



**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**

**FACULTAD DE INFORMÁTICA Y ELECTRÓNICA**

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

**“DISEÑO E IMPLEMENTACIÓN DE UNA RED DE COMUNICACIÓN  
ETHERNET DIDÁCTICA CON PLC’S PARA EL CONTROL Y  
MONITOREO DE PROCESOS MODULARES”**

**TESIS DE GRADO**

Previo a la obtención del título de:

**INGENIERO EN ELECTRÓNICA CONTROL Y REDES  
INDUSTRIALES**

Presentado Por

**LEONARDO MANUEL ASQUI PAGUAY  
IVÁN MOISÉS LEMA HOLGUÍN**

**RIOBAMBA- ECUADOR**

2013

## AGRADECIMIENTO

Mi más grande agradecimiento a mis padres y hermanos por ser la fuente de sabiduría, perseverancia y alegría que han guiado mi vida en los peores y mejores momentos, por ser los mejores amigos.

A Dios por haberme dado salud, vida, la oportunidad de seguir esta carrera y por poner en mi vida un gran número de preciados amigos.

**Leonardo**

A mis padres y familiares que han sido una guía en mi vida y un apoyo incondicional. A mis amigos que siempre estuvieron alentándonos para que siguiéramos adelante y nos ayudaron sin decir que no. A mi enamorada Jessica Bedoya que ha sido aquella persona que me ha acompañado en las buenas en las malas y ha estado todo este tiempo a mi lado frente a cualquier cosa. Y a todas aquellas personas que dieron un apoyo para la culminación de esta meta.

**Iván**

## DEDICATORIA

A mis padres por guiarme y apoyarme, durante el trayecto de esta carrera y de mi vida, por estar siempre en los momentos difíciles.

A mis hermanos por haberme enseñado y dado lo mejor de ellos.

A dios por haberme dado el carácter suficiente, y la valentía para afrontar todos los problemas durante este recorrido.

**Leonardo**

Dedico el presente trabajo a mi papá Moisés Lema, que ha sido mi ejemplo a seguir y siempre ha estado ahí, apoyándome desde que era un niño, enseñándome que siempre hay seguir adelante frente a las adversidades. También a mi hermana Verónica Lema porque es una de las personas más especiales en mi vida. A Jessica Bedoya por el apoyo incondicional y sincero.

**Iván**

**FIRMAS RESPONSABLES Y NOTA**

<b>NOMBRE</b>	<b>FIRMA</b>	<b>FECHA</b>
<b>Ing. Iván Menes</b> <b>DECANO DE LA FACULTAD</b> <b>DE INFORMÁTICA Y ELECTRÓNICA</b>	.....	.....
<b>Ing. Paúl Romero</b> <b>DIRECTOR DE LA ESCUELA DE</b> <b>INGENIERÍA ELECTRÓNICA EN</b> <b>CONTROL Y REDES INDUSTRIALES</b>	.....	.....
<b>Ing. Marco Viteri</b> <b>DIRECTOR DE TESIS</b>	.....	.....
<b>Ing. Diego Barba Maggi</b> <b>MIEMBRO DEL TRIBUNAL</b>	.....	.....
<b>Tec. Carlos Rodríguez Carpio</b> <b>DIRECTOR DPTO</b> <b>DOCUMENTACIÓN</b>	.....	.....
<b>NOTA DE LA TESIS</b>	.....	

“Nosotros LEONARDO MANUEL ASQUI PAGUAY, IVÁN MOISÉS LEMA HOLGUÍN somos responsables de las ideas, doctrinas y resultados expuestos en esta tesis; y, el patrimonio intelectual de la Tesis de Grado pertenece a la ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO”

---

**Leonardo Manuel Asqui Paguay**

---

**Iván Moisés Lema Holguín**

## ÍNDICE DE ABREVIATURAS

<b>CA</b>	Canal de acceso
<b>COM</b>	Component Object Model (Modelo de Objetos Componentes)
<b>CSMA/CD</b>	Carrier Sense Multiple Access with Collision Detect
<b>DBMS</b>	Database Management System (Sistema de Gestión de Base de Datos)
<b>ODBC</b>	Open DataBase Connectivity
<b>OLE DB</b>	Object Linking and Embedding Database
<b>PFC</b>	Programable Fieldbus Controller (Controlador programable Fieldbus)
<b>TCP</b>	Transmission Control Protocol (Protocolo de Control de Transmisión)
<b>HMI</b>	Human Machine Interface
<b>SCADA</b>	Supervisory Control and Data Acquisition
<b>CIM</b>	Computer Integrated Manufacturing
<b>mA</b>	Mili amperios.
<b>E/S</b>	Entradas/salidas
<b>I/O</b>	Input/output
<b>OSI</b>	Open System Interconnection
<b>CIP</b>	Common Industrial Protocol
<b>PFC</b>	Programmable Fieldbus Controller
<b>MAC</b>	Media Access Control
<b>ADO</b>	ActiveX Data Objects
<b>LAN</b>	Local Area Network
<b>IP</b>	Internet Protocol
<b>OPC</b>	OLE for Process Control

# ÍNDICE GENERAL

## Contenido

AGRADECIMIENTO .....	
DEDICATORIA.....	
ÍNDICE DE ABREVIATURAS .....	
ÍNDICE GENERAL.....	
ÍNDICE DE FIGURAS .....	
ÍNDICE DE TABLAS .....	
INTRODUCCIÓN .....	
CAPÍTULO I	
MARCO REFERENCIAL.....	23
1.1. ANTECEDENTES .....	23
1.2. JUSTIFICACIÓN DEL PROYECTO DE TESIS .....	25
1.3. OBJETIVOS .....	27
1.3.1. OBJETIVO GENERAL .....	27
1.3.2. OBJETIVOS ESPECÍFICOS.....	27
1.4. HIPÓTESIS .....	28
CAPÍTULO II	
MARCO TEÓRICO .....	29
2.1. INTRODUCCIÓN A LA AUTOMATIZACIÓN .....	29
2.2. AUTOMATIZACIÓN INDUSTRIAL .....	30
2.2.1. Tipos de automatización industrial.....	31
2.2.2. Tecnologías empleadas en la automatización .....	32
2.2.3. Herramientas para la automatización.....	33
2.3. CONTROLADORES PROGRAMABLES .....	34
2.3.1. PC.....	34
2.3.1.1. PC Industriales .....	34
2.3.1.2. PC-based Controller .....	35
2.3.1.3. Controladores Embebidos .....	35
2.3.2. PAC.....	35
2.3.2.1. Compact FieldPoint .....	37

