunto de elementos dinámicamente relacionados, que permiten producir, transmitir, regular o modificar movimiento. Cada operador cumple una función específica dentro del sistema.

### 2.4.1. ¿Estructura Metálica construida en aluminio

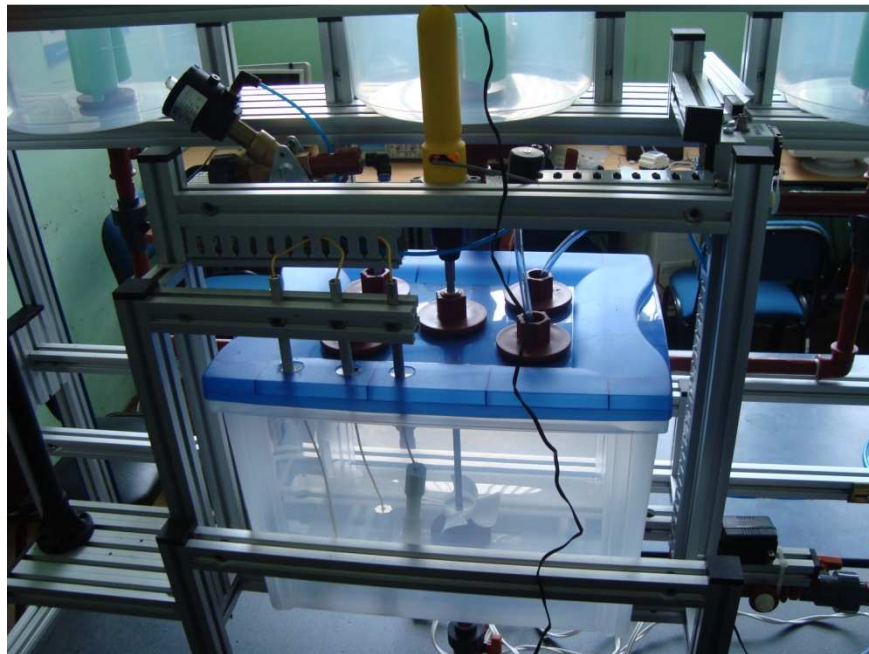


Figura. II.36. Estructura de aluminio del sistema de mezclado de líquidos

### 2.4.2. Características del aluminio

El aluminio es uno de los elementos más abundantes en la tierra (8%), siendo aventajado en cantidad solamente por el oxígeno (45%) y el silicio(28%). El hecho de que el aluminio es el metal más abundante de la corteza terrestre se debe a que se halla presente en todas las rocas, excepto en las calcáreas puras y las de tipo silíceo (cuarzos). Con toda la gama de aleaciones de aluminio se obtiene una amplia variedad de combinaciones de resistencia mecánica,

ductilidad, conductividad eléctrica y resistencia a la corrosión, por lo que se ha adoptado un sistema en el cual la designación de la aleación se hace con un número de cuatro dígitos. A continuación mostramos características de las aleaciones más comunes:

- Aluminio con pureza de 99.0% mínimo. Es la aleación más comúnmente usada para fabricación de utensilios, envases, aletas de refrigeración, capsulas y en general en aquellas aplicaciones donde se requiere realizar grandes deformaciones.
- Aluminio con pureza de 99.0% mínimo. Es la aleación más comúnmente usada para la fabricación de utensilios domésticos, envases, piezas decorativas, recubrimientos y en general, aquellas aplicaciones donde no se requiere una resistencia mecánica importante.
- Aleación con pureza de 99.5% mínimo. Esta aleación está destinada fundamentalmente a los usos de conducción eléctrica.
- Resistencia mecánica superior a la aleación 6003, buena resistencia a la corrosión, suelda satisfactoriamente, se puede obtener máximo de propiedades desarrollando tratamientos térmicos.
- **Ligereza:** El aluminio tiene un peso muy reducido: pesa solamente un tercio que el acero con el mismo volumen, permitiendo, de esta manera, obtener importantes ahorros de peso en casi todos los tipos de **aplicaciones**, sobre todo, la mecánica.
- **Duración:** El aluminio, gracias a la capacidad de desarrollar una película en la que no puede penetrar el óxido en las superficies expuestas, no está sometido a problemas de corrosión atmosférica, habituales en el hierro y el

cobre, y no requiere ningún tipo de pintura de protección. Por este motivo, está especialmente indicado para aplicaciones arquitectónicas y navales, así como la fabricación de **cerramientos** y fachadas continuas.

- **Conductividad:** El aluminio tiene una extraordinaria conductividad eléctrica que lo hace indispensable para la electrónica y las aplicaciones eléctricas. Permite realizar líneas de gran longitud con cables de aluminio capaces de conducir la corriente eléctrica en cantidad doble que las de cobre del mismo peso. De hecho, se presta también a aplicaciones de calefacción y refrigeración.
- **No tóxico:** Es un material atóxico, muy utilizado para conservar alimentos y bebidas.
- **Maleabilidad:** El aluminio es muy maleable y puede modelarse, con todas las técnicas habituales de **tratamiento**, con más facilidad que la mayoría de otros metales. Se puede forjar, laminar hasta obtener una hoja muy fina, extrusar en **perfiles** complejos o plegar.
- **Versatilidad:** La posibilidad de utilizar el metal en aleaciones, que pueden ser rígidas o elásticas, especialmente sólidas y resistentes a la corrosión, permite adaptar el aluminio a una amplia gama de necesidades.
- **Reciclabilidad:** El aluminio se recicla con gran facilidad y con un coste energético reducido: actualmente, una cuarta parte de las necesidades de aluminio en Europa se abastece utilizando metal de segunda fusión que, a su vez, puede reciclarse indefinidamente.
- **Aplicaciones:** Ingeniería estructural, transporte, maquinados en general de barras cilíndricas, etc.

- Aplicaciones: Arquitectura, marcos para ventanas, puertas, cancelas, etc., tubos para muebles, irrigación, industria química, electrodoméstica, electrónica, disipadores de calor, etc.
- Excelente extrudabilidad, mediana resistencia mecánica, alta resistencia a la corrosión, buen acabado superficial, especial para anodizar.

### **2.4.3. Ventajas del Aluminio**

Las ventajas: protegen el contenido durante largos periodos, son muy ligeras, son difíciles de romper, presentan una gran comodidad de manejo y ocupan muy poco espacio. Y lo más importante: son cien por cien reciclables.

En definitiva, "el aluminio es el elemento más abundante de la corteza terrestre después del oxígeno y el silicio y además puede ser reciclado infinitamente sin por ello perder un ápice de sus cualidades", lo que conlleva que "las aplicaciones son infinitas y su demanda crece día a día", ya que se trata de un material "idóneo para el mundo actual y que respeta el medio en el que vivimos".

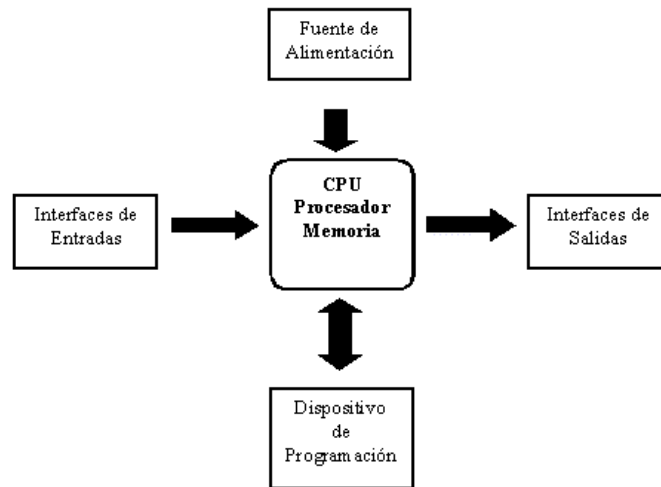
## **2.5. Sistema Informático**

### **2.5.1. PLC**

#### **2.5.1.1. Definición**

Un aparato electrónico operado digitalmente, que usa una memoria programable para el almacenamiento interno de instrucciones para implementar funciones específicas, tales como lógica, secuenciación, registro y control de tiempos, conteo y operaciones aritméticas para

controlar, a través de módulos de entrada/salida digitales (ON/OFF) o analógicos (1 5 VDC, 4 20 mA, etc.), varios tipos de máquinas o procesos.



**Figura. II.37. Funcionamiento PLC**

#### 2.5.1.2. Marcas – Tipos

Debido a la gran variedad de tipos distintos de PLC, tanto en sus funciones, en su capacidad, en su aspecto físico y otros, es que es posible clasificar los distintos tipos en varias categorías.

**PLC tipo Nano:** Generalmente PLC de tipo compacto ( Fuente, CPU e I/O integradas ) que puede manejar un conjunto reducido de I/O, generalmente en un número inferior a 100. Permiten manejar entradas y salidas digitales y algunos módulos especiales.

**PLC tipo Compactos:** Estos PLC tienen incorporado la Fuente de Alimentación, su CPU y módulos de I/O en un solo módulo principal y permiten manejar desde unas pocas I/O hasta varios cientos ( alrededor de 500 I/O ) , su tamaño es superior a los Nano PLC y soportan una gran variedad de módulos especiales, tales como:

- Entradas y salidas análogas
- Módulos contadores rápidos
- Módulos de comunicaciones
- Interfaces de operador
- Expansiones de i/o

**PLC tipo Modular:** Estos PLC se componen de un conjunto de elementos que conforman el controlador final, estos son:

- Rack
- Fuente de Alimentación
- CPU
- Módulos de I/O

De estos tipos existen desde los denominados Micro PLC que soportan gran cantidad de I/O, hasta los PLC de grandes prestaciones que permiten manejar miles de I/O.

### 2.5.1.3. Estructura del PLC

El PLC esta conformado por una estructura externa e interna.

**Estructura Externa** o configuración externa de un autómatas programable industrial se refiere al aspecto físico exterior del mismo, bloques o elementos en que está dividido.

Actualmente son tres las estructuras más significativas que existen en el mercado:

- Estructura compacta.
- Estructura semimodular. ( Estructura Americana)
- Estructura modular. (Estructura Europea)

### **1.- Estructura compacta**

Este tipo de autómatas se distingue por presentar en un solo bloque todos sus elementos, esto es, fuente de alimentación, CPU, memorias, entradas/salidas, etc..

Son los autómatas de gama baja o nanoautómatas los que suelen tener una estructura compacta. Su potencia de proceso suele ser muy limitada dedicándose a controlar máquinas muy pequeñas o cuadros de mando.



**Figura. II.38. Estructura del PLC compacta**

### **2.- Estructura semimodular**

Se caracteriza por separar las E/S del resto del autómata, de tal forma que en un bloque compacto están reunidas las CPU, memoria de usuario o de programa y fuente de alimentación y separadamente las unidades de E/S .

Son los autómatas de gama media los que suelen tener una estructura semimodular (Americana).

### **3.- Estructura modular**

Su característica principal es la de que existe un módulo para cada uno de los diferentes elementos que componen el autómata como puede ser una fuente de alimentación, CPU, E/S, etc. La sujeción de los mismos se hace por carril DIN, placa perforada o sobre RACK, en donde va alojado el BUS externo de unión de los distintos módulos que lo componen.



Son los autómatas de gama alta los que suelen tener una estructura modular, que permiten una gran flexibilidad en su constitución

**Estructura Interna** de cada uno de los diferentes elementos que componen el autómata, las funciones y funcionamiento de cada una de ellas.

El autómata está constituido por diferentes elementos, pero tres son los básicos:

- CPU
- Entradas y Salidas

Con las partes mencionadas podemos decir que tenemos un autómata pero para que sea operativo son necesarios otros elementos tales como:

- Fuente de alimentación
- Interfaces
- La unidad o consola de programación
- Los dispositivos periféricos

### **1.- CPU**

La CPU(Central Processing Unit) es la parte inteligente del sistema. Interpreta las instrucciones del programa de usuario y consulta el estado de las entradas. Dependiendo de dichos estados y del programa, ordena la activación de las salidas deseadas.

La CPU está constituida por los siguientes elementos:

- Procesador
- Memoria monitor del sistema
- Circuitos auxiliares

### **Procesador**

Está constituido por el microprocesador, el reloj(generator de onda cuadrada) y algún chip auxiliar.

El micropocesador es un circuito integrado (chip), que realiza una gran cantidad de operaciones, que podemos agrupar en:

- Operaciones de tipo lógico.
- Operaciones de tipo aritmético.
- Operaciones de control de la transferencia de la información dentro del autómata.

Para que el microprocesador pueda realizar todas estas operaciones está dotado de unos circuitos internos que son los siguientes:

- **Circuitos de la unidad aritmética y lógica o ALU:** Es la parte del  $\mu p$  donde se realizan los cálculos y las decisiones lógicas para controlar el autómata.
- **Circuitos de la unidad de control (UC) o Decodificador de instrucciones:** Decodifica las instrucciones leídas en memoria y se generan las señales de control.
- **Acumulador:** Es la encargada de almacenar el resultado de la última operación realizada por el ALU.
- **Flags:** Flags, o indicadores de resultado, que pueden ser consultados por el programa.
- **Contador de programa:** Encargada de la lectura de las instrucciones de usuario.

- **Bus(interno):** No son circuitos en si, sino zonas conductoras en paralelo que transmiten datos, direcciones, instrucciones y señales de control entre las diferentes partes del mp.

### **Memoria monitor del sistema**

Es una memoria de tipo ROM, y además del sistema operativo del autómata contiene las siguientes rutinas, incluidas por el fabricante.

- Inicialización tras puesta en tensión o reset.
- Rutinas de test y de respuesta a error de funcionamiento.
- Intercambio de información con unidades exteriores.
- Lectura y escritura en las interfaces de E/S.

### **Funciones básicas de la CPU**

En la memoria ROM del sistema, el fabricante ha grabado una serie de programas ejecutivos, software del sistema y es a estos programas a los que accederá el  $\mu$ p para realizar las funciones.

El software del sistema de cualquier autómata consta de una serie de funciones básicas que realiza en determinados tiempos de cada ciclo.

En general cada autómata contiene y realiza las siguientes funciones:

- Vigilar que el tiempo de ejecución del programa de usuario no exceda de un determinado tiempo máximo. A esta función se le denomina Watchdog.
- Ejecutar el programa usuario.
- Crear una imagen de las entradas, ya que el programa de usuario no debe acceder directamente a dichas entradas.

- Renovar el estado de las salidas en función de la imagen de las mismas, obtenida al final del ciclo de ejecución del programa usuario.
- Chequeo del sistema.

## **2.- Entradas y Salidas**

La sección de entradas mediante el interfaz, adapta y codifica de forma comprensible para la CPU las señales procedentes de los dispositivos de entrada o captadores.

Hay dos tipos de entradas:

- Entradas digitales
- Entradas analógicas

La sección de salida también mediante interfaz trabaja de forma inversa a las entradas, es decir, decodifica las señales procedentes de la CPU, y las amplifica y manda con ellas los dispositivos de salida o actuadores como lámparas, relés... aquí también existen unos interfaces de adaptación a las salidas de protección de circuitos internos.

Hay dos tipos de salidas:

- Salidas digitales
- Salidas analógicas

### **Entradas digitales**

Los módulos de entrada digitales permiten conectar al autómeta captadores de tipo todo o nada como finales de carrera pulsadores...

Los módulos de entrada digitales trabajan con señales de tensión, por ejemplo cuando por una vía llegan 24 voltios se interpreta como un "1" y cuando llegan cero voltios se interpreta como un "0"

El proceso de adquisición de la señal digital consta de varias etapas.

- Protección contra sobretensiones
- Filtrado
- Puesta en forma de la onda
- Aislamiento galvánico o por optoacoplador.

### **Entradas analógicas**

Los módulos de entrada analógicas permiten que los autómatas programables trabajen con accionadores de mando analógico y lean señales de tipo analógico como pueden ser la temperatura, la presión o el caudal.

Los módulos de entradas analógicas convierten una magnitud analógica en un número que se deposita en una variable interna del autómata. Lo que realiza es una conversión A/D, puesto que el autómata solo trabaja con señales digitales. Esta conversión se realiza con una precisión o resolución determinada (número de bits) y cada cierto intervalo de tiempo (período muestreo).

Los módulos de entrada analógica pueden leer tensión o intensidad.

El proceso de adquisición de la señal analógica consta de varias etapas:

- Filtrado
- Conversión A/D
- Memoria interna

### **Salidas digitales**

Un módulo de salida digital permite al autómata programable actuar sobre los preaccionadores y accionadores que admitan ordenes de tipo todo o nada.

El valor binario de las salidas digitales se convierte en la apertura o cierre de un relé interno del autómata en el caso de módulos de salidas a relé.

En los módulos estáticos (bornero), los elementos que conmutan son los componentes electrónicos como transistores o triacs, y en los módulos electromecánicos son contactos de relés internos al módulo.

Los módulos de salidas estáticos al suministrar tensión, solo pueden actuar sobre elementos que trabajan todos a la misma tensión, en cambio los módulos de salida electromecánicos, al ser libres de tensión, pueden actuar sobre elementos que trabajen a tensiones distintas.

El proceso de envío de la señal digital consta de varias etapas:

- Puesta en forma
- Aislamiento
- Circuito de mando (relé interno)
- Protección electrónica
- Tratamiento cortocircuitos

### **Salidas analógicas**

Los módulos de salida analógica permiten que el valor de una variable numérica interna del autómata se convierta en tensión o intensidad.

Lo que realiza es una conversión D/A, puesto que el autómata solo trabaja con señales digitales. Esta conversión se realiza con una precisión o resolución determinada (numero de bits) y cada cierto intervalo de tiempo (periodo muestreo).

Esta tensión o intensidad puede servir de referencia de mando para actuadores que admitan mando analógico como pueden ser los variadores de velocidad, las etapas de los tiristores de los hornos, reguladores de temperatura... permitiendo al autómata realiza funciones de regulación y control de procesos continuos.

El proceso de envío de la señal analógica consta de varias etapas:

- Aislamiento galvánico
- Conversión D/A
- Circuitos de amplificación y adaptación
- Protección electrónica de la salida

Como hemos visto las señales analógicas sufren un gran proceso de adaptación tanto en los módulos de entrada como en los módulos de salida. Las funciones de conversión A/D y D/A que realiza son esenciales. Por ello los módulos de E/S analógicos se les consideran módulos de E/S especiales.

### **3.- Fuente de alimentación**

La fuente de alimentación proporciona las tensiones necesarias para el funcionamiento de los distintos circuitos del sistema.

La alimentación a la CPU puede ser de continua a 24 Vcc, tensión muy frecuente en cuadros de distribución, o en alterna a 110/220 Vca. En cualquier caso es la propia CPU la que alimenta las interfaces conectadas a través del bus interno.

La alimentación a los circuitos E/S puede realizarse, según tipos, en alterna a 48/110/220 Vca o en continua a 12/24/48 Vcc.

La fuente de alimentación del autómata puede incorporar una batería tampón, que se utiliza para el mantenimiento de algunas posiciones internas y del

programa usuario en memoria RAM, cuando falla la alimentación o se apaga el autómata.

#### **4.- Interfaces**

En el control de un proceso automatizado, es imprescindible un dialogo entre operador-máquina junto con una comunicación entre la máquina y el autómata, estas comunicaciones se establecerán por medio del conjunto de entradas y salidas del citado elemento.

Los autómatas son capaces de manejar tensiones y corrientes de nivel industrial, gracias a que disponen un bloque de circuitos de interfaz de E/S muy potente, que les permite conectarse directamente con los sensores y accionamientos del proceso.

De entre todos los tipos de interfaces que existen, las interfaces específicas permiten la conexión con elementos muy concretos del proceso de automatización. Se pueden distinguir entre ellas tres grupos bien diferenciados:

- Entradas / salidas especiales.
- Entradas / salidas inteligentes.
- Procesadores periféricos inteligentes.

Las interfaces especiales del primer grupo se caracterizan por no influir en las variables de estado del proceso de automatización. Únicamente se encargan de adecuar las E/S, para que puedan ser inteligibles por la CPU, si son entradas, o para que puedan ser interpretadas correctamente por actuadores (motores, cilindros, etc.), en el caso de las salidas.



Las del segundo grupo admiten múltiples modos de configuración, por medio de unas combinaciones binarias situadas en la misma tarjeta. De esta forma se descarga de trabajo a la unidad central, con las ventajas que conlleva.

Los procesadores periféricos inteligentes, son módulos que incluyen su propio procesador, memorias y puntos auxiliares de entrada / salida. Estos procesadores contienen en origen un programa especializado en la ejecución de una tarea concreta, a la que le basta conocer los puntos de consigna y los parámetros de aplicación para ejecutar, de forma autónoma e independiente de la CPU principal, el programa de control.

#### **5.- La unidad o consola de programación**

Es el conjunto de medio hardware y software mediante los cuales el programador introduce y depura sobre las secuencias de instrucciones (en uno u otro lenguaje) que constituyen el programa a ejecutar.

#### **6.- Los dispositivos periféricos**

##### **Memoria**

La memoria es el almacén donde el autómata guarda todo cuanto necesita para ejecutar la tarea de control.

Datos del proceso:

- Señales de planta, entradas y salidas.
- Variables internas, de bit y de palabra.
- Datos alfanuméricos y constantes.

Datos de control:

- Instrucciones de usuario (programa)

- Configuración del autómata (modo de funcionamiento, número de e/s conectadas, ...)

Existen varios tipos de memorias:

- RAM. Memoria de lectura y escritura.
- ROM. Memoria de solo lectura, no reprogramable.
- EPROM. Memoria de solo lectura, reprogramables con borrado por ultravioletas.
- EEPROM. Memoria de solo lectura, alterables por medios eléctricos.

La memoria RAM se utiliza principalmente como memoria interna, y únicamente como memoria de programa en el caso de que pueda asegurarse el mantenimiento de los datos con una batería exterior.

La memoria ROM se utiliza para almacenar el programa monitor del sistema como hemos visto en el apartado dedicado a la CPU.

Las memorias EPROM se utilizan para almacenar el programa de usuario, una vez que ha sido convenientemente depurada.

Las memorias EEPROM se emplean principalmente para almacenar programas, aunque en la actualidad es cada vez más frecuente el uso de combinaciones RAM + EEPROM, utilizando estas últimas como memorias de seguridad que salvan el contenido de las RAM. Una vez reanudada la alimentación, el contenido de la EEPROM se vuelca sobre la RAM. Las soluciones de este tipo están sustituyendo a las clásicas RAM + batería puesto que presentan muchos menos problemas.

### **Memoria interna**

En un autómata programable, la memoria interna es aquella que almacena el estado de las variables que maneja el autómata: entradas, salidas, contadores, relés internos, señales de estado, etc. Esta memoria interna se encuentra dividida en varias áreas, cada una de ellas con un cometido y características distintas.

La clasificación de la memoria interna no se realiza atendiendo a sus características de lectura y escritura, sino por el tipo de variables que almacena y el número de bits que ocupa la variable. Así, la memoria interna del autómata queda clasificada en las siguientes áreas.

**Área de imágenes de entradas/salidas y Área interna (IR).**

En esta área de memoria se encuentran:

- Los canales (registros) asociados a los terminales externos (entradas y salidas).
- Los relés (bit) internos (no correspondidos con el terminal externo), gestionados como relés de E/S.
- Los relés E/S no usados pueden usarse como IR.
- No retienen estado frente a la falta de alimentación o cambio de modo de operación.

**Área especial (SR).**

Son relés de señalización de funciones particulares como:

- Servicio (siempre ON, OFF)
- Diagnósis (señalización o anomalías)
- Temporizaciones (relojes a varias frecuencias)
- Cálculo
- Comunicaciones

- Accesible en forma de bit o de canal.
- No conservan su estado en caso de fallo de alimentación o cambio de modo.

### **Área auxiliar (AR).**

Contienen bits de control e información de recursos de PLC como: Puerto RS232C, puertos periféricos, casetes de memoria...

- Se dividen en dos bloques:
  - Señalización: Errores de configuración, datos del sistema.
  - Memorización y gestión de datos
- Es un área de retención.
- Accesible en forma de bit o de canal.
- No conservan su estado en caso de fallo de alimentación o cambio de modo.

### **Área de enlace (LR).**

- Se utilizan para el intercambio de datos entre dos PLC's unidos en forma PC
- Dedicados al intercambio de información entre PLC's.
- Si no se utilizan como LR pueden usarse como IR.
- Accesible en forma de bit o canal.
- No conservan su estado en caso de fallo de alimentación o cambio de modo.

### **Área de retención (HR).**

- Mantienen su estado ante fallos de alimentación o cambio de modo de PLC.

- Son gestionados como los IR y direccionables como bit o como canal.

### **Área de temporizadores y contadores (TIM/CNT).**

- Es el área de memoria que simula el funcionamiento de estos dispositivos.
- Son usados por el PLC para programar retardos y contajes.

### **Área de datos (DM).**

- Se trata de memoria de 16 bits ( palabra).
- Utilizable para gestión de valores numéricos.
- Mantiene su estado ante cambios de modos de trabajo o fallo de alimentación.
- Direccionables como Canal(palabra).
- Esta área suele contener los parámetros de configuración del PLC(setup).

Las variables contenidas en la memoria interna, pueden ser consultadas y modificadas continuamente por el programa, cualquier número de veces. Esta actualización continua de los datos obliga a construir la memoria con dispositivos RAM.

### **Memoria de programa**

La memoria de programa, normalmente externa y enchufable a la CPU mediante casete de memoria, almacena el programa escrito por el usuario para su aplicación.

Cada instrucción del usuario ocupa un paso o dirección del programa.

Las memorias de programa o memorias de usuario son siempre de tipo permanente RAM + batería o EPROM/EEPROM . Por lo general la mayoría de

los fabricantes de autómatas ofrecen la posibilidad de utilizar memorias RAM con batería para la fase de desarrollo y depuración de los programas, y de pasar estos a memorias no volátiles EPROM o EEPROM una vez finalizada esta fase.

La ejecución del programa en el módulo es siempre prioritaria, de forma que si se da tensión al autómata con un módulo conectado, la CPU ejecuta su programa y no el contenido en memoria RAM interna.

#### 2.5.1.4. Principio de funcionamiento del PLC

##### Ciclo de ejecución

- **Tratamiento Interno.** El sistema supervisa el PLC (ejecutabilidad del programa, gestión de tiempo, actualización indicadores RUN, I/O, ERR, COM, detección de pasos RUN/STOP, Tratamiento de ordenes del terminal de programación etc..
- **Confirmación de las entradas.**-Escritura en memoria del estado que presentan las entradas.
- **Tratamiento del programa.**- Se ejecuta el programa de usuario.
- **Actualización de las salidas:** escritura de bits de salidas %Q

##### Ciclo de Funcionamiento

- **Autómata en run.**- El procesador ejecuta el tratamiento interno, la confirmación de las entradas, tratamiento del programa y actualización de las salidas.
- **Autómata en stop.**- Únicamente se ejecuta el tratamiento interno y la actualización de las salidas

La duración del programa será 500 ms de lo contrario se provoca la parada – watchdog



**Figura. II.39. Ejecucion normal (ciclica) del PLC**

### **Ejecución periódica**

En este caso. La confirmación de las entradas, el tratamiento del programa y la actualización de las salidas, se realiza de forma periódica dependiendo de un tiempo definido por el usuario durante la configuración (de 2 a 150 ms).

Al comienzo del ciclo, un temporizador de programa se ajusta al valor configurado. El ciclo habrá de terminar antes que finalice el tiempo programado en el temporizador. AL finalizar el ciclo el temporizador se reinicia.

Si el tiempo de ciclo sobrepasa el programado, el bit de sistema %S19 pasa a 1.

Su comprobación y reinicialización correrá a cargo del programa de usuario



**Figura. II.40. Ejecución periódica del PLC**

#### 2.5.1.5. Clases de Programación

Se definen cuatro lenguajes de programación normalizados. Esto significa que su sintaxis y semántica ha sido definida, no permitiendo particularidades distintivas (dialectos). Una vez aprendidos se podrá usar una amplia variedad de sistemas basados en esta norma.

Los lenguajes consisten en dos de tipo literal y dos de tipo gráfico:

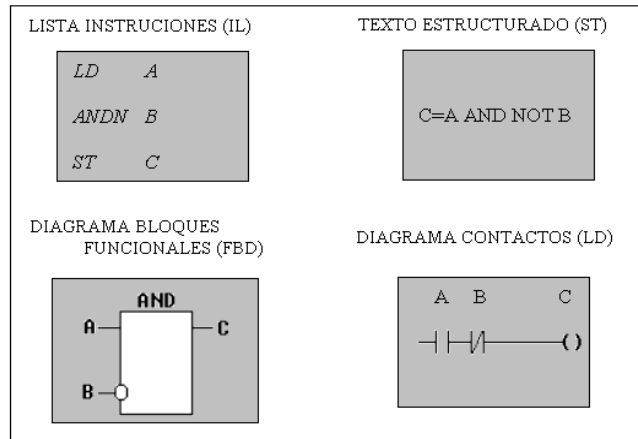
Literales:

- Lista de instrucciones (IL).
- Texto estructurado (ST).

Gráficos:

- Diagrama de contactos (LD).
- Diagrama de bloques funcionales (FBD).





**Figura. II.41. Tipos de Lenguajes de Programación**

### 2.5.1.6. Aplicación

EL PLC por sus especiales características de diseño tiene un campo de aplicación muy extenso. La constante evolución del Hardware y Software amplía continuamente este campo para poder satisfacer las necesidades que se detectan en el aspecto de sus posibilidades reales.

El PLC por sus especiales características de diseño tiene un campo de aplicación muy extenso. La constante evolución del hardware y software amplía constantemente este campo para poder satisfacer las necesidades que se detectan en el espectro de sus posibilidades reales.

Su utilización se da fundamentalmente en aquellas instalaciones en donde es necesario un proceso de maniobra, control, señalización, etc., por tanto, su aplicación abarca desde procesos de fabricación industriales de cualquier tipo a transformaciones industriales, control de instalaciones, etc.

Sus reducidas dimensiones, la extremada facilidad de su montaje, la posibilidad de almacenar los programas para su posterior y rápida utilización, la modificación o alteración de los mismos, etc., hace que su eficacia se aprecie fundamentalmente en procesos en que se producen necesidades tales como:

- Maniobra de máquinas
- Maquinaria industrial de plástico
- Máquinas transfer
- Maquinaria de embalajes
- Maniobra de instalaciones:
- Instalación de aire acondicionado, calefacción...
- Instalaciones de seguridad
- Señalización y control:
- Chequeo de programas
- Señalización del estado de procesos

### 2.5.2 Diseño

#### **El diseño del sistema de mezclado de líquidos se realiza en Labview**

LabVIEW (Laboratory Virtual Instrument Engineering Workbench) es un sistema de desarrollo basado en programación gráfica, orientado a desarrollar aplicaciones para instrumentación, el cual integra una serie de librerías para comunicación con instrumentos electrónicos como GPIB, RS232 o RS485 con tarjetas de adquisición de datos, con sistemas de acondicionamiento como VXI o SCXI, comunicaciones en redes TCP/IP, UDP, o en los estándares de software

.COM, .OLE, .DDE, .DDL o ActiveX para Windows, así como AppleEvents para MacOS o PIPE para UNIX.

Es un lenguaje de programación gráfico para el diseño de sistemas de adquisición de datos, instrumentación y control. Labview permite diseñar interfaces de usuario mediante una consola interactivo basado en software. Usted puede diseñar especificando su sistema funcional, su diagrama de bloques o una notación de diseño de ingeniería. Labview es a la vez compatible con herramientas de desarrollo similares y puede trabajar con programas de otra área de aplicación, como por ejemplo Matlab. Tiene la ventaja de que permite una fácil integración con hardware, específicamente con tarjetas de medición, adquisición y procesamiento de datos (incluyendo adquisición de imágenes).

Los programas utilizados en LabVIEW se llaman instrumentos virtuales “VIs”, ya que tienen la apariencia de los instrumentos reales, sin embargo, poseen analogías con funciones provenientes de lenguajes de programación convenientes.

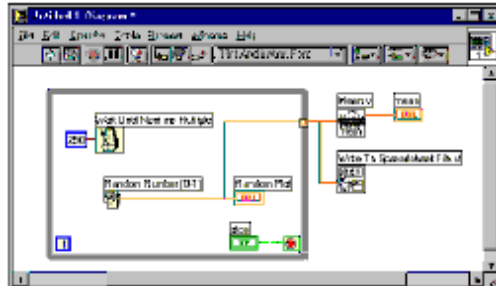
Los VIs se compones de:

**Panel frontal:** Interfase gráfica que simula el panel de un instrumento real, el cual permite entrada y salida de datos, incluye; perillas, pulsadores, botones, gráficos, controles e indicadores.



Figura. II.42. Panel frontal de LabVIEW

**Diagrama de Bloques:** Contiene el código fuente gráfico de VI, posee funciones y estructuras que relacionan las entradas con las salidas creadas en el panel frontal.



**Figura. II.43. Diagrama de bloques de LabVIEW**

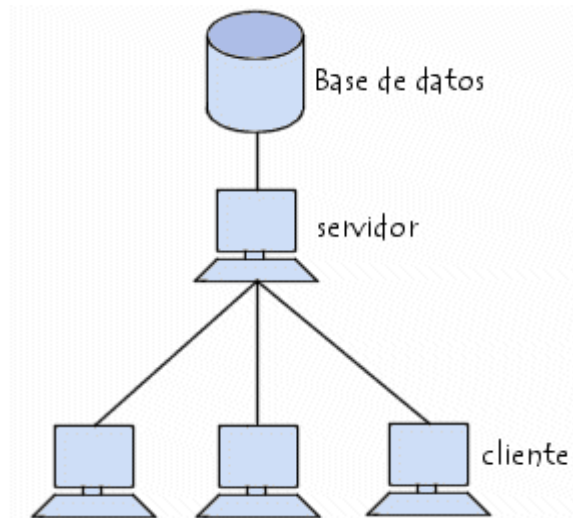
### Aplicaciones de LabVIEW

Labview tiene su mayor aplicación en sistemas de medición, como monitoreo de procesos y aplicaciones de control, un ejemplo de esto pueden ser sistemas de monitoreo en transportación, Laboratorios para clases en universidades, procesos de control industrial. Labview es muy utilizado en procesamiento digital de señales (wavelets, FFT, Total Distorsion Harmonic TDH), procesamiento en tiempo real de aplicaciones biomédicas, manipulación de imágenes y audio, automatización, diseño de filtros digitales, generación de señales, entre otras, etc

### 2.5.3. Base de Datos

Una base de datos (cuya abreviatura es BD) es una entidad en la cual se pueden almacenar datos de manera estructurada, con la menor redundancia posible. Diferentes programas y diferentes usuarios deben poder utilizar estos datos. Por lo tanto, el concepto de base de datos generalmente está relacionado con el de [red](#) ya que se debe poder compartir esta información. De allí el término **base**. "Sistema de

información" es el término general utilizado para la estructura global que incluye todos los mecanismos para compartir datos que se han instalado.



**Figura. II.44. Ejemplo de una Base de datos**

### **¿Por qué utilizar una base de datos?**

Una base de datos proporciona a los usuarios el acceso a datos, que pueden visualizar, ingresar o actualizar, en concordancia con los derechos de acceso que se les hayan otorgado. Se convierte más útil a medida que la cantidad de datos almacenados crece.

Una base de datos puede ser local, es decir que puede utilizarla sólo un usuario en un equipo, o puede ser distribuida, es decir que la información se almacena en equipos remotos y se puede acceder a ella a través de una red.

La principal ventaja de utilizar bases de datos es que múltiples usuarios pueden acceder a ellas al mismo tiempo.

### **Administración de bases de datos**

Rápidamente surgió la necesidad de contar con un sistema de administración para controlar tanto los datos como los usuarios. La administración de bases de datos se realiza con un sistema llamado **DBMS** (Database management system [Sistema de

administración de bases de datos]). El DBMS es un conjunto de servicios (aplicaciones de software) para administrar bases de datos, que permite:

- Un fácil acceso a los datos
- El acceso a la información por parte de múltiples usuarios
- La manipulación de los datos encontrados en la base de datos (insertar, eliminar, editar)

## CAPITULO III PARTE INVESTIGATIVA

### 3.1. Introducción

En este capítulo se realiza un análisis del control y almacenamiento de datos en respuesta a las necesidades que presentan las empresas industriales se han venido creando día a día nuevos sistemas industriales para ayudar a resolver los problemas manuales que atraviesan dichas empresas para lo cual nos valemos del estándar IEC 611 31-3 para que nos permita realizar de manera correcta este tipo de sistemas mecatrónicos y además se identifican y analizan las metodologías de desarrollo que contengan los métodos, las herramientas y los procedimientos específicos para la implementación de un sistema mecatrónico con el objetivo de identificar las características más apropiadas de las metodologías de desarrollo para sistemas mecatrónicos escogidas y a las cuales incluiremos actividades que consideramos necesarias trabajando con mayor profundidad.

### 3.2. Análisis del estándar IEC 61131-3

El estándar internacional IEC 61131 es una colección completa de estándares referentes a controladores programables y sus periféricos asociados.

#### **IEC 61131-3: UN RECURSO DE PROGRAMACIÓN ESTANDAR**

En la actualidad aún siguen persistiendo sistemas de control específicos del fabricante, con programación dependiente y conexión compleja entre distintos sistemas de control. Esto significa para el usuario costos elevados, escasa flexibilidad y falta de normalización en las soluciones al control industrial.

IEC 61131 es el primer paso en la estandarización de los autómatas programables y sus periféricos, incluyendo los lenguajes de programación que se deben utilizar. Esta norma se divide en cinco partes:

- Parte 1: Vista general.
- Parte 2: Hardware.
- Parte 3: Lenguaje de programación.
- Parte 4: Guías de usuario.
- Parte 5: Comunicación.

#### **3.2.1 La Norma IEC-61131 se refiere a:**

Los autómatas programables (AP ó PLC's) y a sus periféricos correspondientes, tales como:



- Los equipos de programación y depuración (PADT's)
- Los equipos de ensayo (TE's)
- Los interfaces hombre-máquina (MMI's)

Esta norma no trata del sistema automatizado, del cual el autómata programable es un componente básico.

PADT: Programming And Debugging Tool

TE: Test Equipment

MMI: Man-Machine Interface

### **3.2.2. La finalidad de esta Norma IEC-61131 es:**

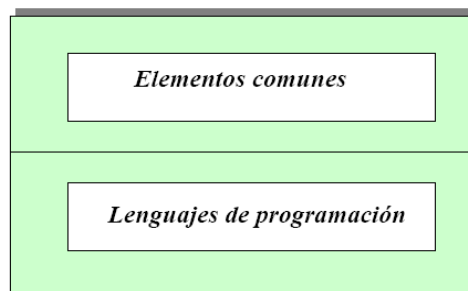
- Definir e identificar las características principales que se refieren a la selección y aplicación de los PLC's y sus periféricos.
- Especificar los requisitos mínimos para las características funcionales, las condiciones de servicio, los aspectos constructivos, la seguridad general y los ensayos aplicables a los PLC's y sus periféricos.
- Definir los lenguajes de programación de uso más corriente, las reglas sintácticas y semánticas, el juego de instrucciones fundamental, los ensayos y los medios de ampliación y adaptación de los equipos.
- Dar a los usuarios una información de carácter general y unas directrices de aplicación.
- Definir las comunicaciones entre los PLC's y otros sistemas.

IEC 61131-3 pretende ser la base real para estandarizar los lenguajes de programación en la automatización industrial, haciendo el trabajo independiente de cualquier compañía.

Hay muchas maneras de describir el trabajo desarrollado en la tercera parte de esta norma, indicaremos algunas de ellas son:

- IEC 61131-3 es el resultado del gran esfuerzo realizado por 7 multinacionales a los que se añaden muchos años de experiencia en el campo de la automatización industrial.
- Incluye 200 páginas de texto aproximadamente, con más de 60 tablas.
- IEC 61131-3 son las especificaciones de la sintaxis y semántica de un lenguaje de programación, incluyendo el modelo de software y la estructura del lenguaje.

**Otra visión distinta es dividir el estándar en dos partes**



**Figura. III.1. El estándar IEC-61131**

- Elementos comunes.
- Lenguajes de programación.

<p><b><u>Elementos comunes</u></b></p> <ul style="list-style-type: none"> <li>• Tipos de datos y variables</li> <li>• Modelo de software</li> <li>• Modelo de comunicación de datos</li> <li>• Modelo de programación</li> <li>• Unidades de organización del programa</li> <li>• Gráfico Funcional Secuencial (SFC)</li> <li>• Elementos de configuración</li> </ul>
<p><b><u>Lenguajes de programación</u></b></p> <ul style="list-style-type: none"> <li>• Lista de instrucciones (IL)</li> <li>• Texto estructurado (ST)</li> <li>• Diagrama de bloques funcionales (FBD)</li> <li>• Diagrama de contactos (LD)</li> </ul>

**Figura. III.2. Detalle Elementos comunes y lenguajes de programación**

## **Elementos Comunes**

### **Tipos de datos**

Dentro de los elementos comunes, se definen los tipos de datos. Los tipos de datos previenen de errores en una fase inicial, como por ejemplo la división de un dato tipo fecha por un número entero. Los tipos comunes de datos son: variables booleanas, número entero, número real, byte y palabra, pero también fechas, horas del día y cadenas (strings).

Basado en estos tipos de datos, el usuario puede definir sus propios tipos de datos, conocidos como tipos de datos derivados. De este modo, se puede definir por ejemplo un canal de entrada analógica como un tipo de dato.

### **Variables**

Las variables permiten identificar los objetos de datos cuyos contenidos pueden cambiar, por ejemplo, los datos asociados a entradas, salidas o a la memoria del autómata programable. Una variable se puede declarar como uno de los tipos de datos elementales definidos o como uno de los tipos de datos derivados. De este

modo se crea un alto nivel de independencia con el hardware, favoreciendo la reusabilidad del software.

La extensión de las variables está normalmente limitada a la unidad de organización en la cual han sido declaradas como locales. Esto significa que sus nombres pueden ser reutilizados en otras partes sin conflictos, eliminando una frecuente fuente de errores. Si las variables deben tener una extensión global, han de ser declaradas como globales utilizando la palabra reservada `VAR_GLOBAL`.

Pueden ser asignados parámetros y valores iniciales que se restablecen al inicio, para obtener la configuración inicial correcta.

### **Configuración, recursos y tareas**

Para entender esto mejor, vamos a ver el modelo de software, que define IEC 61131-3

Al más alto nivel, el elemento software requerido para solucionar un problema de control particular puede ser formulado como una configuración. Una configuración es específica para un tipo de sistema de control, incluyendo las características del hardware: procesadores, direccionamiento de la memoria para los canales de I/O y otras capacidades del sistema.

Dentro de una configuración, se pueden definir uno o más recursos. Se puede entender el recurso como un procesador capaz de ejecutar programas IEC.

Con un recurso, pueden estar definidas una o más tareas. Las tareas controlan la ejecución de un conjunto de programas y/o bloques de función. Cada una de ellos puede ser ejecutado periódicamente o por una señal de disparo especificada, como el cambio de estado de una variable.

Los programas están diseñados a partir de un diferente número de elementos de software, escrito en algunos de los distintos lenguajes definidos en IEC 61131-3. Típicamente, un programa es una interacción de Funciones y Bloques Funcionales, con capacidad para intercambiar datos.

Funciones y bloques funcionales son las partes básicas de construcción de un programa, que contienen una declaración de datos y variables y un conjunto de instrucciones..

Comparado esto con un PLC convencional, éste contiene un solo recurso, ejecutando una tarea que controla un único programa de manera cíclica.

IEC 61131-3 incluye la posibilidad de disponer de estructuras más complejas. El futuro que incluye multi-procesamiento y gestión de programas por eventos ¡Y no está muy lejos!, observar simplemente las características de los sistemas distribuidos o los sistemas de control de tiempo real. IEC 61131-3 está disponible para un amplio rango de aplicaciones, sin tener que conocer otros lenguajes de programación adicionales.

### **Unidades de Organización de Programa**

Dentro de IEC 1131-3, los programas, bloques Funcionales y funciones se denominan Unidades de Organización de Programas, POU's.

### **Funciones**

IEC 61131-3 especifica funciones estándar y funciones definidas por usuario. Las funciones estándar son por ejemplo ADD (suma), ABS (valor absoluto), SQRT (raíz cuadrada), SIN (seno), y COS (coseno). Las funciones definidas por usuario, una vez implementadas pueden ser usadas indefinidamente en cualquier POU (program organization units).

Las funciones no pueden contener ninguna información de estado interno, es decir, que la invocación de una función con los mismos argumentos (parámetros de entrada) debe suministrar siempre el mismo valor (salida).

### **Bloques Funcionales, FB's**

Los bloques funcionales son los equivalentes de los circuitos integrados, IC's, que representan funciones de control especializadas.

Los FB's contienen tanto datos como instrucciones, y además pueden guardar los valores de las variables (que es una de las diferencias con las funciones). Tienen un interfaz de entradas y salidas bien definido y un código interno oculto, como un circuito integrado o una caja negra.

De este modo, establecen una clara separación entre los diferentes niveles de programadores, o el personal de mantenimiento. Un lazo de control de temperatura, PID, es un excelente ejemplo de bloque funcional. Una vez definido, puede ser usado una y otra vez, en el mismo programa, en diferentes programas o en distintos proyectos. Esto lo hace altamente reutilizable.

Los bloques funcionales pueden ser escritos por el usuario en alguno de los lenguajes de la norma IEC, pero también existen FB's estándar (biestables, detección de flancos, contadores, temporizadores, etc.).

Existe la posibilidad de ser llamados múltiples veces creando copias del bloque funcional que se denominan instancias. Cada instancia llevará asociado un identificador y una estructura de datos que contenga sus variables de salida e internas.

### **Programas**

Los programas son “un conjunto lógico de todos los elementos y construcciones del lenguaje de programación que son necesarios para el tratamiento de señal previsto que se requiere para el control de una máquina o proceso mediante el sistema de autómatas programables”. Un programa puede contener, aparte de la declaración de tipos de datos, variables y su código interno, distintas instancias de funciones y bloques funcionales.

### **Gráfico Funcional Secuencial (SFC)**

SFC describe gráficamente el comportamiento secuencial de un programa de control. Esta definición deriva de las Redes de Petri y Grafset (IEC 848), con las modificaciones adecuadas para convertir las representaciones de una norma de documentación en un conjunto de elementos de control de ejecución para una POU de un autómatas programables.

SFC ayuda a estructurar la organización interna de un programa, y a descomponer un problema en partes manejables, manteniendo simultáneamente una visión global. Los elementos del SFC proporcionan un medio para subdividir una POU de un autómatas programables en un conjunto de etapas y transiciones interconectadas por medio de enlaces directos.

Cada etapa lleva asociados un conjunto de bloques de acción y a cada transición va asociada una condición de transición que cuando se cumple, causa la desactivación de la etapa anterior a la transición y la activación de la siguiente. Los bloques de acción permiten realizar el control del proceso. Cada elemento puede ser programado en alguno de los lenguajes IEC, incluyéndose el propio SFC. Dado que los elementos del SFC requieren almacenar información, las

únicas POU's que se pueden estructurar utilizando estos elementos son los bloques funcionales y los programas.

Se pueden usar secuencias alternativas y paralelas, comúnmente utilizadas en muchas aplicaciones. Debido a su estructura general, de sencilla comprensión, SFC permite la transmisión de información entre distintas personas con distintos niveles de preparación y responsabilidad dentro de la empresa.

### 3.2.3 Lenguajes de Programación

Se definen cuatro lenguajes de programación normalizados. Esto significa que su sintaxis y semántica ha sido definida, no permitiendo particularidades distintivas (dialectos). Una vez aprendidos se podrá usar una amplia variedad de sistemas basados en esta norma.

Los lenguajes consisten en dos de tipo literal y dos de tipo gráfico:

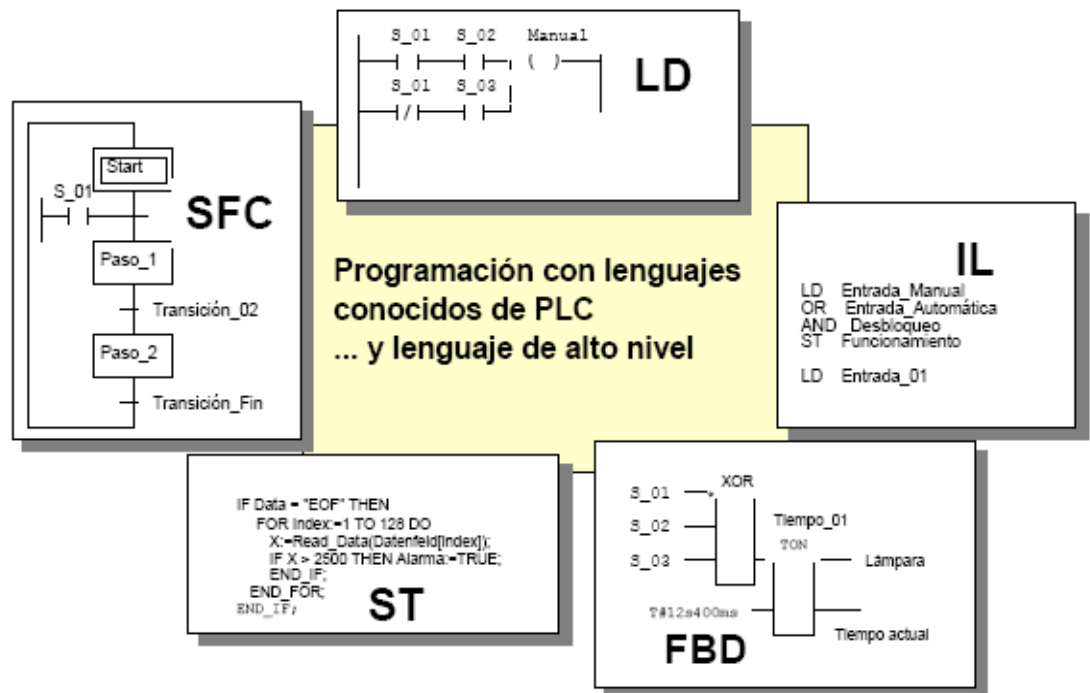
Literales:

- Lista de instrucciones (IL).
- Texto estructurado (ST).

Gráficos:

- Diagrama de contactos (LD).
- Diagrama de bloques funcionales (FBD).





**Figura. III.3. Lenguajes de Programación**

El ambiente de programación **IEC 61131-3** provee los cinco lenguajes de programación especificados por el estándar IEC 61131-3 para la programación de PLCs, con un sexto lenguaje llamado "Flow Char" (FC). Los cinco lenguajes principales son Gráficos de Funciones Secuenciales (GFS), Diagrama de Bloques Funcionales (DBF), Diagrama de Escalera, Texto Estructurado y Lista de Instrucciones.

**Las ventajas principales de IEC 61131-3 son:**

- Compatibilidad con muchos proveedores
- Gran capacidad para documentación y depuración
- La habilidad de usar diferentes lenguajes en un programa para tomar las ventajas de cada uno. Por favor, refiérase a la siguiente descripción de cada uno de los lenguajes del IEC 61131-3

La selección del lenguaje a ser usado para un desarrollo puede estar basado en la naturaleza de la aplicación, y viene determinado por las preferencias del programador.

**La elección del lenguaje de programación depende de:**

- Los conocimientos del programador,
- El problema a tratar,
- El nivel de descripción del proceso,
- La estructura del sistema de control,
- La coordinación con otras personas o departamentos.

Los cuatro lenguajes están interrelacionados y permiten su empleo para resolver conjuntamente un problema común según la experiencia del usuario.

El Diagrama de contactos (LD) tiene sus orígenes en los Estados Unidos.

Está basado en la presentación gráfica de la lógica de relés. Lista de Instrucciones (IL) es el modelo de lenguaje ensamblador basado un acumulador simple; procede del alemán 'Anweisungsliste, AWL.

El Diagramas de Bloques Funcionales (FBD) es muy común en aplicaciones que implican flujo de información o datos entre componentes de control.

Las funciones y bloques funcionales aparecen como circuitos integrados y es ampliamente utilizado en Europa. El lenguaje Texto estructurado (ST) es un lenguaje de alto nivel con orígenes en el Ada, Pascal y 'C'; puede ser utilizado para codificar expresiones complejas e instrucciones anidadas; este lenguaje dispone de estructuras para bucles (REPEAT-UNTIL; WHILE-DO), ejecución condicional (IF-THEN-ELSE; CASE), funciones (SQRT, SIN, etc.).

### **Top-down vs. Bottom-up-**

La norma también permite dos formas de desarrollar tu programa de control (ver figura 5): de arriba a abajo (Top-down) y de abajo a arriba (bottom-up). Puedes especificar inicialmente la aplicación completa y dividirla en partes, declarar las variables y demás. También puedes comenzar la programación desde abajo, por ejemplo, por medio de funciones y bloque funcionales. Por cualquiera de los caminos que elijas, IEC 61131-3 te ayudará durante todo el proceso.

### **Lenguaje Gráficos (LD y FBD)**

#### **Características comunes**

- Los **elementos del lenguaje gráfico** se representan mediante caracteres
- ISO/CEI 646 o elementos gráficos o semigráficos.
- Una **red** es un conjunto de elementos gráficos interconectados, que debe ser identificada (un número, p.e.).
- **Reglas de evaluación de redes. Vías de realimentación.**
- Posibilidad de **saltos y retornos** (condicionales e incondicionales).

#### **Diagrama de escalera (“Ladder Diagram”, LD)**

El Diagrama de Escalera Ladder Diagram (**LD**) es un excelente lenguaje gráfico para Lógicas discretas. También tiene la habilidad de incluir instrucciones de Funciones de bloque dentro de una línea. Los contactos y bobinas del diagrama

de escaleras pueden ser usados en el lenguaje Diagrama de Bloques Funcionales para control discreto o funciones.

**Características:**

- Barras de alimentación.
- Elementos de enlace y estados: flujo de energía.
- Contactos, bobinas y bloques funcionales
- Orden de evaluación de las redes.

**Diagrama de Bloques funcionales (FBD)**

Diagramas de Bloques Funcionales Function Block Diagram, (**FBD**) es un lenguaje gráfico usado para construir procedimientos complejos a partir de una librería de funciones. Las librerías estándar, tales como de Matemática o Control, pueden ser combinadas con librerías de funciones personalizadas, tales como llamada de modem, interfase HART, controladores PID y Modbus Maestro para crear programas de aplicación de Diagramas de Bloques Funcionales

Las funciones personalizadas son similares a aquellas disponibles en el diagrama de escalera con las funciones del TelePace y TelePACE C.

**Características:**

- La representación es coherente con la Norma CEI 617-12.
- Las salidas de los bloques funcionales no se conectarán entre sí (se precisa bloque “OR”).

- La evaluación de una red estará terminada antes de la siguiente.

### Lenguajes Literales (IL y ST)

- La **representación de literales.**
- Los siguientes **elementos de estructuración de las POU's:**

TYPE ... END\_TYPE

VAR ... END\_VAR

VAR\_INPUT ... END\_VAR

VAR\_OUTPUT ... END\_VAR

VAR\_IN\_OUT... END\_VAR

VAR\_EXTERNAL .. END\_VAR

FUNCTION ... END\_FUNCTION

FUNCTION\_BLOCK ... END\_FUNCTION\_BLOCK

PROGRAM .. END\_PROGRAM

STEP ... END\_STEP

TRANSITION ... END\_TRANSITION

ACTION ... END\_ACTION

```
LD Entrada_Manual
OR Entrada_Automática
AND Desbloqueo
ST Funcionamiento

LD Entrada_01
```

**IL**

```
IF Data = "EOF" THEN
  FOR index:=1 TO 128 DO
    X:=Read_Data(Datenfeld[Index]);
    IF X > 2500 THEN Alarma:=TRUE;
  END_IF;
END_FOR;
END_IF;
```

**ST**

**Figura. III.4. Nomenclatura de IL Y ST**

### Lista de Instrucciones (“Instruction List”, IL)

Lista de Instrucciones Instruction List (IL) es un lenguaje de bajo nivel, similar a código de máquina. Es útil para pequeñas aplicaciones que requieran rápida ejecución para optimizar una aplicación

#### Características:

- **Instrucciones:**

Etiqueta	Operador	Operando	Comentario
START:	LD	Arranque	(*Pulsar botón *)
	ANDN	%MX5	(* No inhibidor *)
	ST	SALIDA	(* Conecta calefactor *)

Tabla III.1. Instrucciones

- **Operadores:**

LD, ST, S, R, AND, OR, XOR, ADD, SUB, MUL, DIV, GT, GE, EQ, NE, LE, LT, JMP, CAL, RET, )

- **Funciones y bloques funcionales.** Se coloca el nombre en el campo del operando y los parámetros, si los lleva, en entre paréntesis. Se emplea la operación CAL.

### Texto estructurado (“Structured Text”, ST)

Texto Estructurado Structured Text (ST) es un lenguaje tipo Basic que es usado para procedimientos complejos o cálculos que no pueden ser fácilmente implementados utilizando lenguajes gráficos.

#### Características:

- Similar al Pascal. Alto nivel, estructurado en bloques.

- Expresiones: Es una construcción sintáctica que al se evaluada proporciona un valor. Está compuesta por operadores y operandos.
- La evaluación se efectúa según una tabla de prioridades.
- **Sentencias:**
  - De asignación (:=),
  - De control del flujo del programa: llamada a bloque funcional, RETURN
  - De selección: IF, CASE
  - **De iteración: FOR, WHILE, REPEAT, EXIT**

### **Diagrama Funcional Secuencial**

Los Gráficos de Funciones Secuenciales (SFC) es un lenguaje gráfico que provee estructura general y coordinación a las secuencias del programa. Soporta selecciones alternativas y secuencias paralelas

### **Características**

- En sus orígenes fue el GRAFCET (Grafico funcional de control de etapa transición)
- Eficaz técnica para describir el comportamiento secuencial de un proceso y un programa
- Se usa para distribuir el problema de control
- Permite un diagnostico rápido

### **GRAFCET**

- Las etapas o estados implican acciones asociadas

- Las transiciones gobiernan los cambios del estado
- Las flechas indican la dirección de cambio
- Pueden darse esquemas menos lineales
- Los elementos básicos son los steps con bloques de acciones y transiciones
- Soporta alternativas y secuencias paralelas

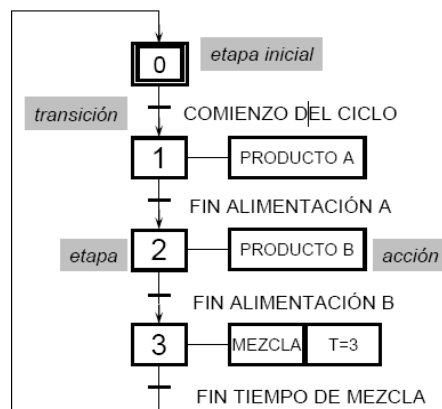


Figura. III.5. GRAFCET

### Salto condicional de etapa

Direccionamiento específico hacia atrás y adelante

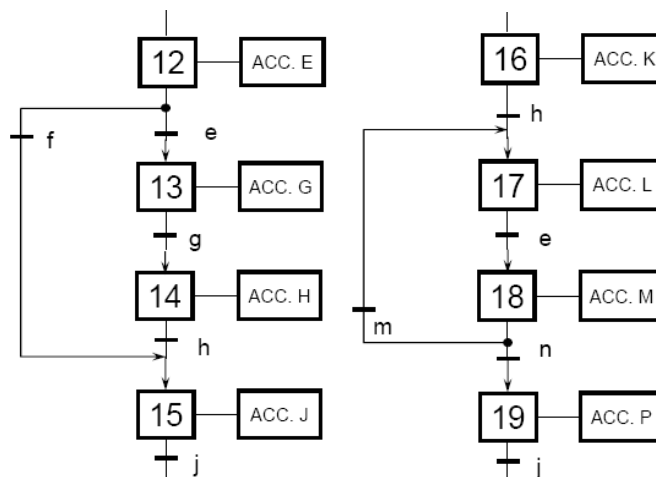
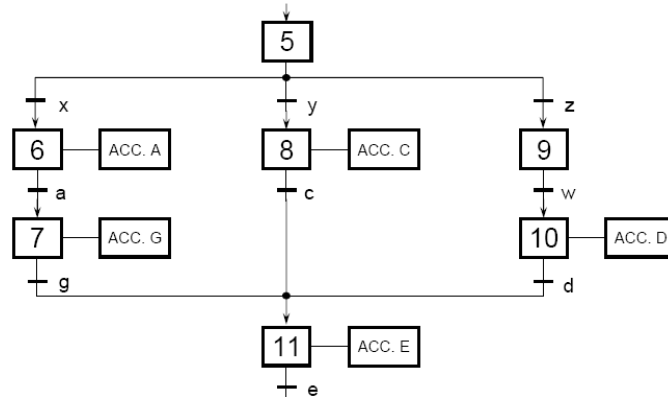


Figura. III.6. GRAFCET Salto condicional de etapa

### Direccionamiento condicional



Elección condicional entre varias secuencias posibles



**Figura. III.7. GRAFCET Direccionamiento condicional**

Gráfico de Flujo es un lenguaje gráfico que es usado para describir operaciones secuenciales en una aplicación. Un diagrama de Grafico de Flujo está compuesto por acciones a ser ejecutadas y pruebas de las acciones ejecutadas.

Adicionalmente, el usuario tiene la capacidad de crear librerías estándar, escritas en C o en lenguajes IEC 61131-3, que pueden ser llamadas desde la aplicación. El programador puede escoger utilizar tantos lenguajes IEC como desee en su aplicación. Esta modularidad y capacidad de re-utilización del código de aplicación, combinado con interfases fáciles de usar, funciones de búsqueda poderosas, depuración en línea, Gerencia de proyecto y generación de documentación, reducen el tiempo de diseño, desarrollo, pruebas, arranque y mantenimiento. La aplicación puede ser cargada en la SCADAPack, ya sea a través de una conexión directa o a través de la red de comunicaciones del SCADA.

**Características claves de la calidad IEC 61131-3**

- **Software Estructurado** - a través de configuraciones, Recursos y Unidades de Organización de Programa (POUs).
- **Tipeado de Datos** - a través de lenguajes que restringen las operaciones a los tipos de datos adecuados.
- **Control de la Ejecución** - a través del uso de Tareas.
- **Comportamiento Secuencial Complejo** - a través de los Esquemas Funcional Secuencial.
- **Encapsulación del Software** - a través del uso de POU, estructuras y tipos de datos complejos.

#### Beneficios del estándar para los usuarios

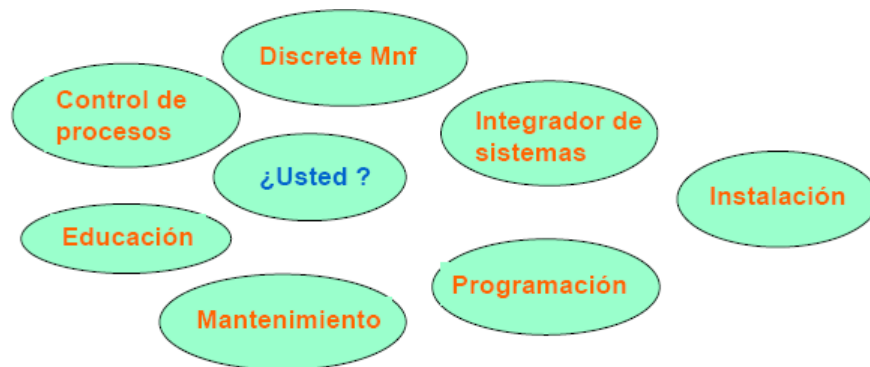


Figura. III.8. Beneficios del estándar para los usuarios

1. Se reduce el gasto en recursos humanos, formación, mantenimiento y consultoría.
2. Evita las fuentes habituales de problemas por el alto nivel de flexibilidad y reusabilidad del software.
3. Las técnicas de programación son utilizables en amplios sectores (control industrial en general).

4. Combinan adecuadamente diferentes elementos que pueden provenir de diferentes fabricantes, programas, proyectos.

5. Incrementa la conectividad y comunicación entre los distintos departamentos y compañías.

### **Implementaciones**

Cumplir todos los requerimientos de la norma IEC 61131-3 no es fácil, por eso se permiten implementaciones parciales en varios aspectos. Esto hace referencia al número de lenguajes que soportan las herramientas de desarrollo disponibles, y al número de funciones y de bloques funcionales. Con ello se deja libertad al suministrador, pero el usuario debe tener cuidado durante el proceso de selección de la herramienta adecuada. Incluso una actualización del software puede dar lugar a un nivel muy alto de trabajo durante la implementación.

Muchos entornos de programación IEC actuales ofrecen aquello que se espera a nivel de interface de usuario: uso de ratón, menús desplegables, pantallas de programación gráfica, múltiples ventanas, ayuda en línea, verificación durante el diseño, etc. Debe hacerse notar que estos detalles no están especificados en la norma por lo que es una de las partes donde los proveedores pueden diferenciarse.

### **Conclusiones**

Las implicaciones técnicas de la norma IEC 61131-3 son altas, dejando bastante espacio para el crecimiento y la diferenciación. Esto la hace adecuada para entrar óptimamente en el próximo siglo.

La norma IEC 61131-3 tendrá un gran impacto en el mundo del control industrial y éste no se restringe al mercado convencional de los PLC's.

Ahora mismo, se pueden ver adoptada en aplicaciones para control de movimiento, sistemas distribuidos y sistemas de control basados en PC (SoftPLC), incluyendo los paquetes SCADA. Y las áreas de su utilización siguen creciendo.

El uso de IEC 61131-3 proporciona numerosos beneficios para usuarios/programadores. Los beneficios de la adopción de este estándar son varios, dependiendo de las áreas de aplicación: control de procesos, integrador de sistemas, educación, programación, mantenimiento, instalación, etc.

**Para la elección del lenguaje de programación se tomo en cuenta los siguientes aspectos:**

- **Los conocimientos del programador:** los conocimientos adquiridos durante la carrera de Ingeniería en sistemas fue Graficas de funciones secuencias y GRAFCET ya que la primero nos permite realizar un Eficaz técnica para describir el comportamiento secuencial de un proceso y un programa y el segundo nos permite realizar el grafico en TwidoSuite para tener la comunicación correspondiente con el PLC a través del protocolo MODBUS
- **El problema a tratar:** es proceso de mezclado de líquidos no es complejo, pero hay que tener en cuenta las transiciones.
- **El nivel de descripción del proceso:**
  - Bombeo de liquido
  - Llenado de tanques
  - Mezclado de liquido
  - Detección de nivel

- Sistema de llenado
  - Modulo de control (Eléctrico, Neumático e Informático)
- **La coordinación con otras persona:** el grupo que realiza la tesis en conjunto con el Ing. Marco Viteri

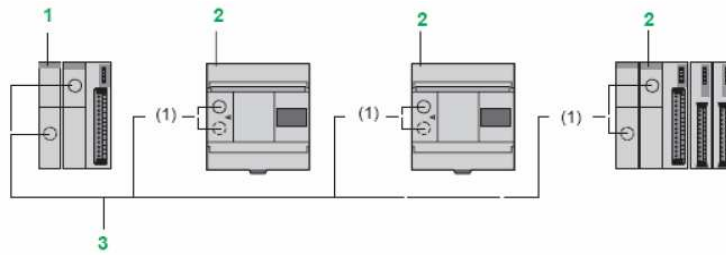
### 3.3. Control y monitoreo

#### 3.3.1 Protocolos de comunicación

Los controladores Twido disponen de un puerto serie, o de un segundo puerto opcional, que se utiliza para servicios en tiempo real o de administración de sistemas. Los servicios en tiempo real proporcionan funciones de distribución de datos para intercambiar datos con dispositivos de E/S, así como funciones de administración para comunicarse con dispositivos externos. Los servicios de administración de sistemas controlan y configuran el controlador por medio de Twido. Cada puerto serie se utiliza para cualquiera de estos servicios, pero sólo el puerto serie 1 es válido para comunicarse con Twido.

Para poder utilizar estos servicios, existen protocolos implícitos disponibles en cada controlador:

**Remote Link (Conexión remota):** permite realizar una comunicación entre autómatas Twido vía RS-485, utilizado para ver E/S a distancia (sin programa en las CPUs deportadas) o para red de Twidos con programa, con una longitud máxima de 200 m y hasta 8 equipos en una red (maestro + 7 esclavos).



**Figura. III.9. Ejemplo de una conexión remota**

- 1.- controlador de base compacta o modular denominada “maestro”
- 2.- Base compacta o modular Twido utilizadas en ampliación de E/S o “reflex” local
- 3.- cable RS 485, 3 hilos a partir del puerto de enlace serie o del 2. Puerto de enlace serie opcional

**ASCII:** permite comunicar el autómata, vía **RS-485** y **RS-232**, con un gran número de dispositivos: **impresoras** (para la impresión periódica de reports de producción), lectores de códigos de barras y módems.

- **El M.P.I. (Interface Multi Punto)**
- **El P.P.I. (Interface Punto por Punto)**
- **El Profibus-DP**

Existen además a nivel industrial otras redes tales como la Profibus-FMS, Industrial Ethernet, etc., pero no intervendrán en nuestro trabajo a pesar de que también puede ser conectado a cualquiera de ellas.

#### **Interface punto por punto (P.P.I)**

Esta interface permite la comunicación de nuestro dispositivo con otros tales como modems, scanners, impresoras, etc., situados a una cierta distancia del

PLC. En la parte frontal del módulo de la CPU posee fichas DB 9 o DB 25 para la comunicación serial vía RS 232 y RS 485.

La conexión Punto a Punto puede ser establecida económicamente y convenientemente por medio del procesador de comunicaciones CP 340. Hay varios protocolos disponibles por debajo de las tres interfaces de conexión:

- 20 mA (TTY)
- RS 232 C/V.24
- RS 422 / RS485

Los siguientes dispositivos pueden ser conectados:

- Controladores programables SIMATIC S7 y SIMATIC S5
- Impresoras
- Robots controladores
- Modems
- Scanners, lectores de códigos de barras, etc.

### **Interface Multipunto (M.P.I.)**

Todas las CPU's (312, 313, 314, 315 y 315 -2DP) lo incorporan desde fábrica. Con éste puerto se puede comunicar fácilmente a distancias reducidas sin requerir módulos adicionales, por ejemplo hacia equipos de M+V (manejo + visualización), unidades de programación y otros autómatas S7-300 o S7- 400 para probar programas o consultar valores de estado.

Se pueden enviar datos a 4 distintos aparatos al mismo tiempo y utilizando siempre el mismo puerto a una velocidad de 187,5 Kbits / seg o 187,5 Kbaudios.

Para pequeñas redes de comunicación o pequeños volúmenes de datos la CPU

ofrece el servicio de Datos Globales, que permite intercambiar cíclicamente cantidades de datos en paquetes de hasta 22 bytes como máximo.

Distancia máxima entre dos estaciones o nudos de red de MPI adyacentes: 50 metros (sin repetidores); 1100 metros (con dos repetidores); 9100 metros (con más de 10 repetidores en serie); por encima de los 500 Klm. (cable de fibra óptica, con módulos de conexión ópticas)

Capacidad de expansión: los componentes comprobadores de campo son usados para configurar la comunicación de interface multipunto: cables LAN, conectores LAN y repetidores RS485, desde el PROFIBUS y la línea de productos de entradas/salidas distribuidas.

Estos componentes permiten una óptima utilización de la configuración.

### **Profibus Dp**

Esta interface de comunicación es usada para gran capacidad de transmisión de datos, llamada Simatic Net o Sinec L2. El S7 300 mantiene una relación muy estrecha con él. Un módulo de comunicación permite conectarlo al Sinec L2 para comunicarse con otros autómatas Simatic y dispositivos de campo. La CPU 315 – 2DP ya la trae integrada. De éste modo, el autómata se adapta armoniosamente en arquitecturas descentralizadas que integran componentes de automatización y dispositivos de campo. El PLC puede desenvolverse aquí como maestro – esclavo, además también se dispone de los prácticos servicios de comunicación llamados Datos Globales.

Para entablar comunicación se utilizan cables LAN, conectores LAN, repetidores, etc.



Digamos entonces que es una red suplementaria que ofrece un gran rendimiento, arquitectura abierta o descentralizada y gran robustez o confiabilidad.

Existe además la gran ventaja del Manejo + Visualización (paneles de operador, llamados Coros) que permite tanto en ésta interface como en las otras de la búsqueda de errores a partir de cualquier dispositivo y así por ejemplo generar una base de datos con los errores (hora y tipo) que puedan existir.

### 3.3.2. protocolo Modbus

**Modbus** es un protocolo de comunicaciones situado en el nivel 7 del Modelo OSI, basado en la arquitectura maestro/esclavo o cliente/servidor, diseñado en 1979 por Modicon para su gama de controladores lógicos programables (PLCs). Convertido en un protocolo de comunicaciones estándar de facto en la industria es el que goza de mayor disponibilidad para la conexión de dispositivos electrónicos industriales. Las razones por las cuales el uso de Modbus es superior a otros protocolos de comunicaciones son:

1. Es público
2. Su implementación es fácil y requiere poco desarrollo
3. Maneja bloques de datos sin suponer restricciones

Modbus permite el control de una red de dispositivos, por ejemplo un sistema de medida de temperatura y humedad, y comunicar los resultados a un ordenador.

Modbus también se usa para la conexión de un ordenador de supervisión con una unidad remota (RTU) en sistemas de supervisión adquisición de datos

(SCADA). Existen versiones del protocolo Modbus para puerto serie y Ethernet (Modbus/TCP).

Existen dos variantes, con diferentes representaciones numéricas de los datos y detalles del protocolo ligeramente desiguales. Modbus RTU es una representación binaria compacta de los datos. Modbus ASCII es una representación legible del protocolo pero menos eficiente. Ambas implementaciones del protocolo son serie. El formato RTU finaliza la trama con un suma de control de redundancia cíclica (CRC), mientras que el formato ASCII utiliza una suma de control de redundancia longitudinal (LRC). La versión Modbus/TCP es muy semejante al formato RTU, pero estableciendo la transmisión mediante paquetes TCP/IP.

Modbus Plus (Modbus+ o MB+), es una versión extendida del protocolo que permanece propietaria de Modicon. Dada la naturaleza de la red precisa un coprocesador dedicado para el control de la misma. Con una velocidad de 1 Mbit/s en un par trenzado sus especificaciones son muy semejantes al estándar EIA/RS-485 aunque no guarda compatibilidad con este.

Cada dispositivo de la red Modbus posee una dirección única. Cualquier dispositivo puede enviar órdenes Modbus, aunque lo habitual es permitirlo sólo a un dispositivo maestro. Cada comando Modbus contiene la dirección del dispositivo destinatario de la orden. Todos los dispositivos reciben la trama pero sólo el destinatario la ejecuta (salvo un modo especial denominado "Broadcast"). Cada uno de los mensajes incluye información redundante que asegura su integridad en la recepción. Los comandos básicos Modbus permiten controlar un

dispositivo RTU para modificar el valor de alguno de sus registros o bien solicitar el contenido de dichos registros.

**Modbus:** comunicación Modbus Maestro/Esclavo por ambos puertos (RS485 o 232), permite conectar Twido a un gran número de equipos industriales, basado en mensajería aperiódica entre equipos.

Nota: El controlador compacto TWDLCAE40DRF proporciona un puerto de comunicación RJ45 Ethernet integrado. Las comunicaciones **Ethernet** implementan el protocolo **TCP/IP Modbus**.

Además, se pueden implementar más comunicaciones agregando módulos de comunicación, existen varios tipos en función del protocolo que se desee:

- Modulo de comunicación Maestro **ASI**.
- Modulo de comunicación Maestro **CANOpen**.
- Modulo de comunicación Maestro/Esclavo **Modbus**.



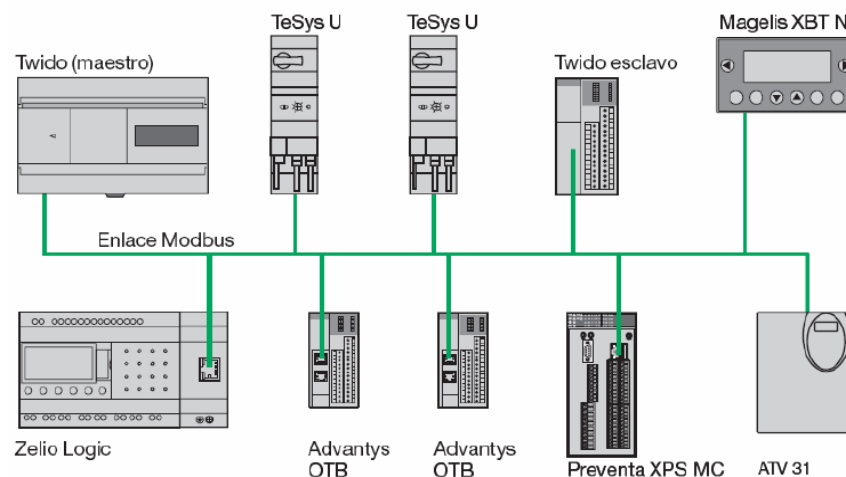
**Figura. III.10. Funciones del protocolo modbus**

#### **Bus de comunicación Modbus:**

El enlace serie Modbus permite responder a las arquitecturas maestro/esclavo (no obstante, es necesario comprobar que los servicios Modbus útiles para la aplicación se implanten en los equipos implicados).

El bus está constituido por una estación maestro y por estaciones esclavo. Sólo la estación maestro puede iniciar el intercambio (la comunicación directa entre estaciones esclavo no es posible). Existen dos mecanismos de intercambio:

- Pregunta/respuesta, las peticiones del maestro se dirigen a un esclavo determinado. El esclavo interrogado espera de vuelta la respuesta.
- Difusión, el maestro difunde un mensaje a todas las estaciones esclavo del bus. Éstas últimas ejecutan la orden sin emitir respuesta.



**Figura. III.11. Ejemplo de comunicación modbus**

**Modo maestro de Modbus:** el modo maestro de Modbus permite que el controlador pueda iniciar una transmisión de solicitudes Modbus, esperando una respuesta desde un esclavo Modbus.

**Modo esclavo Modbus:** el modo esclavo Modbus permite que el controlador pueda responder a las solicitudes de Modbus desde un maestro Modbus. Se trata

del modo de comunicación predeterminado si no existe ninguna comunicación configurada.

La comunicación Modbus Maestro/Esclavo se puede realizar por ambos puertos (RS485 o 232).

Este protocolo permite conectar Twido a un gran número de equipos industriales, como variadores de velocidad, arrancadores de motor, sensores...etc.

### Características

**Topología:** línea, estrella, árbol, red con segmentos

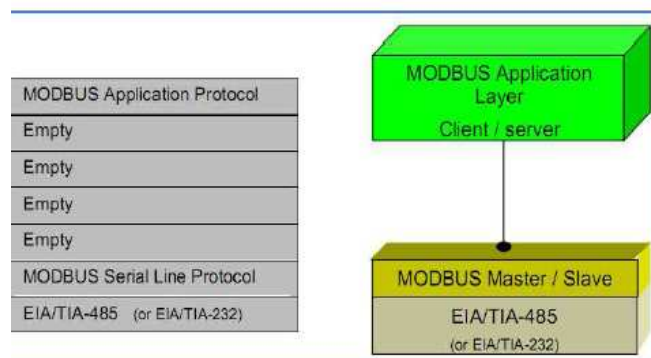
**Soporte:** par trenzado, coaxial, radio

**Máximo dispositivos:** 250 p/segm

**Rate transmisión bps:** 1.2 a 115.2K

**Distancia máxima en km:** 0.35

**Comunicación:** Master/Slave



**Figura. III.12. Protocolo Modbus**

### 3.4. Lenguajes de Programación

Existe una infinidad de lenguajes de programación que pueden ser empleados para la comunicación con diferentes dispositivos, manejo de base de datos, monitoreo, etc. En nuestro caso el dispositivo que queremos comunicar con el PC es un PLC, por lo cual en esta sección se detallarán los lenguajes de programación más habituales y las respectivas características de cada uno de ellos, para finalmente seleccionar el lenguaje de programación que mejor se adapte a nuestras necesidades para llevar a cabo el presente trabajo.

Como se detalló anteriormente el PLC a utilizar en nuestra investigación es de la marca Schneider de la familia Telemecanique, y la única forma de que este se pueda comunicar con otros dispositivos es a través del protocolo Modbus, por lo cual para la elección del lenguaje de programación tanto para la implementación del panel de control, así como para el manejo de base de datos, el cual es un objetivo en nuestro proyecto, se deberá tener en cuenta estos dos aspectos claves para dicha elección.

A continuación se analizan los principales lenguajes de programación y sus características:

#### 3.4.1. Basic / Visual Basic

El lenguaje BASIC fue creado en 1964 por los norteamericanos John Kemeny y Thomas Kurtz, en el Dartmouth College. Su intención era diseñar un lenguaje adecuado para principiantes. No tenía, por tanto, pretensiones de gran potencia, pero era fácil de aprender y permitía abordar pronto pequeños proyectos. BASIC son las iniciales de Beginner's All-purpose Symbolic Instruction Code.

El BASIC tuvo una gran expansión en los años 80, cuando se impuso como lenguaje estándar para los microordenadores domésticos de ocho bits. Se usó ampliamente en la enseñanza y también en la programación profesional. Sin embargo, era un lenguaje poco estructurado y recibió muchas críticas por promover malos hábitos de programación. Pero en lugar de desaparecer, fueron surgiendo versiones nuevas con características más avanzadas. Aún hoy, Visual Basic y su sucesor, Visual Basic .NET (aunque son variantes ya muy alejadas del original) son lenguajes de programación muy usados en entorno Windows. Podríamos definir programación en Visual Basic como el método que se utiliza para desarrollar la interfaz gráfica de usuario.

Con la programación en Visual Basic podrá desarrollar prácticamente cualquier programa que se imagine.

Además con la tecnología ActiveX, Visual Basic le proporciona una herramienta ilimitada para crear aplicaciones en la red.

Programación en Visual Basic permite construir de forma rápida aplicaciones de Windows. No es necesario conocer otros lenguajes de programación para poder saber programar en Visual Basic.

Programación en Visual Basic es una manera sencilla de crear aplicaciones, además el lenguaje le ofrece una cantidad de herramientas para simplificar esta labor, como proyectos, formularios, plantillas de objetos, controles personalizados, add-ins y un gestor de base de datos.

La versión 6.0 de programación en Visual Basic está especialmente diseñada para Internet. Usando los controles ejecutables ActiveX, funcionan como una

aplicación de Visual Basic independiente y es a través de los cuales que se accede a Internet Explorer.

Características fundamentales de Programación en Visual Basic 6.0:

- La posibilidad de acceder a datos de la base de datos, aplicaciones cliente/servidor y aplicaciones escalables como Microsoft SQL Server.
- Tecnología ActiveX
- Controles ActiveX
- Programación en Internet, servidores web, DHTML, etc..
- Creación de archivos .exe, lo que permite distribuir la aplicación con gran libertad.

### 3.4.2. Java

Java es un lenguaje muy moderno (se presentó en 1995) desarrollado por la empresa Sun Microsystems. Al parecer su historia es bastante curiosa, iniciándose como un lenguaje para el control de aparatos electrónicos (con el nombre de Oak). Cuando el proyecto estaba prácticamente abandonado, Bill Joy, cofundador de Sun, viendo Internet como el terreno idóneo para competir con Microsoft, retomó este lenguaje, que con los cambios pertinentes se transformó en Java.

Java es famoso por las applets, pequeñas aplicaciones gráficas que se insertan dentro de una página web. Uno de sus puntos fuertes es su capacidad multiplataforma, que permite que el mismo código pseudocompilado (llamado bytecode) se ejecute en cualquier sistema (cualquier sistema con soporte Java, claro).

Esta programación Java tiene muchas similitudes con el lenguaje C y C++, así que si se tiene conocimiento de este lenguaje, el aprendizaje de la programación Java



será de fácil comprensión por un programador que haya realizado programas en estos lenguajes.

Con la programación en Java, se pueden realizar distintos aplicativos, como son applets, que son aplicaciones especiales, que se ejecutan dentro de un navegador al ser cargada una página HTML en un servidor WEB, Por lo general los applets son programas pequeños y de propósitos específicos.

Otra de las utilidades de la programación en Java es el desarrollo de aplicaciones, que son programas que se ejecutan en forma independiente, es decir con la programación Java, se pueden realizar aplicaciones como un procesador de palabras, una hoja que sirva para cálculos, una aplicación gráfica, etc. en resumen cualquier tipo de aplicación se puede realizar con ella. Java permite la modularidad por lo que se pueden hacer rutinas individuales que sean usadas por más de una aplicación, por ejemplo tenemos una rutina de impresión que puede servir para el procesador de palabras, como para la hoja de cálculo.

La programación en Java, permite el desarrollo de aplicaciones bajo el esquema de Cliente Servidor, como de aplicaciones distribuidas, lo que lo hace capaz de conectar dos o más computadoras u ordenadores, ejecutando tareas simultáneamente, y de esta forma logra distribuir el trabajo a realizar.

### **3.4.3. Pascal / Delphi**

Pascal fue creado inicialmente por Niklaus Wirth como un lenguaje para la enseñanza de la programación al final de los años 60. Es un descendiente de Algol y a su vez influyó mucho en otros lenguajes posteriores, como Ada. Destaca por la claridad del código y por promover «buenos hábitos» de programación.

Pascal alcanzó una gran popularidad cuando la empresa Borland lanzó Turbo Pascal, un compilador para IBM-PC rápido y barato, que además introducía mejoras importantes al lenguaje. Su sucesor, también de gran éxito, es Delphi, una herramienta de programación visual basada en Object Pascal, una versión de Pascal con objetos.

La programación en Delphi es un entorno de desarrollo flexible y potente. Además es intérprete de un lenguaje llamado Object Pascal. La programación Delphi no es solo un intérprete, sino que además incluye otras herramientas para facilitar la escritura del código y el diseño de la aplicación.

Una de las mayores ventajas de la programación delphi es que es una programación orientada a objetos.

Se conoce como Objeto a los diferentes componentes visuales con que trabaja Delphi. Con lo que son objetos una ficha, un botón, una lista, etc.

La ventana principal de programación Delphi se denomina Paleta de Componentes.

Un programa en Delphi consta de las siguientes secciones:

- Un compilador: el cual crea el ejecutable
- Una librería: conocida como VCL, la cual es una librería de clases.
- El IDE: Que viene siendo el ambiente de desarrollo integrado (integrated development environment)

La programación Delphi almacena la aplicación con archivos de extensión .dpr, que define al proyecto en sí. Y archivos .dfm el que guarda el formulario creado en Delphi. .

#### 3.4.4. C#

C# es el nuevo lenguaje de propósito general orientado a objetos creado por Microsoft para su nueva plataforma .NET.

Microsoft.NET es el conjunto de nuevas tecnologías en las que Microsoft ha estado trabajando estos últimos años con el objetivo de mejorar tanto su sistema operativo como su modelo de componentes (COM) para obtener una plataforma con la que sea sencillo el desarrollo de software en forma de servicios web.

Los servicios web son un novedoso tipo de componentes software que se caracterizan a la hora de trabajar por su total independencia respecto a su ubicación física real, la plataforma sobre la que corre, el lenguaje de programación con el que hayan sido desarrollados o el modelo de componentes utilizado para ello.

El acceso a estos servicios se realiza en base a estándares de Internet, como son diferentes mecanismos del protocolo HTTP (GET y PUT) o el novedoso protocolo RPC conocido como SOAP (Simple Access Object Protocol), que no es más que una combinación de estándares como HTTP y XML para realizar llamadas a los miembros de estos servicios web. La idea detrás de SOAP consiste sencillamente en utilizar HTTP como medio de transporte para el envío de los mensajes de solicitud de ejecución de los miembros de servicios web remotos (lo que permite atravesar barreras tales como firewalls) y utilizar XML como lenguaje con el que escribir los cuerpos de estos mensajes.

Pero la plataforma .NET no son sólo los servicios web, sino que también ofrece numerosos servicios a las aplicaciones que para ella se escriban, como son un recolección de basura, independencia de la plataforma, total integración entre

lenguajes (por ejemplo, es posible escribir una clase en C# que derive de otra escrita en Visual Basic.NET que a su vez derive de otra escrita en Cobol).

Como se deduce del párrafo anterior, es posible programar la plataforma .NET en prácticamente cualquier lenguaje, pero Microsoft ha decidido sacar uno nuevo porque ha visto conveniente poder disponer de un lenguaje diseñado desde 0 con vistas a ser utilizado en .NET, un lenguaje que no cuente con elementos heredados de versiones anteriores e innecesarios en esta plataforma y que por tanto sea lo más sencillo posible para programarla aprovechando toda su potencia y versatilidad.

C# combina los mejores elementos de múltiples lenguajes de amplia difusión como C++, Java, Visual Basic o Delphi. De hecho, su creador Anders Heljsberg fue también el creador de muchos otros lenguajes y entornos como Turbo Pascal, Delphi o Visual J++. La idea principal detrás del lenguaje es combinar la potencia de lenguajes como C++ con la sencillez de lenguajes como Visual Basic, y que además la migración a este lenguaje por los programadores de C/C++/Java sea lo más inmediata posible.