2.3.3. PLC (PLC CONTROLADOR LÓGICO PROGRAMABLE) .....	38
2.3.3.1. Estructura del PLC .....	39
2.3.3.2. LENGUAJES DE PROGRAMACIÓN DE PLC´S .....	42
2.4. REDES INDUSTRIALES .....	46
2.4.1. INTRODUCCIÓN .....	46
2.4.2. NIVELES DE COMUNICACIÓN INDUSTRIAL .....	48
2.4.3. Estructuras de comunicación de las redes industriales:.....	49
2.4.4. REDES INDUSTRIALES Y EL MODELO OSI .....	51
2.4.4.1. Modelo OSI .....	51
2.4.4.2. Relación con el Modelo OSI .....	51
2.4.5. ETHERNET INDUSTRIAL .....	54
2.4.5.1. CARACTERÍSTICAS .....	55
2.4.5.2. TOPOLOGÍA DE ETHERNET INDUSTRIAL .....	57
2.4.5.3. PROTOCOLOS DE LA CAPA DE APLICACIÓN PARA ETHERNET INDUSTRIAL .....	59
FIELDBUS FOUNDATION .....	60
PROFINET (Profibus).....	64
ETHERNET/IP .....	70
MODBUS TCP/IP .....	74
2.4.5.4. DISPOSITIVOS .....	79
PLC´S ETHERNET.....	79
ROUTERS O ENRUTADOR .....	80
SWITCH .....	81
2.5. OPC SERVER .....	82
2.5.1. Cómo trabaja la comunicación OPC (Conceptualmente).....	83
2.5.2. Como funciona OPC (Funcionalmente) .....	83
2.5.3. Beneficios de utilizar conectividad OPC.....	84
2.5.4. Tipos de datos que soporta OPC .....	85
2.5.5. Servidor OPC.....	86
2.5.6. Cliente OPC .....	86
2.6. HMI .....	86
2.6.1. Funciones de un HMI .....	87
2.6.2. SCADA (Supervisory Control And Data Acquisition).....	87

2.6.2.1. Esquema Básico de un sistema SCADA .....	88
2.6.3. BASE DE DATOS (DB).....	89
2.6.3.1. Tipos de Bases de datos .....	89
2.6.3.2. Administración de bases de datos.....	90
CAPÍTULO III	
HARWARE Y SOFTWARE UTILIZADO EN LA IMPLEMENTACIÓN .....	93
3.1. PLC WAGO 750-842 .....	93
3.1.1. ETHERNET TCP/IP Programmable Fieldbus Controller (PFC) .....	94
3.1.2. Intercambio de datos.....	95
3.1.2.1. Área de Memoria .....	97
3.1.2.2. Intercambio de Datos entre Maestro Modbus/TCP y Módulos E/S... 98	
3.1.2.3. Intercambio de datos entre la función del PLC de PFC (CPU) y los módulos de E/S .....	99
3.1.2.4. Intercambio de Datos entre el Maestro y la función PLC del PFC (CPU) .....	100
3.1.3. COMUNICACIÓN FIELDBUS.....	101
3.1.3.1. Ethernet.....	102
Arquitectura de Procesos de Datos para MODBUS/TCP .....	104
3.1.4. Módulos de Entradas y Salidas .....	107
3.1.4.1. Módulos de entradas digitales.....	107
3.1.4.2. Módulos de salidas digitales.....	107
3.1.4.3. Módulos de entradas analógicas .....	108
3.1.4.4. Módulos de salidas analógicas.....	108
3.1.4.5. Direccionamiento de los Módulos de E/S.....	109
3.2. FIELD POINT FP-1601 .....	110
3.2.1. Aplicaciones.....	111
3.2.2. Comunicación de Red.....	112
3.2.3. Interface de Red Ethernet para FieldPoint.....	113
3.2.4. Software de Fieldpoint .....	114
3.3. TELEMECANIQUE TWDLMDA20DTK.....	115
3.3.1. Descripción del controlador.....	115
3.3.2. Comunicación de PLC Twido.....	117
3.3.2.1. Características funcionales de las bases modulares.....	118

3.3.2.2. Arquitectura de comunicaciones con los protocolos.....	118
3.3.2.3. Protocolos de Comunicación.....	119
3.3.2.4. Comunicación MODBUS.....	121
3.4. PASARELA O CONVERTOR DE MODBUS SERIAL A ETHERNET.....	123
3.4.1. Definición y Funcionamiento de Pasarela.....	123
3.4.2. Twido Port.....	125
3.4.2.1. Configuración del TwidoPort.....	126
3.5. SIEMENS S7-1200.....	127
3.5.1. Módulos de Comunicación.....	128
3.5.2. Profinet.....	129
3.5.2.1. Arquitectura Profinet.....	130
3.5.3. Profinet en Siemens.....	131
3.5.3.1. Número máximo de conexiones para el puerto PROFINET.....	132
3.5.3.2. Protocolos de Comunicación en S7-1200.....	133
3.5.4. Comunicación con una programadora.....	135
3.5.5. Comunicación entre PLC's.....	136
3.5.5.1. Configurar los parámetros de conexión.....	137
3.3.5.2. Instrucciones de comunicación.....	137
3.6. SWITCH INDUSTRIAL NO GESTIONADO (WEIDMULLER IE-SW8-WAVE) .....	140
3.7. CABLE ETHERNET INDUSTRIAL.....	142
3.7.1. Tipología de cables para Ethernet.....	143
3.7.1.1. Cable Industrial Ethernet, 2x2x24 AWG (conductores monolíticos) UTP.....	143
3.7.1.2. Cable Industrial Ethernet, 4x2x24 AWG (conductores monolíticos) UTP.....	144
3.7.1.3. Cable diseñado para redes industriales Industrial Ethernet, 4x2x24 AWG FTP.....	146
3.7.2. Conectores para Ethernet.....	147
3.8. OPC Kepsriver.....	148
3.8.1. Componentes.....	148
3.8.2. Modos de Proceso.....	150
3.8.3. Menú de Administración.....	150
3.8.4. Componentes Básicos del Servidor.....	160

3.9. BASE DE DATOS.....	171
3.9.1. WampServer .....	171
3.9.1.1. Características de WampServer:.....	172
3.9.1.2. Utilidad .....	172
3.9.1.3. Funcionalidades .....	172
3.9.2. PHP MyAdmin.....	173
3.9.2.1. Características.....	174
3.9.2.2. phpMyAdmin en Wampserver .....	174
3.10. LABVIEW.....	176
3.10.1. I/O SERVER DE LABVIEW.....	176
3.10.1.1. Clientes de I/O Server .....	176
3.10.2. Base de Datos en Labview .....	178
3.10.2.1. ODBC .....	178
3.10.2.2. OLE DB .....	179
3.10.2.3. ADO (ActiveX Data Objects).....	180
3.10.3. Database Connectivity Toolkit.....	180
3.11. PROCESOS MODULARES.....	187
3.11.1. MESA DE INDEXACION.....	187
3.11.2. ENVASADORA .....	189
3.11.3. SEMÁFORO .....	190
3.11.4. CASA AUTOMATIZADA.....	192
CAPÍTULO IV	
IMPLEMENTACIÓN.....	195
4.1. DISEÑO DE LA RED .....	195
4.1.1. Requerimientos.....	196
4.1.2. Escalabilidad.....	197
4.1.3. Redundancia.....	197
4.1.4. Rendimiento.....	197
4.1.6. Arquitectura.....	198
4.1.6.1. TOPOLOGÍA.....	199
4.1.6.1.1. Topología Física .....	199
4.1.6.1.2. Topología Lógica .....	200
4.1.7. Direccionamiento .....	200

4.2. DISEÑO E IMPLEMENTACION ESTRUCTURAL.....	201
4.2.1. MODULAR DE PLC´S.....	201
4.2.2. CIRCUITOS DISEÑADOS .....	202
4.2.2.1. Circuitos de Borneros .....	202
4.2.2.2. Circuitos de la Casa automatizada .....	204
4.2.3. CABLE DE CONEXIÓN ENTRE EL MÓDULO Y EL PLC .....	206
4.3. PROGRAMACIÓN.....	208
4.3.1. PROGRAMACIÓN DEL SIEMENS S7-1200.....	208
4.3.1.1. Funcionamiento del Módulo de Envasadora. ....	208
4.3.1.2. Señales de Entradas y Salidas asignadas al S7-1200 .....	210
4.3.1.3. Grafcet del Módulo Envasadora .....	211
4.3.1.4. Ecuaciones .....	212
4.3.2. PROGRAMACIÓN DEL WAGO 750-842.....	212
4.3.2.1. Funcionamiento de la Mesa Indexadora.....	212
4.3.2.2. Señales de las Entradas y Salidas asignadas al Wago 750-842. ....	214
4.3.2.3. Grafcet del módulo Mesa Indexadora.....	214
4.3.2.4. Ecuaciones .....	215
4.3.3. PROGRAMACIÓN DEL PLC TWIDO .....	216
4.3.3.1. Secuencia del Semáforo.....	216
4.3.3.2. Señales de las Entradas y Salidas asignadas al Twido.....	217
4.3.3.3. Grafcet del Semáforo .....	218
4.3.3.4. Ecuaciones .....	219
4.3.4. PROGRAMACIÓN DEL FIELDPOINT FP-1601 .....	219
4.3.4.1. Funcionamiento de la Casa Automatizada .....	219
4.3.4.2. Señales de las Entradas y Salidas asignadas al FieldPoint .....	221
4.3.4.3. Grafcet de la Casa Automatizada.....	222
4.3.4.4. Ecuaciones .....	224
4.4. BASE DE DATOS.....	225
4.4.1. DISEÑO .....	225
4.4.2. CONFIGURACIÓN.....	226
4.4.2.1. CREACIÓN BASE DE DATOS EN MySQL PHP.....	226
4.5. HMI .....	231
4.5.1. OPC .....	231

4.5.1.1. Configuración S7-1200.....	232
4.5.1.2. Configuración Wago 750-842.....	240
4.5.1.3. Configuración TWDLMDA20DTK.....	252
4.5.1.4. Configuración FieldPoint 16-01.....	259
4.5.2. LABVIEW.....	264
4.5.2.1. <i>Modo de operación del HMI</i> .....	264
4.5.2.2. <i>Enlace del OPC con LabVIEW</i> .....	270
4.5.2.3. <i>Enlace de la Base de Datos con LabVIEW</i> .....	274
CAPÍTULO V	
RESULTADOS.....	282
5.1. INTRODUCCION.....	282
5.2. ANALISIS DE LA ENCUESTA.....	282
CONCLUSIONES.....	
RECOMENDACIONES.....	
RESUMEN.....	
SUMMARY.....	
GLOSARIO.....	
BIBLIOGRAFIA.....	
BIBLIOGRAFÍA DE INTERNET.....	
ANEXOS.....	