Además de C#, Microsoft proporciona Visual Studio.NET, la nueva versión de su entorno de desarrollo adaptada a la plataforma .NET y que ofrece una interfaz común para trabajar de manera cómoda y visual con cualquiera de los lenguajes de la plataforma .NET (por defecto, C++, C#, Visual Basic.NET y JScript.NET, aunque pueden añadirse nuevos lenguajes mediante los plugins que proporcionen sus fabricantes).

### **3.4.5. LabVIEW**

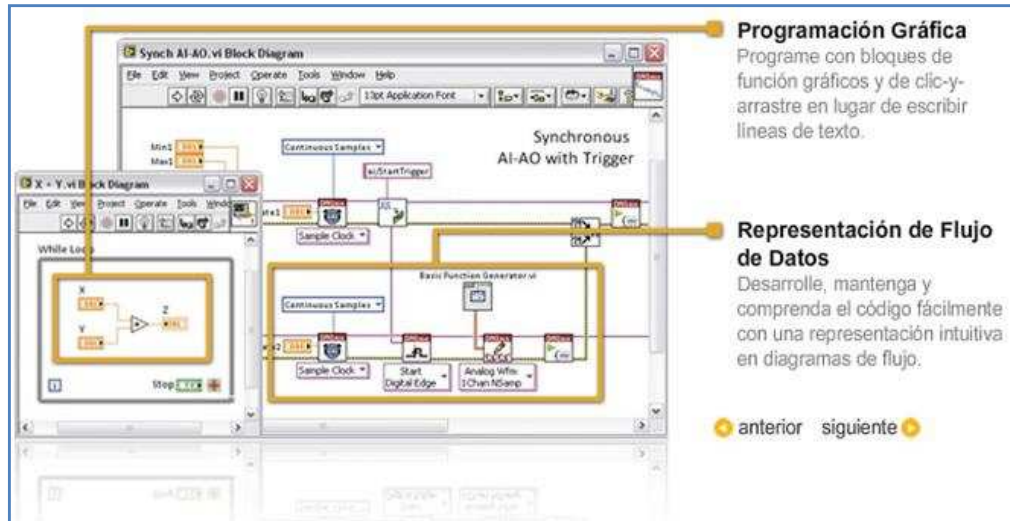
LabVIEW es un entorno de programación gráfica usado por miles de ingenieros e investigadores para desarrollar sistemas sofisticados de medida, pruebas y control

usando íconos gráficos e intuitivos y cables que parecen un diagrama de flujo. LabVIEW ofrece una integración incomparable con miles de dispositivos de hardware y brinda cientos de bibliotecas integradas para análisis avanzado y visualización de datos. La plataforma LabVIEW es escalable a través de múltiples objetivos y sistemas operativos, desde su introducción en 1986 se ha vuelto un líder en la industria.

Creado por la empresa National Instruments, <http://www.ni.com/>, LabVIEW nace para la conexión de instrumentos de medición a una computadora personal. En la actualidad LabVIEW, tiene una orientación hacia la instrumentación, el control y el procesamiento de señales.

La potencia y bajos precios de las PCs de la actualidad, unida a LabVIEW, resulta una herramienta ideal para los estudiantes y profesionales de las ciencias y la ingeniería.

Tradicionalmente los lenguajes de programación fueron por sentencias. Es decir, palabras que invocaban una rutina determinada. Un conjunto de rutinas hacen un programa. Por el contrario, programar con un lenguaje gráfico (G) como LabVIEW, tiene la ventaja de que las rutinas están representadas con íconos. Estos íconos, están agrupados en librerías temáticas que hacen fácil y rápida su ubicación. Por esta razón, la depuración de los programas es visual y por lo tanto rápida. [19]



**Figura. III.13. Apariencia de LabVIEW**

### **Ventajas:**

LabVIEW está altamente integrado con el hardware de medida, por lo que se puede configurar y usar rápidamente cualquier dispositivo de medida que se tenga. Con LabVIEW puede conectarse a miles de instrumentos para construir sistemas de medida completos, incluyendo desde cualquier tipo de instrumento autónomo hasta dispositivos de adquisición de datos, controladores de movimiento y sistemas de adquisición de imagen. Además LabVIEW trabaja con más de 1000 librerías de instrumentos de cientos de fabricantes, y muchos fabricantes de dispositivos de medida incluyen también herramientas de LabVIEW con sus productos.

### **Módulos y ToolKits adicionales**

Además LabVIEW cuenta con un sinnúmero de Toolkits y Módulos adicionales para potenciar su funcionalidad, como son conectividad con Sistemas de Gestión de Bases de Datos (DBMS) a través de el DataBase Connectivity Toolkit, el NI OPC Server, un servidor OPC que ofrece conectividad con protocolos industriales como Modbus, OPC, entre otros permitiendo la interconexión de cientos de decenas de marcas y cientos de dispositivos.

El monitoreo y supervisión de tareas en las plantas industriales repartidas en diferentes áreas o puntos, están caracterizadas por la dificultad en la toma de mediciones o por la continua necesidad del monitoreo de datos. Este hecho ha permitido el desarrollo de herramientas computacionales como los sistemas SCADA por sus siglas en inglés (Supervisory Control And Data Acquisition), los cuales no son más que cualquier software que permita el acceso a datos remotos de un proceso utilizando herramientas de comunicación.

No se trata de un sistema de control sino de una utilidad software de monitoreo o supervisión, que realiza la tarea de interfaz entre los niveles de control y los de gestión a un nivel superior.

Un sistema SCADA es principalmente una herramienta de supervisión y mando. Entre sus objetivos podemos resaltar: economía, accesibilidad, mantenimiento, ergonomía, gestión y conectividad.

Para la implementación de un SCADA, LabVIEW cuenta con el Módulo DSC (Datalogging and Supervisory Control), con el cual como su nombre lo dice nos permite realizar monitoreo y control, todo esto a través de la utilización de un servidor OPC con el de NI que se mencionó anteriormente, pero además podría implementarse un servidor OPC de otro fabricante.

### **3.5. Elección del lenguaje de programación**

Una vez que se han detallado los principales lenguajes de programación y sus principales características, se ha llegado a la conclusión de que el lenguaje más apto

para cumplir con nuestros objetivos y además para implementar un programa que realice el control es LabVIEW.

Se ha escogido LAVBIEW ya que este es un lenguaje de programación especialmente diseñado para la automatización de procesos industriales, que ofrece conectividad con miles de dispositivos y además una capacidad de procesamiento muy poderosa, lo que permite implementar un SCADA para realizar el monitoreo y control de de una forma sencilla y eficaz, a través del módulo DSC para este fin y el DataBase Connectivity Toolkit para la gestión con bases de datos, lo que lo convierte en el lenguaje de programación más apto para la implementación de aplicaciones de carácter industrial, ubicándolo muy por encima de los lenguajes de programación de propósito general como Visual Basic o sus similares.

Por lo anteriormente dicho, determinamos que el lenguaje de programación más apropiado para la programación del Panel de Control del sistema de almacenamiento inteligente es LabVIEW, ya que se garantiza la conectividad entre los diferentes dispositivos así como la parte de gestión de base de datos que se utilizarán en nuestra investigación.

### **3.6. Análisis de las metodologías existentes para sistemas mecatrónicos**

En esta sección se va a realizar el estudio de unas pocas metodologías que se encontraron para poder combinar y aplicar en nuestro sistema mecatrónico ya que no existe una metodología que sea apta por completo para nuestro proyecto.



### 3.6.1. Según La Universidad De Sevilla

Según esta Universidad en su Metodología para el desarrollo de sistemas mecatrónicos plantea la necesidad de cinco etapas como se muestra a continuación:

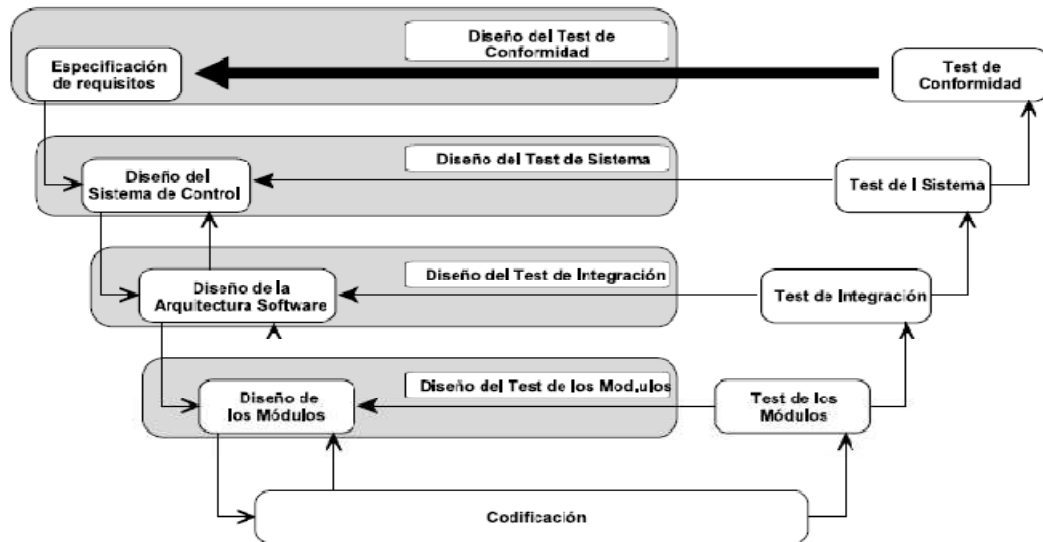


Figura. III.14. Etapas de la metodología Según La Universidad De Sevilla

#### 3.6.1.1. Fase de Especificación de los requisitos software.

##### Análisis de requisitos software.

En esta fase se debe especificar el contenido describiendo el funcionamiento del proceso y en sí del sistema de control para los diferentes escenarios en los que debe operar: funcionamiento normal en producción, situaciones de imperfección, paradas de mantenimiento, etc.

Se hayan utilizado o no de diseño, los estados de un automatismo representan en general escenarios o problemas diferentes, independientes, ya que no se pueden dar simultáneamente. Por ejemplo, las acciones de control durante el modo

normal de producción serán distintas a las que requiere un modo manual, o una situación de avería.

La descripción de requisitos debe por tanto realizarse para cada uno de estos estados de forma independiente, como si se tratara de problemas distintos.

### **Diseño de un test de conformidad.**

Es especialmente importante la definición de un plan de ensayos de conformidad que permitan validar el resultado final. El plan debe considerar aspectos como:

- El equipamiento que requieren las pruebas
- Quién y cuándo se ejecutan
- Modos de operación que van a ser testados (arranque, modo normal, parada, reset, mantenimiento, etc)
- Condiciones anormales que pueden lograrse previsiblemente en los ensayos
- Criterios para definir si las pruebas son superadas o no.

#### **3.6.1.2. Fase de Diseño del Sistema de Control**

En sistemas basados en PLCs, y en general, para todos los sistemas de automatización industrial, la fase de diseño del sistema comienza por la definición de la arquitectura hardware más adecuada para el control del proceso. En la siguiente lista se indican algunas:

- **DCS-Distributed Control System.** Se compone de múltiples bucles de control de variables analógicas, tales como temperatura, caudal en línea, pH, etc.  
  
Son muy habituales en procesos continuos, donde se requiere mantener en todo instante esas magnitudes dentro de un rango de trabajo. Los dispositivos que ejecutan los bucles de control tienen una escasa capacidad de programación. En realidad, tan sólo se configuran con un conjunto de parámetros de supervisión (niveles de alarma), y control (valores de consigna, constantes PID, valores límite, etc). En esta arquitectura, sensores, actuadores y controladores se conectan habitualmente mediante buses de comunicaciones.
- **Sistemas Centralizados.** En ellos todos los elementos del sistema de control, sensores y actuadores se conectan a un único equipo de control. Esta conexión puede ser cableada, con buses de campo, o ambas simultáneamente. Esta arquitectura es adecuada para procesos pequeños, con pocos sensores y actuadores, que se distribuyen sobre distancias cortas. Los PLCs son una solución ideal para esta arquitectura.
- **Sistemas Maestro-Eslavo.** Se trata de una variante del sistema centralizado, donde por necesidades de distancia se necesita un sistema anexo al principal (el esclavo) que ejecute las órdenes del maestro, pero que tenga la inteligencia suficiente para poder actuar por su cuenta si la comunicación con el maestro se pierde. La mayoría de los PLCs comerciales poseen módulos esclavos. Para el programa que ejecuta el maestro, el esclavo es transparente.

- **Sistemas multi-Centralizados.** Consisten en sistemas de estructura centralizada que trabajan de forma autónoma pero coordinada. **Esta coordinación puede implementarse mediante señales de campo, o transfiriendo datos a través de buses de comunicaciones.** Es una estructura muy adecuada en sistemas complejos, donde la distancia entre elementos sea grande, o cuando existan áreas diferentes de proceso que trabajan de forma autónoma. La existencia de varios sistemas semi-independientes aumenta la disponibilidad del proceso ante un fallo del sistema de control. Los PLCs están perfectamente diseñados para soportar esta arquitectura.
- **Sistemas Híbridos.** Son aquellos que integran control continuo y discreto, es decir bucles de control, con secuenciamiento de acciones y control lógico. Se instalan con frecuencia en procesos de fabricación por lotes (batch), como en la industria farmacéutica o alimentaria. La potencia de cálculo de los PLCs actuales les permite ejecutar bucles de control continuos, y por tanto, pueden integrar funciones de los DCS. Los autómatas programables son, por tanto, una tecnología más que adecuada para un sistema híbrido.

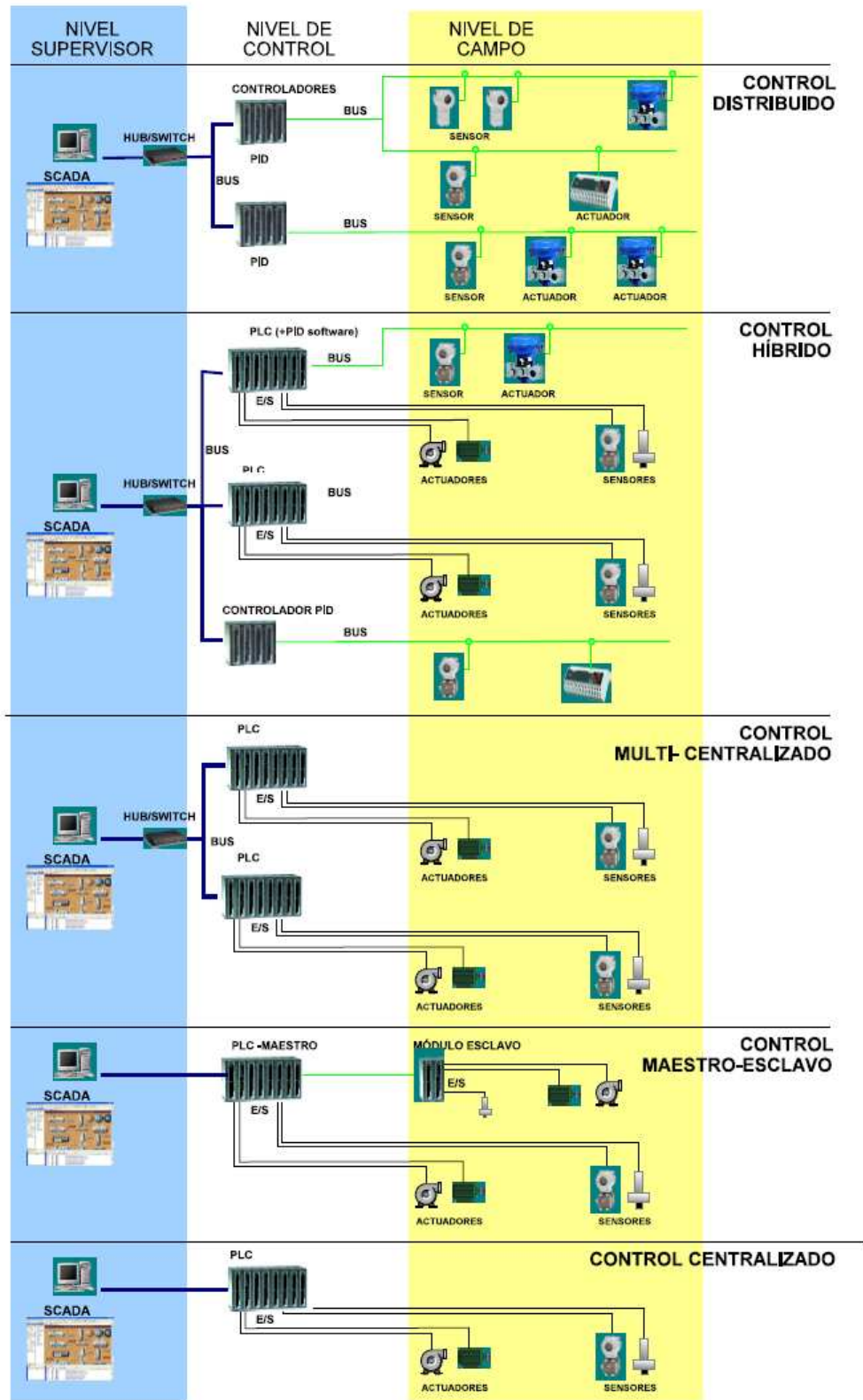


Figura. III.15. Estructuras de control

El estándar IEC 61131 es el marco conceptual de los sistemas de control basados en PLCs. Define sus características físicas, lógicas, y su conectividad.

También ofrece un modelo de arquitectura que abarca desde los sistemas centralizados a los multi-centralizados.

Los sistemas distribuidos se salen del ámbito de este texto, por lo que utilizaremos el IEC 61131 como guía en la descripción de la estructura hardware, y para el desarrollo del programa. Al diseñar la arquitectura hardware debemos decidir aspectos como:

- Cuántos PLCs son necesarios
- Funciones del proceso que ejecuta cada PLC
- Ampliaciones necesarias
- Cómo se interconectan
- Con qué buses,
- Fabricantes, modelos y tecnología.

Los conceptos con los que IEC 61131 describe la arquitectura hardware son tres:

1. Las **Configuraciones** o PLCs que constituyen el sistema de control , así como su interconexión.
2. Para cada configuración hay que definir qué **Recursos** incluye (CPUs o cualquier dispositivo capaz de ejecutar un programa IEC 61131).
3. Cada recurso contiene características fijas que deben declararse (capacidad de memoria, E/S integradas, temporizadores, modos de ejecución, etc), así como capacidades ampliables que deben diseñarse de manera específica (E/S en slots de ampliación, E/S descentralizadas en buses de comunicaciones, módulos de función especial, etc).

## **Especificación y diseño del test de Sistema**

En esta fase también es especialmente importante definir pruebas que verifiquen la interconexión hardware y la integración software/hardware, sobre todo cuando se utilizan dispositivos de diferentes fabricantes. Los test se realizaran a varios niveles, como por ejemplo los siguientes:

- Comprobación del cableado con los elementos de proceso (activando manualmente las salidas del PLC, o desarrollando un programa al efecto).
  - Verificación de la configuración de todos los dispositivos
  - Analizando la correcta instalación de ampliaciones y periféricos a los PLCs.
  - Comprobando las comunicaciones entre dispositivos (usando el sistema de auto diagnósticos, herramientas ofrecidas por el fabricante, o programas desarrollados para ello).
  - Verificando la integración del software dentro de las distintas configuraciones y la coordinación de los módulos software entre distintos PLCs.

### **3.6.1.3. Fase de Diseño de la Arquitectura Software**

Se describe la organización del software, cómo se divide y qué módulos se desarrollan, y cómo se integran formando diferentes programas o aplicaciones. La arquitectura software depende en gran medida de esa tecnología de programación escogida; en nuestro caso, el IEC 61131-3. De las ventajas que se logran con una correcta estructuración, significamos las siguientes:

- Una mejor visión no sólo para los programadores, sino también para los operadores, y el personal de instalación y mantenimiento.
- Una buena base de comunicación dentro del equipo de desarrollo.
- Generación de código más comprensible, reutilizable, verificable y fácil de mantener.
- Una documentación mejor organizada.

La norma IEC 61131-3 contiene conceptos y lenguajes de programación muy potentes a la hora de estructurar el software.

El IEC 61131 posee una gran capacidad de organización que se basa en la estructuración y la descomposición de un programa.

### **Estructuración de un programa**

Según la norma 61131-3, un programa se define como “una interconexión lógica de unidades de programa (POUs) a la que se le asocia un modo de ejecución (tarea)”.

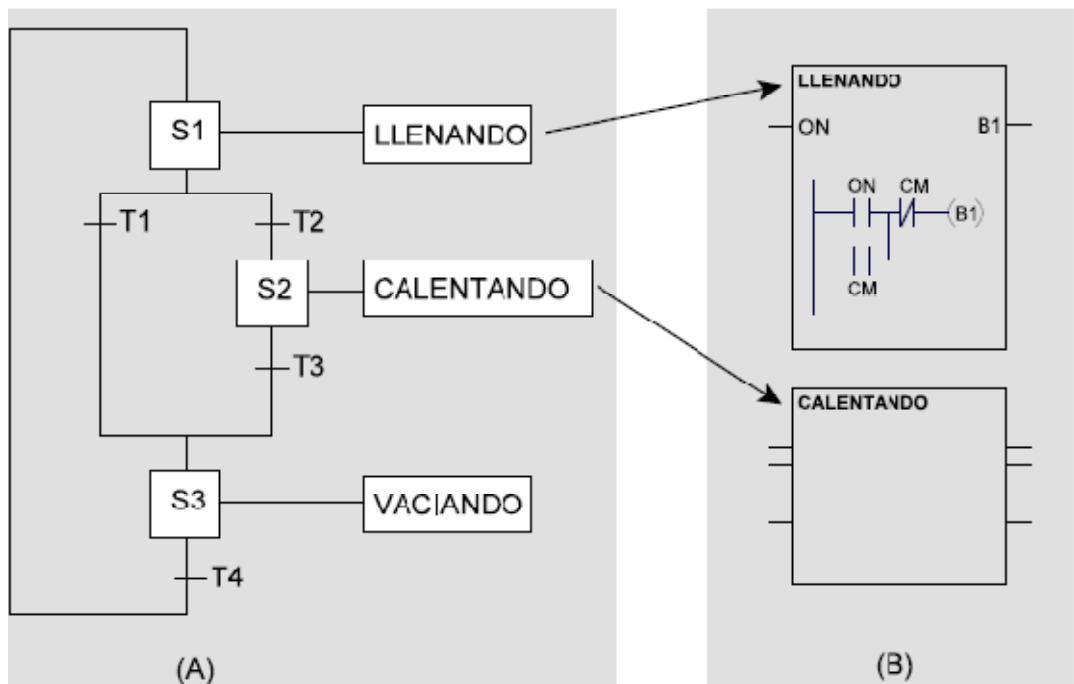
### **Descomposición de un programa**

IEC 61131-3 define 5 lenguajes de programación distintos. Uno de ellos, el SFC (Sequential Function Chart), se diseña para descomponer un programa secuencial y concurrente en un conjunto de operaciones más sencillas. El SFC es un lenguaje gráfico, heredero del GRAFCET con el que puede describirse de forma relativamente sencilla secuencias complejas de operaciones, secuencias



independientes que se ejecutan de forma paralela, o incluso pueden sincronizarse secuencias paralelas.

El SFC consta de etapas unidas mediante transiciones. Las etapas reflejan estados particulares del sistema. Cada etapa tiene asociadas acciones que ejecutan cierta función de control. Estas acciones pueden programarse en cualquiera de los lenguajes IEC 61131-3. Las transiciones poseen condiciones que controlan la evolución de unas etapas a otras. Las condiciones de transición se programan normalmente empleando lenguaje de contactos o de funciones.



**Figura. III.16. Lenguaje de programación CFS**

### Descripción funcional

Una vez definida la arquitectura del programa, debe realizarse una lista de todos los bloques de acción y unidades de programa. Para cada uno de ellos se redactará una descripción funcional, que en términos genéricos, indique la función que ejecuta en

alguno o varios de los siguientes ámbitos: sobre los elementos del proceso, en el estado del automatismo, o en la coordinación del programa.

### **Especificación y diseño de un test de integración**

El objetivo principal de las pruebas en este punto es determinar que el funcionamiento conjunto de las unidades de programa se corresponde con el deseado. En particular, los test deberán diseñarse para verificar la correcta elección y funcionamiento de las señales que interconectan los bloques entre sí.

Para cada tarea que podría ejecutarse, se debería seguir los siguientes pasos:

- Testar uno a uno la coordinación entre las POUs de los bloques de acción y el SFC principal.
- Verificar que las POUs de acción funcionan correctamente cuando se detiene y se reinicia su ejecución como consecuencia del cambio de etapa del SFC.
- Testar una a una la interconexión entre el Bloque de la secuencia principal y las unidades de programa de los actuadores.
- Testar el funcionamiento completo

#### **3.1.6.4. Fase de Diseño de los Módulos o Unidades de Programa**

El diseño de una unidad de programa comienza por la definición de sus interfaces. Para ello es necesario tener en cuenta que, en general podemos encontrar tres tipos de señales:

- Señales de campo, que provienen directamente de los sensores o actuadores.

- Señales de mando, de las que se reciben órdenes de los operadores (cuadros de maniobra o SCADAs).
- Señales de coordinación entre diferentes POU's (indicaciones de fin de ejecución de un trabajo, errores, etc).
- En la arquitectura software se puede observar fácilmente el número de unidades de programa que existe.
- Bloques dedicados al control directo de elementos de proceso.
- Bloques que gestionan el comportamiento del proceso y del automatismo.

Al definir una señal deben establecerse tres puntos:

- Tipo de variable (BOOLEAN, BYTE, INTEGER, FLOAT, TIME, etc).
- Mecanismo de señalización. Para señales de tipo binario, especialmente las de coordinación, debe decidirse entre tres formas de señalización:
  1. Activas por nivel. La información se indica por el valor de la variable. P.e un detector de fuego señala con un 0 la presencia de llamas. Hasta que se extinga completamente el detector seguirá dando ese valor.
  2. Activas por flanco. Un cambio en el valor de la señal (de 0 a 1 en flancos positivos, o de 1 a 0 en flancos negativos) indica la ocurrencia de un suceso que debe ser conocida. P.e en estado normal un relé térmico ofrece un valor de 1. Si se produce el sobrecalentamiento de un motor, el relé se abre y se lee como un 0. El fallo térmico se señala con un flanco negativo. Para que una señal de flanco pueda avisar de un nuevo evento, ésta debe volver a su estado de reposo, el previo al flanco.

3. Pulso. El evento se avisa con un pulso. Para este tipo de señales, existe un estado de reposo (0 ó 1). Cuando se desea señalar un evento, la línea cambia de estado, aunque ese cambio es temporal, ya que retorna al estado de reposo transcurrido un cierto tiempo (pulso).
- Terminada la interfaz, la función que debe ejecutar la POU debe definirse con claridad traduciendo la descripción funcional, que se hizo en la fase de diseño de la arquitectura (función general) a otra “más tecnológica”, empleando términos que incluyan las señales de su interfaz. Por ejemplo, en lugar de decir “abre la válvula de vaciado”, diremos “activa (pon a 1) la salida AbrirValvula”.

### **Diseño de un test para los módulos**

Para cada una de las unidades de programa desarrolladas se diseñará un test que permita validar la funcionalidad de las POUs comenzando por las más simples, en el sentido de que no emplean dentro de su código otras POUs que deban ser testadas. Sí podrían contener POUs de sistema, librerías estándar, o librerías del propio equipo de desarrollo que ya han sido suficientemente probadas y utilizadas.

Los test más básicos son de tipo estático. El comportamiento del programa se verifica forzando el cambio en alguna variable de entrada de su interfaz. Este procedimiento puede ejecutarse desde el entorno de programación empleando un simulador del autómata sobre el que forzamos su E/S de

forma manual. Para efectuar todo este proceso sobre un PLC real, hay que asegurarse de que éste no esté operando sobre el proceso.

Más complejos son los test dinámicos, donde se generan patrones de prueba a modo de secuencia temporal de cambios que afectan a distintas entradas de forma simultánea.

El resultado del test se obtiene del análisis de la evolución temporal de las salidas, y de los cambios de estado de la unidad de programa. Este tipo de ensayos pueden ser muy complejos, por lo que en muchas ocasiones, es necesario ejecutarlos mediante simulaciones, o desarrollando una unidad de programa específica que genere esos patrones de prueba.

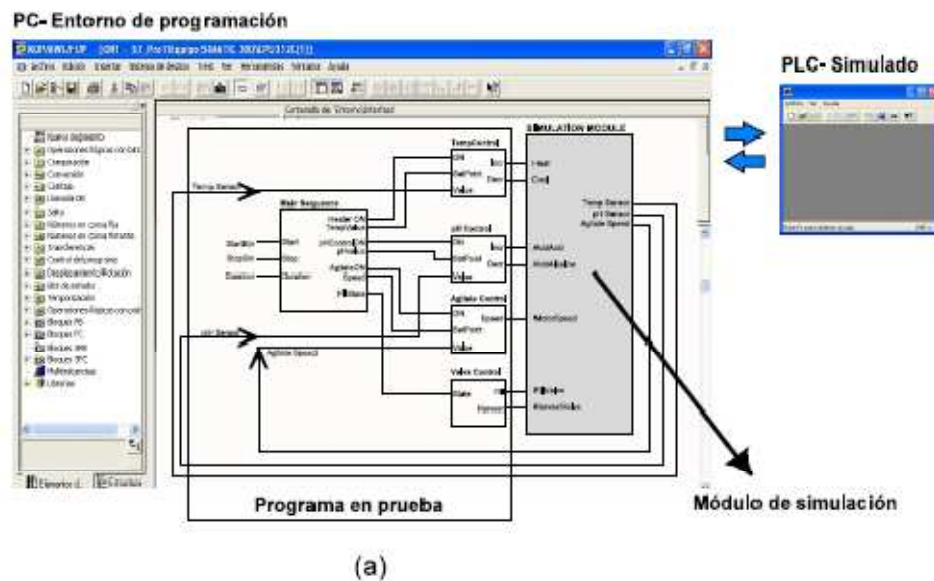


Figura. III.17. PC – entorno de programación 1 (a)

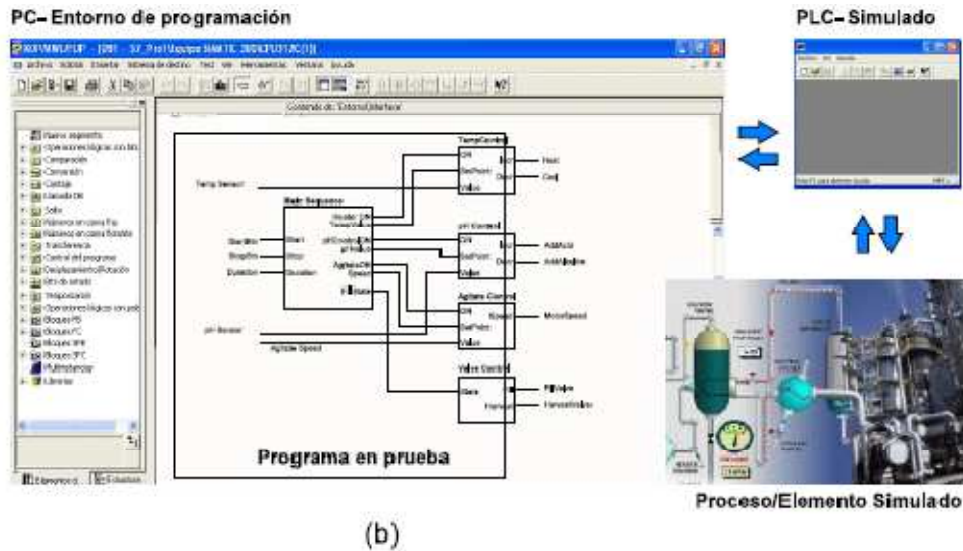


Figura. III.18. PC – entorno de programación 1 (b)

### 3.6.1.5. Fase de Codificación (programación)

En esta fase se procede a desarrollar el código en alguno de los lenguajes estándar o se modifican programas ya existentes. La mejor opción en este caso es la reutilización de código, es decir, emplear programas ya utilizados en otros proyectos, y por tanto ya testados. Esto ahorra tiempo y costes de desarrollo y validación. En segundo lugar, la modificación de programas ya existentes es una opción válida si se trata tan sólo de pequeñas modificaciones, añadir una o dos señales, o modificar una o dos de las existentes, etc. Si los cambios son amplios, es recomendable desarrollar un nuevo programa.

El estilo de programación es también importante. El código debe ser legible. Cualquier miembro del equipo debe ser capaz de comprenderlo, partiendo de las especificaciones del módulo, simplemente “leyendo” el programa. Para lograrlo deben seguirse al menos las siguientes recomendaciones:

- Declarar las variables internas con nombres que reflejen el significado de la información que contienen. Por ejemplo en un lugar que utiliza “fallo Térmico” o “sobrecalentamiento”.
- En la mayoría de los casos, es mejor emplear “lógica positiva” en las variables binarias. Siguiendo el ejemplo anterior, esto significa que cuando se produce un sobrecalentamiento, la variable “sobrecalentamiento” está en 1.
- Deben comentarse todas, o la mayoría de las líneas de programa, explicando lo que se pretende hacer empleando los nombres de variables internas o de la interfaz, nunca en términos del elementos de proceso.
- No emplear NUNCA variables globales. De otro modo el código podría no ser reutilizable.
- El lenguaje de programación debe escogerse en relación con la funcionalidad del programa y los conocimientos del programador. Por ejemplo:
  - LADDER.- Es el más empleado. Muy adecuado para programas con pocas señales y pocas variables internas.
  - FUNCIONES.- Muy adecuado para estructurar cualquier unidad de programa compuesto de otras POU's más pequeñas que se ejecutan de forma concurrente (simultánea y coordinada). Por ejemplo el programa principal de muchas arquitecturas (PLCOPEN).
  - SFC.- Adecuado para descomponer programas con complejas secuencias de operación. Por ejemplo para desarrollar los modos GEMMMA del automatismo.
  - INSTRUCCIONES - Adecuado para desarrollar código muy compacto y rápido de ejecución, o donde conocer el tiempo de

ejecución de las instrucciones puede ser importante. Por ejemplo en la gestión de alarmas. Es un lenguaje similar al ensamblador que requiere conocimientos muy especializados. No está muy extendido.

- **TEXTO ESTRUCTURADO.** Similar al PASCAL. Es adecuado para programar algoritmos matemáticos de todo tipo. Simples, como un comparador con histéresis, o complejos como un algoritmo de regulación PID.

**3.6.2. Según José Guadalupe Castro Lugo, Juan José Padilla Ybarra, Eduardo Romero A.**

## **METODOLOGÍA PARA REALIZAR UNA AUTOMATIZACIÓN UTILIZANDO PLC**

### **3.6.2.1. Descripción Del Sistema**

En esta actividad se debe tomar en cuenta la siguiente información:

- Procedimiento que se debe seguir en la operación (arranque, paro).
- Dispositivos que intervienen en el proceso (sensores, transductores, motores, variadores, etc.)
- Variables a medir, variables a controlar, variables a monitorear, rangos de operación
- Función de los dispositivos, entradas y salidas.



Esta actividad se lleva a cabo mediante entrevistas con los operadores y encargados de mantenimiento del proceso, visitas de campo y la experiencia del integrador.

### **3.6.2.2. Diagrama De Flujo**

Son los pasos que se va seguir para realizar un proceso específico. Dicho diagrama es útil para determinar cómo funciona realmente el proceso. El diagrama se utiliza en gran parte de las fases del proceso de mejora continua, sobretodo en definición de proyectos, diagnóstico, diseño e implantación de soluciones, mantenimiento de las mejoras, traslado de materiales, pasos para ventas y procedimientos del proceso.

### **3.6.2.3. Descripción De Los Equipos Del Sistema**

Aquí se agrupan todos los dispositivos que intervienen en el proceso, se describe bien su función e identifica las entradas y salidas del sistema. Esto ayuda a conocer con mayor detalle el sistema y las funciones para los cuales fueron diseñados los dispositivos. Además sirve para conocer más a detalle el proceso y entenderlo mejor; es decir, tener una amplia visión para la siguiente etapa.

### **3.6.2.4. Requerimientos Del Cliente**

Estos se obtienen, de las entrevistas realizadas con los operadores y jefes de mantenimiento, los cuales indican características de operación, características de

los equipos, rango de operación y en algunos casos el rango del costo de los equipos a utilizar.

### **3.6.2.5. Selección Del Autómata Programable**

Para llevar a cabo la selección del autómata se deben realizar dos evaluaciones como se describe a continuación:

1. Para seleccionar el tipo de autómata
2. Otro para seleccionar la marca.

Todo esto es debido a las diferentes opciones que brinda el mercado actualmente.

#### **Matriz de decisiones para la selección de PLC**

Para realizar la matriz de selección se deben seguir los siguientes pasos:

- Elaborar una lista de características de selección
- Ordenar la lista de características
- Asignación de ponderación relativa a cada característica de la selección
- Establecer parámetros de rendimiento o calificación de utilidad para cada una de las características
- Calcular los valores de utilidad relativa de los diseños alternativos además de comparar los valores de utilidad relativa.

#### **Matriz de decisiones para la selección de la marca del PLC**

Para generar esta matriz se debe realizar un procedimiento similar al del paso anterior solamente que aquí se compararan por lo menos cuatro marcas diferentes de autómatas programables seleccionado en la etapa anterior.

### 3.6.2.6. Programación del plc.

Existen dos formas de programación para el PLC:

1. El método heurístico o informal (función memoria).
2. El método formal (redes de Petri o GRAFCET).

Primero se debe de realizar el diagrama GRAFCET, el cual consiste en un diagrama gráfico de etapas y transiciones, por medio del cual se puede llevar a cabo con facilidad la programación del PLC elegido de acuerdo con el software del mismo.

#### **GRAFCET del proceso**

Los pasos esenciales que se debe realizar son:

- Se debe caracterizar el funcionamiento del automatismo con total independencia de los componentes con los que vaya a ser construido. Esto equivale a centrar el interés no tanto en la estructura física o en la tecnología empleada para implementar el automatismo, sino en la función que debe realizar.
- El elemento fundamental de un proceso es la operación (denominada etapa en el lenguaje de GRAFCET), entendiendo como tal una acción realizada por el automatismo.
- Se debe dividir el proceso en macro etapas y éstas en etapas más elementales, hasta conseguir que las acciones a realizar en cada una de ellas dependa sólo de relaciones combinatorias entre entradas y salidas.
- Establecer un gráfico de evolución que indique la secuencia de las operaciones (secuencia de etapas) y las condiciones lógicas para pasar de

una etapa a otra. (denominada condiciones de transición en el lenguaje de GRAFCET) Como resultado de este paso se obtienen las ecuaciones lógicas y queda resuelta la parte secuencial del automatismo.

- Establecer para cada operación elemental (etapa) las relaciones lógicas entre entradas y salidas, utilizando eventualmente otras variables internas combinacionales.
- Finalmente implementar el sistema utilizando el programa de las relaciones lógicas de los dos puntos anteriores.

### **Descripción de entradas y salidas del proceso**

De acuerdo con el diagrama del GRAFCET se describen las entradas y salidas que intervienen en el programa para tener una mejor visualización del mismo. Esto ayuda en mayor parte para la siguiente etapa que se refiere a la programación del PLC.

### **Programa del PLC**

Este se realiza en el software del autómatas elegido, usando el diagrama de GRAFCET realizado anteriormente. Cada línea de programa se puede ir leyendo directamente del GRAFCET.

## **3.7. Descripción de la solución planteada en la investigación**

Una vez analizada cada una de las metodologías antes mencionadas se procede a escoger, combinar e incrementar las fases necesarias para proponer la metodología

más aplicable para sistemas mecatrónicos pero especialmente para nuestro sistema de mezclado de líquidos.

Como en todo sistema mecatrónico, dicho sistema no podría ser conseguida si no fuese alimentado con información proveniente del mundo exterior, ya sea por sensores, electroválvulas, bombas eléctricas o neumáticas, etc. Lo cual ha permitido que la robótica, demótica, automatización industrial hayan avanzado de forma vertiginosa en las últimas décadas, todo aquello no sería posible sin la utilización de estos elementos de vital importancia en este tipo de sistemas.

Por tal razón, la forma en la que el presente sistema de mezclado de líquidos interactuará con el mundo exterior a través de sensores, los mismos que tomarán las señales externas percibidas y emitirlas al PLC y por ende al computador, para que estos de acuerdo a la lógica de funcionamiento con la que se les programe actúen de una u otra forma, haciendo que el módulo de mezclado de líquidos, así como los módulos con los que este interactúa, como son el módulo de la envasadora y la banda transportadora, interactúen entre sí consiguiendo finalmente el mezclado de líquidos de manera automático todo esto en base a las señales recibidas.

### **3.8. Propuesta metodológica para la implementación de proyectos mecatrónicos.**

Para poder realizar nuestro sistema mecatrónico nos hemos encontrado en un problema ya que durante la investigación que hemos realizado no hemos encontrado las metodologías necesarias para la implementación de sistemas mecatrónicos, es así que las metodologías existentes no están enfocadas en su totalidad a todos los

aspectos que presenta nuestro sistema de mezclado de líquidos además la metodología que nosotros propongamos debe contener el diseño e implementación mecánica, eléctrica, neumática, etc. Por tal motivo nosotros hemos hecho la combinación de estas dos metodologías para desarrollo de sistemas mecatrónicos y además con otra metodología ágil que es el XP que para ello no nos hizo falta describirlo ya que durante la carrera ya lo hemos estudiado.

Luego de haber dicho todo esto nos vimos en la necesidad de crear o modificar una metodología para la implementación de nuestro proyecto, que cubra todas las etapas de proyectos mecatrónicos, pero que además me permitirá incorporar también el desarrollo de software, por lo cual luego de estudiar el problema, llegamos a la conclusión de adaptar todas las metodología antes estudiadas para que abarque las partes mecatrónicos, con lo cual garantizaríamos y lograremos el correcto desarrollo de nuestro proyecto de tesis, pues se tratan de metodologías ya probadas, y también se tendría en cuenta las demás etapas que nos interesa o que incrementemos.

La metodología que adaptemos deberá estar acorde a las necesidades que tiene un proyecto mecatrónico las mismas que son:

- Implementación rápida y sencilla
- Pruebas continuas durante el desarrollo
- Trabajo en equipos
- Entrega en etapas de la solución de acuerdo a la priorización de necesidades.
- Susceptible a cambios durante el desarrollo.
- Escalable

- Desarrollo incremental
- Permitir realizar el diseño eléctrico, mecánico, neumático e informático.

Luego de estudiar las características anteriores que debe nuestra metodología, y analizar las metodologías existentes para el desarrollo de proyectos mecatrónicos y también para desarrollo de software, llegamos a la conclusión que las metodologías que más se adaptan a nuestras necesidades es:

- La metodología que propone de la Universidad de Sevilla que contiene cinco etapas pero en si se realizó el estudio necesario y se seleccionó los que más se adaptan a nuestro sistema mecatrónico.
- La de desarrollo de software que se escogió es la metodología ágil XP (Programación Extrema), debido a que es una metodología ágil, se prioriza las entregas que se harán del proyecto, es sencilla de seguir, se realiza desarrollo por prueba y error, lo cual se adapta más a nuestro proyecto, ya que al no ser especialistas en electrónica, ni diseño mecánico, es una característica que nos interesa.

Por tal razón la metodología que proponemos es la siguiente:

## **1. Planificación del Proyecto**

- 1.1. Descripción del Sistema
- 1.2. Especificación de Requerimientos
- 1.3. Planificación Inicial
- 1.4. Plan de Iteraciones

## **2. Diseño**

- 2.1. Diseño Mecatrónico
  - 2.1.1. Diseño Mecánico

2.1.2. Diseño Eléctrico

2.1.3. Graficet

2.2. Diseño Software

2.2.1. Diagramas de Secuencia

2.2.2. Diagramas de Actividades

2.2.3. Diagramas de Clases

2.2.4. Diagrama de Componentes

2.2.5. Diagrama de Despliegue

2.2.6. Diccionario de Datos

2.2.7. Modelo Lógico de la Base de Datos

### **3. Implementación**

3.1. Implementación Mecatrónica

3.1.1. Sistema Mecánico

3.1.2. Sistema Neumático

3.1.3. Sistema Eléctrico

3.1.4. Programación del PLC

3.1.5. Control y Monitoreo del PLC

3.2. Implementación Software

3.2.1. Modelo Físico de la Base de Datos

3.2.2. Prototipos de Interfaces de Usuario

### **4. Pruebas del Sistema**

4.1. Pruebas Hardware

4.2. Pruebas Software

### **5. Implantación**



### **5.1. Manual de Usuario**

La metodología aquí planteada posee las mismas etapas que XP con la combinación de la metodología investigada que aplica la Universidad de Sevilla para el desarrollo de un proyecto mecatrónico, estas combinaciones se realizan en algunas de las fases la misma que se detallan a continuación:

**Segunda Etapa (Diseño):** En esta etapa nosotros hemos dividido en dos partes: Diseño Mecatrónico y Diseño Software.

En el diseño mecatrónico se realizara el diseño de la estructura mecánica, ubicación de los tanques y los motores neumáticos como el electrónico, diseño eléctrico y diseño del programa para el control del PLC a través del lenguaje Grafcet.

En cambio el diseño software se empleará en los principales diagramas de modelado UML, así como la parte del diseño de base de datos.

**Tercera Etapa (Implementación):** Al igual que la etapa anterior se la ha dividido en dos partes: Implementación Mecatrónica e implementación Software.

En implementación Mecatrónico se realizará la respectiva construcción de la estructura mecánica, ubicación de sensores, válvulas y cilindros neumáticos, cableado de las borneras al PLC y etc.

La programación del PLC se lo hará en el respectivo lenguaje de programación con el que venga el PLC, de la misma manera se realizará la implementación del programa para el control del PLC si la situación lo amerita.

**Cuarta Etapa (Pruebas):** En esta etapa se realizara las pruebas hardware y software para el correcto funcionamiento del proyecto de tesis, hay que recalcar que siempre se debe estar realizando pruebas durante la implementación, por lo cual

durante la implementación del proyecto siempre se estará realizando cambios constantes en la Tercera y Cuarta etapa, y con esto se cumple la parte incremental.

**Quinta Etapa (Implantación):** En esta etapa se realizará el desarrollo del manual de configuración, manual de usuario y la puesta en marcha del sistema es decir el lanzamiento del sistema ya terminado y en funcionamiento listo para ser utilizado para otros módulos.

# **CAPITULO IV DESARROLLO DE UN SISTEMA MECATRÓNICO PARA EL MEZCLADO DE LÍQUIDOS**

## **4.1. Introducción**

En este capítulo se va detallar todo lo referente a nuestro proyecto en sí del proceso de planificación, diseño, desarrollo e implementación de la tesis “Implementación de un Sistema de Mezclado de Líquidos (SIMEL), el mismo que fue realizado siguiendo la metodología planteada que se propone en el capítulo 3, lo cual ha permitido que el proyecto se lo realice de forma rápida, sencilla, sin complicaciones y con resultados excelentes.

Para poder realizar este proyecto de tesis hemos asignado mucho tiempo al diseño e implementación mecánica, eléctrica, neumática e informática para lograr el objetivo propuesto que es implementar un sistema mecatrónico para el mezclado de líquidos,

Además en este proyecto se va a realizar la implementación en LabView que trabajará como un todo para de esta manera conseguir la implementación de este sistema con la utilización de Base de Datos implementada en el propio Labview la misma que almacenará datos tales como el nivel máximo y mínimo del tanque 1, del tanque 2, el tanque 3 y por último el tanque 4 el más importante ya que es el resultado del líquido mezclado.

#### **4.1.1. Componentes Del Modulo De La Mezcladora Automática**

Para el diseño mecatrónico se utilizará los siguientes materiales listados a continuación:

- Estructura de aluminio
- Sensores
  - Sonda
  - Flotador Camsco PD-76AB
- 1 Bomba eléctrica Flojet modelo: P46XD094
- PLC Telemecanique TWDLCAE40DRF
- 1 Válvulas 4V120-06
- 1 Valvula AV210
- Sócalos 3UG05-02-2AF00-A7B10000052296
- 2 Relé controlador de nivel Siemens RXM4AB1BD
- 1 Relé controlador de nivel Siemens RXM4AB2BD
- 4 Envases de Plástico
- Concentrador de señales: ABE7 H16R31

- Pretul
- Compresor.
- Mezclador
- Tubería de plástico
- Sistema de Control.
- Breaker para el control del voltaje de 110 voltios

#### 4.1.2. Descripción de los dispositivos utilizados para el ensamblaje del mezclador

##### 4.1.2.1. Sensor sonda

El sensor sonda nos permite establecer los niveles máximo, mínimo y referencia del líquido.



**Figura. IV.1. Sensor sonda**

#### 4.1.2.2. Sensor Flotador Camsco PD-76AB



**Figura. IV.2. Sensor Flotador**

##### Aplicación

El regulador llano líquido de PD-76AB es un modelo nuevo del dispositivo de control de nivel de líquido de la bola de flotador con características de la reacción sensible, del control exacto, de la novedad y de práctico. No. de la patente: ZL01 2 646008, en el principio de la fuerza flotante de las bolas de flotador dobles, control encendido-apagado del contacto del doble de la caja sellada cambia en zona de seguridad y control auto del sistema de fuente de nivel líquido, es el primer producto bien escogido de las fábricas, hoteles, apartamentos, edificios, altas construcciones de viviendas, transporte, sistema automático para controlar la posición líquida

##### principio 2.Construction

El modelo L placa muerta para el regulador llano líquido es fijo en la piscina líquida paralelamente y el alambre en caja de lacre está conectado con la puerta de la energía. Ajuste la longitud de conexión del pozo doble del flotador de bola y controle el nivel líquido en el 1/2 del flotador de bola doble. Encienda para la operación cuando el nivel líquido está en el 1/2 del flotador de una bola más baja. Pare la operación cuando el nivel líquido se levanta hasta el 1/2 del flotador

de bola superior. Contrario, puede ser utilizado para drenar la contaminación y para reducir el nivel líquido.

Fuente de alimentación:	CA 220V, 50Hz o 60Hz
Temperatura de Ambient:	-30~+80
Cambie la corriente de salida:	CA 220V/5ª
Nota:	Dirija comenzado debajo de 1KW, contactor Cambiado de puesto sobre 1KW.

**Figura. IV.3. Características técnicas Sensor Flotador Camsco PD-76AB**

<http://spanish.alibaba.com/product-gs/pd-76ab-automatic-liquid-level-controller-267017673.html>

#### 4.1.2.3. Bomba eléctrica Flojet modelo: P46XD094



**Figura. IV.4. Bomba eléctrica Flojet**

#### **Características**

Presión: 150 PSI

La bomba se sella en material resistente a la corrosión

El sistema de puente elimina la pulsación y el ciclo

La última tecnología del diafragma del molde de la co - inyección elimina virtualmente las trayectorias potenciales del escape

Los materiales mojados fueron diseñados para soportar mucho elementos químicos y corrosivos

Impulsión de velocidad variable

#### 4.1.2.4. PLC Telemecanique TWDLCAE40DRF

Twido PLC 110/240Vac, 40I/O Ethernet

Especificaciones

Entradas	24
Salidas	16
Serie	Twido
Tensión	24Vd.c.
Tipo	PLC

Tabla IV.1. Especificaciones técnicas del PLC Telemecanique TWDLCAE40DRF

#### 4.1.2.5. Válvulas 4V120-06



Figura. IV.5. Válvula 4V120-06

**Fluido:** Aire

**Puertos:** In = Out = Exhaust = 1/8"

**T. Valvula:** 5 vias 2 posic.

**O. Presion:** 1.5 a 8 Bar ( 21 a 114 psi)

**F. Maxima :** 5 ciclos/seg



**Voltaje:** DC12, DC24, AC24, AC110, AC220 volts

<b>Operación</b>	<b>Piloto interno</b>
<b>Área del orificio (CV)</b>	<b>10 mm<sup>2</sup> (0.56)</b>
<b>Lubricación</b>	<b>No requiere lubricación</b>
<b>Presión de la operación</b>	<b>1,5 ~ 8 kg/cm<sup>2</sup> (21~114 PSI)</b>
<b>Máxima presión</b>	<b>10,5 kg/cm<sup>2</sup> (150 PSI)</b>
<b>Temperatura</b>	<b>5 ~ 60°C (41~140°F)</b>
<b>Corriente alterna</b>	<b>50/60Hz</b>
<b>Variación del voltaje</b>	<b>±10%</b>
<b>Consumo de potencia</b>	<b>AC:3V A DC:2,5W</b>
<b>Aislamiento de la bobina</b>	<b>Clase F</b>
<b>Protección bobina</b>	<b>IP65 (DIN40050)</b>
<b>Conector</b>	<b>Tipo DIN</b>
<b>Máxima frecuencia</b>	<b>5 Ciclo /seg</b>
<b>Tiempo de respuesta</b>	<b>0,05 seg</b>

**Tabla IV.2- Características específicas de la válvula 4V120-06**

#### **4.1.2.6. Válvula AV210**



**Figura. IV.6. Válvula AV210**

AV210, Válvulas de accionamiento neumático con asiento inclinado

La AV210 es una válvula de asiento inclinado de accionamiento externo, diseñada para utilizarse en equipos industriales robustos. Esta válvula puede funcionar a temperaturas y viscosidades del medio muy elevadas y es resistente a las partículas de suciedad del medio; por eso, a menudo recibe el nombre de

válvula "a prueba de fallos". Está disponible en bronce de cañón (RG5/bronce) y acero inoxidable (AISI316).

- Programa básico de alta capacidad
- 2/2 vías
- Pistón de asiento angular
- Versión NC: Con cierre hacia y en contra de la dirección del caudal
- Versión NO: Cierre en contra de la dirección del flujo
- Cuerpo de la válvula de bronce o acero inoxidable

#### 4.1.2.7. Sócalos 3UG05-02-2AF00-A7B10000052296

#### 4.1.2.8. Relé controlador de nivel Siemens RXM4AB1BD

Tipo	enchufe en miniatura, indicador mecánico
<b>Pines</b>	14
Zócalo	1XZV3, 1XZV4, 1XZV5
Número de contactos	<b>8</b>
contacto con amperaje (resistente)	<b>6</b>
contacto con amperaje (inductivo)	<b>6</b>
Material de los contactos	plata de níquel
Voltaje de bobina	<b>24 VCC</b>
Bobina de valor nominal	<b>12 VA</b>
Hz	<b>50/60</b>
Altura	<b>1.81 Pulgadas</b>
Ancho	<b>9.83 pulgadas</b>
Normas	UL, CE, CCN, RoHS
Peso	0,08

**Tabla IV.3. Características específicas del Relé controlador de nivel Siemens**

**RXM4AB1BD**

#### 4.1.2.9. Relé controlador de nivel Siemens RXM4AB2BD



Figura. IV.7. Relé RXM4AB2BD

Tipo	enchufe en miniatura, LED indicador
<b>Pines</b>	14
Zócalo	1XZV3, 1XZV4, 1XZV5
Número de contactos	<b>8</b>
contacto con amperaje (resistente)	<b>6</b>
contacto con amperaje (inductivo)	<b>6</b>
Material de los contactos	plata de níquel
Voltaje de bobina	<b>24 VCC</b>
Bobina de valor nominal	<b>12 VA</b>
Hz	<b>50/60</b>
Altura	<b>1.81 Pulgadas</b>
Ancho	<b>0.83 pulgadas</b>
Características	Indicador LED
Normas	UL, CE, CCN, RoHS
Peso	0,08

Tabla IV.4. Características específicas del Relé controlador de nivel Siemens  
RXM4AB2BD

#### 4.1.2.10. Concentrador de señales: ABE7 H16R31



**Figura. IV.8. Concentrador de señales: ABE7 H16R31**

### Características técnicas

<b>Tipo de producto o componente</b>	Sub-base E/S discreta pasiva
<b>Tipo de base secundaria</b>	Sub-base E/S
<b>Número de canales</b>	16
<b>Número de filas</b>	3
<b>Conexiones - terminales</b>	Terminales de tipo tornillo 1 x 0,14...1 x 2,5 mm <sup>2</sup> 26...14 flexible sin extremo de cable Terminales de tipo tornillo 1 x 0,09...1 x 1,5 mm <sup>2</sup> 28...16 flexible con extr. cable Terminales de tipo tornillo 1 x 0,14...1 x 2,5 mm <sup>2</sup> 26...12 sólido Terminales de tipo tornillo 2 x 0,09...2 x 0,75 mm <sup>2</sup> 28...20 flexible con extr. cable Terminales de tipo tornillo 2 x 0,2...2 x 2,5 mm <sup>2</sup> 24...14 sólido
<b>Distribución de polaridad</b>	0 o 24 V
<b>Ancho</b>	125 mm

**Tabla IV.5. Características Técnicas Concentrador de señales: ABE7 H16R31**

### Complementario

<b>Tensión de alimentación</b>	19...30 V IEC 61131-2
<b>Tipo de circuito de alimentación</b>	CC
<b>Número de terminales por canal</b>	3
<b>LED de estado</b>	1 LED verde encendido 1 LED por canal verde estado de canal
<b>Protección contra cortocircuito</b>	2 A fusible interno 5 x 20 mm fundido rápido extremo del PLC
<b>Tipo de conector</b>	HE-10
<b>Número de pin</b>	20 patillas
<b>Modo de fijación</b>	Mediante clips perfil DIN simétrico de 35 mm Mediante tornillos plaza maciza+kit de fijación
<b>Corriente de suministro</b>	≤ 1,8 A
<b>Corriente por canal</b>	≤ 0,5 A

<b>Corriente por salida de común</b>	$\leq 1,8$ A
<b>Caída tensión fusible alimentación</b>	0,3 V
<b>Tensión asignada de aislamiento .</b>	2000 V
<b>Categoría de instalación</b>	II IEC 60664-1
<b>Par de apriete</b>	0,6 N.m plano Ø 3,5
<b>Peso</b>	0,346 kg

**Tabla IV.6. Características Técnicas complementarias Concentrador de señales: ABE7**

**H16R31**

**4.1.2.11. Pretul**

**Características técnicas**

**Voltaje:** 2,4 V

**Velocidad:** 150 r/min

**Tiempo de carga:** 3-5 horas

**Corriente:** 300 mA

**Cargador de baterías**

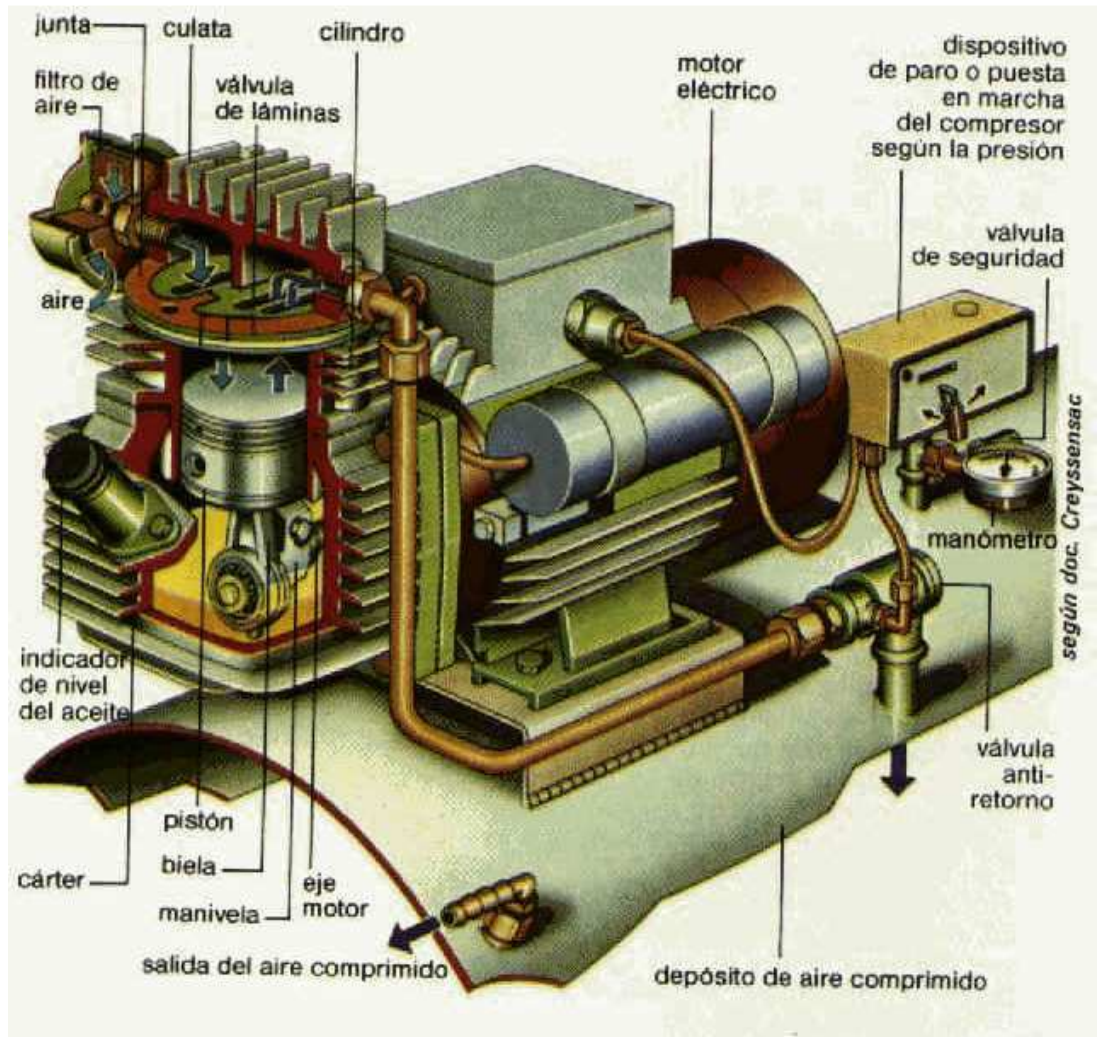
**Entrada:** 120 V ~ /60 Hz

**Corriente:** 140 mA

**Salida:** 2,4 V

**Corriente:** 300 mA

#### 4.1.2.12. Compresor



**Figura. IV.9. Compresor**

**Compresor:** Aparato que sirve para comprimir un fluido, generalmente aire, a una presión dada. Existen dos categorías, las máquinas volumétricas (aumento de presión por reducción de volumen), y los turbocompresores (el aire arrastrado por una rueda móvil adquiere cierta velocidad, que se traduce en un aumento de presión en la rueda y en el difusor de salida).

El compresor de pistón o de émbolo es el más antiguo y extendido, se emplea exclusivamente para presiones elevadas.

En la primera carrera del émbolo, el aire es aspirado a una presión  $p_1$  y el volumen aspirado varía de 0 a  $V_1$ .

Al retroceder el émbolo, este aire es comprimido pasando de la presión  $p_1$  a la presión  $p_2$ , y su volumen varía de  $V_1$  a  $V_2$ .

En la segunda parte o fase de la carrera el aire es expelido a presión  $p_2$ .

Debido al trabajo de compresión, que desprende gran cantidad de calor, debe refrigerarse.

Para presiones muy elevadas, se procede en varias fases, realizándose en cada una un coeficiente de compresión del orden de 6 a 7.

El compresor a émbolos libres del sistema Pescara comprende un motor diesel de dos tiempos, con dos émbolos opuestos, cada uno de los cuales está unido rígidamente a un pistón compresor. Por una de sus caras, este último impele el aire comprimido útil. El volumen de aire comprimido que queda en el cilindro, al final de la carrera forma un colchón neumático y provoca el retroceso.

Por su otra cara, el pistón compresor, durante la carrera motriz, aspira aire atmosférico que, en el retroceso, y merced al empuje del colchón neumático citado, impele hacia el motor; es el aire de barrido y el aire comburente del motor.

#### **4.1.2.13. Mezclador**

Es un elemento elaborado para la mezcla de los líquidos

#### 4.1.2.14. Break para el control del voltaje de 110 voltios



Figura. IV.10. Breaker

<b>Estándar</b>	Conforme con IEC60898
<b>Protección</b>	Contra sobrecarga y cortocircuito
<b>Corriente nominal, In</b>	1, 2, 3, 4, 6, 10, 16, 20, 25, 32, 40, 50, 63 <sup>a</sup>
<b>Tensión nominal</b>	1 polo 230/400VAC 50/60Hz 2, 3, 4 polo 400VAC 50/60Hz
<b>Temperatura ambiente</b>	-5_to +40_pursuant to IEC60898
<b>Poder de corte</b>	10kA
<b>Características</b>	Curvas B, C, D
<b>Número de polos</b>	1P, 1P+N, 2P, 3P, 3P+N y 4P
<b>Tipo de disparo</b>	Magneto-térmico
<b>Tipo de terminal</b>	Bornes
<b>Sección del cable</b>	Cable hasta 25mm <sup>2</sup>
<b>Instalación</b>	Montaje en Raíl DIN de 35mm
<b>Anchura</b>	17.8mm por polo
<b>Endurancia eléctrica</b> <b>In: 1 ... 32A 40 ... 63A</b> <b>Endurancia: 8000 ciclos 8000 ciclos</b> <b>Frecuencia máx.: (Ciclos/hora) 240 120</b>	

Tabla IV.7. Características técnicas Breaker

## 4.2. Planificación del Proyecto

### 4.2.1. Descripción del Sistema



El sistema de mezclado de líquidos, nos permite obtener una mezcla homogénea de tres líquidos almacenados en tanques distintos y el nivel del líquido es controlado por sensores en este caso son solo sensores flotadores, la mezcla se obtiene en el cuarto tanque el cual en un futuro el líquido resultado de la mezcla será llevado al módulo de la envasadora.

#### **4.2.2. Especificación de Requerimientos**

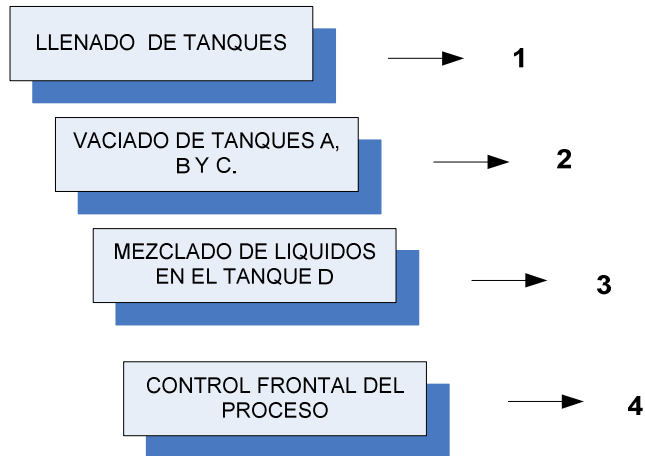
1. El sistema debe permitir el mezclado de líquidos
2. El sistema contará con un panel frontal, desde el cual se podrá controlar el sistema de mezclado de líquidos.
3. El administrador podrá almacenar los datos en la base de datos del propio LabVIEW y mostrar posteriormente en el mismo.

#### **4.2.3. Planificación Inicial**

Para ver la planificación inicial ver Anexo 1 (Cronograma)

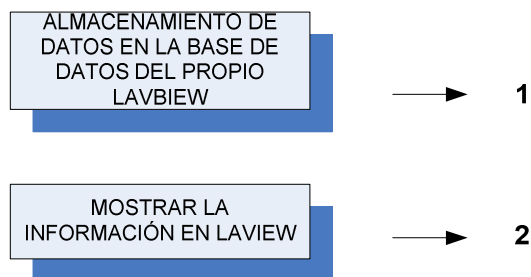
#### **4.2.4. Plan de Iteraciones (Descripción de cada Iteración)**

##### **Primera iteración**



**Figura. IV.11. Plan de Iteración 1**

### Segunda Iteración



**Figura. IV.12. Plan de Iteración 2**

### Descripción de las iteraciones

**Primera Iteración:** Se trata de obtener las funcionalidades básicas del sistema de mezclado de líquidos, programación del PLC, así como el desarrollo del panel frontal, para gestionar la ejecución del sistema del mezclado de líquidos.

**Segunda iteración:** En esta iteración se trata implementar las opciones para el manejo, almacenamiento y visualización de la información que se obtenga durante el proceso.

**Tercera Iteración:** Siempre es necesario ir revisando con frecuencia si el sistema está quedando como los requisitos que se tuvieron inicialmente.

## 4.3. Diseño

### 4.3.1. Diseño Mecatrónico

#### 4.3.1.1. Diseño Mecánico

El Sistema de mezclado de líquidos está equipado con un armazón de Aluminio reforzado, en las dos siguientes figuras se puede observar que los espacios se encuentran bien distribuidos con respecto a los tanques y la parte eléctrica del sistema.

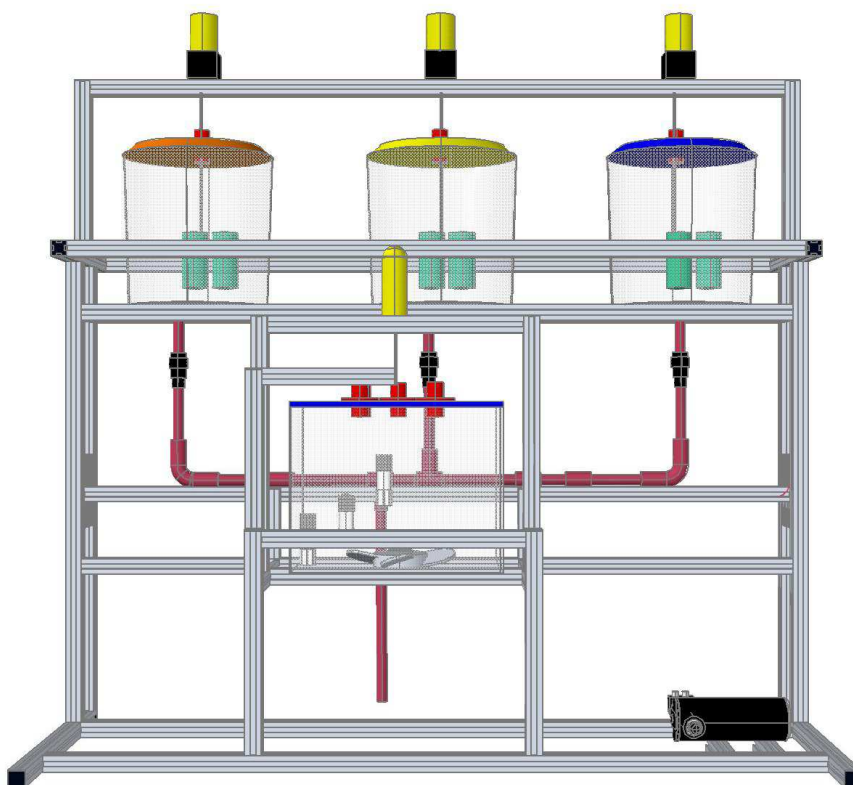


Figura. IV.13. Vista frontal del Sistema de mezclado de líquidos

En la parte posterior se puede observar cómo se encuentra distribuido la parte mecánica y neumática y las tuberías de agua

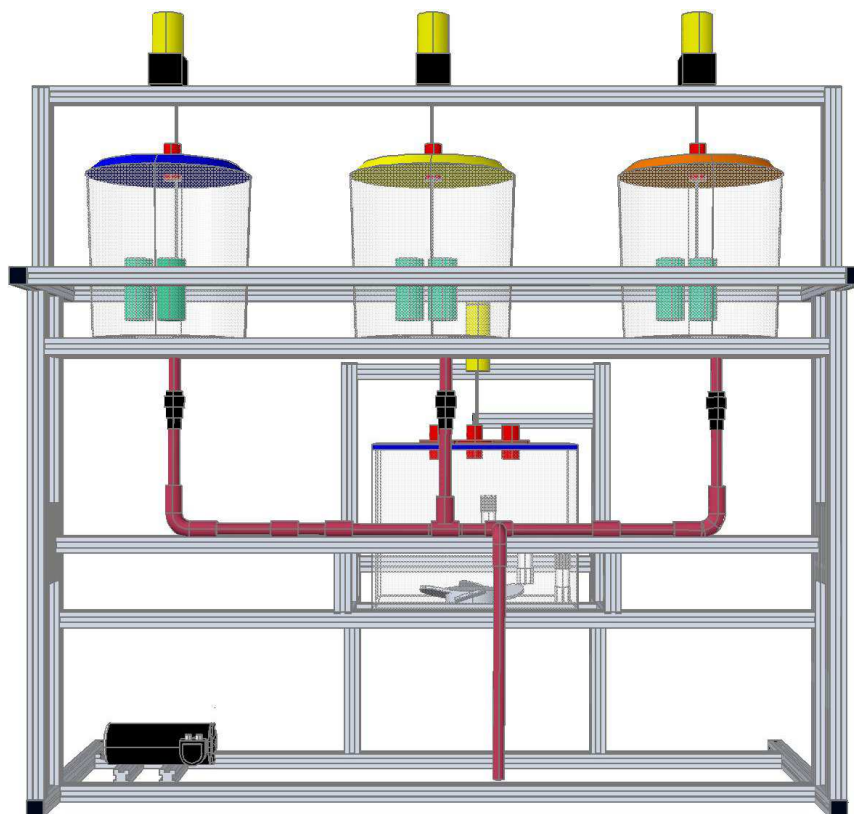


Figura. IV.14. Vista posterior del sistema de mezclado de líquidos

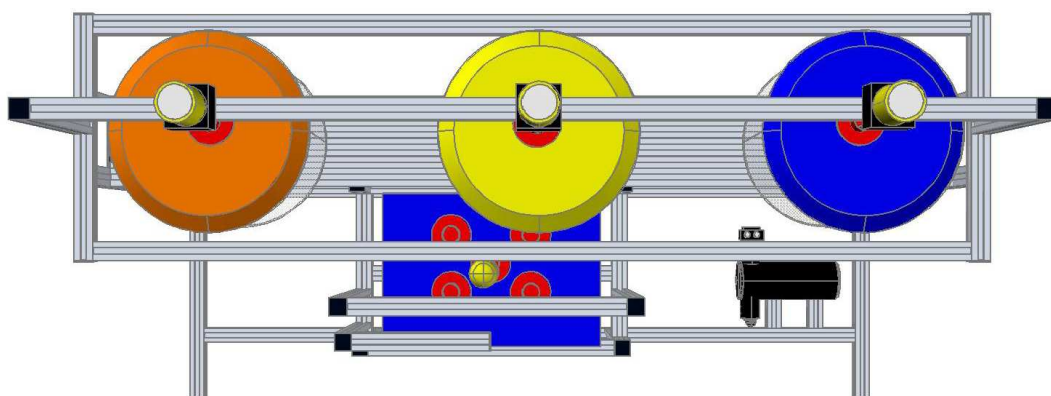


Figura. IV.15. Vista Superior del Sistema de mezclado de líquidos

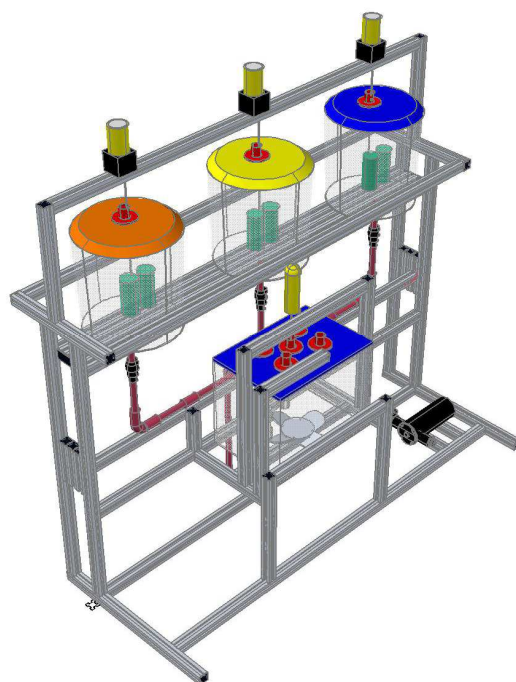


Figura. IV.16. Vista lateral izquierda del Sistema de mezclado de Líquidos

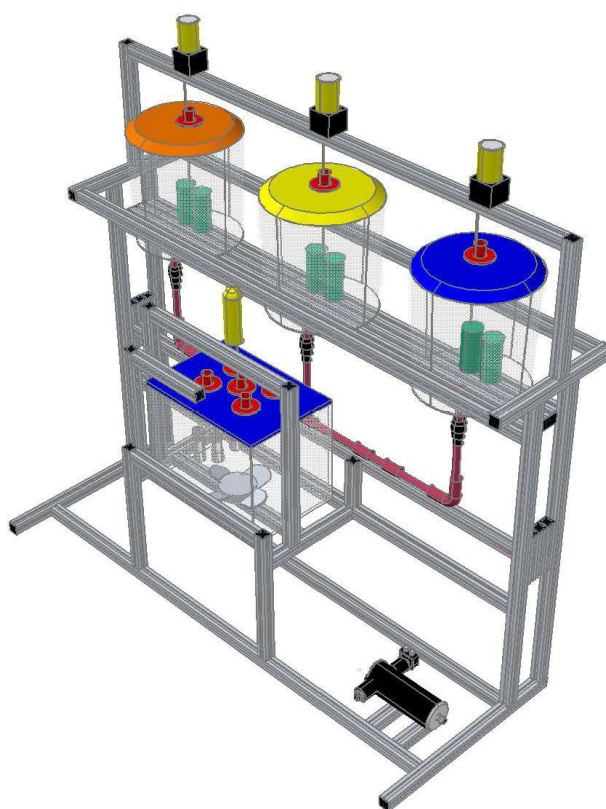


Figura. IV.17. Vista lateral derecha del sistema de mezclado de líquido

En las vista laterales se puede observar cómo va a viajar el líquido de los tanques A, B, C hacia el tanque D, además los sensores flotador y sonda en el tanque de abajo también se encuentra en mezclados el cual nos permite obtener una mezcla homogénea.

#### 4.3.1.2. Diseño Eléctrico

El diseño eléctrico del sistema se encuentran borneras las cuales se encuentran instaladas junto a su respectivo modulo como son PLC, los cables utilizados para la conexión fueron construidos utilizando dos conectores DB25 Macho y Hembra respectivamente y cable UTP.

El diseño eléctrico esta marcado para las Entradas y Salidas con las que cuenta el PLC Telemecanique TWDLMDA40DRF, el cual tiene 24 entradas y 16 salidas.

##### **Entradas**

- Start o inicio
- Stop
- Sensor Flotador 1
- Sensor Flotador 2
- Sensor Flotador 3
- Sensor sonda

##### **Salidas**

- Válvula 1(tanque 1)
- Válvula 2(tanque 2)

- Válvula 3 (tanque 3)
- Mezclador
- Bomba Eléctrica(envía el liquido mezclado a la envasadora)

Entradas/Salidas				
Tabla de entradas				
Uso	Dirección	Símbolo	Utilizado por	Filtrado
<input checked="" type="checkbox"/>	%I0.0	STOP	Lógica aplicación	3 ms
<input checked="" type="checkbox"/>	%I0.1	START_1	Lógica aplicación	3 ms
<input type="checkbox"/>	%I0.2	START_2		3 ms
<input type="checkbox"/>	%I0.3	SELECTOR_IZQ		3 ms
<input type="checkbox"/>	%I0.4	SELECTOR_DER		3 ms
<input checked="" type="checkbox"/>	%I0.5	SF_1	Lógica aplicación	3 ms
<input checked="" type="checkbox"/>	%I0.6	SF_2	Lógica aplicación	3 ms
<input checked="" type="checkbox"/>	%I0.7	SF_3	Lógica aplicación	3 ms
<input checked="" type="checkbox"/>	%I0.8	SONDA	Lógica aplicación	3 ms

**Tabla IV.8. Asignacion de las entradas del sistema de mezclado**

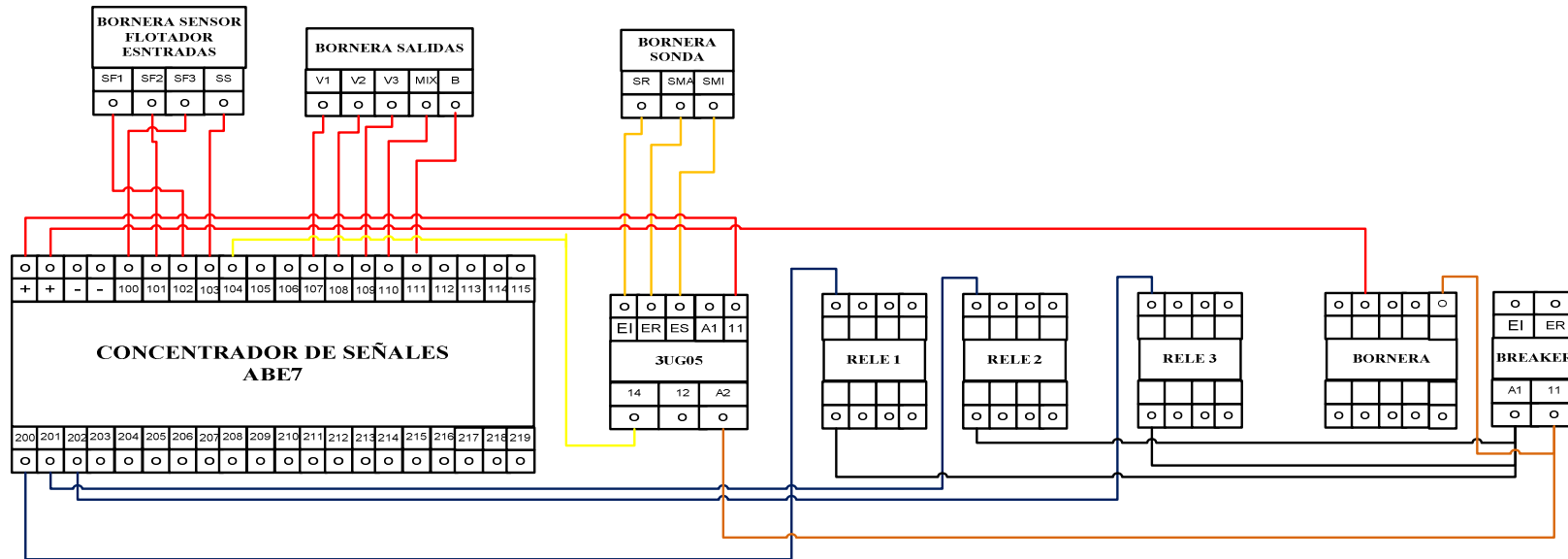
Tabla de salidas				
Uso	Dirección	Símbolo	¿Estado?	Utilizado por
<input type="checkbox"/>	%Q0.0	LAMP_1		
<input type="checkbox"/>	%Q0.1	LAPM_2	<input type="checkbox"/>	
<input type="checkbox"/>	%Q0.2	LAMP_3	<input type="checkbox"/>	
<input checked="" type="checkbox"/>	%Q0.3	VALV_1	<input type="checkbox"/>	Lógica aplicación
<input checked="" type="checkbox"/>	%Q0.4	VALV_2		Lógica aplicación
<input checked="" type="checkbox"/>	%Q0.5	VALV_3		Lógica aplicación
<input checked="" type="checkbox"/>	%Q0.6	MIX		Lógica aplicación
<input checked="" type="checkbox"/>	%Q0.7	BOMBA		Lógica aplicación

**Tabla IV.9. Asignacion de las salidas del sistema de mezclado**

Uso	%M	Símbolo
<input checked="" type="checkbox"/>	%M0	M0
<input checked="" type="checkbox"/>	%M1	M1
<input checked="" type="checkbox"/>	%M2	M2
<input checked="" type="checkbox"/>	%M3	M3
<input checked="" type="checkbox"/>	%M4	M4
<input checked="" type="checkbox"/>	%M5	M5
<input checked="" type="checkbox"/>	%M6	M6
<input checked="" type="checkbox"/>	%M7	M7
<input type="checkbox"/>	%M8	M8
<input type="checkbox"/>	%M9	M9
<input type="checkbox"/>	%M10	M10

**Tabla IV.10. Asignacion de las memorias del sistema de mezclado**

## SISTEMA ELECTRICO DEL SISTEMA DE MEZCLADO DE LIQUIDO



**3UG05:** CONTROLA EL SENSOR SONDA  
**RELE 1:** CONTROLA LA VALVULA 2  
**RELE 2:** CONTROLA LA BOMBA  
**RELE 3:** CONTROLA EL MEXCLADOR  
**SD:** SENSOR SONDA SEÑAL DE REFERENCIA  
**SMI:** SENSOR SONDA SEÑAL MINIMA  
**SMA:** SENSOR SONDA SEÑAL MAXIMA  
**SF1:** SENSOR FLOTADOR 1  
**SF2:** SENSOR FLOTADOR 2  
**SF3:** SENSOR FLOTADOR 3  
**SS:** SENSOR SONDA

Figura. IV.18. Diseño del Sistema Eléctrico del Mezclado de Líquidos



**SISTEMA DE CONTROL DEL SISTEMA DEMEZCLADO DE LIQUIDOS**

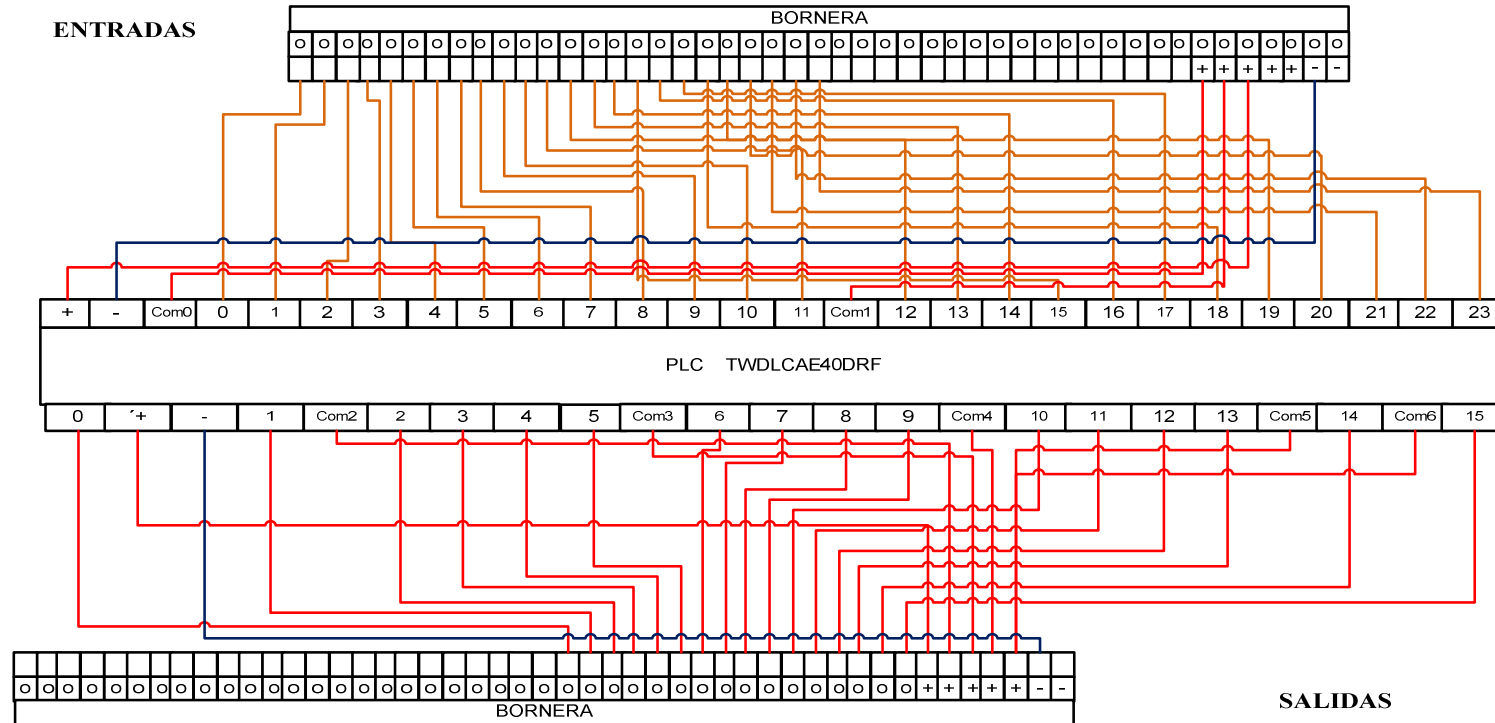


Figura. IV.19. Diseño del sistema de Control del Sistema de Mezclado de Líquidos 1