## ÍNDICE DE FIGURAS

<b>Figura II.1.</b> Automatización en diferentes niveles de la pirámide de control.....	31
<b>Figura II.2.</b> PC industrial de panel PPC-5012-ACE855 .....	34
<b>Figura II.3.</b> Imagen del interior de un módem/enrutador ADSL.....	35
<b>Figura II.4.</b> PAC, convergencia entre PC, PLC y hardware.....	36
<b>Figura II.5.</b> COMPACT FIELDPOINT .....	37
<b>Figura II.6.</b> PLC (Controlador Lógico Programable).....	38
<b>Figura II.7.</b> PLC Compacto y PLC Modular.....	40
<b>Figura II.8.</b> Ejemplo de programación mediante diagrama de funciones .....	45
<b>Figura II.9.</b> Diagrama Del PLC.....	46
<b>Figura II.10.</b> Pirámide CIM .....	48
<b>Figura II.11.</b> Cable Filar AS-i .....	51
<b>Figura II.12.</b> Redundancia de una red Ethernet.....	55
<b>Figura II.13.</b> Topología Bus.....	57
<b>Figura II.14.</b> Topología Anillo .....	58
<b>Figura II.15.</b> Topología Estrella .....	58
<b>Figura II.16.</b> Estructura de estratos del Foundation Fieldbus.....	61
<b>Figura II.17.</b> Ejemplo de señal Fieldbus en modo de tensión .....	62
<b>Figura II.18.</b> Estructura de dispositivo Profinet .....	65
<b>Figura II.19.</b> Modelos de operación de Profinet.....	65
<b>Figura II.20.</b> Tiempos de respuesta de Profinet.....	67
<b>Figura II.21.</b> Creación e interconexión de componentes.....	69
<b>Figura II.22.</b> Integración de una red PROFinet con una red Ethernet industrial.....	70
<b>Figura II.23.</b> Encapsulamiento de la trama Modbus en TCP.....	78
<b>Figura II.24.</b> Utilización de un Router en una red Ethernet Industrial.....	81
<b>Figura II.25.</b> Red Industrial con Switch .....	82
<b>Figura II.26.</b> Función del servidor OPC entre la fuente de datos y el cliente de datos.....	83
<b>Figura II.27.</b> Arquitectura Cliente/Servidor OPC.....	84
<b>Figura II.28.</b> Esquema de un sistema SCADA. ....	88
<b>Figura II.29.</b> Esquema y niveles de una Base de Datos. ....	89
<b>Figura II.30.</b> Administración de una Base de Datos. ....	90
<b>Figura III.31.</b> PLC WAGO-I/O-SYSTEM 750 .....	94
<b>Figura III.32.</b> Áreas de memoria e Intercambio de Datos.....	97
<b>Figura III.33.</b> Intercambio de datos entre el Maestro Modbus y módulos de E/S. ....	99
<b>Figura III.34.</b> Intercambio de datos entre la función del PLC de PFC (CPU) y los módulos de E/S. ....	100
<b>Figura III.35.</b> Intercambio de datos entre el maestro Modbus y la función PLC del PFC. ....	101
<b>Figura III.36.</b> Ejemplo de topología Modbus con WAGO 750.....	105
<b>Figura III.37.</b> Dispositivo FieldPoint FP-1601 .....	111
<b>Figura III.38.</b> Conexión de FP-1601 a una red Ethernet.....	112
<b>Figura III.39.</b> Esquema de la arquitectura Publish/Suscriber.....	113
<b>Figura III.40.</b> Controlador Twido TWDLMDA20DTK.....	115
<b>Figura III.41.</b> Arquitectura de comunicaciones con los tres protocolos anteriormente mencionados. .....	119
<b>Figura III.42.</b> Configuración Modbus en Twido Suite.....	122
<b>Figura III.43.</b> Funcionamiento de pasarela. ....	124

<b>Figura III.44.</b> TwidoPort. ....	125
<b>Figura III.45.</b> Conexión de TwidoPort con PLC Twido y Switch Ethernet.....	126
<b>Figura III.46.</b> PLC Siemens S7-1200.....	128
<b>Figura III.47.</b> Ubicación de conector para comunicación. ....	129
<b>Figura III.48.</b> Tipos de comunicación Profinet del S7-1200.....	132
<b>Figura III.49.</b> Conexión entre PC y S7-1200.....	135
<b>Figura III.50.</b> Switch Industrial No Gestionado IE SW8 WAVE. ....	140
<b>Figura III.51.</b> Tipos de Cable para conexión Ethernet.....	143
<b>Figura III.52.</b> Cable UTP 2 par trenzado para conexión Ethernet. ....	143
<b>Figura III.53.</b> Cable UTP 4 par trenzado para conexión Ethernet. ....	144
<b>Figura III.54.</b> Cable FTP 4 par trenzado para conexión Ethernet. ....	146
<b>Figura III.55.</b> Conectores RJ-45 y M12 para Ethernet.....	147
<b>Figura III.56.</b> OPC KepServer.....	148
<b>Figura III.57.</b> Menú de Administration del Kepserver.....	151
<b>Figura III.58.</b> Pestaña Settings-Administration. ....	152
<b>Figura III.59.</b> Pestaña Settings-Configuration. ....	153
<b>Figura III.60.</b> Pestaña Settings-Runtime Process. ....	154
<b>Figura III.61.</b> Pestaña Settings-Runtime Options.....	155
<b>Figura III.62.</b> Pestaña Settings-Event Log.....	157
<b>Figura III.63.</b> Pestaña Settings-ProgID Redirect.....	158
<b>Figura III.64.</b> Pestaña Settings-User Manager. ....	159
<b>Figura III.65.</b> Propiedades de un Canal – Pestaña General.....	160
<b>Figura III.66.</b> Propiedades de un Canal – Pestaña Comunicaciones. ....	161
<b>Figura III.67.</b> Propiedades de un Canal – Pestaña Interfaz de Red.....	162
<b>Figura III.68.</b> Propiedades de un Canal – Pestaña Optimización de Escritura.....	163
<b>Figura III.69.</b> Propiedades de un Canal – Pestaña Avanzado.....	164
<b>Figura III.70.</b> Propiedades de Dispositivo – Pestaña General. ....	165
<b>Figura III.71.</b> Propiedades de Dispositivo – Pestaña Scan Mode.....	167
<b>Figura III.72.</b> Propiedades de Dispositivo – Pestaña Timing.....	168
<b>Figura III.73.</b> Propiedades de Dispositivo – Pestaña Auto-Demotion.....	169
<b>Figura III.74.</b> Propiedades de Dispositivo – Pestaña Database Creation.....	170
Figura III.75. WampServer.....	171
Figura III.76. phpMyAdmin.....	173
<b>Figura III.77.</b> Acceso a phpMyAdmin por el menu de WampServer.....	175
<b>Figura III.78.</b> Acceso a phpMyAdmin por el LocalHost. ....	175
<b>Figura III.79.</b> Intefaz phpMyAdmin.....	176
<b>Figura III.80.</b> Labview y red OPC con DSC. ....	177
<b>Figura III.81.</b> Labview y red Modbus.....	178
<b>Figura III.82.</b> Paleta General de Operaciones de Base de Datos. ....	181
<b>Figura III.83.</b> Paleta Avanzada de Base de Datos.....	181
<b>Figura III.84.</b> Paleta Avanzada de Base de Datos.....	182
<b>Figura III.85.</b> Icono DB Tools Open Connection. ....	182
<b>Figura III.86.</b> Icono DB Tools Close Connection. ....	183
<b>Figura III.87.</b> Icono DB Tools Insert Data. ....	183
<b>Figura III.88.</b> Icono DB Tools Select Data. ....	184
<b>Figura III.89.</b> Icono DB Tools Create Table.....	184