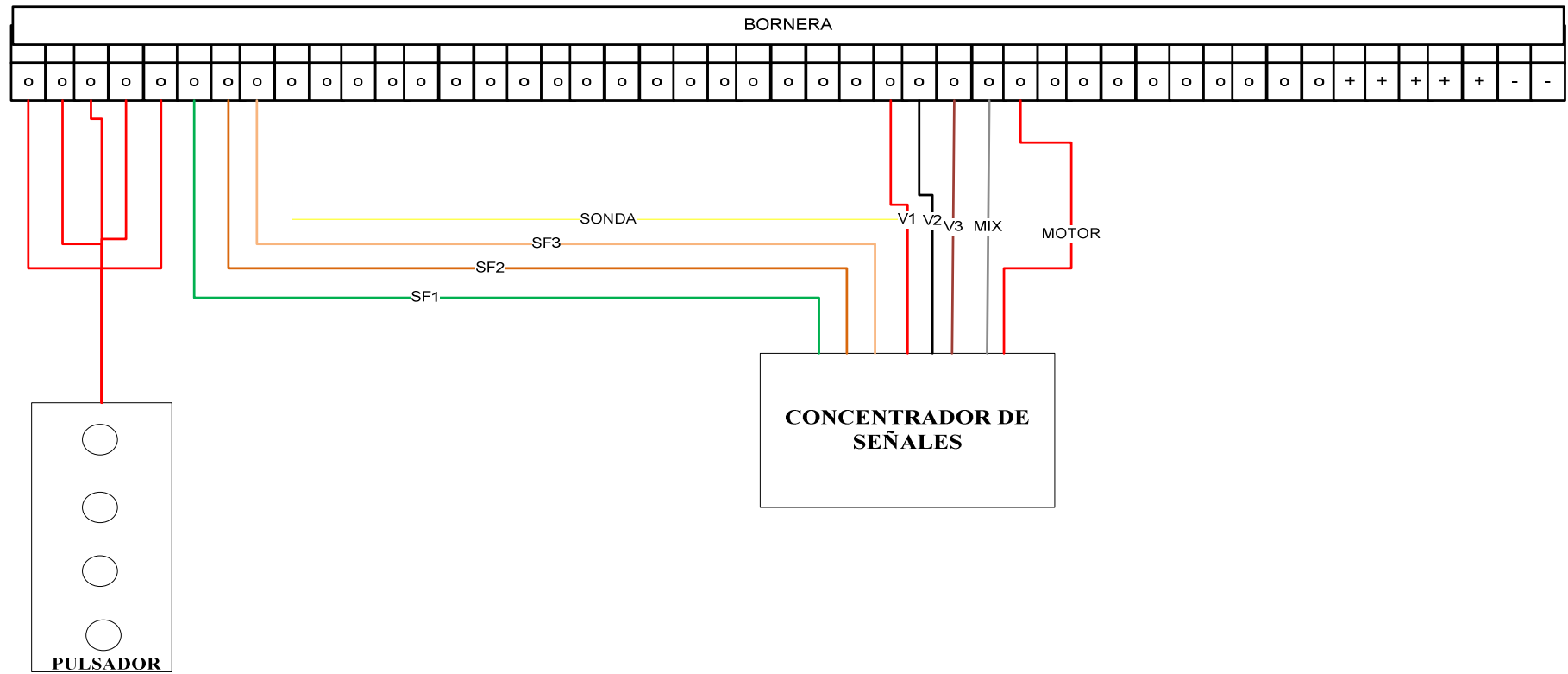


Figura. IV.20. Diseño del sistema de Control del Sistema de Mezclado de Líquidos 2

## 4.3.1.3. Grafcet

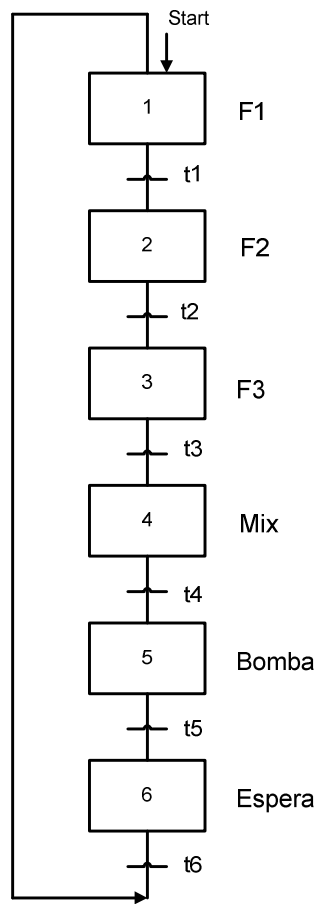


Figura. IV.21. GRAFCET del sistema de mezclado de líquidos

Ecuaciones

$$M1 = M0 \sim M7 + M6t6 + M1 \sim M2$$

$$M2 = M1t1 + M2 \sim M3$$

$$M3 = M2t2 + M3 \sim M4$$

$$M4 = M3t3 + M4 \sim M5$$

$$M5 = M4t4 + M5 \sim M6$$

$$M6 = M5t5 + M6 \sim M1$$

Condiciones para el funcionamiento de la mezcladora

$$M0 = (start + \sim SF1 \sim SF2 \sim SF3 + M0) \sim SONDIA$$

$$M7 = (start + \sim SF1 \sim SF2 \sim SF3 \sim M1 + M7) \sim SONDA$$

### 4.3.2. Diseño Software

#### 4.3.2.1. Diagrama de casos de uso

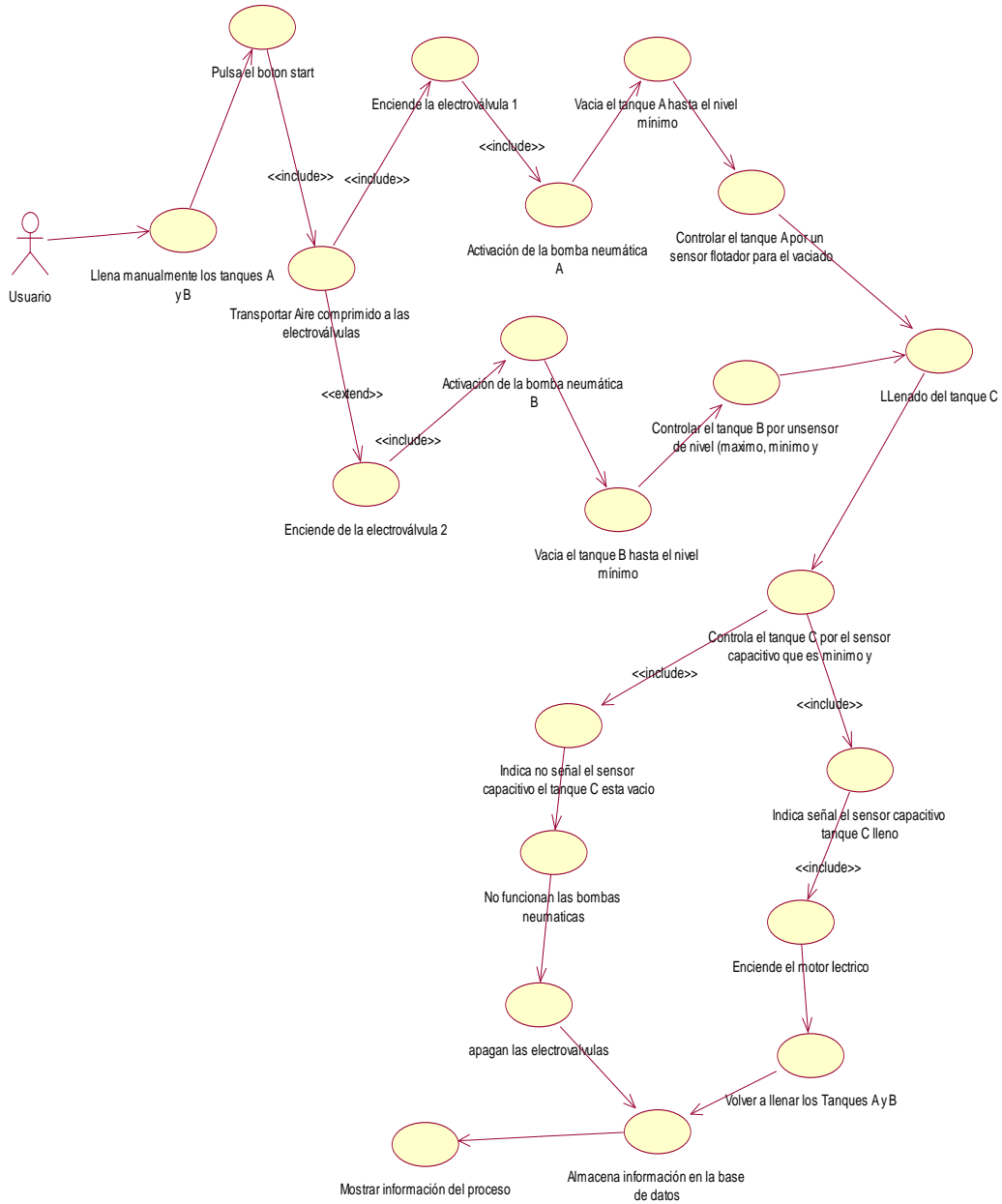


Figura. IV.22. Casos de usos del sistema de mezclado de líquidos

#### 4.3.2.2. Diagramas de Secuencia

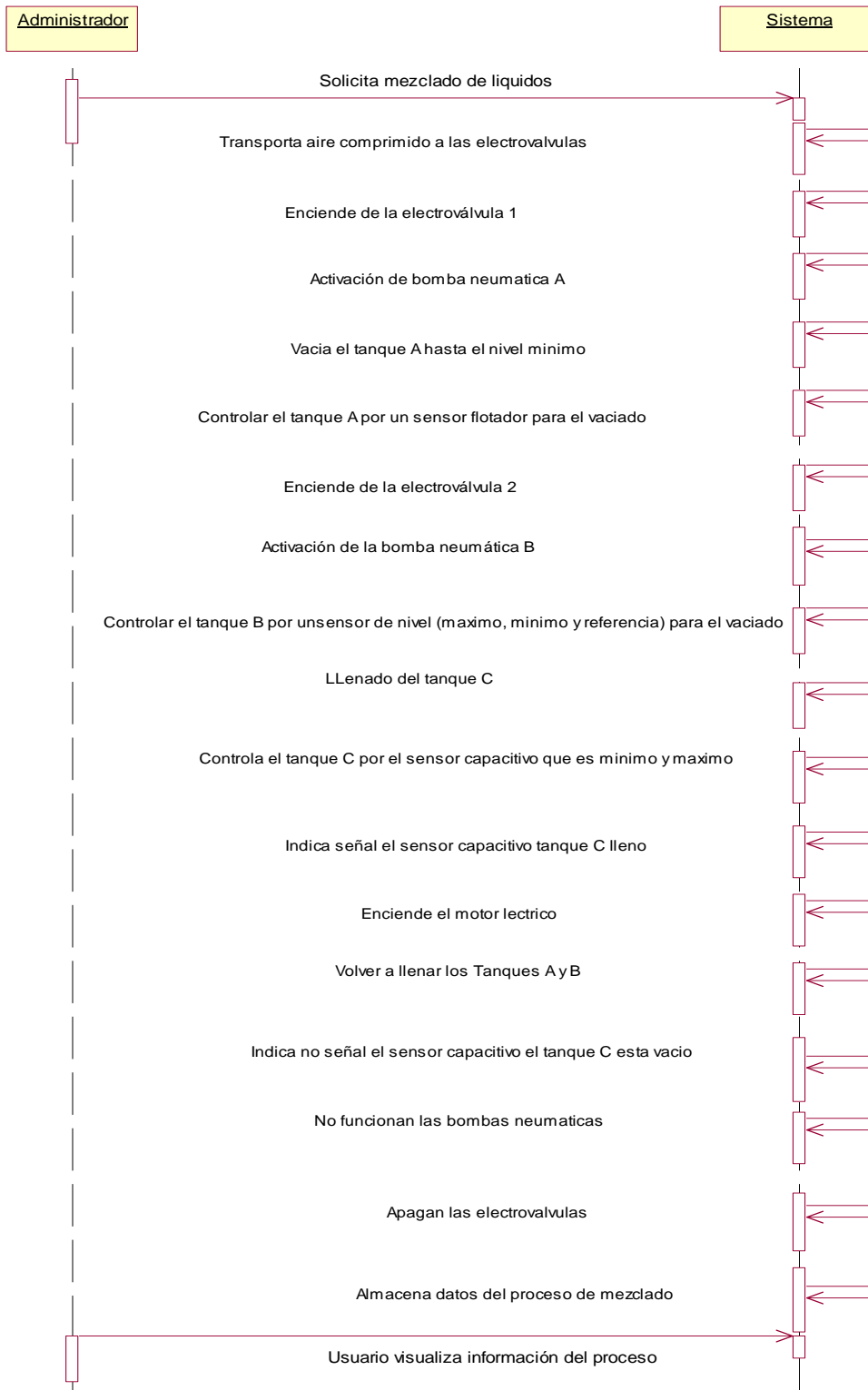


Figura. IV.23. Diagrama de secuencia sistema de mezclado de líquidos

4.3.2.3. Diagramas de Actividades

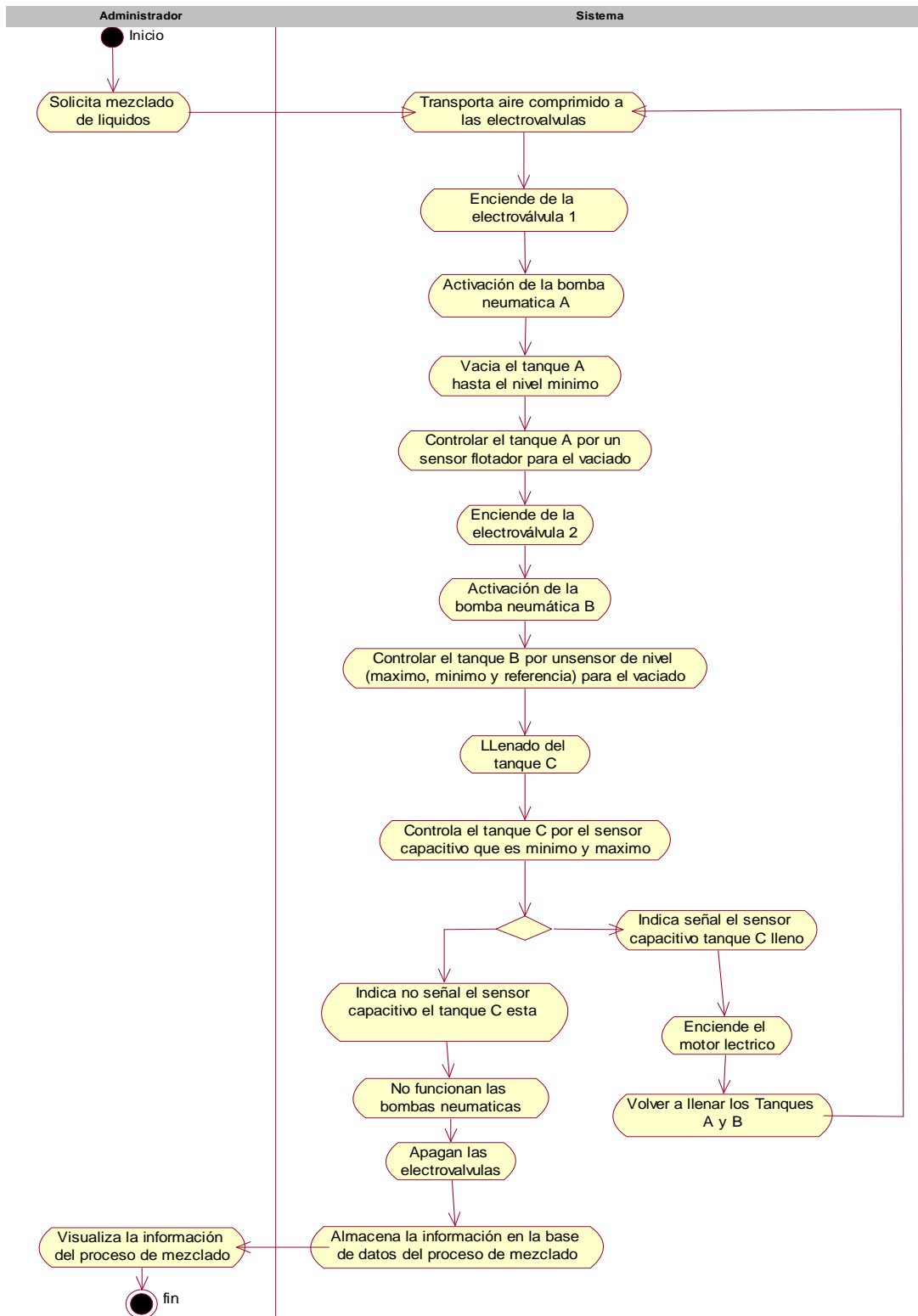


Figura. IV.24. Diagrama de actividades sistema de mezclado de líquidos

#### 4.3.2.4. Diagrama de Clases

#### 4.3.2.5. Diagrama de Componentes

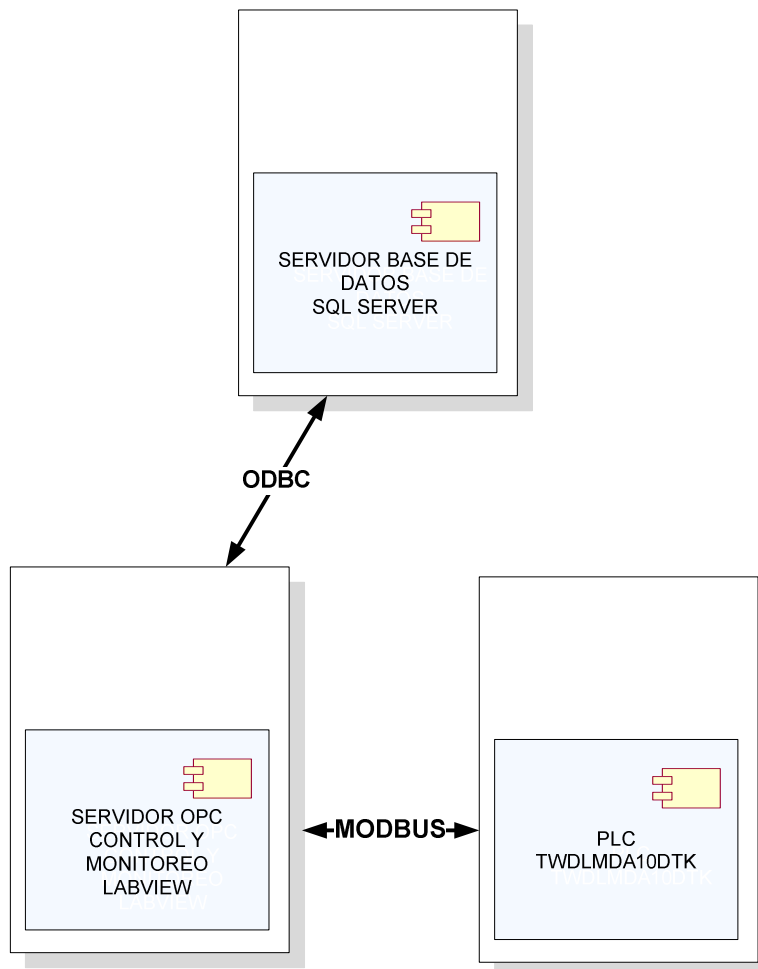


Figura. IV.25. Diagrama de componentes sistema de mezclado de líquidos

### 4.3.2.6. Diccionario de Datos

### 4.3.2.7. Modelo Lógico de la Base de Datos

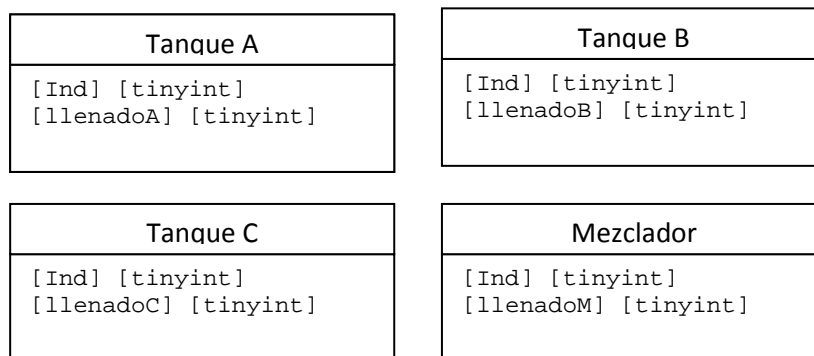


Figura. IV.26. Modelo lógico de BD del sistema de mezclado de líquidos

## 4.4. Implementación

### 4.4.1. Procesos del Mezclado

En la actualidad las empresas industriales sean éstos lácteos, farmacéuticos entre otros realizan el mezclado de líquidos de manera homogénea y equivalente ya que es de vital importancia estos aspectos.



Es así que para tener un proceso de mezclado eficiente se deberá contar con los siguientes aspectos primordiales los mismos que serán un prerrequisito para iniciar el funcionamiento de nuestro módulo y de esta manera podamos automatizar los procesos humanos en empresas industriales pero no se puede lograr automatizar en su totalidad porque algún proceso debe ser manual, en nuestro caso será el llenado de los tanques A, B y C.

### **Prerrequisitos**

Para iniciar el mezclado de líquidos los tanques A, B y C deben tener el nivel máximo de líquido y el tanque D debe estar vacío, de esta manera empezar el proceso con la seguridad de que el proceso se ejecute correctamente.

Presión: 4 Bar

Voltaje de entrada para el PLC: 110 VAC

Voltaje de entrada para la sonda: 110/220 VAC

Voltaje para sensores: 24 VDC

Voltaje para la bomba eléctrica de vaciado: 110 VAC

#### **4.4.1.1. Llenado de tanques**

Cabe recalcar que el llenado de los tanques A, B y C se lo realiza de manera manual es decir que antes de que se pulse el botón START los tres tanques ya deben estar llenos.

Los niveles de dichos tanques son controlados por los sensores flotadores como se explica a continuación:

- El tanque A, B y C se lo controla con el sensor flotador que detecta la presencia o ausencia del líquido.

#### 4.4.1.2. Bombeo de líquido(Revisar no entiendo)

Al pulsar el botón inicio START de la botonera empieza el proceso del mezclado, es así que inicia con el bombeo del líquido a su vez la activación de las tres válvulas que permitirá que empiece el funcionamiento de los tres sensores flotadores del tanque a, b y c, todo este proceso permite seguir con la siguiente etapa.

#### 4.4.1.3. Mezclado de líquido

Finalmente en esta etapa se procede a la mezcla en un tiempo de 5 segundos, con esto se logró el objetivo que es el mezclado de líquidos en el tanque D que se procede a realizarlo de la siguiente forma: el 37% de líquido del tanque A , el 37 % de líquido del tanque B y el 37 % de líquido del tanque C aproximadamente, para lo cual debe funcionar correctamente las válvulas que corresponda a cada uno de los tanques, y cuando llegue al nivel máximo del mezclado de líquido en el tanque D dejará de funcionar dichas válvulas e iniciará actuar la bomba que permitirá enviar el líquido mezclado al módulo siguiente que es la envasadora.

#### 4.4.1.4. Detección de nivel

Para la detección de nivel se utiliza el sensor flotador en cada uno de los tanques de la parte superior y en el tanque de la parte inferior que es el líquido resultado de la mezcla se detecta el nivel con un sensor de sonda.

Es así que en el tanque A, B y C se detectan el nivel del líquido con el sensor flotador y por último en el tanque D se detecta con el sensor de sonda.

En el tanque A la detección se logra con el sensor flotador que de igual manera la señal es enviada a la entrada del PLC sea este el nivel bajo o el nivel alto

En el tanque B la detección se logra con el sensor flotador que de igual manera la señal es enviada a la entrada del PLC sea este el nivel bajo o el nivel alto y.

En el tanque C la detección se lo realiza con el mismo sensor flotador donde la señal es enviada a la entrada de PLC.

En el tanque D la detección del nivel se realiza con el sensor de nivel o sonda de 3 hilos y si tiene señal significa que está lleno y vacío de acuerdo a la circunstancia para lo cual se controla a través de las entradas del PLC.

Para controlar el nivel en el tanque D se utiliza el sensor de sonda que funcionan así: si tiene señal con el hilo de nivel alto significa que el tanque está lleno y por ende debe prenderse el motor eléctrico para sacar el líquido mezclado hacia el módulo de la envasadora y si no tiene señal significa que el tanque D está vacío y por ende debe dejar de funcionar la bomba eléctrica y por consiguiente se apaga el agitador.

#### 4.4.1.5. Sistema de llenado

Esta etapa es cuando ya se ha logrado el mezclado de líquido y está listo para ser enviado al siguiente módulo que es la envasadora pero en este se va enviar nuevamente al tanque A, B y C para lo cual empieza a funcionar la bomba que envía el líquido mezclado.

#### 4.4.1.6. Modulo de control (Eléctrico, Neumático e Informático).



**Figura. IV.27.** Modulo de control del sistema de mezclado de líquidos

**Para el control eléctrico se utiliza los siguientes elementos:**

- Sócalo (RXZE2M114M) que es para controlar el funcionamiento de los sensores de nivel o sonda de 220 voltios.
- El relé RXM4AB1BD para controlar la bomba eléctrica de 110 voltios
- Tres válvulas.
- La entrada de aire del compresor.
- Borneras ABE7 H16R31
- Botonera

**Para el control informático se utiliza los siguientes elementos:**

- El PLC Telemecanique TWD40DRF para las Entradas y salidas del sistema.

**Entradas**

- Sensor Flotador 1
- Sensor Flotador 2
- Sensor Flotador 3
- Sensor de Nivel
- Start / Stop

**Salidas**

- Válvula 1
- Válvula 2
- Válvula 3
- Motor para agitador
- Bomba

**Para el control neumático se utiliza los siguientes elementos:**

- Tuberías de plástico
- Mangueras desde el compresor hacia la entrada del aire comprimido
- Mangueras de la bomba eléctrica.

**4.4.2. Estructura de Aluminio**



**Figura. IV.28.** Estructura de Aluminio

Este sistema de mezclado de líquidos está equipado con un armazón de Aluminio reforzado el mismo que será la base fundamental para la colocación de otros elementos neumáticos, eléctricos, entre otros.

#### **4.4.3. Implementación Mecatrónica**

##### **4.4.3.1. Sistema Mecánico**

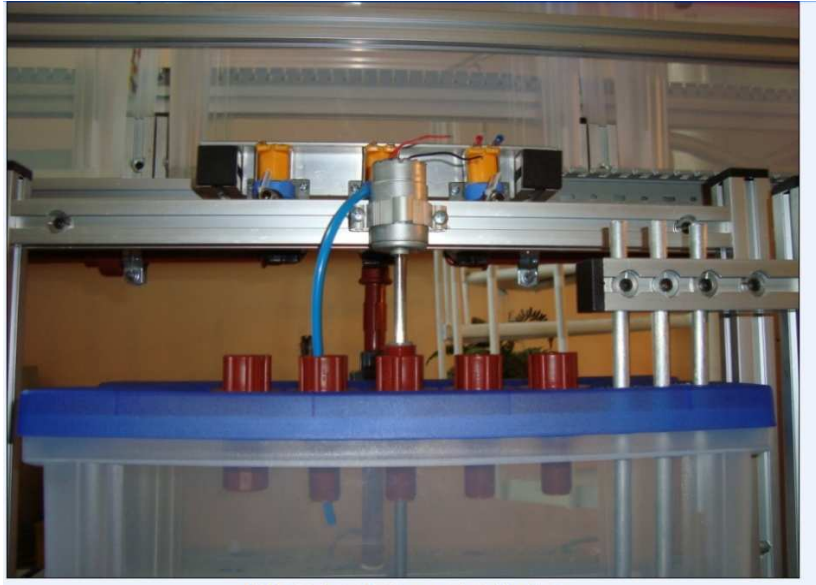


**Figura. IV.29.** Implementación mecánica

#### 4.4.3.2. Sistema Neumático



**Figura. IV.30.** Manguereado de la Bomba eléctrica



**Figura. IV.31.** Manguereado de las válvulas



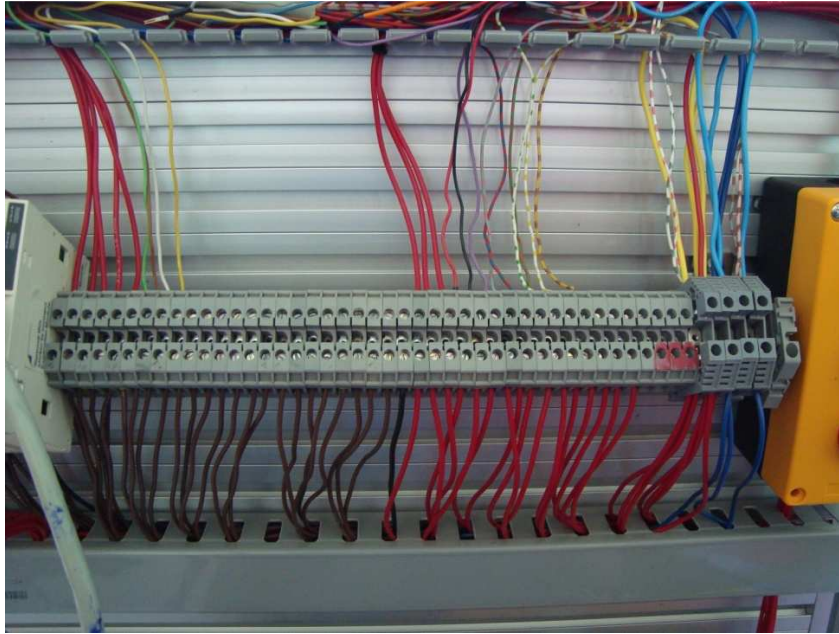
**Figura. IV.32.** Tubería plástica

#### 4.4.3.3. Sistema Eléctrico



**Figura. IV.33.** Conexión de Entradas y Salidas al PLC





**Figura. IV.34. Conexión de entradas y salidas a las borneras**

#### **4.4.3.4. Programación del PLC.**

Para el correcto funcionamiento de nuestro sistema de mezclado de líquidos se utilizó el PLC **TWDLCAE40DRF** que específicamente será para la implementación del proyecto de tesis, su programación se hará con el software que trae consigo, que es Twido Soft v3.1, el cual utiliza Diagramas Ladder como lenguaje de programación.

Para ver el programa en lenguaje Ladder revisar el Anexo 2.

#### **4.4.3.5. Control del PLC**

Para el correcto funcionamiento de nuestro sistema de mezclado de líquidos se utilizó el PLC **CDA40DRF** que específicamente será para la implementación del proyecto de tesis, su programación se hará con el software que trae consigo, que es Twido Suite, el cual utiliza Diagramas Ladder como lenguaje de programación.

Para ver el programa en lenguaje Ladder revisar el Anexo 2.

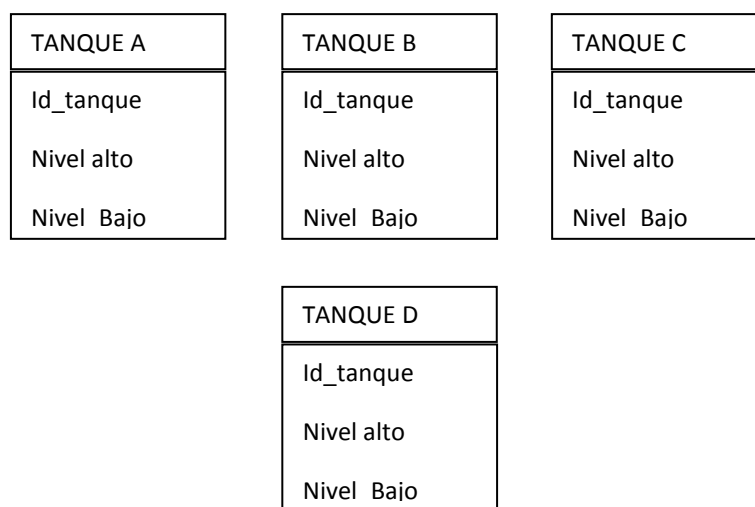
#### 4.4.4. Implementación Software

Para implementar el proyecto de tesis se realizó el panel de control que se desarrolló en el lenguaje LabView la misma que permitirá controlar todo el proceso de mezclado de una forma simple y rápida.

Para la implementación del proyecto de mezclado de líquidos se utilizará Labview 7.1 Professional Edition.

Cabe señalar que el correcto funcionamiento del proyecto mecatrónico en sí dependerá del programa de panel de control de LabView, el mismo que siempre deberá estar en ejecución y es el encargado de interactuar con la base de datos dependiendo de las entradas y salidas del sistema de mezclado leídas y escritas por el PLC, por lo cual se utilizará el Sistema de Administración de base de datos.

##### 4.4.4.1. Modelo Físico de la Base de Datos



**Figura. IV.35.** Modelo físico del Sistema de Mezclado de líquidos

#### 4.4.4.2. Prototipos de Interfaces de Usuario

En la primera pantalla se presenta un menú en el cual se muestra una presentación y el menú el cual muestra la monitorización y el almacenamiento de datos en la base de datos

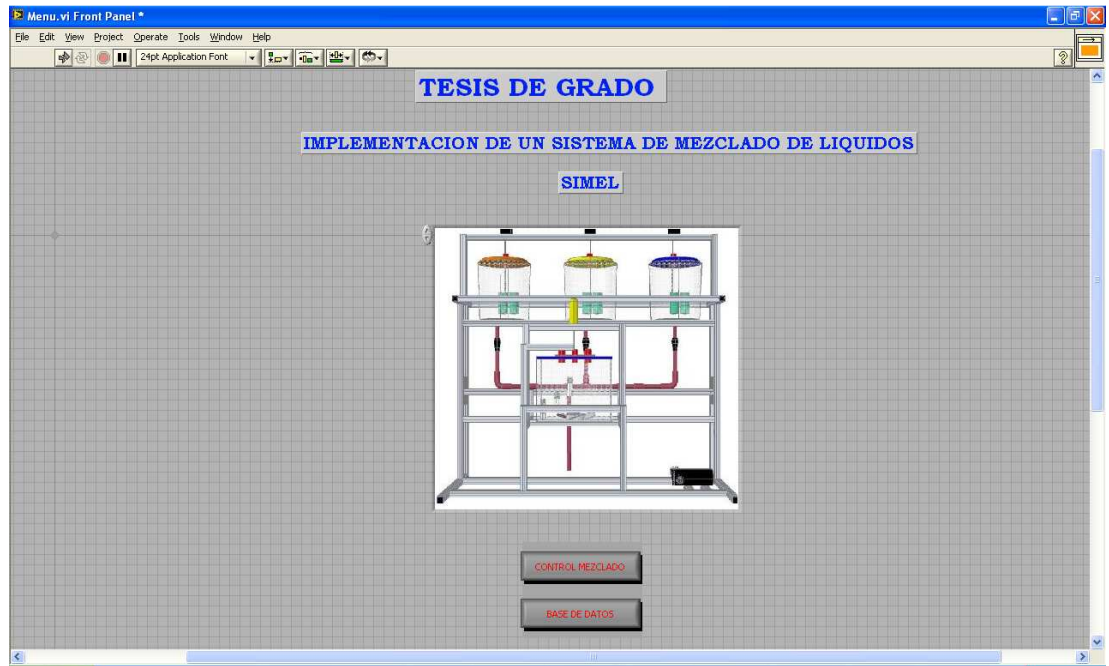


Figura. IV.36. Menú principal del Sistema de Mezclado de liquido

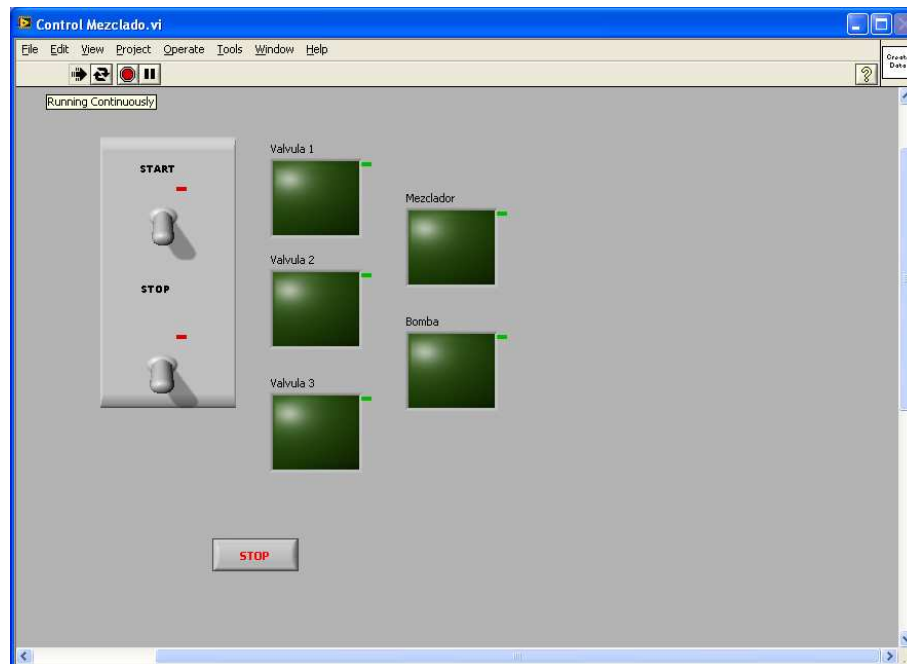
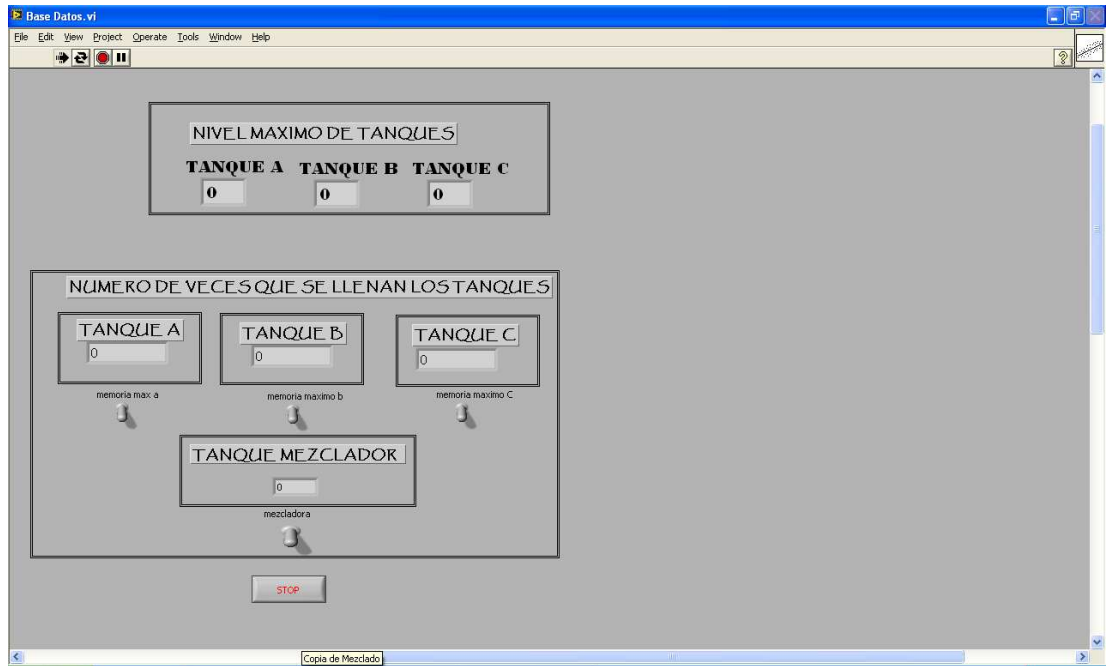


Figura. IV.37. Monitorización del Sistema de Mezclado de líquido



**Figura. IV.38. Extracción de los datos de la base de datos del Sistema de Mezclado de líquido**

## 4.5. Pruebas del Sistema

Las pruebas se las debe llevar a cabo a medida que se va configurando el hardware y desarrollando el software.

### 4.5.1. Pruebas Hardware

Las pruebas hardware se las realizó a medida que se fue instalando y configurando el hardware, con esto se determina que todo va funcionando correctamente, y en caso de existir algún error, se corrige sobre el recorrido, es de vital importancia graduar apropiadamente los sensores, ya que de su correcta graduación depende el buen funcionamiento del sistema.

### 4.5.2. Pruebas Software

Al igual que la implementación hardware, en la implementación software se la debe ir probando tan pronto como se lo va implementando, esto debido a que la metodología de desarrollo que se utilizó está basada en la metodología XP, con algunas implementaciones para que a mas de programación se pueda realizar tareas típicas de un sistema mecatrónico esto se podrá lograr con el panel de control en LabView.

## 4.6. Implantación

### 4.6.1. Manual de Configuración

#### 4.6.1.1. Requerimientos

Para que el sistema funcione de forma eficiente es necesario que se cumpla con los siguientes requerimientos mínimos.

#### **Hardware**

- PC con procesador Pentium 4 de 3 Ghz o superior
- 2 Gb de memoria RAM o superior.
- Disco duro con 80 Gb de espacio libre.
- PLC CDA40DRF
- Puerto RS232
- Cable de datos RS232 para comunicación del PLC con el PC.

- Fuente de alimentación de 24 Voltios para alimentación del PLC y del módulo del sistema de mezclado de líquidos.

### Software

- Microsoft Windows XP o superior.
- Adobe Flash Player 9.0
- Labview Profesional 7.1
- Twido Soft 3.1

#### 4.6.1.2. Configuración del equipo que albergará el Panel de Control.

La carpeta Mezclado\_L debe ser copiada a la unidad C:/, esto debido a que toda la configuración esta referenciada a dicha ruta, esto puede verse a continuación:

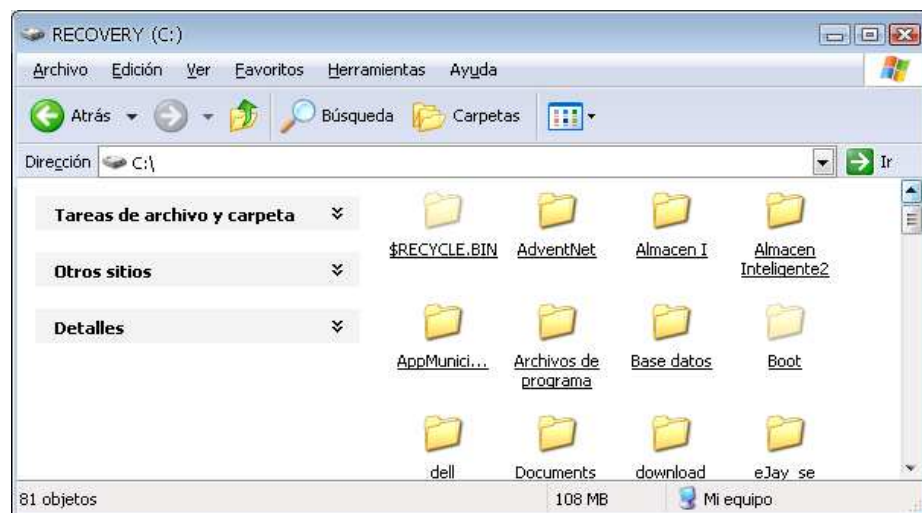


Figura. IV.39. Carpeta Almacen\_I copiada a la unidad C.

A continuación procedemos a configurar el archivo UDL llamado Mezclado.udl, ubicado dentro de la carpeta UDL con la configuración de base de datos:

Debe estar correctamente instalado Labview 7.1 Professional para que el panel de control pueda funcionar perfectamente.

#### 4.6.2. Manual de Usuario

Para el Manual de Usuario ver Anexo 3.

### 4.7. Análisis De Datos Y Comprobación De Hipótesis

**HIPOTESIS.-** La implementación de un sistema mecatrónico para el mezclado de líquidos, caso práctico laboratorio de automatización de la EIS permitirá conseguir el prototipo básico con un diseño sencillo y sólido, mediante la programación de los de los procesos de control del módulo.

Hay algunas técnicas para realizar el análisis y la comprobación de la hipótesis pero la que vamos a utilizar son dos: la de juicios de expertos y las encuestas a estudiantes de la EIS.

Para el juicio de expertos hemos contado con la colaboración de los ingenieros Pablo Montalvo Jaramillo y Humberto Matheu Aguilar, docentes de la facultad de Mecánica de la ESPOCH, con más de 20 años de experiencia en el área de Mecatrónica y Automatización Industrial.(Ver Anexo 4)

### Valoración Cualitativa

EXPERTO PARAMETRO	Pablo Montalvo Jaramillo	Humberto Matheu Aguilar	Conclusiones
Con respecto a la Automatización	Es un sistema operativo lógico que maneja procesos, tiempos y costos	El avance de la industria obliga a las empresas automatizar sus procesos.	En base al criterio emitido, se concluye que la automatización brinda mejores oportunidades de crecimiento a las empresas.
Con respecto a la Mecatrónica	Se puede integrar a las industrias a favor de sistemas informáticos que mezcle la electrónica, mecánica y neumática	Es un área en la que el país está adentrándose desde hace unos pocos años y es importante para la producción.	Se concluye que la Mecatrónica brinda la oportunidad de incrementar la calidad y cantidad de producción en una empresa.
Factor Tiempo	Es lógico que se reduzca el tiempo, ya que el proceso manual siempre es más lento que un proceso automático	Reduce el tiempo y mejora las utilidades para las empresas.	De acuerdo a la opinión emitida se evidencia claramente la reducción de tiempos de respuesta al utilizar un Sistema de Mezclado de Líquidos.
Factor Espacio Físico	Se pueden desarrollar sistemas de mezclado de líquidos muy eficaces, permitirá optimizar el espacio.	Se optimiza el espacio físico así como todos los recursos que intervienen en este proceso.	Los expertos concluyen que con la implementación del Sistema de Mezclado de Líquidos se reduce el espacio físico
Factor Recurso Humano	Se debe analizar el impacto que causa la automatización y reubicar al personal en otras áreas.	Optimiza el recurso humano y obliga a la capacitación del mismo.	Según sus criterios no es un objetivo de la automatización reducir el recurso humano, pero es indudable que esto ocurre.
Implementación de un módulo didáctico	Esto permite a los estudiantes tener una idea de lo que está sucediendo en	Es excelente que se implemente en la FIE, a pesar de los altos costos	Apoyan la implementación de este tipo de investigaciones, ya



	el campo industrial.	que ello requiere	que con ello se brinda la oportunidad a los estudiante de ampliar su conocimiento en la parte práctica
--	----------------------	-------------------	--

**Tabla IV.11. Análisis Cualitativo de Juicio de Expertos**

### Valoración Cuantitativa

La valoración se hará en base a la siguiente tabla.

<b>Criterio</b>	<b>Valor</b>
No estoy de acuerdo	1
Medianamente de acuerdo	5
Estoy de acuerdo	10

**Tabla IV.12. Tabla Parámetros de calificación del análisis cuantitativo**

Experto	Ing. Pablo Montalvo	Ing. Humberto Matheu	Valoración
Con respecto a la Automatización	10	10	20
Con respecto a la mecatrónica	10	10	20
Factor tiempo	10	10	20
Factor espacio físico	10	10	20
Factor Recurso Humano	5	10	15
Implementación de un módulo didáctico	10	10	20

**Tabla IV.13. Tabla Análisis Cuantitativo del Juicio de Expertos**

En las tablas anteriores se puede observar los criterios vertidos por los expertos sobre cada uno de los factores que intervienen en la implementación de un sistema de Mezclado de Líquidos, como son:

Tiempo de respuesta.- En el análisis cualitativo se tuvo una opinión favorable en cuanto a este parámetro, y en el análisis cuantitativo se tuvo una valoración de 20, lo cual refleja que los expertos está, completamente de acuerdo que los tiempos de respuesta del Sistema de Mezclado de Líquidos serán menores que realizando de manera manual.

Espacio Físico.- Del análisis cualitativo y cuantitativo se determinó que los expertos coinciden, en que optimizará el espacio físico con la implementación del sistema.

Recurso Humano.- Este es el parámetro que los expertos no coincidieron completamente en el análisis cualitativo y cuantitativo, sin embargo en la valoración conjunta de los dos expertos, se puede decir que con la implementación del Sistema de Mezclado de Líquidos si se reduce el recurso humano necesario para la operación del mismo.

Unificando las valoraciones cualitativas y cauntitativas de los expertos se avala plenamente la hipótesis planteada en la presente investigación, lo que permite concluir que el sistema de mezclado de líquidos permitirá optimizar el control del Sistema Implementado.

Para la comprobación de la hipótesis anteriormente citada se presenta a continuación los resultados obtenidos del análisis de las **encuestas** realizadas a lo estudiantes de

la EIS exactamente a los estudiantes de Mecatrónica y Automatización Industrial los mismos que se presentan a continuación. (Ver Anexo 4)

Pregunta 1	Pregunta 2	Pregunta 3	Pregunta 4	Pregunta 5
17	16	18	14	19

Tabla IV.14. Resultados de la encuesta

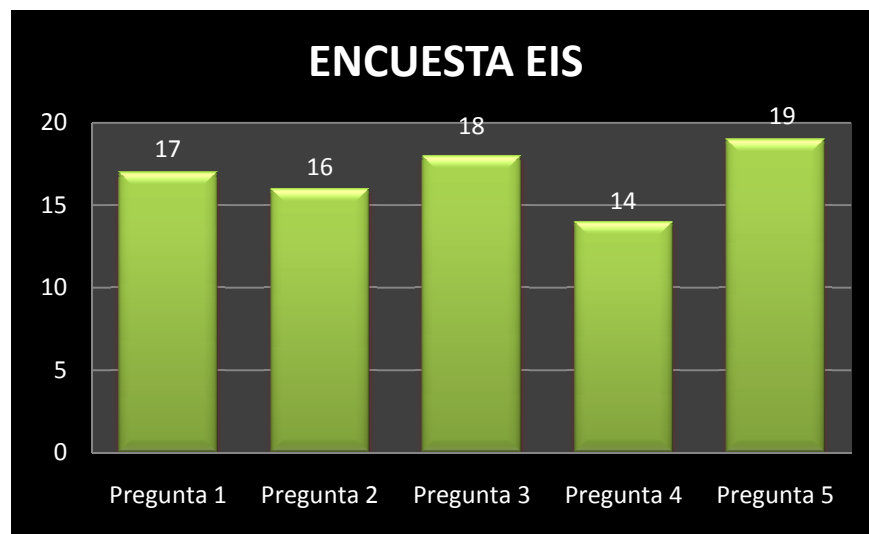


Figura. IV.40. Representación Barras

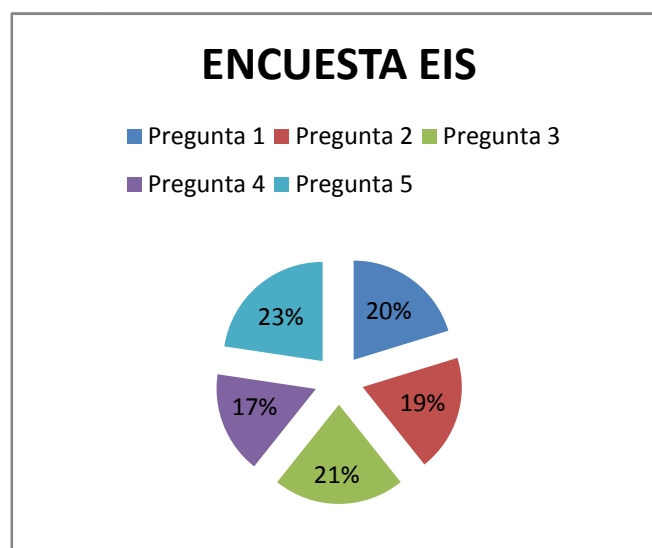


Figura. IV.41. Representación Pastel

## Conclusión del Análisis

Al realizar el estudio de las encuestas realizadas dirigidas a los estudiantes de sexto y séptimo nivel de la EIS se pudo determinar que la implementación del sistema de mezclado de líquidos cumple con las expectativas de los estudiantes que es mejorar la didáctica de la enseñanza aprendizaje y por lo tanto permite adquirir nuevos conocimientos y experiencias para tener un buen desempeño en el medio laboral como profesional.

## CONCLUSIONES

- ✓ Mediante la integración de Aplicaciones Informáticas y la Mecatrónica para la industria, hoy el control industrial se lo realiza por medio de las máquinas automatizadas, enviando así al hombre asumir el papel de supervisor, esto hace que exista una manipulación directa hombre - maquina pero únicamente cuando la maquina entra en estado crítico o de error.
- ✓ El programa implementado para el Sistema de Mezclado de Líquidos (SIMEL) se trata de una lógica de mezclado de líquidos, mediante el cual

podemos darnos cuenta la ayuda que brinda para que la gestión del mezclado sea eficiente, homogénea y precisa ya que estos tipos de sistemas son utilizados en las industrias farmacéuticas y lácteas.

- ✓ Se determinó que lenguaje de programación más idóneo para la implementación del Panel de control es LabVIEW de la empresa National Instruments (NI), dadas las características y facilidades que ofrece para el control de equipos y procesos industriales.
- ✓ Se realizó un estudio y análisis de las metodologías existentes para proyectos mecatrónicos combinándolo con la metodología de software ágil XP y finalmente realizar la adaptación de una metodología propia para nuestro proyecto, los mismos que cubren las etapas del diseño mecatrónico en proyectos de este tipo.
- ✓ Mediante la utilización de LabVIEW y el servidor Industrial OPC Servers, se logró la comunicación entre el PLC y PC, se desarrollo también un Panel Frontal en LabVIEW, el cual permite el control del sistema de mezclado de líquidos.
- ✓ Con la implementación de una base de datos para realizar el control de forma lógica, la misma que almacenará la cantidad máxima del líquido almacenado en cada tanque pero especialmente el tanque D ya que es el resultado de la mezcla de los tres líquidos del tanque A, B y C.

## RECOMENDACIONES

- ✓ Para empezar el desarrollo de un proyecto mecatrónico se recomienda analizar las diferentes metodologías mecatrónicas existentes, luego de esto se debe seleccionar aquellas que más se adapte a las características del proyecto, teniendo en cuenta que deberá realizarse modificaciones a dichas fases de las metodologías para combinarlos y finalmente adaptarlos.
- ✓ Antes de poner en marcha el sistema de mezclado, se recomienda verificar que la instalación física del sistema eléctrico, neumático y mecánico, se encuentre de acuerdo a los diseños especificados en el

capítulo de desarrollo, para evitar daños en los equipos así como posibles lesiones en las personas que lo manipulan.

- ✓ Se recomienda que durante todo el proceso de desarrollo, tanto físico como lógico, se vayan realizando pruebas continuas, ya que de esta forma se evita la pérdida de tiempo y se asegura el éxito del proyecto.
  
- ✓ En caso de que se necesiten realizar modificaciones en el sistema, se recomienda a los técnicos estudiar el capítulo de desarrollo del presente proyecto y a los usuarios leer detenidamente el manual de usuario para una correcta utilización del Sistema de Mezclado de Líquidos (SIMEL).

## RESUMEN

Se construyó un sistema mecatrónico, que se lo implementó en la ESPOCH, Laboratorio de Automatización Industrial, con la finalidad de mejorar el proceso de enseñanza – aprendizaje en los estudiantes.

Se adaptó una combinación de tres metodologías: para la automatización; para el desarrollo de sistemas mecatrónicos y para el desarrollo de software, incluyéndose fases de desarrollo e implementación mecánica, electrónica empleando el estándar IEC 61131-3.

El sistema se implementó en una estructura de aluminio 1.40 cm de alto y 1.60 cm de alto, cuenta con cuatro tanques cada uno con sensores que determinan la cantidad de líquido a almacenarse y las señales a ser enviadas a la base de datos; para el control automático se desarrolló un panel frontal en LabView, para el funcionamiento secuencial se implementó ecuaciones Grafcet con programación Ladder en TwidoSuite y para controlar el llenado / vaciado de tanques se utilizó base de datos de 2 tablas.

Para proceder al mezclado el tanque 4 debe encontrarse vacío e inmediatamente los tanques 1, 2 y 3 envían el líquido en un tiempo de 30s al tanque 4, la mezcla se lo realiza en 5s, es decir, el producto está listo para ser enviado al siguiente módulo (envasadora). Todo el proceso se realiza en un total de 35 segundos.

Se logró obtener un prototipo sencillo que permite desarmar, armar, entender y manejar la aplicación software para prácticas estudiantiles, recomendándose dar el uso y manipulación adecuada al sistema mecatrónico de mezclado de líquidos construido.

## **SUMARY**

A mechatronic system, implemented at the ESPOCH, Industrial Automation Lab, was constructed, to improve the student teaching – learning process. A combination of three methodologies was adapted: for automation, for the development of mechatronic systems and for the software development, including development phases and mechanical and electronic implementation using the standard IEC 61131-3. The system was implemented in an aluminum structure 1.40 cm high and 1.60 cm thick, it consists of four tanks each with sensors determining the liquid quantity to be



stored and the signals to be sent to the database; for the automated control a front panel was developed in the Labview; for the sequential functioning equations Grafcet were implemented with the Ladder Programming in TwidoSuite and to control the filling / emptying of tanks the database of two tables was used. To mix the tank 4 must be empty and immediately tanks 1,2 and 3 send the liquid in 30 s to tank 4; mixing is carried out in 5 s, i.e. the product is ready to be sent to the following module. All the process is carried out in 35 seconds. It was possible to obtain a simple prototype which permits to disassemble, assemble, understand and handle the software application for student practices. It is recommended to use and handle adequately the mechatronic system for liquid mixing.

## **GLOSARIO**

### **Archivo UDL**

Es la extensión del archivo de la conexión de labview con SQL Server

### **ASCII**

Código estándar americano para el intercambio de información (del inglés "American Standard Code for Information Interchange"). Protocolo de

comunicación que representa caracteres alfanuméricos, incluidos números y letras.

## **AUTÓMATA**

Controlador programable Twido. Existen dos tipos de controladores: compacto y modular.

## **AUTOMATIZACIÓN**

Automatización es la tecnología que trata de la aplicación de sistemas mecánicos, electrónicos y de bases computacionales para operar y controlar la producción.

## **DATABASE CONNECTIVITY TOOLKIT EN LABVIEW**

Permite establecer la conexión entre LabVIEW y SQL Server

## **EPROM.**

Memoria de solo lectura, reprogramables con borrado por ultravioletas.

## **EEPROM.**

Memoria de solo lectura, alterables por medios eléctricos.

## **GRAFCET**

Grafcet permite representar gráficamente y de forma estructurada el funcionamiento de una operación secuencial. Método analítico que divide cualquier sistema de control secuencial en una serie de pasos a los que se asocian acciones, transiciones y condiciones.

## **Lenguaje Ladder**

Programa escrito en lenguaje Ladder compuesto por una representación gráfica de instrucciones de un programa de controlador con símbolos para contactos, bobinas y bloques en una serie de escalones ejecutados de forma secuencial por un controlador.

## **MODBUS**

Protocolo de comunicaciones que hace posible que exista la comunicación del PCL - PC.

## **PLC**

Es la unidad de control mínima en un proceso automatizado; Con el cual se pueden programar eventos resultantes de acuerdo a un estado específico del sistema.

## **BIBLIOGRAFIA**

- 1.- ALCIATORE, D. Sistema Mecánico: introducción a la mecatrónica y los sistemas de medición. 2ª .ed. Italia: McGraw-Hill Interamericana, 2007. 426 p.

2.- BOLTON, W. Mecatrónica: sistemas de control electrónico en la ingeniería mecánica y eléctrica. 3ª.ed. México: Alfaomega, 2007. 596 p.

3.- BASE DE DATOS

<http://es.kioskea.net/contents/bdd/bddintro.php3>

20101013

4.- GARCÍA, E. Automatización Industrial: automatización de procesos industriales. Bogotá: s. edit. 2006. 253 p.

5.- ELEMENTOS DEL SISTEMA DE MEZCLADO

[http://www.ra.danfoss.com/TechnicalInfo/Literature/Manuals/04/ICP\\_D400F105indd.pdf](http://www.ra.danfoss.com/TechnicalInfo/Literature/Manuals/04/ICP_D400F105indd.pdf)

20101005

<http://bibdigital.epn.edu.ec/bitstream/15000/2071/1/CD-2874.pdf>

20101005

<http://pdf.directindustry.es/pdf/ls-industrial-systems/miniature-circuit-breakers/Show/bkn-b-19851-4057.html>

20101005

<http://www.drillspot.com/brand/telemecanique>

20100903

<http://equipment.tradepad.net/products-277019-liquid-level-controller-level-switch.html>

20100922

<http://www.euskalnet.net/j.m.f.b./neunaticabasica.htm>

20100922

## 6.- LENGUAJES DE PROGRAMACION

[http://www.inele.ufro.cl/apuntes/LabView/Manuales/Sistema%20\\_De%20\\_Adquisici%F3n%20\\_De%20\\_Datos%20\\_Para%20\\_Biomedica.pdf//manual](http://www.inele.ufro.cl/apuntes/LabView/Manuales/Sistema%20_De%20_Adquisici%F3n%20_De%20_Datos%20_Para%20_Biomedica.pdf//manual)

20100910

<http://books.google.com.ec/books?id=ZYAYyO8CmiIC&pg=PA11&lpg=PA11&dq=como+enlazar+paneles+de+presentacion+en+labview&source=bl&ots=mXTMvprSvb&sig=32zpFRGJuHUj8DuCiSrvKN4maVY&hl=es&ei=->

[F2jTOKZOCkclgrfX\\_Aw&sa=X&oi=book\\_result&ct=result&resnum=3&ved=0CA4Q6AEwAg#v=onepage&q&f=false//](http://books.google.com.ec/books?id=F2jTOKZOCkclgrfX_Aw&sa=X&oi=book_result&ct=result&resnum=3&ved=0CA4Q6AEwAg#v=onepage&q&f=false//)

20100910

<http://www.casadellibro.com/capitulos/8426714269.pdf//8426714269>

20100910

## 7.- METODOLOGÍAS MECATRONICAS

[http://www.dte.us.es/tec\\_ind/electron/ai/proyecto/Metodologia.pdf](http://www.dte.us.es/tec_ind/electron/ai/proyecto/Metodologia.pdf)

20100511

[http://antiguo.itson.mx/revistaimpulso/vol1p1\\_files/articulos/V1\\_art4.pdf](http://antiguo.itson.mx/revistaimpulso/vol1p1_files/articulos/V1_art4.pdf)

20100511

## 8. OPC

<http://blog.s21sec.com/2009/02/opc-estandar-en-las-redes-industriales.html>

20101410

<http://bibing.us.es/proyectos/abreproy/11477/fichero/Memoria%252F02.pdf>

20101410

## 9.- PLC

[http://www.plcopen.org/pages/.../intro\\_iec\\_61131\\_3\\_spanish.doc](http://www.plcopen.org/pages/.../intro_iec_61131_3_spanish.doc)

20091228

<http://www.controlmicrosystems.com/>

20091228

<http://www.buenastareas.com/temas/norma-iec-61131-3/0>

20101004

<http://www.grupo->

[maser.com/PAG\\_Cursos/Auto/auto2/auto2/PAGINA%20PRINCIPAL/index.htm](http://www.grupo-maser.com/PAG_Cursos/Auto/auto2/auto2/PAGINA%20PRINCIPAL/index.htm)

20101110

[http://galia.fc.uaslp.mx/~cantocar/automatas/APUNTES\\_CURSO/CAPITULO2.PDF](http://galia.fc.uaslp.mx/~cantocar/automatas/APUNTES_CURSO/CAPITULO2.PDF)

20101003

## 10.- PROTOCOLOS DE COMUNICACIÓN

<http://www.dte.upct.es/personal/manuel.jimenez/investigacion/congresos/pdfs/TAEE2000%20Comunic%20Ind.pdf>

20101006

[http://www.equiposdidacticos.com/pdf/catalogos/Manual\\_Twido.pdf](http://www.equiposdidacticos.com/pdf/catalogos/Manual_Twido.pdf)

20101011

## 11.- SISTEMA ELÉCTRICO

<http://www.scribd.com/doc/32351971/Medidores-de-nivel-Sensores>

20101003

[http://www.fluidcomponents.es/Aerospace/Products/LiquidLevel/A\\_ProdLiquid.asp](http://www.fluidcomponents.es/Aerospace/Products/LiquidLevel/A_ProdLiquid.asp)

20101005

<https://www.u->

[cursos.cl/ingenieria/2009/1/EL41B/1/material\\_alumnos/objeto/29763](https://www.u-cursos.cl/ingenieria/2009/1/EL41B/1/material_alumnos/objeto/29763).

20100510

<https://www.u->

[cursos.cl/ingenieria/2010/1/EL3003/2/material\\_docente/previsualizar?id\\_material=280136](https://www.u-cursos.cl/ingenieria/2010/1/EL3003/2/material_docente/previsualizar?id_material=280136)

20101011

[http://www.cablesenoferta.com/Tienda/\(S\(bzzljb555syu0h55rrvknvei\)\)/catalog/productinfo.aspx?id=1&cid=2&AspxAutoDetectCookieSupport=1](http://www.cablesenoferta.com/Tienda/(S(bzzljb555syu0h55rrvknvei))/catalog/productinfo.aspx?id=1&cid=2&AspxAutoDetectCookieSupport=1)

20101011

## 12.- SISTEMA NEUMÁTICO

<http://www.sapiensman.com/neumatica/nueutica11.htm>

20100310

[http://www.ehu.es/inwmooqb/NEUMATICA/Neumatica%20y%20el  
electroneumatica/CAP4.%20Valvulas%20neumaticas.pdf](http://www.ehu.es/inwmooqb/NEUMATICA/Neumatica%20y%20el%20electroneumatica/CAP4.%20Valvulas%20neumaticas.pdf)

20100510



# ANEXOS

## **ANEXO 1**

### **Planificación Inicial**

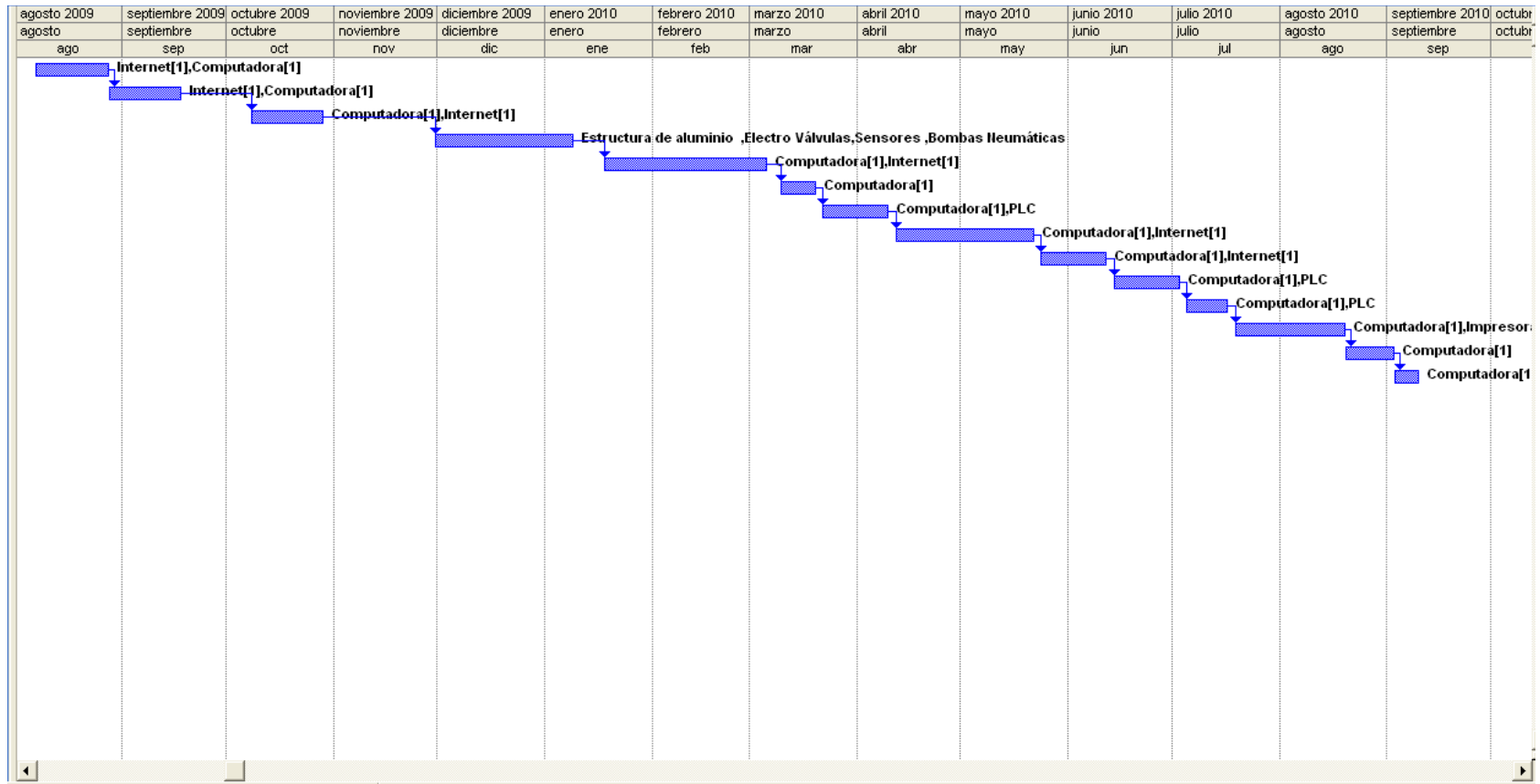
El plan general de trabajo que inicialmente se ha planteado para el desarrollo de las actividades del presente anteproyecto se especifica en las siguientes fases:

- FASE I: Análisis, estudio e investigación del tema a desarrollar.

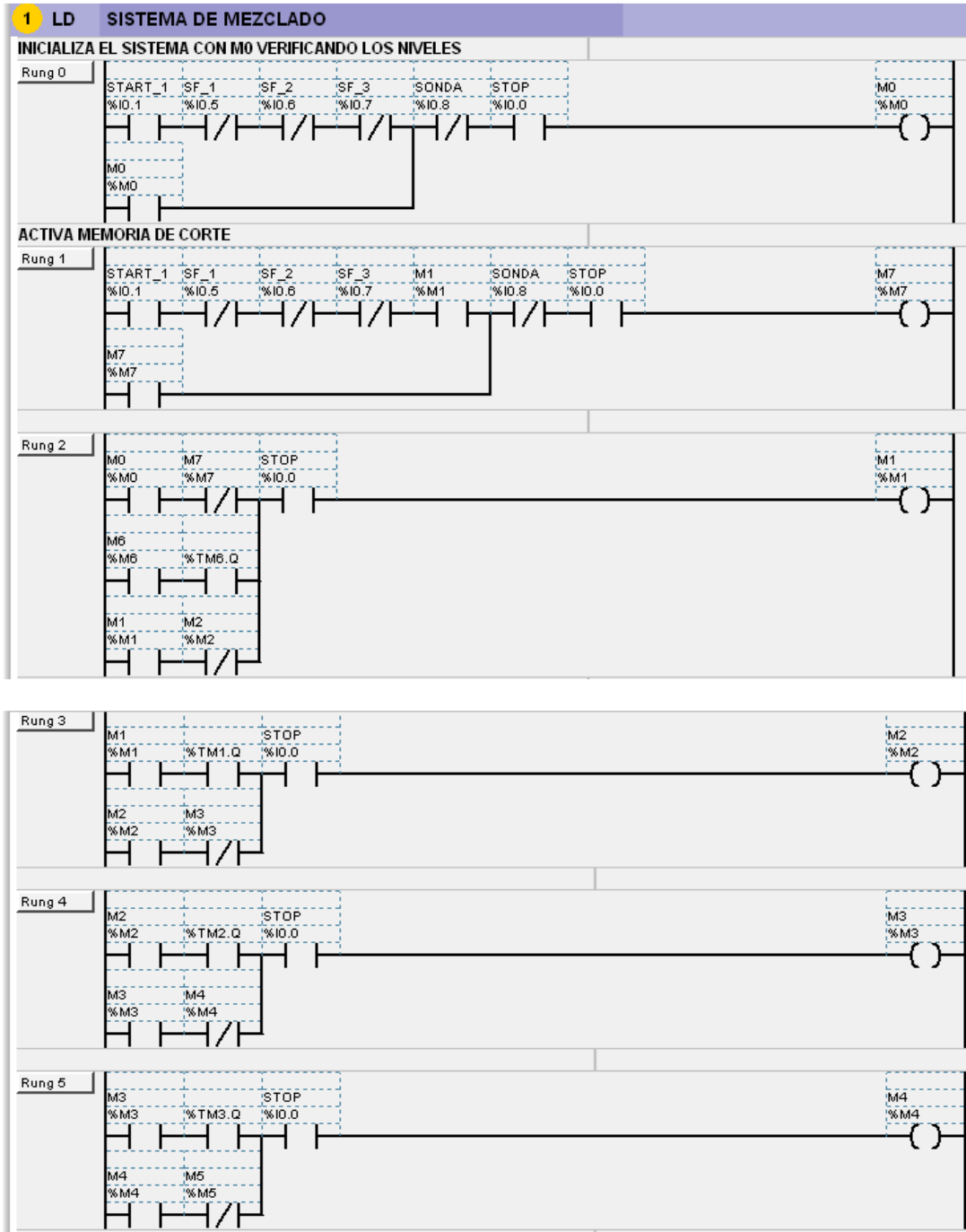
- FASE II: Recopilación y almacenamiento de datos (primarios y/o secundarios), basados en la investigación
- FASE III: Planteamiento y diseño del módulo de una mezcladora automática.
- FASE IV: Ejecución (construcción y ensamblaje) de una mezcladora automática.
- FASE V: Programación para los procesos a ejecutarse del módulo de una mezcladora automática.
- FASE VI: Supervisar el sistema de acuerdo a los requerimientos establecidos.
- FASE VII: Pruebas de funcionamiento y mantenimiento.
- FASE VIII: Diseño e implementación de la Interfaz de usuario para el control de la producción.
- FASE IX: Implementación de la Base de Datos
- FASE X: Pruebas y corrección del módulo final.
- Fase XI: Determinación de Resultados
- FASE XII: Corrección del Trabajo Final
- Fase XIII: Defensa Final.
- Fase XIV: Corrección y presentación final.

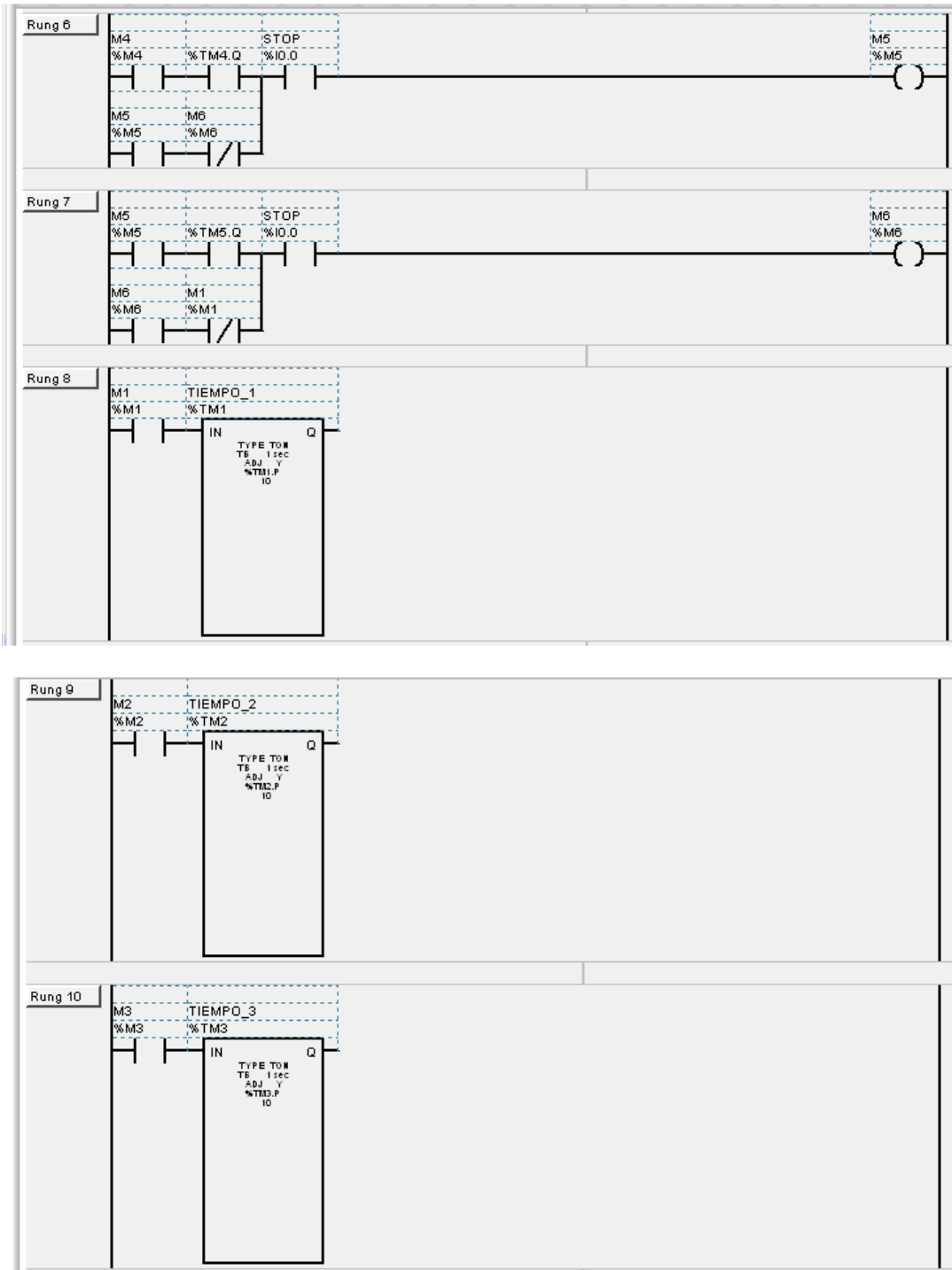
	Nombre del recurso	Tipo	Etiqueta de material	Iniciales	Grupo	Capacidad máxima	Tasa estándar	Tasa horas extra	Costo/Usd	Acumular
1	Computadora	Material		C			\$ 0,00		\$ 0,50	Prorratio
2	Impresora	Material		I			\$ 0,00		\$ 0,20	Prorratio
3	Hojas	Material		H			\$ 0,00		\$ 0,04	Prorratio
4	tinta	Material		t			\$ 0,00		\$ 0,15	Prorratio
5	Internet	Material		I			\$ 0,00		\$ 0,80	Prorratio
6	Telefono	Material		T			\$ 0,00		\$ 5,00	Prorratio
7	Luz	Material		L			\$ 0,00		\$ 2,00	Prorratio
8	Estructura de aluminio	Trabajo		E		100%	\$ 0,00/hora	\$ 0,00/hora	\$ 1.500,00	Prorratio
9	Electro Válvulas	Trabajo		E		100%	\$ 0,00/hora	\$ 0,00/hora	\$ 450,00	Prorratio
10	Sensores	Trabajo		S		100%	\$ 0,00/hora	\$ 0,00/hora	\$ 1.200,00	Prorratio
11	Bombas Neumáticas	Trabajo		B		100%	\$ 0,00/hora	\$ 0,00/hora	\$ 600,00	Prorratio

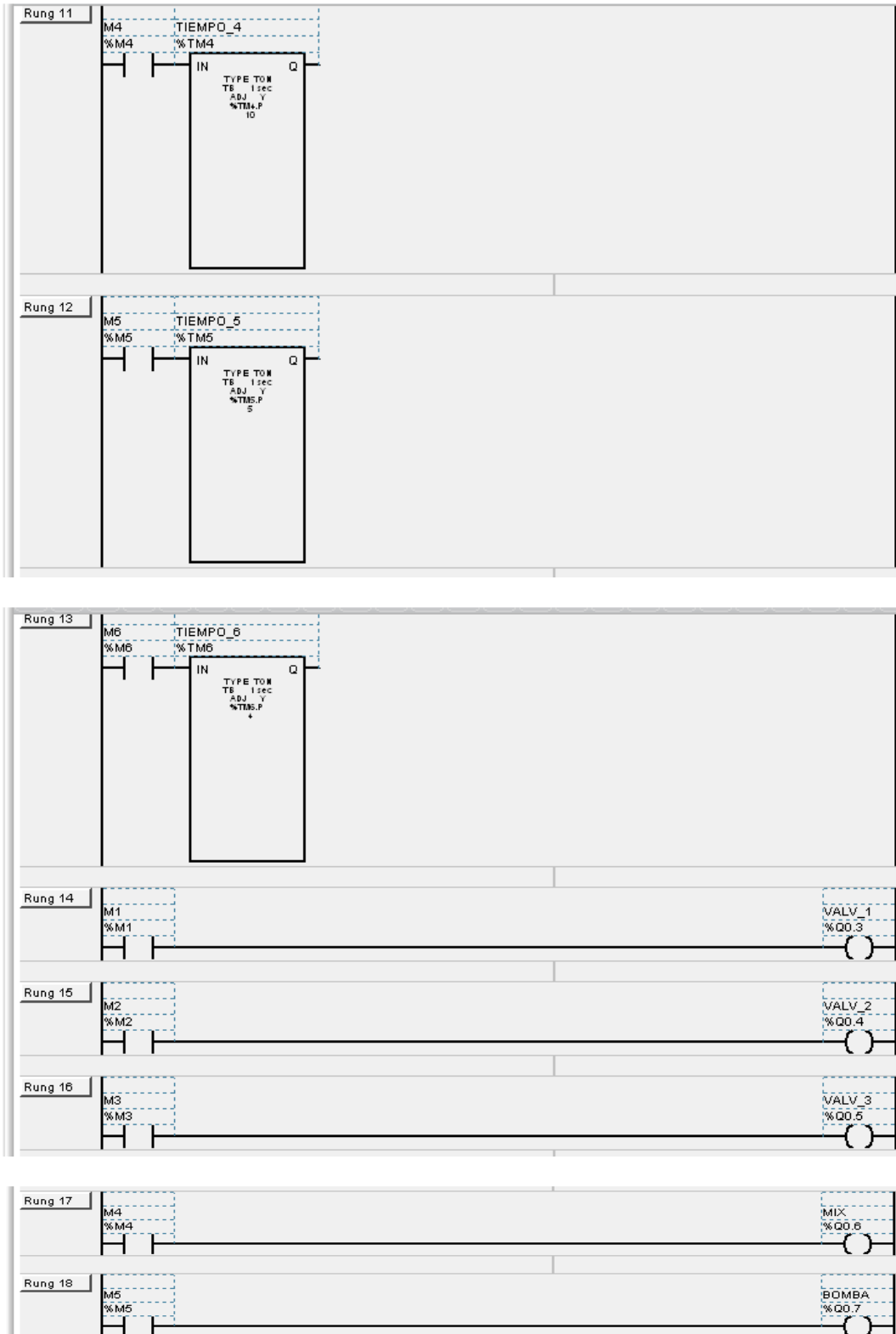
	Nombre de tarea	Duración	Comienzo	Fin	Predecesor	Nombres de los recursos
1	Análisis, estudio e investigación del tema a desarrollar	15 días	vie 07/08/09	jue 27/08/09		Internet[1],Computadora[1]
2	Recopilación y almacenamiento de datos (primarios y/o secundarios), basados en la investigación	15 días	vie 28/08/09	jue 17/09/09	1	Internet[1],Computadora[1]
3	Planteamiento y diseño del módulo de una mezcladora automática.	15 días	jue 08/10/09	mié 28/10/09	2	Computadora[1],Internet[1]
4	Ejecución (construcción y ensamblaje) de una mezcladora automática.	30 días	lun 30/11/09	vie 08/01/10	3	Estructura de aluminio ,Ele
5	Programación para los procesos a ejecutarse del módulo de una mezcladora automática.	35 días	lun 18/01/10	vie 05/03/10	4	Computadora[1],Internet[1]
6	Supervisar el sistema de acuerdo a los requerimientos establecidos.	8 días	mié 10/03/10	vie 19/03/10	5	Computadora[1]
7	Pruebas de funcionamiento y mantenimiento	15 días	lun 22/03/10	vie 09/04/10	6	Computadora[1],PLC
8	Diseño e implementación de la Interfaz de usuario para el control de la producción	30 días	lun 12/04/10	vie 21/05/10	7	Computadora[1],Internet[1]
9	Diseño e implementación de la Base de Datos	15 días	lun 24/05/10	vie 11/06/10	8	Computadora[1],Internet[1]
10	Pruebas y corrección del módulo final.	15 días	lun 14/06/10	vie 02/07/10	9	Computadora[1],PLC
11	Determinación de Resultados	10 días	lun 05/07/10	vie 16/07/10	10	Computadora[1],PLC
12	Corrección del Trabajo Final	24 días	lun 19/07/10	jue 19/08/10	11	Computadora[1],Impresora
13	Defensa Final.	10 días?	vie 20/08/10	jue 02/09/10	12	Computadora[1]
14	Corrección y presentación final.	5 días?	vie 03/09/10	jue 09/09/10	13	Computadora[1],Impres



### ANEXO 2 PROGRAMA LADDER







## **ANEXO 3**

### **MANUAL DE USUARIO**

#### **1. Presentación**

El presente manual de usuario de la aplicación de: “Implementación de un Sistema de Mezclado de Líquidos, Caso Práctico: Laboratorio de Automatización Industrial de la EIS”; contiene las informaciones técnicas pertinentes que permiten al usuario instalar y trabajar en el módulo de manera correcta.

Una condición preliminar a cumplir por el personal que trabaje en el sistema de mezclado de líquidos es que disponga de conocimientos técnicos, ya que la vida útil del módulo, su rendimiento y disponibilidad de operación dependen en alto grado de la correcta ejecución de los trabajos de limpieza, del manejo y mantenimiento del mismo.

#### **2. Introducción**

En este documento se describirá el funcionamiento en si de la aplicación, el cual proporcionará al usuario facilidad al encontrar detallado cada uno de los pasos que se siguió para realizar la instalación, configuración, implementación, manejo y ejecución de la aplicación en forma correcta.

Entre las opciones constará lo siguiente:

- ✓ Instalación de LabView



- OPC Server
  - Configuración del OPC Server(Creación del Modbus)
  - Database Connectivity Toolkit
- ✓ Instalación del software TwidoSuite

La presente aplicación está destinada a los estudiantes que cursan la materia optativa de Automatización Industrial de la EIS, para que tengan mayor accesibilidad a dispositivos industriales. Se realizará un control automático del mezclado de líquido en el software LabView en la cual se simulará secciones de una fábrica.

El mezclador de líquidos puede ser usado para tareas de programas prácticos como un solo mecanismo o en conexión con el módulo de la envasadora ya que pueden trabajar en conjunto para simular a una verdadera empresa industrial farmacéutica u otro que utilice este tipo de sistema; para lo cual se requiere un PLC de 24 entradas y 16 salidas para la operación.

### **3. Objetivos del sistema**

- Programar el proceso de control del módulo bajo el lenguaje de programación seleccionado y diseñar la interfaz de usuario para controlar el proceso de mezclado de líquidos.
- Diseñar e implementar una base de datos que permita registrar datos para la inspección del proceso.

- Construir el módulo final en el laboratorio de Automatización industrial para obtener el mezclado de líquidos.

#### **4. Capacidades del sistema**

La Implementación del sistema de mezclado de líquidos tendrá la capacidad de realizar la mezcla de manera homogénea y equitativa en un tiempo no mas de 1 minuto ya que el módulo está compuesto por 4 tanques cada una de ellas cuenta con un sensor para realizar el control manual y automático; es así que el control manual se lo realiza por medio de las botoneras que está en un tablero y el control automático está implementado en el lenguaje gráfico LabView que es fácil de manejar el sistema. Además nuestro sistema tiene la posibilidad de trabajar en conjunto con otro sistema que es la envasadora automática y demostrar que nuestro sistema es escalable y de fácil manejo.

#### **5. Requisitos mínimos indispensables**

##### **5.1. Hardware**

- **Características del PC**

PC con procesador Pentium 4 de 3 Ghz o superior

2 Gb de memoria RAM o superior.

Disco duro con 80 Gb de espacio libre.