<b>Figura III.90.</b> Icono DB Tools Execute Query.....	185
<b>Figura III.91.</b> Icono DB Tools Fetch Recordset Data.....	185
<b>Figura III.92.</b> Icono DB Tools Fetch Element Data. ....	186
<b>Figura III.93.</b> Icono DB Tools Free Object. ....	186
<b>Figura III.94.</b> Icono DB Tools Drop Table. ....	187
<b>Figura III.95.</b> Módulo Mesa de Indexación. ....	187
<b>Figura III.96.</b> Módulo Envasadora.....	189
<b>Figura III.97.</b> Semáforo. ....	190
<b>Figura III.98.</b> Casa Automatizada .....	192
<b>Figura IV.99.</b> Topología de la Red Didáctica Ethernet .....	196
<b>Figura IV.100.</b> Disposición de la Red Didáctica Ethernet.....	198
<b>Figura IV.101.</b> Modular de PLC's.....	202
<b>Figura IV. 102.</b> Pista de la Plaqueta de Borneros.....	204
<b>Figura IV.103.</b> Circuito Base Puente H.....	205
<b>Figura IV.104.</b> Placa de los circuitos Puente H y de Temperatura.....	205
<b>Figura IV.105.</b> Placa de Alimentación.....	206
<b>Figura IV.106.</b> Cable de conexión .....	207
<b>Figura IV.107.</b> Grafcet del Módulo Envasadora.....	211
<b>Figura IV.108.</b> Grafcet del Módulo Mesa de Indexación.....	215
<b>Figura IV.109.</b> Grafcet del semáforo.....	218
<b>Figura IV.110.</b> Grafcet de la Casa Automatizada. Activación o desactivación de luces.....	222
<b>Figura IV.111.</b> Grafcet de la Casa Automatizada. Apertura o cierre de garaje.....	223
<b>Figura IV.111.</b> Grafcet de la Casa Automatizada. Activación o desactivación de alarma.....	223
<b>Figura IV.112.</b> Modelado de la Base de Datos.....	225
<b>Figura IV.113.</b> Diseño de la Base de Datos .....	226
<b>Figura IV.114.</b> Menú WampServer .....	227
<b>Figura IV.115.</b> Creación de una base de datos .....	227
<b>Figura IV.116.</b> Creación de una tabla.....	228
<b>Figura IV.117.</b> Configuración de campos de una tabla .....	229
<b>Figura IV.118.</b> Creación de un INDEX.....	230
<b>Figura IV.119.</b> Campo de INDEX.....	230
<b>Figura IV.120.</b> Creación de una relación.....	231
<b>Figura IV.121.</b> Selección de Dispositivos y Redes. En TIA Portal .....	232
<b>Figura IV.122.</b> Selección del dispositivo empleado en la implementación. En TIA Portal.....	233
<b>Figura IV.123.</b> Reseteo del PLC. En TIA Portal .....	234
<b>Figura IV.124.</b> Asignación de la dirección IP en el PLC. En TIA Portal.....	235
<b>Figura IV.125.</b> Bloque de Datos con las variables a utilizar ya compilado. En TIA Portal.....	236
<b>Figura IV.126.</b> Tabla de Datos con las variables del Bloque de Datos. En TIA Portal .....	237
<b>Figura IV.127.</b> Elección del modelo del dispositivo. En Keepserver .....	238
<b>Figura IV.128.</b> Asignación de la dirección IP del dispositivo. En Keepserver.....	238
<b>Figura IV.129.</b> Direcciones de la Tabla de Observación utilizadas para las tags de OPC. En TIA Portal .....	239
<b>Figura IV.130.</b> Creación de las tags de OPC en Kepsriver.....	240
<b>Figura IV. 131.</b> Pantalla principal del Boot Server .....	242
<b>Figura IV. 132.</b> Creación del nuevo programa en WAGO IO PRO 32.....	244
<b>Figura IV. 133.</b> Elección del PLC WAGO 750-842.....	245

<b>Figura IV. 134.</b> Creación del Nuevo Canal de comunicación. ....	245
<b>Figura IV. 135.</b> Elección del canal Ethernet_TCP_IP. ....	246
<b>Figura IV. 136.</b> Ventana de Parámetros de Comunicación con la dirección IP del Wago. ....	246
<b>Figura IV.137.</b> Elección del Driver del dispositivo para el canal. ....	247
<b>Figura IV. 138.</b> Elección de la tarjeta de red. ....	247
<b>Figura IV. 139.</b> Elección del modelo que se utiliza. ....	248
<b>Figura IV. 140.</b> Asignación de la dirección IP del Wago. ....	248
<b>Figura IV. 141.</b> Ventana de configuración de módulos adicionales para Wago. ....	249
<b>Figura IV. 142.</b> Configuración de las salidas en Wago IO Pro 32. ....	250
<b>Figura IV. 143.</b> Creación del tag de salida en Kepserver. ....	251
<b>Figura IV. 144.</b> Configuración de las entradas en Wago IO Pro 32. ....	251
<b>Figura IV. 145.</b> Creación del tag de Inicio en Kepserver. ....	252
<b>Figura IV.146.</b> Tipo de controlador Twido que se utiliza TWDLMDA20DTK. ....	253
<b>Figura IV. 147.</b> Configuración del puerto de comunicación para el twido. ....	253
<b>Figura IV. 148.</b> Pasarela Ethernet en el espacio de trabajo. ....	254
<b>Figura IV. 149.</b> Configuración de la dirección de la Pasarela. ....	254
<b>Figura IV. 150.</b> Conexión entre Twido y la Pasarela en el espacio de trabajo. ....	255
<b>Figura IV. 151.</b> Configuración de la red. ....	255
<b>Figura IV. 152.</b> Creación de la nueva conexión. ....	256
<b>Figura IV. 153.</b> Selección del driver para Twido. ....	256
<b>Figura IV. 154.</b> Asignación de la tarjeta d red para Twido. ....	257
<b>Figura IV. 155.</b> Elección del modelo del dispositivo para Twido. ....	257
<b>Figura IV. 156.</b> Asignación de la dirección IP del Twido. ....	258
<b>Figura IV. 157.</b> Creación de los tags del Twido. ....	259
<b>Figura IV. 158.</b> Creación de un nuevo proyecto. ....	260
<b>Figura IV. 159.</b> Selección del dispositivo con la red a utilizar. ....	261
<b>Figura IV. 160.</b> Vista del FieldPoint y sus dispositivos. ....	261
<b>Figura IV. 161.</b> Bloques del dispositivo de entradas analógicas en Labview. ....	262
<b>Figura IV. 162.</b> Paleta de FieldPoint. ....	263
<b>Figura IV. 163.</b> Paleta de FieldPoint (Advanced). ....	263
<b>Figura IV. 164.</b> HMI Módulo Envasadora (Panel Frontal) ....	265
<b>Figura IV. 165.</b> Diagrama de Bloques Módulo Envasadora ....	265
<b>Figura IV. 166.</b> HMI Módulo Mesa de Indexación (Panel Frontal). ....	266
<b>Figura IV. 167.</b> Diagrama de Bloques Módulo Mesa de Indexación ....	266
<b>Figura IV. 168.</b> HMI Semáforo (Panel Frontal). ....	267
<b>Figura IV. 169.</b> Diagrama de Bloques Semáforo ....	267
<b>Figura IV. 170</b> HMI Casa Automatizada (Panel Frontal) ....	268
<b>Figura IV. 171.</b> Diagrama de Bloques Casa Automatizada. ....	268
<b>Figura IV. 172.</b> Pantalla adyacente de Tiempo, Historicos, Usuarios y Tags. ....	269
<b>Figura IV. 173.</b> Creación de I/O Server en Labview. ....	270
<b>Figura IV. 174.</b> Selección OPC Client. ....	271
<b>Figura IV. 175.</b> Variables I/O Server utilizadas en el proyecto. ....	271
<b>Figura IV. 176.</b> Creación de una nueva variable ....	272
<b>Figura IV. 177.</b> Selección del canal y dispositivo utilizados. ....	273
<b>Figura IV. 178.</b> Adición de las nuevas variables. ....	273
<b>Figura IV. 179.</b> Nuevas variables adicionadas. ....	274