- **Características del PLC TWDLCAE40DRF**



- ✓ 24 entradas digitales, 14 de relé y 2 salidas de transistor
- ✓ 2 potenciómetros analógicos
- ✓ 1 puerto serie integrado
- ✓ 1 slot para un puerto serie adicional
- ✓ RTC integrado
- ✓ Compartimiento de batería para batería externa reemplazable por el usuario
- ✓ Admite hasta 7 módulos de ampliación de E/S.
- ✓ Admite hasta dos módulos de interface del bus AS-Interface V2
- ✓ Admite un módulo master de interface del bus de campo CANopen:
  - ✓ Admite un cartucho de memoria opcional (de 32 ó 64 KB)
  - ✓ Admite un módulo de monitor de operación opcional
  - ✓ Puerto RJ45 de interface Ethernet integrado

[electric.com/85257578007E5C8A/all/4E9CB1347509F7F8882575780062449B/\\$File/31004123k01001.pdf](http://electric.com/85257578007E5C8A/all/4E9CB1347509F7F8882575780062449B/$File/31004123k01001.pdf)

- **Cable de comunicación de PC al PLC**

## 5.2. Software

- Microsoft Windows XP o superior.
- Labview Profesional 7.1 o superior.
- TwidoSuite

## 6. Instalación

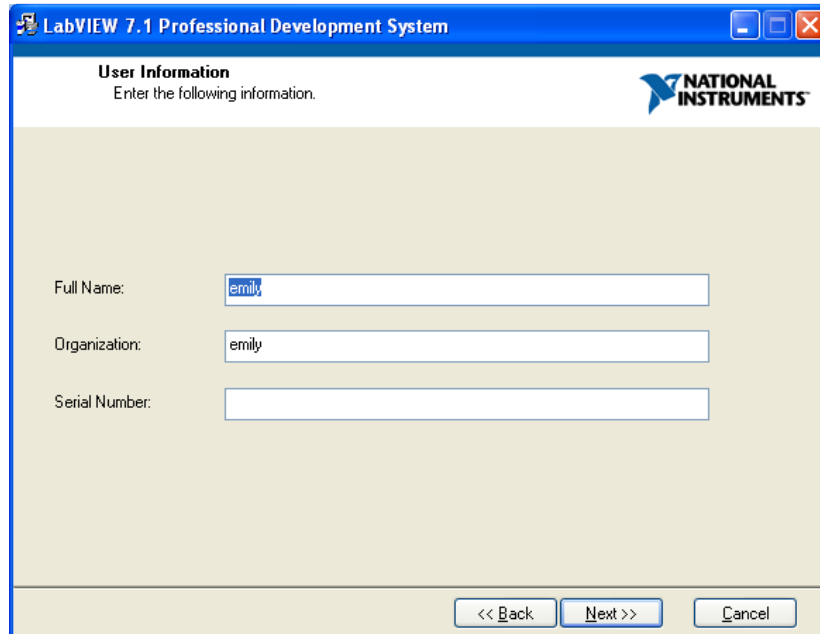
### 6.1. LabView

Es un lenguaje de programación de alto nivel, de tipo gráfico, y enfocado al uso en instrumentación, puede ser usado para elaborar cualquier algoritmo que se desee, en cualquier aplicación, como en análisis, telemática, juegos, manejo de textos, etc.

- a. Se ejecuta el ícono setup y se presenta la siguiente pantalla:

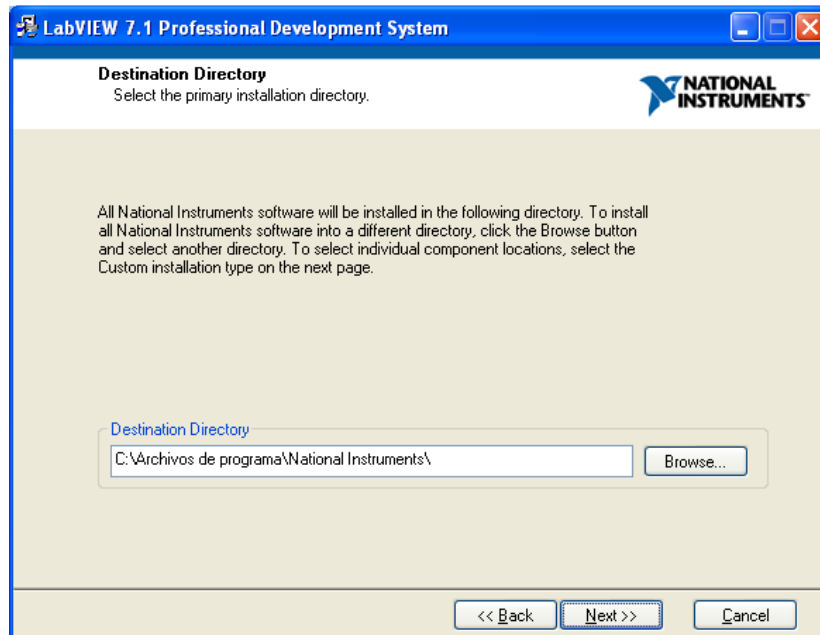


- b. Luego de terminar de cargar todos los archivos se presenta la siguiente pantalla:



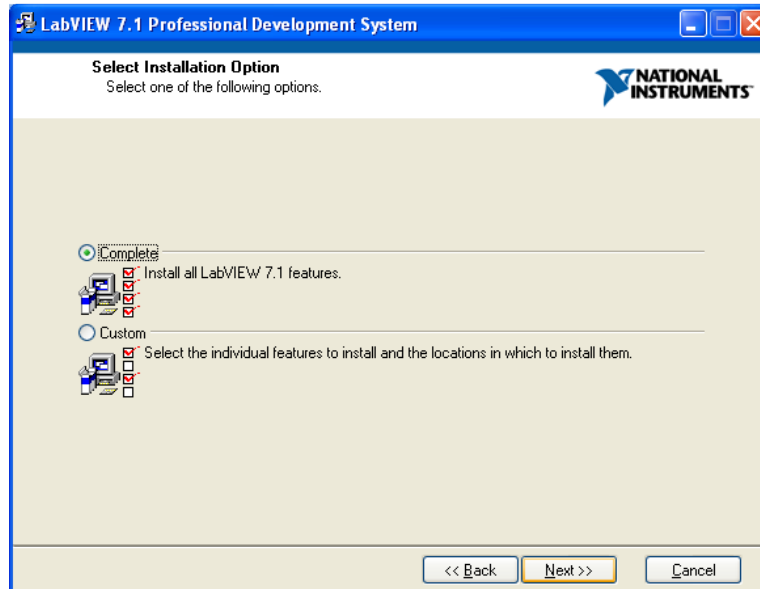
The screenshot shows a dialog box titled "LabVIEW 7.1 Professional Development System" with the "User Information" section. The text "Enter the following information." is displayed. The National Instruments logo is in the top right corner. There are three input fields: "Full Name:" with the text "emily", "Organization:" with the text "emily", and "Serial Number:" which is empty. At the bottom, there are three buttons: "<< Back", "Next >>", and "Cancel".

- c. Luego de llenar los datos del usuario se pasa a la siguiente pantalla haciendo clic en Next como la que sigue:

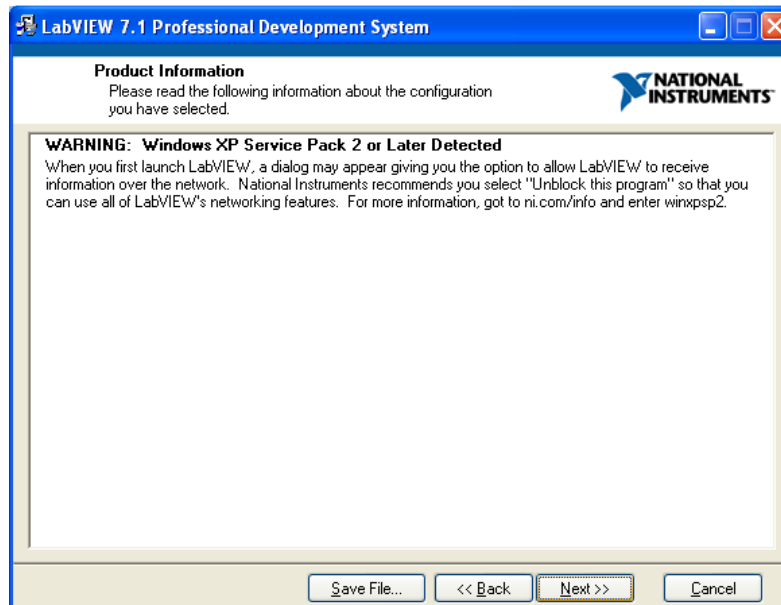


The screenshot shows a dialog box titled "LabVIEW 7.1 Professional Development System" with the "Destination Directory" section. The text "Select the primary installation directory." is displayed. The National Instruments logo is in the top right corner. Below the text, there is a paragraph: "All National Instruments software will be installed in the following directory. To install all National Instruments software into a different directory, click the Browse button and select another directory. To select individual component locations, select the Custom installation type on the next page." Below this text is a text box containing "C:\Archivos de programa\National Instruments\" and a "Browse..." button. At the bottom, there are three buttons: "<< Back", "Next >>", and "Cancel".

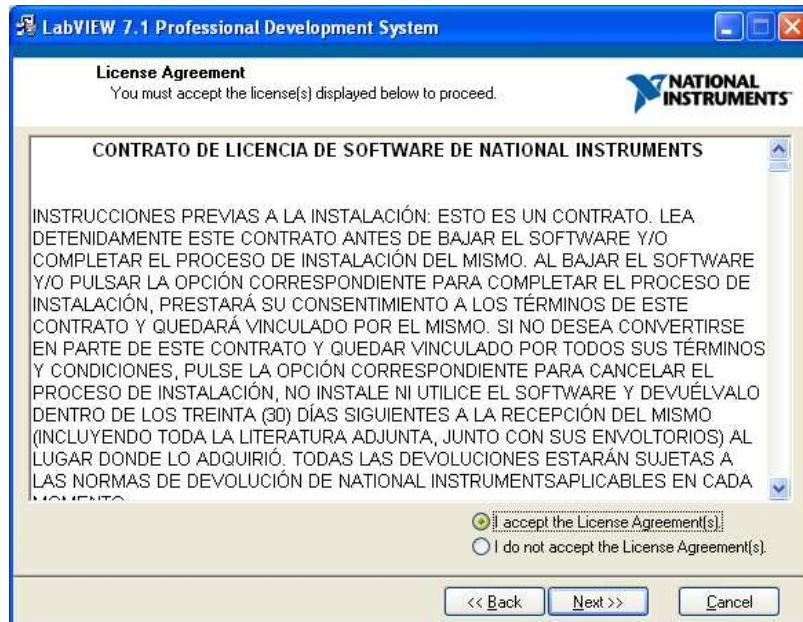
- d. Después de direccionar el lugar de instalación se hace clic en Next y se presenta la siguiente pantalla:



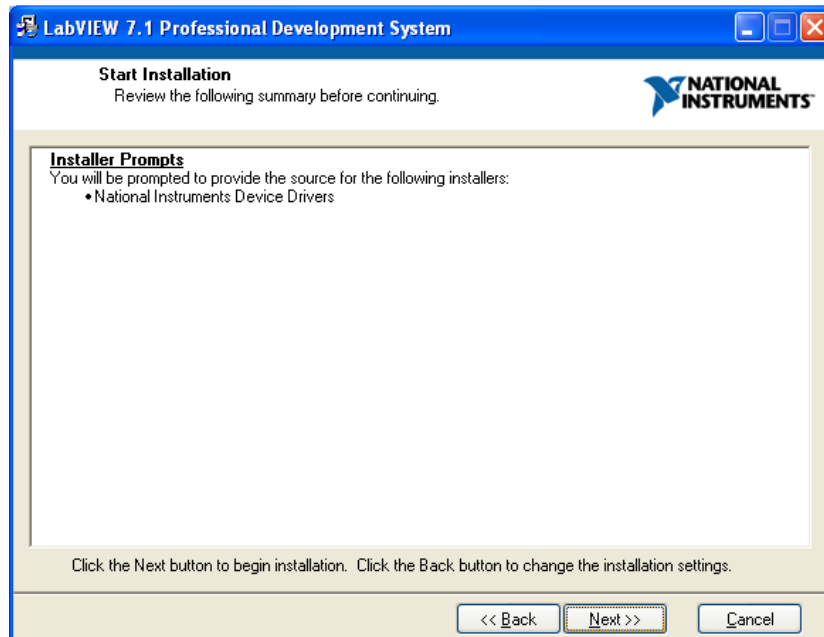
- e. Después de seleccionar el tipo de instalación y haciendo clic en siguiente se muestra la siguiente pantalla con la información del producto:



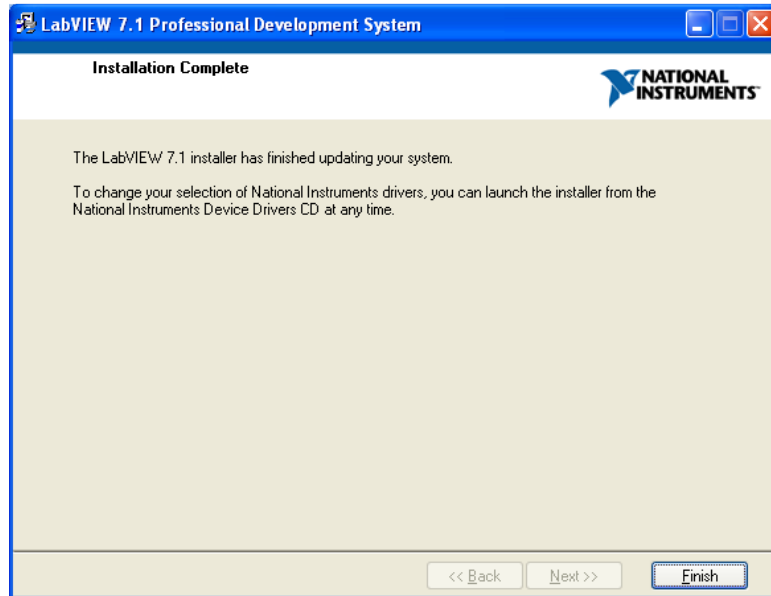
- f. Luego de leer la información del producto se presenta la siguiente pantalla en la cual está la licencia del producto nos informamos de ello y aceptamos el contrato de la licencia y finalmente hacemos clic en Next:



- g. En esta pantalla se inicia la instalación del software Labview:



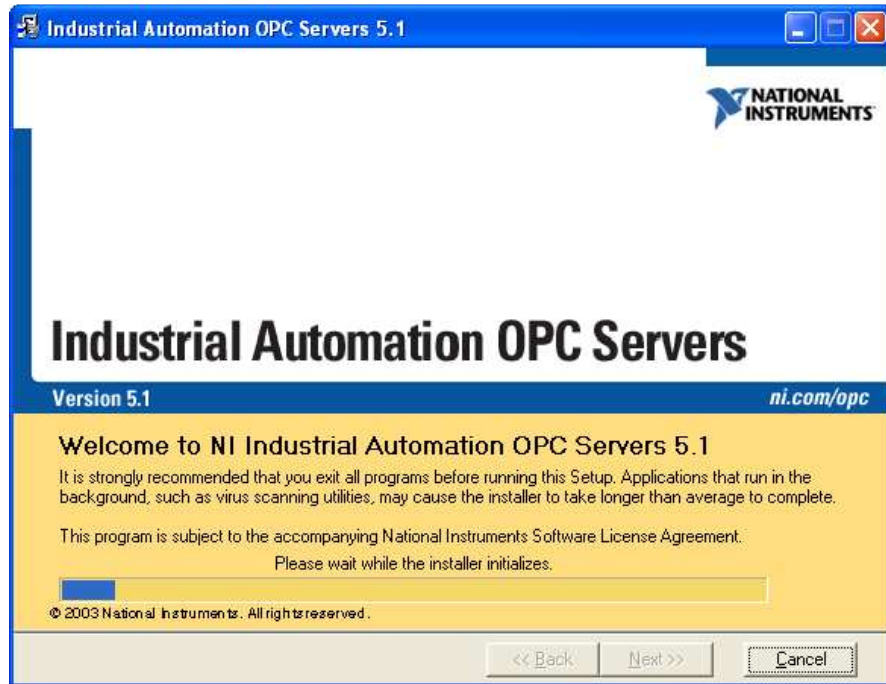
- h. Después de hacer clic en Next se presenta la pantalla de finalización de la instalación:



### 6.1.1. Instalación del OPC Server

- a. Al ejecutar el ícono de instalación del OPC Server se carga la siguiente ventana de bienvenida como se muestra a continuación:





- b. Luego de hacer clic en Next se presenta la siguiente pantalla en la cual se debe llenar la información solicitada del usuario:

Industrial Automation OPC Servers 5.1

NATIONAL INSTRUMENTS

## User Information

Enter the following information.

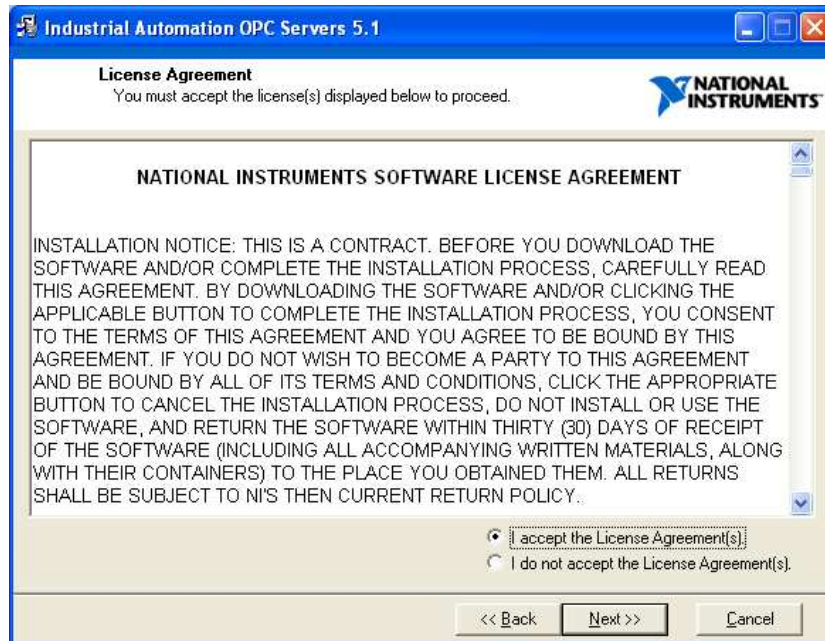
Full Name:

Organization:

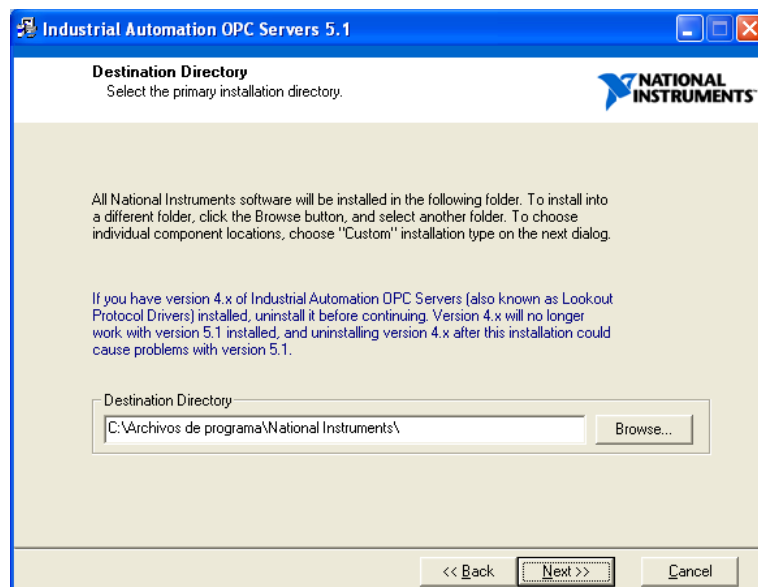
Serial Number:

<< Back Next >> Cancel

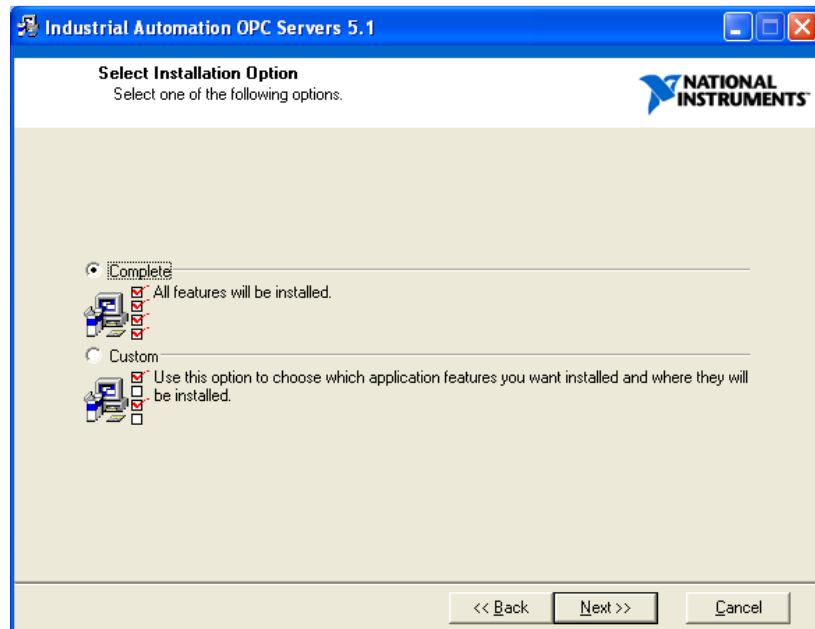
- c. Después de hacer clic en Next se muestra la siguiente ventana en la cual se encuentra el contrato de licencia la misma que la leemos, aceptamos y hacemos clic en Next:



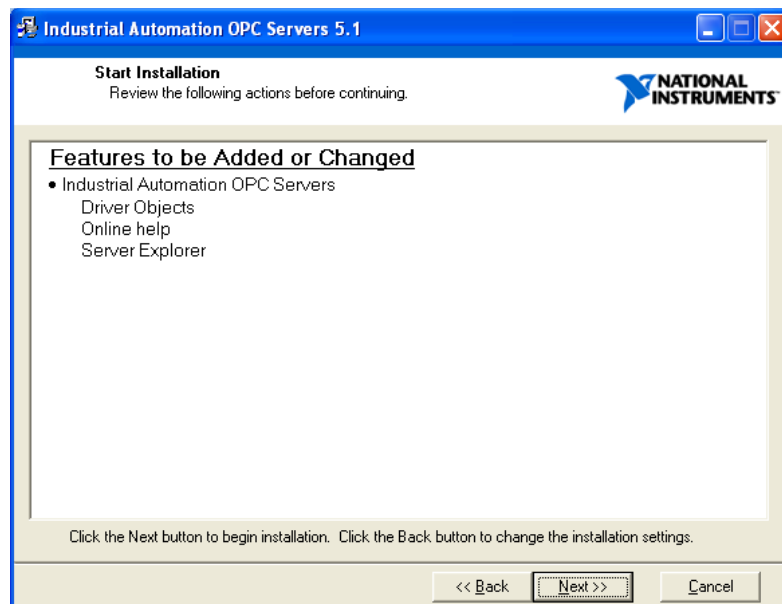
- d. Se muestra la ventana para escoger el destino del directorio en la cual de va instalar:



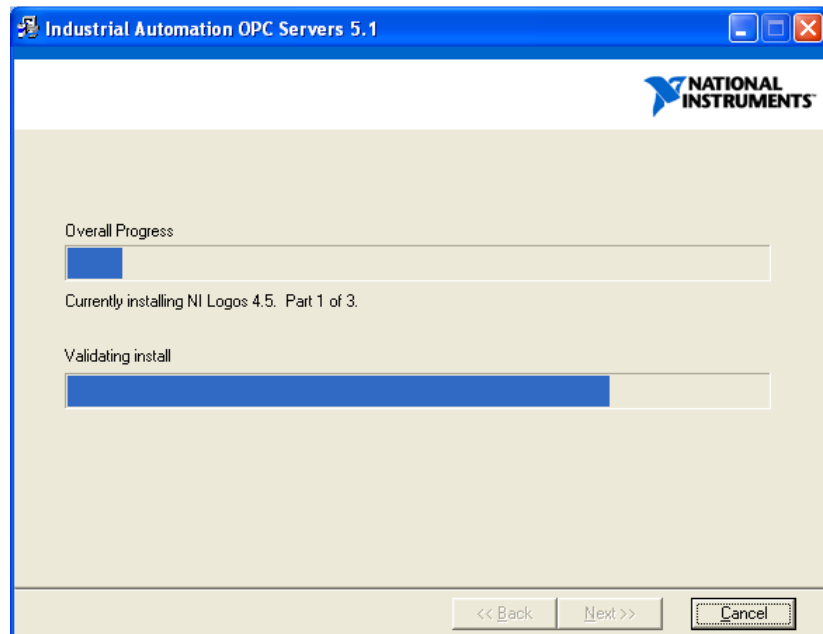
- e. La siguiente ventana presenta el tipo de instalación que se va realizar y hacemos Clic en Next:



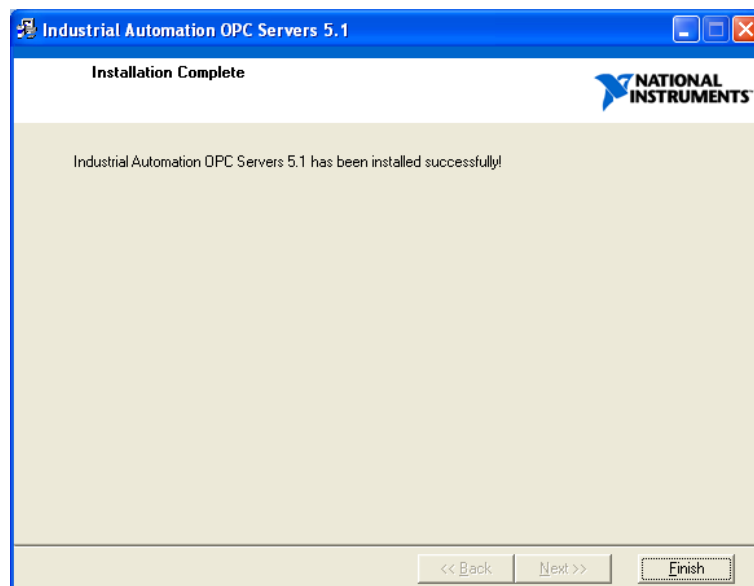
- f. En esta ventana se inicia realmente la instalación del OPC Server en la cual se hace clic en Next para pasar a la ventana siguiente:



- g. Se inicia la instalación del OPC Servers 5.1



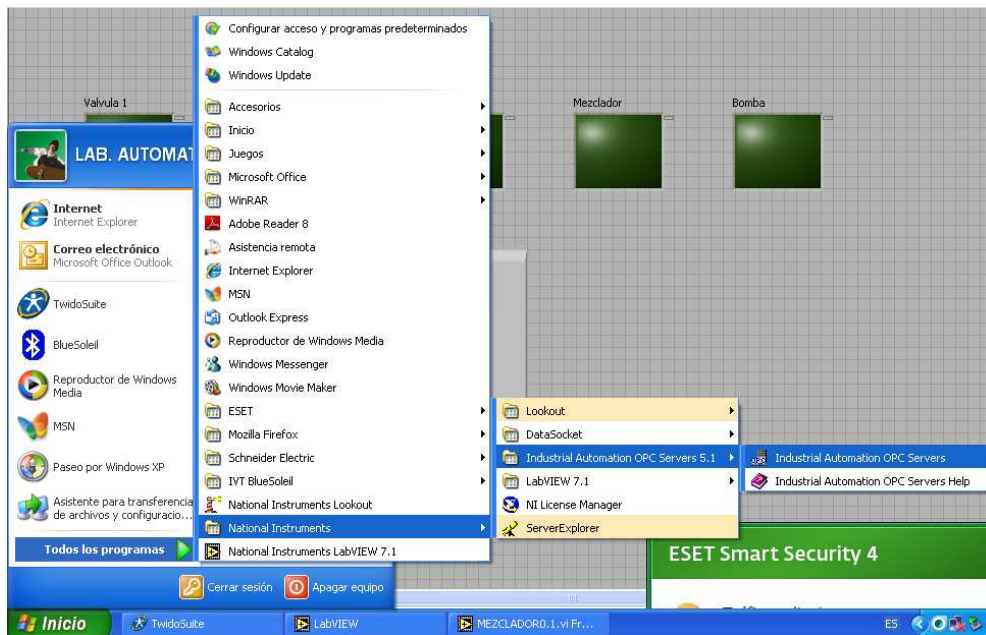
h. Se ha finalizado la instalación del OPC Server.



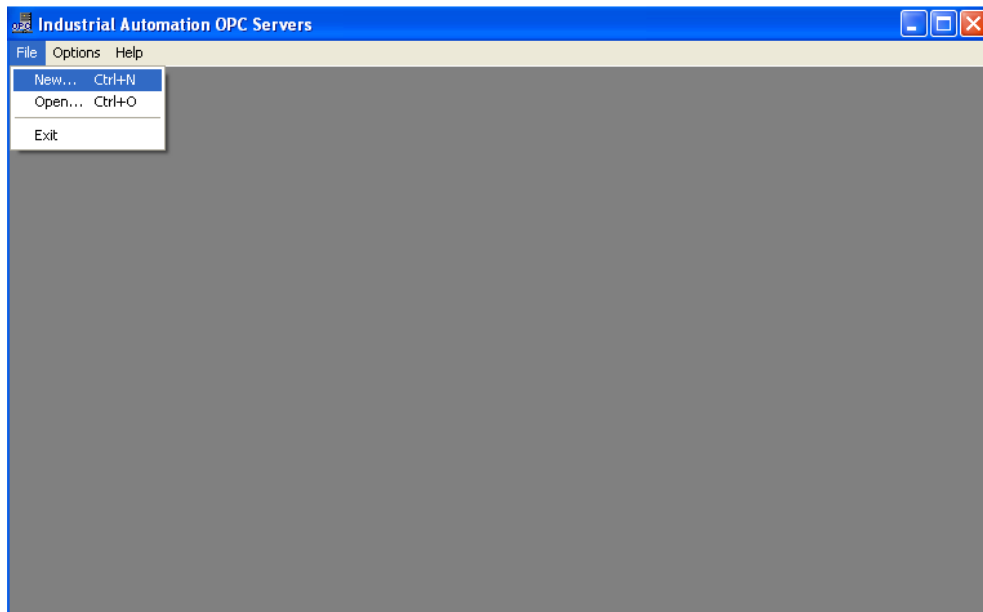
### 6.1.2. Configuración del OPC Server(creación del modbus serial)

- a. Conecte el PLC o dispositivo que actué como maestro en el puerto COM de donde tenga instalado el OPC Server

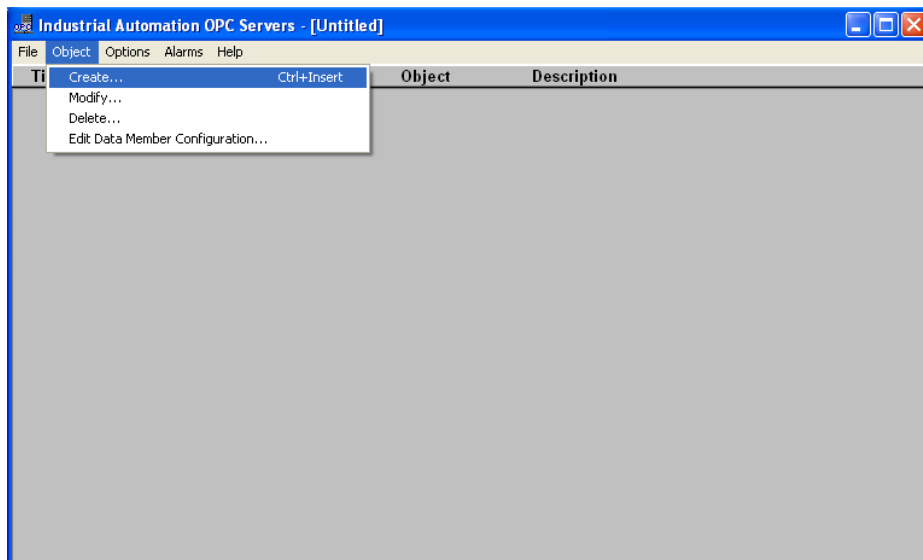
- b. Hacemos clic en Inicio/Todos los Programas /National Instruments/Industrial Automation OPC Server 5.1 / Industrial Automation OPC Server como se muestra a continuación:



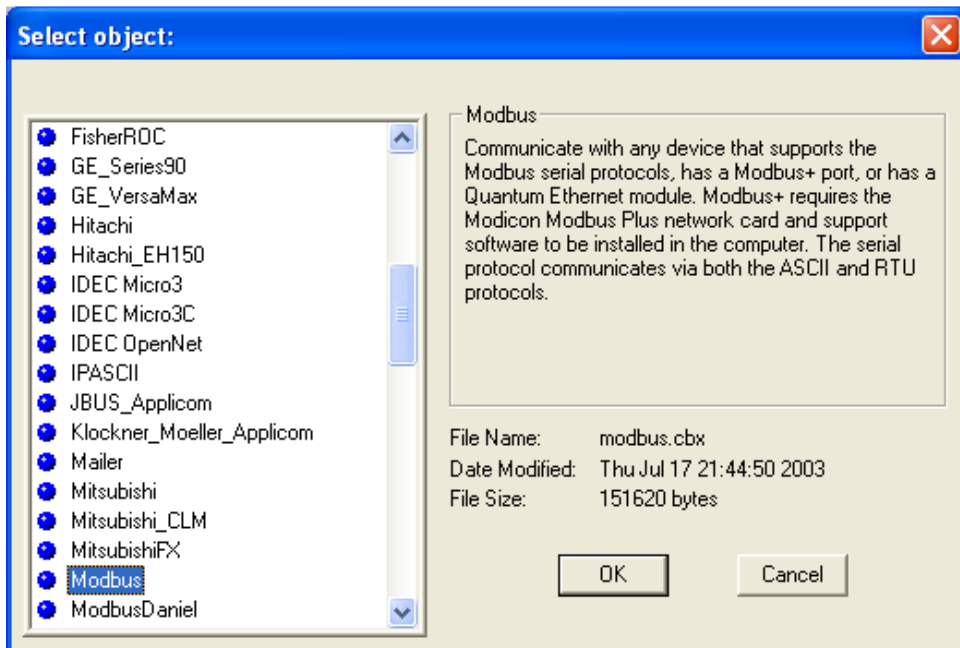
- c. Se abre la ventana en la cual se crea el modbus, hacemos clic en File/New como se muestra en la siguiente pantalla:



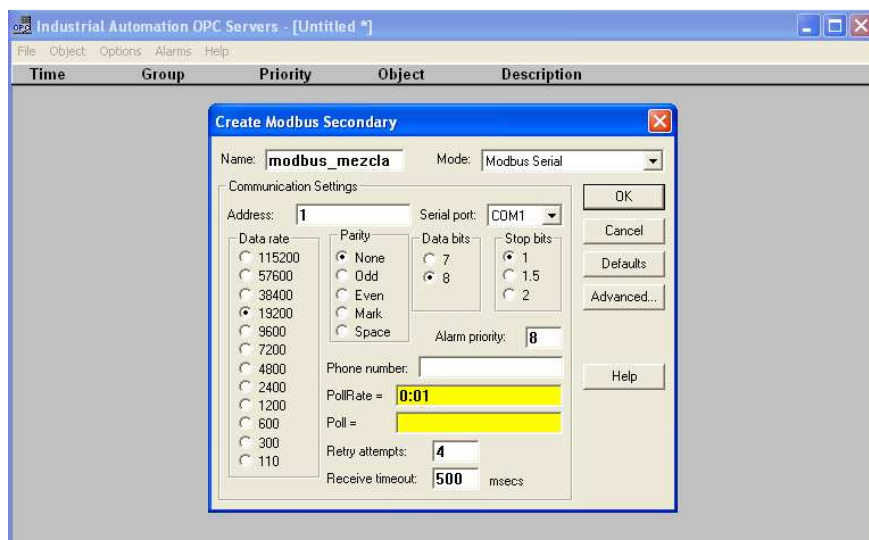
d. Hacemos clic en Object/ Create como se presenta a continuación:



e. Luego de ello se muestra la siguiente ventana en la cual nos da la opción de seleccionar el objeto y hacemos clic en OK como se muestra a continuación:



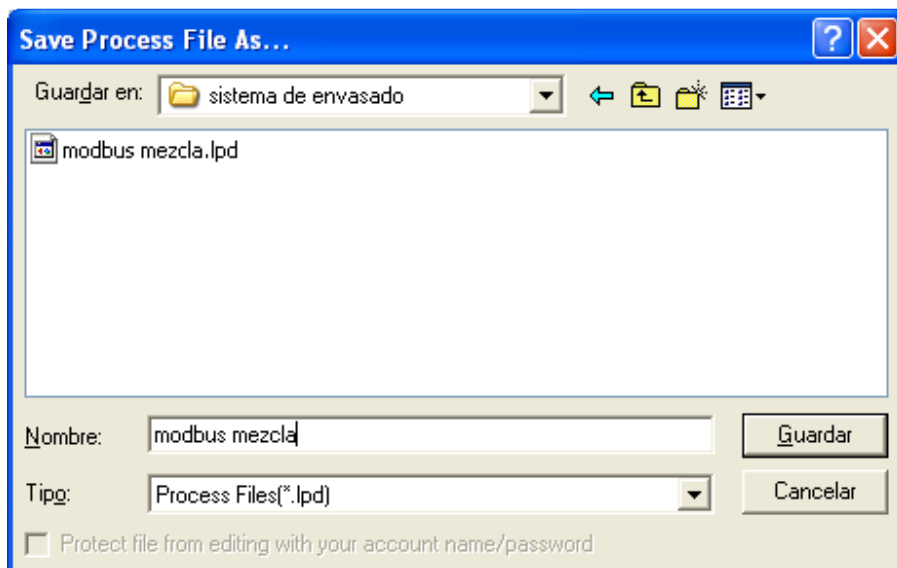
- f. Aquí se presenta la ventana en la cual se da un nombre al modbus , el modo y la velocidad con la que se va conectar y finalmente hacemos clic en OK como se muestra a continuación:



- g. Luego de ello cerramos la ventana OPC Servers y se nos presenta una ventana en la cual nos pregunta si queremos guardar o no el modbus creado como se muestra a continuación:



- Se presenta la ventana en la cual escribimos el nombre del modbus como se muestra a continuación:



- h. Luego de guardar el modbus se presenta otra ventana en la cual nos pregunta si con este proceso se va iniciar el modbus creado como se muestra a continuación:



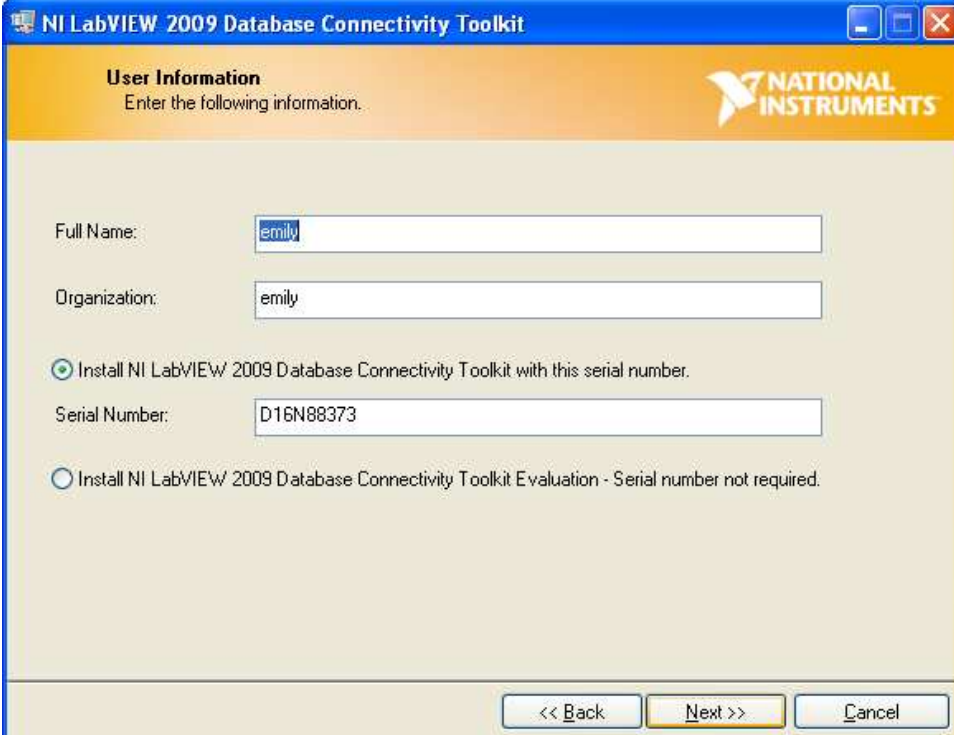


### 6.1.3. Instalación del Componente Database Connectivity Toolkit

- a. Al ejecutar la instalación de este componente se nos presenta la siguiente pantalla en la cual hacemos clic en Next.



- b. Al hacer Next se nos presenta la siguiente pantalla en la cual llenamos todos los datos solicitados y hacemos clic en Next.

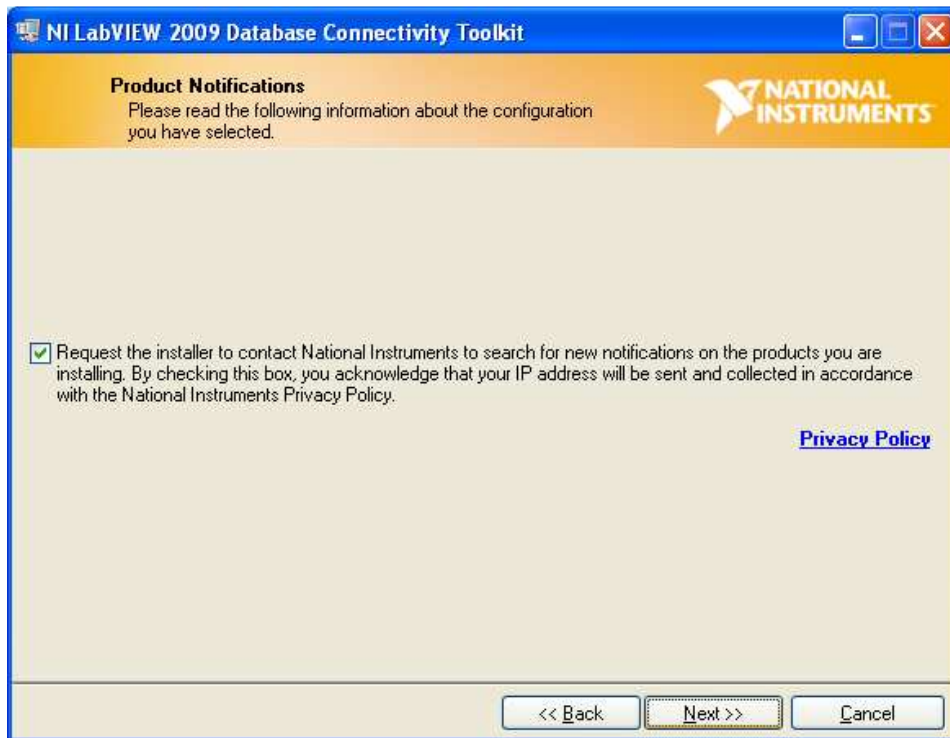


The image shows a Windows-style dialog box titled "NI LabVIEW 2009 Database Connectivity Toolkit". The dialog has a blue header bar with the title and standard window controls (minimize, maximize, close). Below the header is an orange banner with the text "User Information" and "Enter the following information." on the left, and the National Instruments logo on the right. The main area is light beige and contains the following fields and options:

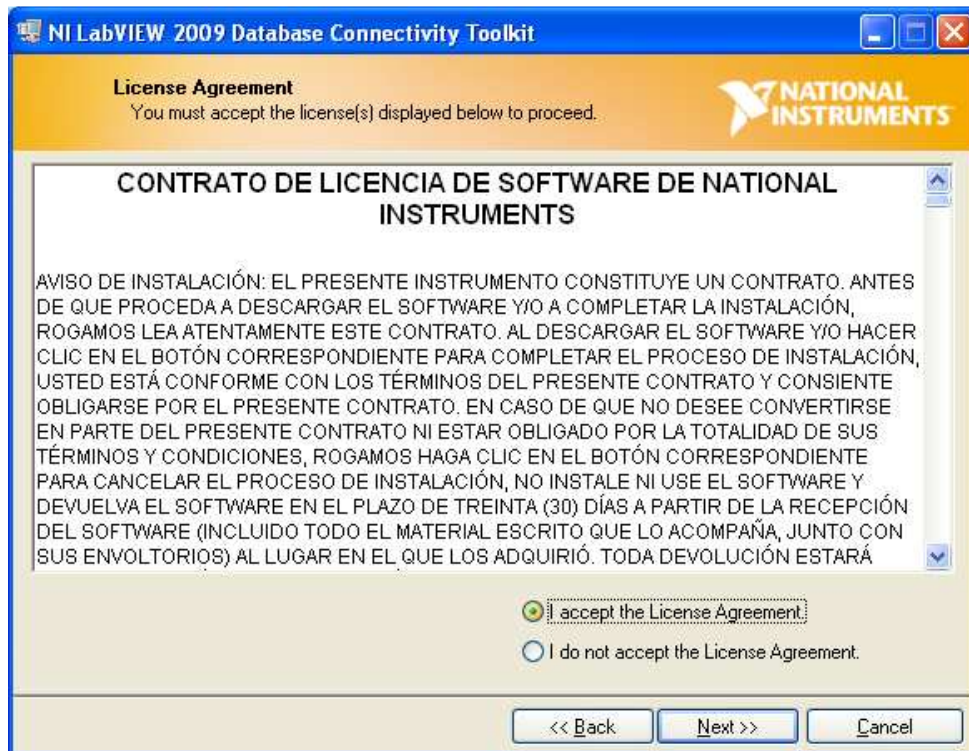
- "Full Name:" with a text box containing "emily".
- "Organization:" with a text box containing "emily".
- A radio button selected for "Install NI LabVIEW 2009 Database Connectivity Toolkit with this serial number.".
- "Serial Number:" with a text box containing "D16N88373".
- An unselected radio button for "Install NI LabVIEW 2009 Database Connectivity Toolkit Evaluation - Serial number not required.".