<b>Figura IV. 180.</b> Nuevas variables mostradas en el proyecto.....	274
<b>Figura IV. 181.</b> Conexión de ODBC con Labview.....	275
<b>Figura IV. 182.</b> Configuración de parámetros para la conexión. ....	276
<b>Figura IV. 183.</b> Creación del archivo DNS. ....	277
<b>Figura IV. 184.</b> Configuración del archivo DNS. ....	277
<b>Figura IV. 185.</b> Configuración final para la conexión. ....	278
<b>Figura IV. 186.</b> Create Data Link para conexión por UDL.....	279
<b>Figura IV. 187.</b> Selección del proveedor para ODBC.....	279
<b>Figura IV. 188.</b> Resultados del nivel de satisfacción de las preguntas de la encuesta .....	289

## ÍNDICE DE TABLAS

Tabla II.I. Tipos y funciones de las interfaces de entrada /salida .....	41
Tabla II.II. Simbología de la programación Ladder .....	43
Tabla II.III. Comparación técnica de los tipos Fieldbus. ....	63
Tabla II.IV. Estructura del prefijo de Modbus/TCP.....	76
Tabla II.V. Estructura del mensaje Modbus/TCP.....	77
Tabla III.VI. Asignación de entradas y salidas digitales para procesar las palabras de datos de acuerdo con el formato de Intel .....	98
Tabla III.VII. Cabecera de Modbus/TCP.....	106
Tabla III. VIII. Direcciones Absolutas .....	110
Tabla III. IX. Rango de direcciones para los datos de los módulos de E / S.....	110
Tabla III. X. Rango de direcciones de datos Fieldbus. ....	110
Tabla III.XI. Características de las funciones de comunicación .....	118
Tabla III.XII. Protocolos soportados en Ethernet .....	137
Tabla III.XIII. Modos de recepción del área de recepción .....	138
Tabla III.XIV. Características Técnicas del cable UTP 2 Par Trenzado. ....	144
Tabla III.XV. Características Eléctricas del cable UTP 2 Par Trenzado.....	144
Tabla III.XVI. Características Técnicas del cable UTP 4 Par Trenzado. ....	145
Tabla III.XVII. Características Eléctricas del cable UTP 4 Par Trenzado.....	145
Tabla III.XVIII. Características Técnicas del cable FTP 4 Par Trenzado .....	146
Tabla III.XIX. Características Eléctricas del cable FTP 4 Par Trenzado.....	147
Tabla IV.XX. Características de la red .....	200
Tabla IV.XXI. Direcciones de la red.....	201
Tabla IV.XXII. Direcciones de los Procesos .....	201
Tabla IV.XXIII. Disposición del cable de conexión .....	206
Tabla IV.XXIV. Asignación de Entradas y Salidas del PLC .....	210
Tabla IV.XXV. Tabla de Ecuaciones.....	212
Tabla IV.XXVI. Asignación de Entradas y Salidas del PLC WAGO .....	214
Tabla IV.XXVII. Tabla de Ecuaciones para el PLC WAGO.....	215
Tabla IV.XXVIII. Asignación de Entradas y Salidas del PLC TWIDO.....	217
Tabla IV.XXIX. Tabla de Ecuaciones para el PLC TWIDO.....	219
Tabla IV.XXX. Asignación de Entradas y Salidas del PAC FIELD POINT FP-1601.....	221
Tabla IV.XXXI. Tabla de Ecuaciones para el FP 1601 para el funcionamiento del garaje.....	224
Tabla IV.XXXII. Tabla de Ecuaciones para el FP-1601 para el funcionamiento de las luces.....	224
Tabla IV.XXXIII. Tabla de Ecuaciones para el FP-1601 para el funcionamiento de las luces.....	225
Tabla IV.XXXIV. Áreas de Direccionamiento IEC-61131-3 .....	240
Tabla V.XXXV. Tabla de resultados y ponderaciones pregunta 1 .....	284
Tabla V.XXXVI. Tabla de resultados y ponderaciones pregunta 2 .....	285
Tabla V.XXXVII. Tabla de resultados y ponderaciones pregunta 3 .....	285
Tabla V.XXXVIII. Tabla de resultados y ponderaciones pregunta 4 .....	286
Tabla V.XXXIX. Tabla de resultados y ponderaciones pregunta 5.....	287
Tabla V.XL. Tabla de resultados y ponderaciones pregunta 6 .....	287
Tabla V.XLI. Tabla de resultados y ponderaciones pregunta 7 .....	288
Tabla V.XLII. Pesos de las variables de las preguntas de la encuesta .....	289

# INTRODUCCIÓN

La comunicación en las plantas industriales se ha hecho imprescindible en la industria actual. Muchos sistemas están conformados por equipos de diferentes fabricantes y funcionan en diferentes niveles de automatización, estos equipos a menudo se encuentran distanciados entre sí, pero se desea que trabajen de forma coordinada para obtener un resultado satisfactorio, ese es el objetivo principal de la comunicación integrada más conocida como CIM (Computer Integrated Manufacturing).

En la industria coexisten una serie de equipos y dispositivos dedicados al control de una maquina o la realización de un proceso. Entre estos dispositivos tenemos a los Controladores programables, sensores, actuadores, etc. El desarrollo de redes industriales ha establecido una forma de unir todos estos dispositivos, aumentando el rendimiento y mejorando su eficiencia.

Una de las redes mayormente utilizadas en la industria actualmente es el Ethernet Industrial, puesto que las secuencias de procesos y producción son controladas por un modelo cliente/servidor con controladores, PLC, etc., ofreciendo las ventajas del Ethernet tradicional aplicado al entorno industrial, además de ser un estándar con variedad de equipos y software desarrollados para su implementación.

# **CAPÍTULO I**

## **MARCO REFERENCIAL**

### **1.1. ANTECEDENTES**

Las comunicaciones industriales han sido un reto a vencer, por la importancia que éstas tienen sobre el desarrollo de los procesos industriales. Hace algunos años la comunicación de datos desde las máquinas hacia el departamento de mantenimiento o producción se la realizaba mediante papeles de oficina o formularios grandes para llenarlos.

Las comunicaciones entre los instrumentos de proceso y el sistema de control se basaban principalmente en señales analógicas tradicionales (neumáticas de 3 a 15 psi en las válvulas de control y electrónicas de 4 a 20 mA de CC). Pero hoy en día las comunicaciones industriales se manejan en la mayoría de los casos con instrumentos digitales capaces de manejar gran cantidad de datos y guardarlos históricamente. En

vez de transmitir cada variable por un par de hilos, transmiten secuencialmente las variables por medio de un cable de comunicaciones llamado bus.

La integración de los diferentes equipos y dispositivos existentes en una planta, se hace dividiendo las tareas entre grupos de diferentes procesadores, formando un modelo jerárquico de fabricación asistida por computador (CIM), que de acuerdo a la función y el tipo de conexiones se dividen en varios niveles de control, permitiendo que los datos de producción y de servicios estén disponibles, prácticamente en toda la planta y a disposición de ser accedidos cuando se desee, por el personal de control desde sus puestos, e influenciar con sus acciones en todo el proceso.

Ethernet Industrial es similar a la Ethernet convencional, pero está rediseñada para ser utilizada en plantas tanto de procesos continuos como de manufactura; utiliza componentes y medios Ethernet estándar para reemplazar a los protocolos especializados tradicionalmente de las redes industriales y para satisfacer los requerimientos, incorpora características de robustez, redundancia y durabilidad, que permiten a los diferentes dispositivos seguir conectados a pesar de las diversas condiciones agresivas que se trabajan en planta.

Hoy en día la utilización de Ethernet Industrial se lo hace con controladores o sistemas que usan un mismo protocolo específico de Ethernet industrial como Modbus TCP/IP, Profinet, Ethernet/IP, entre otros, debido a lo flexible para converger y poder

comunicarse con protocolos basados en la misma tecnología y poder ocupar la misma red sin tener q cambiar su topología.

## **1.2. JUSTIFICACIÓN DEL PROYECTO DE TESIS**

Ethernet es ya la red elegida en el mundo de automatización de oficinas, su presencia a nivel de control industrial proporciona comunicación desde el sensor hasta la oficina, con lo que la integración será una tarea mucho más fácil para los fabricantes. En términos de velocidad y tasa de transferencia de datos, Ethernet excede confortablemente a los buses de campo de alta velocidad como son Profibus y ControlNet.

Ethernet Industrial permite a las empresas tomar datos de una línea de manufactura y utilizarlos en el software corporativo, como por ejemplo las aplicaciones de control de inventarios y de gestión de activos. Todos los dispositivos de entrada/salida pueden trabajar con la Web, con solo colocar una dirección IP en los sistemas, toda la información de diagnóstico sobre ellos estará disponible en tiempo real para ser utilizada por los encargados de las tareas de diagnóstico, mantenimiento y monitorización remota.

Los usuarios pueden implementar topologías de red redundantes y también salvar mayores distancias a través de enlaces de fibra óptica. Si se combina con E/S remotas, Ethernet también permite la comunicación con dispositivos de campo HART. Resulta posible incluso intercambiar variables auxiliares HART junto con otros datos.

Ethernet industrial ofrece una arquitectura de red flexible y rentable para sus instalaciones de producción con un futuro garantizado.

La utilización de una red Ethernet Industrial en el campo académico podrá demostrar su facilidad de uso, la convergencia de diferentes procesos en una sola red, la comunicación entre cada uno de los procesos modulares, y la posibilidad de tener acceso a cada sensor de la red sin la necesidad de trasladarse hasta el sensor, además la excelente escalabilidad que presenta una red Ethernet industrial sin tener que cambiar mayormente la topología de la misma.

En la aplicación a nivel didáctico, la comunicación Ethernet industrial permitirá el control, monitoreo y reprogramación de cada proceso modular existente en la red sin la necesidad de desconectarla de la misma y trasladarse al punto donde se encuentre el controlador del proceso. También se podrá incorporar nuevos procesos modulares a la red sin mayores complicaciones. Los procesos modulares que se utilizara para la red Ethernet Industrial serán una mesa de indexación, una envasadora y llenado de latas, un semáforo, y una casa automatizada, procesos independientes que pueden converger en una sola red.

## **1.3. OBJETIVOS**

### **1.3.1. OBJETIVO GENERAL**

Diseñar e implementar una red de comunicación Ethernet didáctica entre PLC's para el control y monitoreo de procesos modulares.

### **1.3.2. OBJETIVOS ESPECÍFICOS**

- Diseñar y construir el módulo didáctico de comunicaciones vía Ethernet industrial con Switch no gestionados, controladores programables de diferentes fabricantes como son los PLC's, y elementos de conexión para el mejor manejo y configuración de una Red Ethernet Industrial.
- Programar los controladores de diferentes fabricantes para el control independiente de los procesos modulares en su propio software de programación con un lenguaje adecuado dependiendo las características de cada uno.
- Establecer la comunicación entre los controladores programables de diferentes fabricantes para la transmisión de datos mediante la interconexión de Switch's no gestionados con el stack de protocolos TCP/IP.
- Diseñar e implementar una base de datos para el registro de eventos o variables de interés de los procesos modulares mediante la utilización de un sistema de gestión de base de datos relacional con MySQL y Labview.
- Diseñar y programar una interfaz de usuario para el control y monitoreo de todo el sistema didáctico con la utilización de software Labview mediante la configuración de un OPC server.