At the bottom of the dialog are three buttons: "<< Back", "Next >>" (highlighted in yellow), and "Cancel".

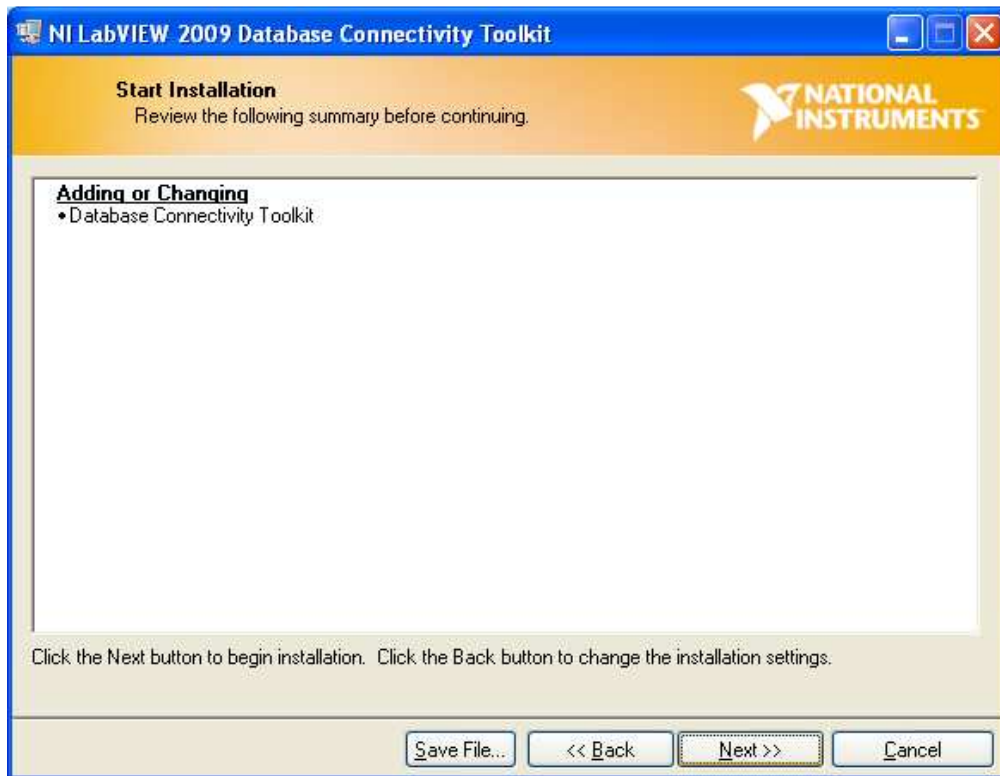
- c. Se presenta la siguiente pantalla en la cual nos informa que se realizó la configuración para la instalación.



- d. Se presenta la información del contrato de licencia la cual aceptamos y hacemos clic en siguiente.



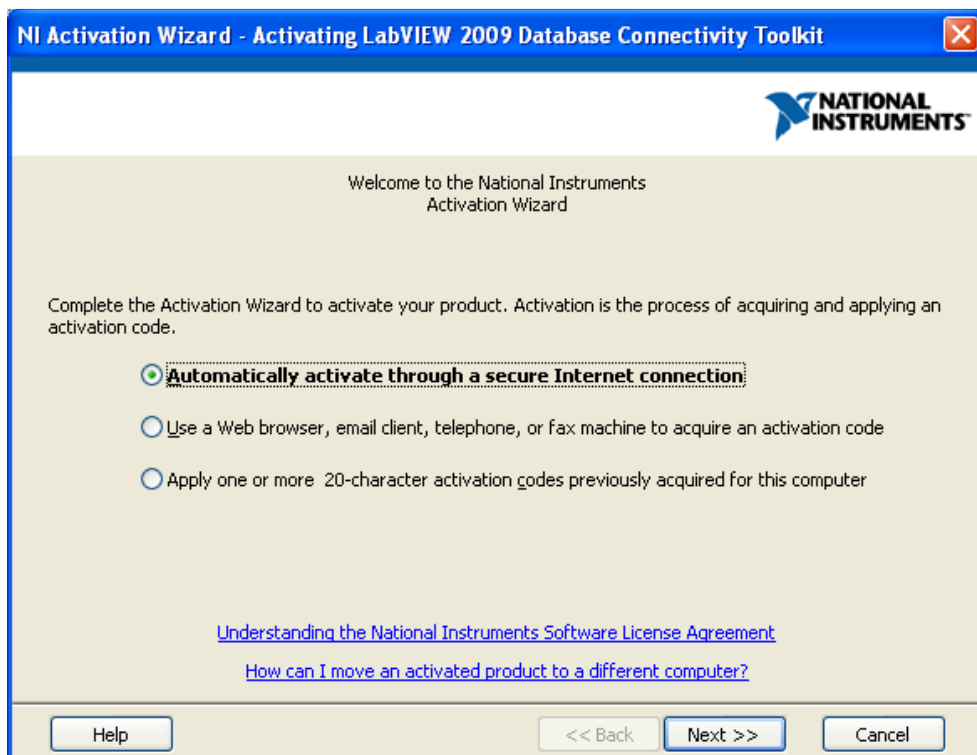
- e. La siguiente pantalla indica que ya se inicia la instalación del componente y hacemos clic en Next.



- f. La siguiente pantalla nos indica que ya se instaló el componente satisfactoriamente y hacemos clic en Next.



g. Se activa el componente instalado y hacemos clic en Next.




- h. En la siguiente pantalla está el serial que utiliza para que empiece a funcionar el componente y hacemos clic en Next.



- i. La siguiente pantalla permite llenar los datos para la activación.

NI Activation Wizard - Activating LabVIEW 2009 Database Connectivity Toolkit



The following information is required to activate your products.

First Name	Last Name	Organization
Mariana	Morocho	emily

Register my product to ensure access to my service benefits from National Instruments, including technical support, software upgrades and updates.

Country: Ecuador

Address: Riobamba

Address (cont.): Riobamba


City	State	Postal Code
Riobamba	24 de Mayo	123

Email	Phone	Fax
mercedess_1984@hotmail.com	032620522	

[National Instruments Product Activation Privacy Statement](#)

Help << Back Next >> Cancel

NI Activation Wizard - Activating LabVIEW 2009 Database Connectivity Toolkit



Send me an email confirmation of this activation.

Email: mercedess\_1984@hotmail.com

Help << Back Next >> Cancel

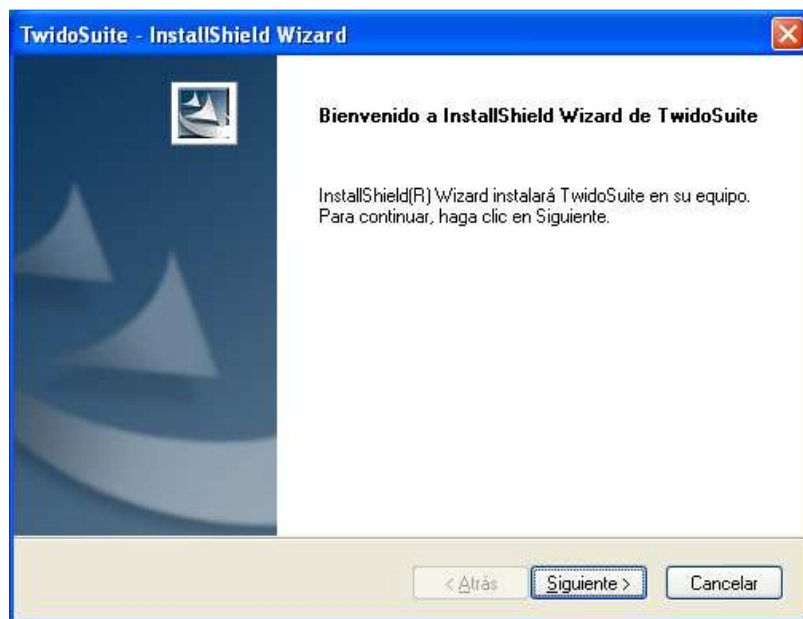


## 6.2. Twido Suite

- a. Para comenzar la instalación se procede a abrir el archivo ejecutable.



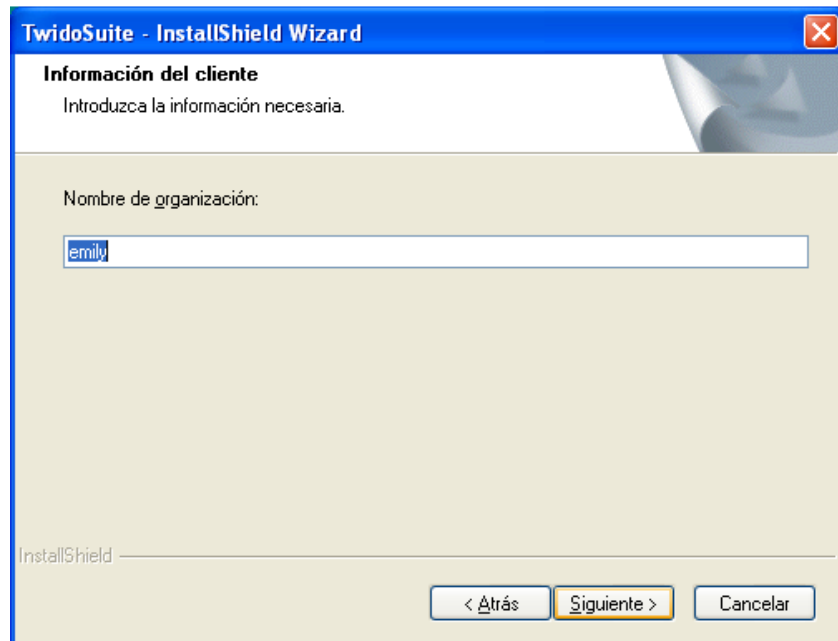
- b. Se presenta la ventana de bienvenida a la instalación del TwidoSuite como se muestra a continuación:



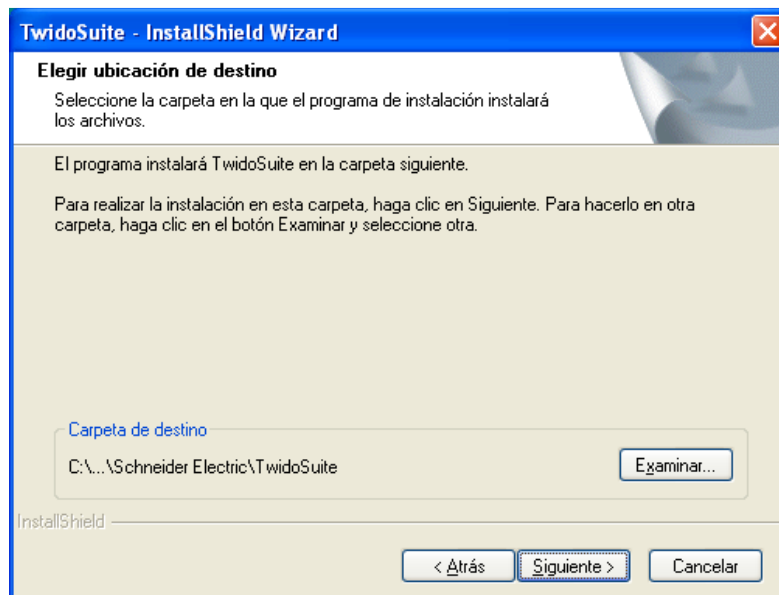
- c. Aparecerá la ventana flotante con la información de los derechos usuario de la licencia del software TwidoSuite, la aceptamos pulsando el botón **“Accept”** como se muestra a continuación:



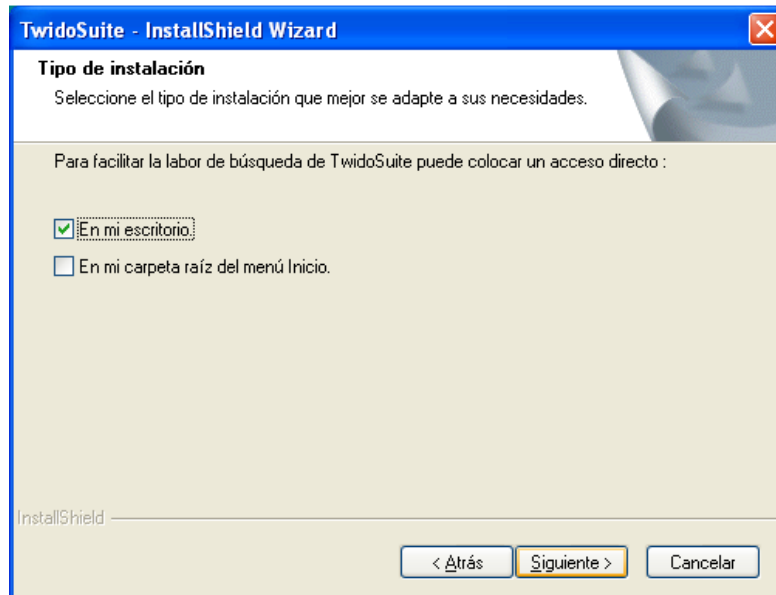
- d. Luego de hacer clic en siguiente en la ventana anterior se presenta la siguiente ventana con la información del cliente:



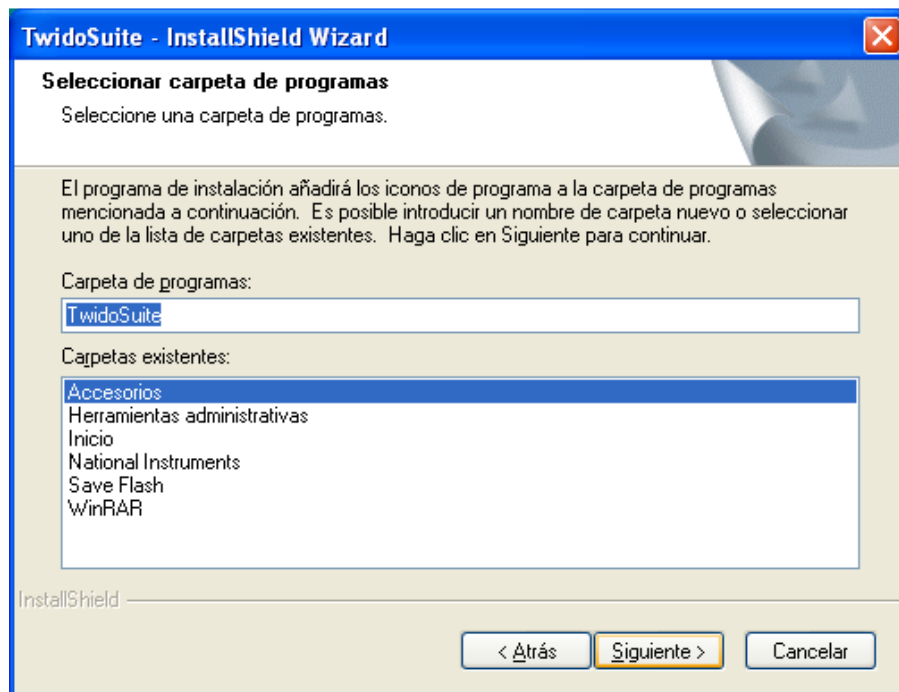
- e. Luego de hacer clic en siguiente se presenta la ventana para elegir la ubicación de destino en la cual se instalará como se muestra a continuación:



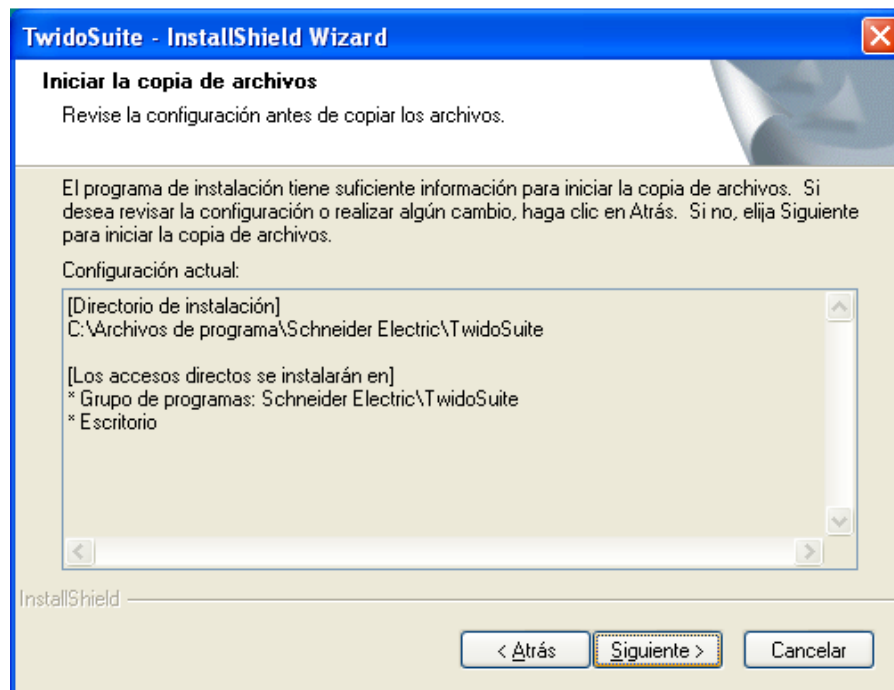
- f. Después de hacer clic en siguiente se presenta la ventana del tipo de instalación para nuestro mejor uso y hacemos clic en siguiente:



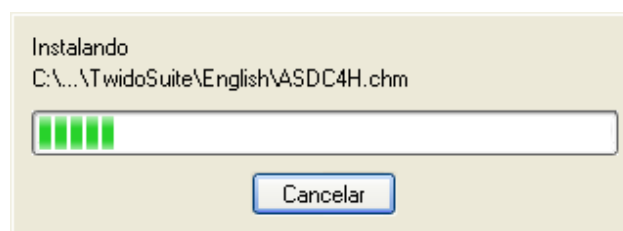
- g. Seleccionamos la carpeta de programas en la cual de añadirá el TwidoSuite y hacemos clic en siguiente como se muestra a continuación:

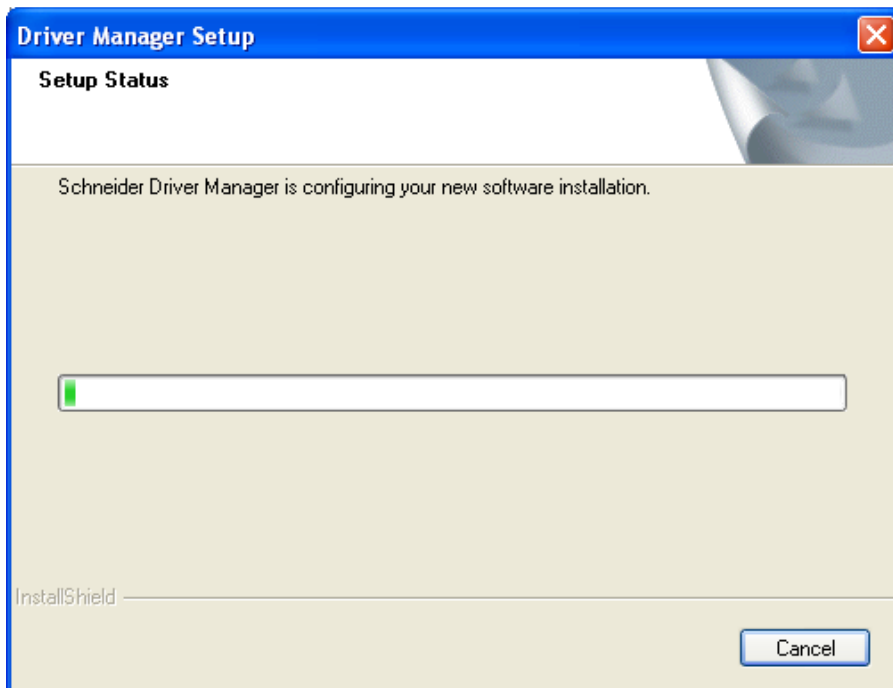
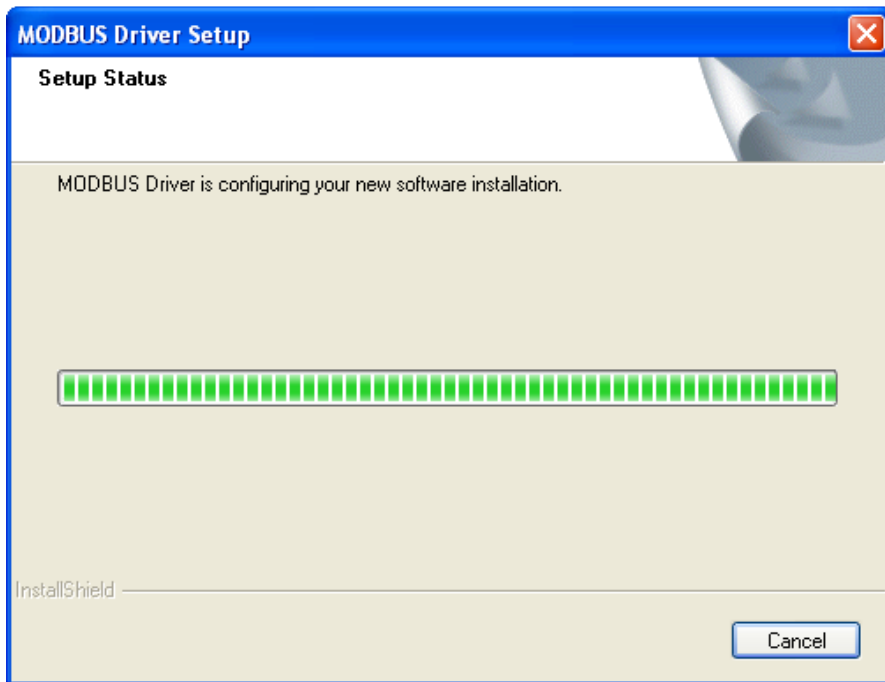


- h. Muestra la información de Inicio de copia de los archivos de instalación y hacemos clic en siguiente como se muestra:

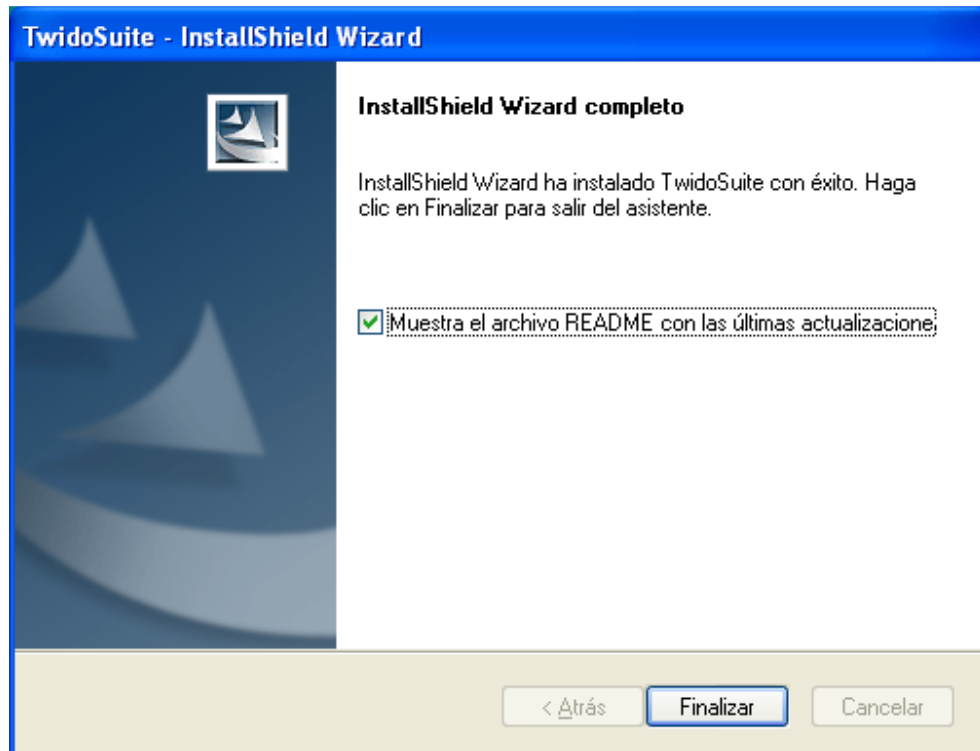


- i. Se muestra las siguientes ventanas en la cual se instala el TwidoSuite en si, el driver modbus y el driver de manejo como se muestra a continuación





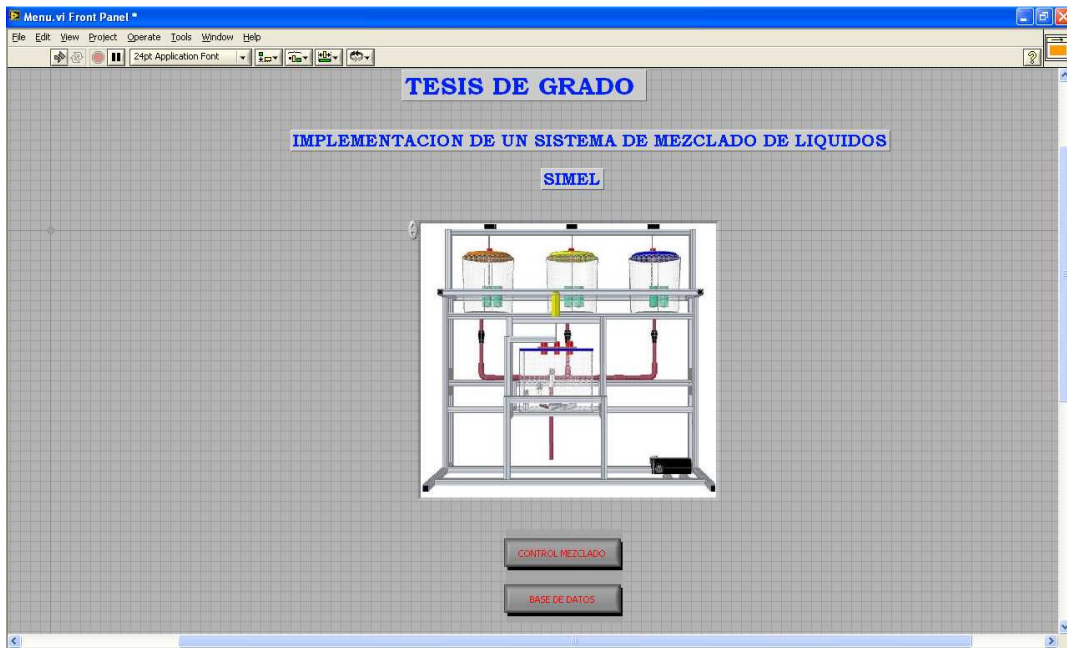
j. Se finaliza la instalación del TwidoSuite.



## 7. Cómo utilizar el programa

Para iniciar a utilizar el programa implementado para el control es necesario tener conocimientos básicos del lenguaje LabView y TwidoSuite.





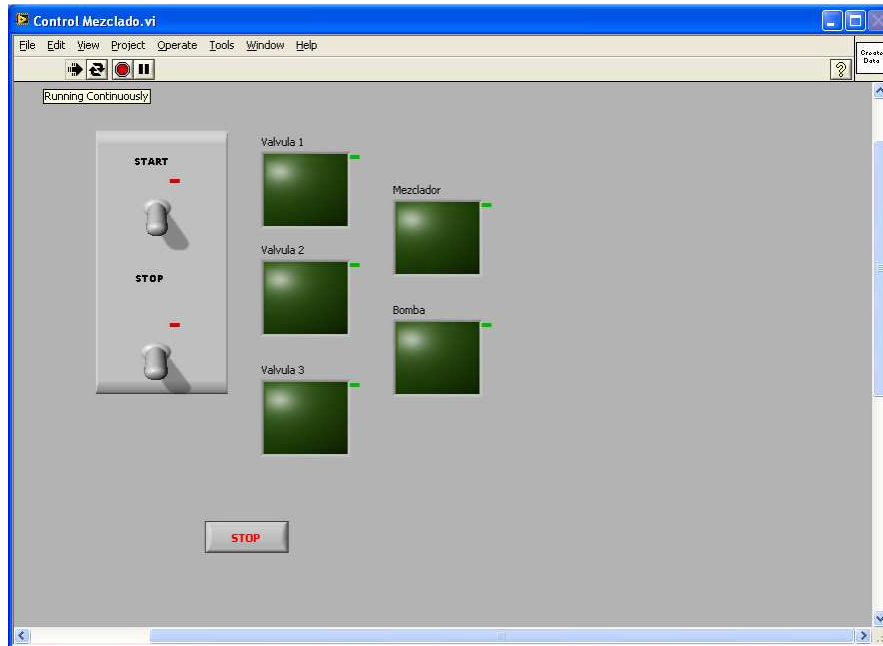
El sistema contará con lo siguiente menú:

- Control Mezclado
- Base de datos

Al hacer clic en control de mezclado nos llevara a la pantalla de monitorización el cual nos permite controlar todo el proceso de mezclado de líquidos como los botones

**Start:** Inicia el proceso de Mezclado

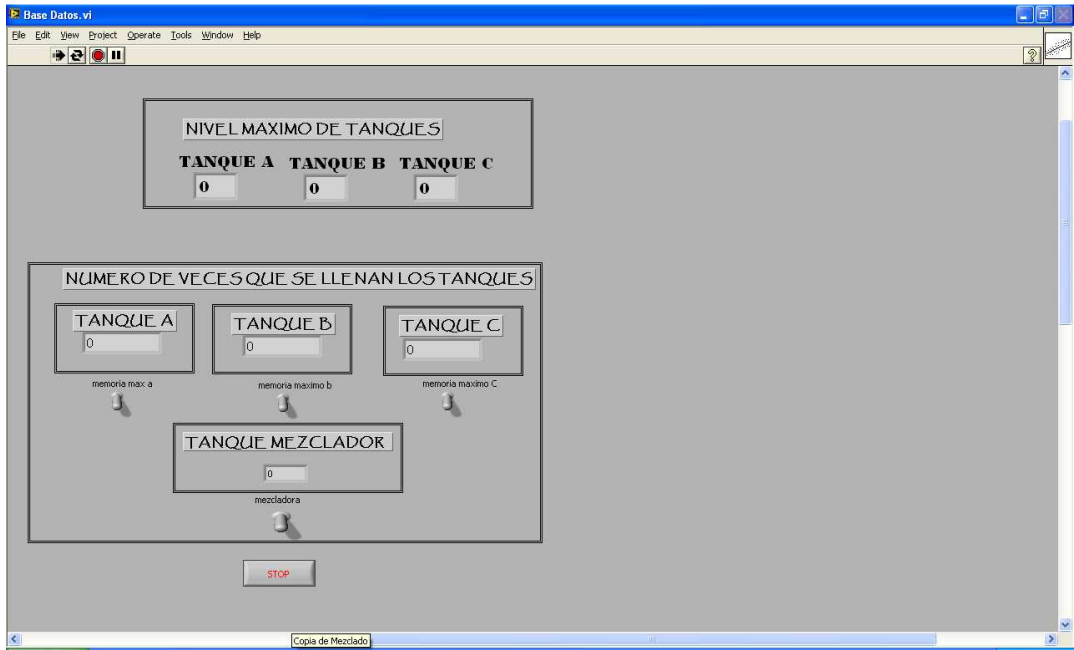
**Stop:** Termina el proceso en cualquier momento que el usuario lo dese.



Luego se irán encendiendo el cuadro de las válvulas 30 segundos y después el mezclador realiza la mezcla durante 5 segundos y finalmente la bomba empieza a enviar el líquido listo a la envasadora; este proceso es opcional ya que no corresponde a nuestro proyecto.

El STOP es para regresar al menú para escoger otra opción.

Al seleccionar la opción Base de Datos se muestra la siguiente pantalla, la misma que es utilizada para mostrar los niveles máximos de cada uno de los tanques A, B,C y D; al mismo tiempo se envía almacenar los datos en la base de datos SQL Server 2005.



ANEXO 4

**a) Mediante Juicio de Expertos**

La presente encuesta tiene como objetivo la comprobación de la hipótesis de la tesis titulada **“IMPLEMENTACION DE UN SISTEMA DE MEZCLADO DE LIQUIDOS. CASO PRÁCTICO: LABORATORIO DE AUTOMATIZACIÓN INDUSTRIAL DE LA EIS”**.

**Nombre:**

Ing. Pablo Montalvo

**Profesión (es):**

Ing. Electromecánico

Ing. Industrial.

**Experiencia docente:**

Cuenta con 24 años en la Facultad de Mecánica de la ESPOCH.

Cuenta con 26 años de experiencia en la Facultad de Mecánica en la ESPOCH.

**1. ¿Cuál es su criterio sobre la automatización industrial?**

-----  
-----  
-----  
-----

**2. ¿Cuál es su criterio sobre la aplicación de la Mecatrónica?**

-----  
-----

-----  
-----

3. **¿Considera Ud. importante que las empresas incorporen sistemas mecatrónicos, destinadas al área de producción para el Mezclado de Líquidos en las Industrias?**

-----  
-----  
-----  
-----

4. **¿Cree Ud. Que mediante la incorporación del Sistema de Mezclado de Líquidos se reducirá el tiempo necesario para el proceso de mezclado en una industria y por qué?**

-----  
-----  
-----  
-----

5. **¿Cree Ud. Que mediante la implementación de un Sistema de Mezclado de Líquidos, se reducirá el talento humano necesario para el mezclado de líquidos?**

-----  
-----  
-----  
-----

6. ¿Cuál es su opinión respecto a la implementación de un módulo didáctico para mejorar el proceso enseñanza – aprendizaje en la materia de Mecatrónica y Automatización Industrial?

-----  
-----  
-----  
-----

7. ¿Qué sugerencias podría hacer respecto a los temas abordados?

-----  
-----  
-----  
-----

Ing. Humberto Matheu

-----

**FIRMA**

CI: 170527929-5

**b) Mediante Encuestas a Estudiantes de la EIS**

Determinación de número de encuestas basadas en la población de estudiantes de sexto y séptimo nivel de la EIS.

**Muestra**

**N= Población: 80 Estudiantes**

**E= 0.1; Límite de error**

**$\alpha$ = 0.5; Varianza**

**Z= 1.96; Nivel de confianza**

$$n = \frac{N \alpha^2 Z^2}{(N - 1)E^2 + \alpha^2}$$

$$n = \frac{80(0.5)^2(1.96)^2}{(80 - 1)(0.1)^2 + (0.5)^2(1.96)^2}$$

$$n = 20$$

Para proceder a realizar las encuestas se tiene el siguiente formato:

La siguiente encuesta es con motivo de verificar si la implementación del módulo de Mezclado de Líquidos que permitirá mejorar la enseñanza – aprendizaje de los estudiantes en un prototipo básico con un diseño sencillo y sólido, adquiriendo nuevos conocimientos de la automatización industrial.

1. Cree Ud. que la materia de automatización industrial está acorde con la formación académica de su carrera.

Si ( )

No ( )

2. Cree ud que es importante que se implemente un laboratorio de automatización industrial para que los estudiantes estén acorde la teoría con la práctica.

Si ( )

No ( )

3. Cree Ud que con la construcción e implementación del Sistema(Mezclado de Líquidos) se logrará mejorar la enseñanza – aprendizaje de los estudiantes en la materia de automatización industrial.

Si ( )

No ( )

4. Cree Ud que el laboratorio de Automatización Industrial implementado con módulos didácticos será suficiente para que los estudiantes adquieran nuevos conocimientos relacionados a la materia para un buen desenvolvimiento en el campo industrial como profesional.

Si ( )

No ( )

5. Cree Ud que el módulo (mezcladora) es sencilla y sde manejar tanto en la parte del hardware como del software?

Si ( )

No ( )

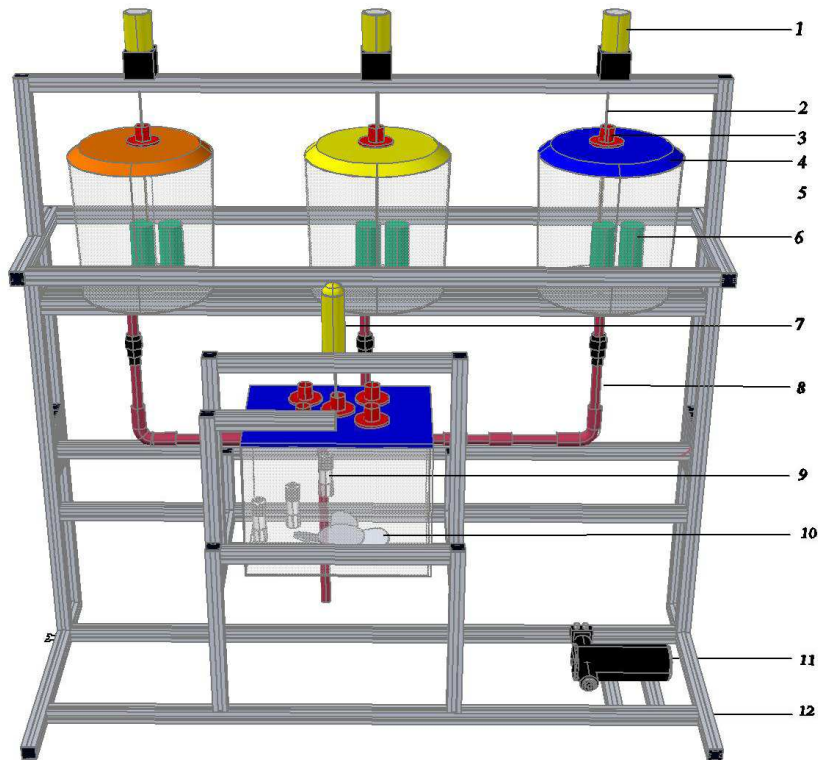
## TABULACION DE DATOS

Pregunta 1	Pregunta 2	Pregunta 3	Pregunta 4	Pregunta 5
1	1	1	1	1
1	1	1	1	1
1		1	1	1
1	1	1		1
1	1		1	1
	1	1		1
1	1	1	1	1
	1			1
1		1	1	1
	1	1		1
1	1	1	1	1
1		1	1	1
1	1	1		1
1	1	1	1	1
1	1	1	1	
1	1	1	1	1
1		1	1	1
1	1	1		1
1	1	1	1	1
1	1	1	1	1
<b>17</b>	<b>16</b>	<b>18</b>	<b>14</b>	<b>19</b>



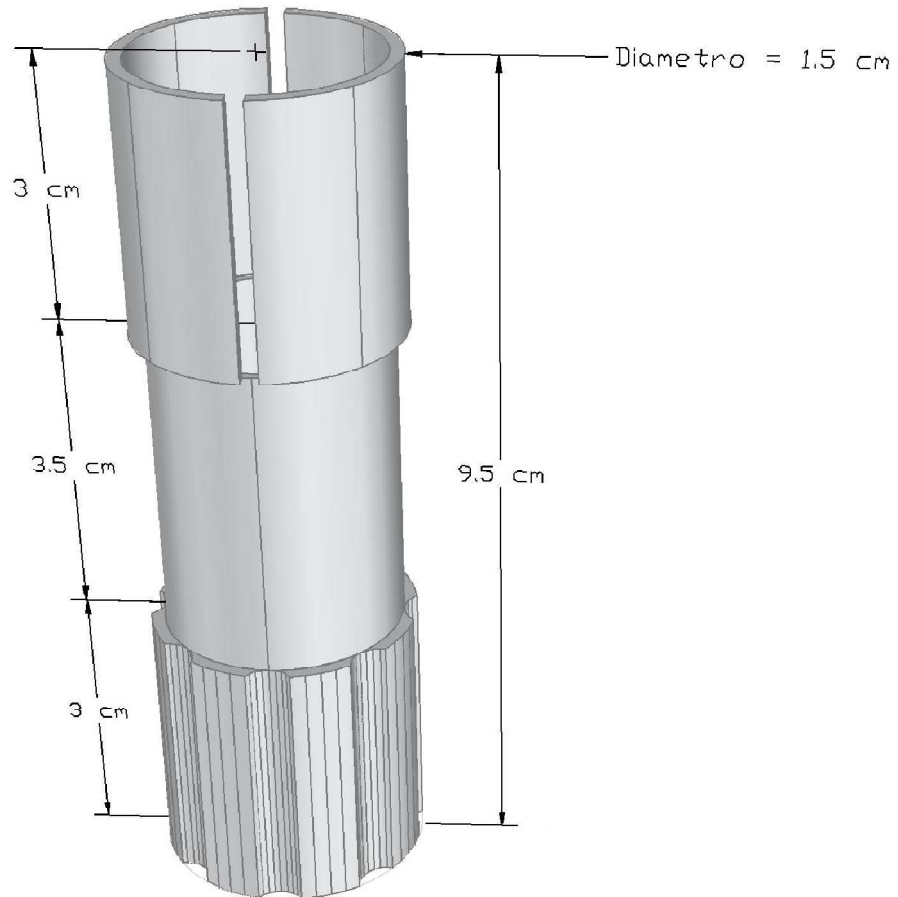
## ANEXO 5

## DISEÑO DEL MODULO

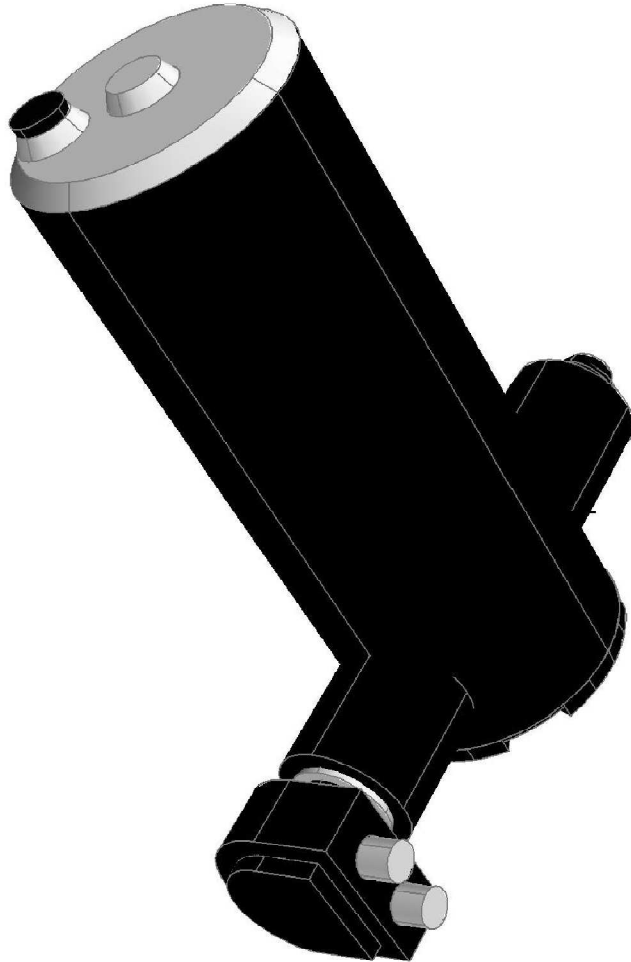


- |                                       |                            |
|---------------------------------------|----------------------------|
| 1. Sensor flotador parte superior     | 7. Pretul                  |
| 2. Hilo conductor del sensor flotador | 8. Tubería                 |
| 3. Tapa                               | 9. Sensor Sonda            |
| 4. Tapa del tanque                    | 10. Mezclador              |
| 5. Frasco                             | 11. Motor                  |
| 6. Sensor flotador parte inferior     | 12. Estructura de Aluminio |

### DISEÑO DEL SENSOR SONDA



**DISEÑO DEL MOTOR**



**DISEÑO DEL MEZCLADOR**