#### **1.4. HIPÓTESIS**

El diseño y la implementación de una Red didáctica de comunicación Ethernet permitirán mejorar el control y monitoreo de procesos modulares con controladores programables de diferentes fabricantes.

## **CAPÍTULO II**

### **MARCO TEÓRICO**

#### **2.1. INTRODUCCIÓN A LA AUTOMATIZACIÓN**

Un operador que realiza una tarea monótona, requiere una gran necesidad mental y sensorial de su parte, por a ello, al perder el interés o entusiasmo puede causar un descuido que podría afectar la producción, causar accidentes, o el desperdicio de material, debido a esto las maquinas reemplazan este tipo de tareas y los hombres realizan trabajos más calificados.

La automatización usa sistemas computarizados, electromecánicos y sistemas de control para mejorar un proceso, mejorando su eficiencia y reduciendo en lo posible la intervención humana, reduce los riesgos que se pueden generar debido a un trabajo monótono en un proceso.

Algunas de las ventajas de aplicar automatización a un proceso son:

- Reemplazo de operador humano en tareas fuera del alcance de sus capacidades, como levantar cargas pesadas, trabajos en ambientes extremos o tareas que necesiten manejo de una alta precisión o de alto riesgo.
- Mejoramiento de la productividad, al mantener la línea de producción automatizada las demoras del proceso son mínimas, el tiempo de ejecución se disminuye según el proceso, pudiendo proveer los productos de buena calidad en cantidades necesarias y en el tiempo establecido manteniendo un stock necesario del producto de fabricación.
- Siendo la automatización como una disciplina de ingeniería es más que un sistema de control, abarca la instrumentación industrial, que incluye los sensores, los transmisores de campo, los sistemas de control y supervisión, los sistemas de transmisión y recolección de datos y las aplicaciones de software en tiempo real para supervisar, controlar las operaciones de plantas o procesos industriales.
- Se reduciría costos en la materia prima, mano de obra, materiales, energía.
- Integrar la gestión y producción (1).

## **2.2. AUTOMATIZACIÓN INDUSTRIAL**

En el enfoque de la industria, la automatización industrial requiere de un eficiencia para aprovechar en su totalidad los recursos al menor coste posible obteniendo un éxito económico, maximizando el rendimiento de las maquinas procesos industriales,

automatizándolos, y aportando con la comunicación entre las diferentes niveles de la pirámide de automatización.

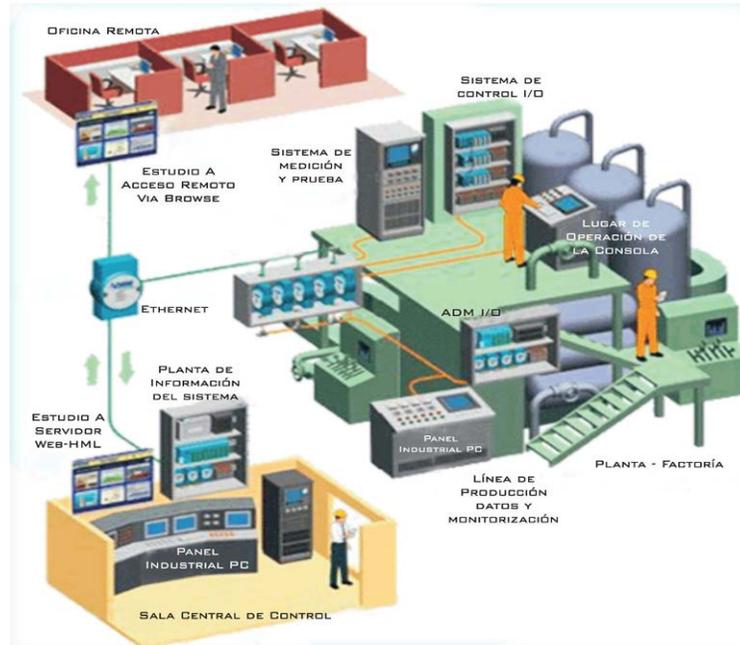


Figura II.1. Automatización en diferentes niveles de la pirámide de control  
Fuente: <http://www.automatizacion-industrial.es/>

### 2.2.1. Tipos de automatización industrial

- **La automatización fija:** Es un sistema en el cual la secuencia del ensamblaje está determinada por la configuración del equipo de producción, se utiliza cuando se fabrica un único producto y tiene una alta demanda. El ciclo de vida de la automatización dependerá del tiempo de permanencia del producto en el mercado por lo tanto hay poca flexibilidad para aceptar cambios en el producto.
- **La automatización programada:** El equipo de producción está diseñado para cambiar las secuencias de operaciones adaptándose a las variaciones del producto, estos cambios se realiza atreves de un programa informático. Se

emplea cuando el volumen de fabricación es bajo y existe diversidad en la producción a realizar.

- **La automatización flexible:** Es aquel que puede producir una variedad de productos (o partes) con ninguna pérdida de tiempo de producción debido a la capacidad de cambiar los programas de las piezas, los sistemas flexibles suelen estar formados por una serie de puestos de trabajo interconectados entre sí por sistemas de almacenamiento y manipulación de materiales, controlados en su conjunto por un ordenador.

### **2.2.2. Tecnologías empleadas en la automatización**

**Tecnología Cableada:** Se realiza interconectando físicamente los elementos que lo integran según la manera de ser conectado los elementos será el funcionamiento que realice.

Dispositivos que se utilizan tecnologías cableadas son:

- Automatismos eléctricos a base de contactores.
- Mandos neumáticos, oleohidráulica.
- Tarjetas electrónicas

La tecnología cableada ha sido extensamente empleada, pero presenta los siguientes inconvenientes

- Difícil a la hora de detectar las fallas
- Es poco flexible ante modificaciones o ampliaciones.

- Es difícil de mantener
- No es útil en aplicaciones en controles complejos

**Tecnología programada**, la parte de control se realiza creando un programa residente en la memoria de una unidad de control. Los controladores Programables pertenecen a la tecnología programada, el cual entre sus ventajas están todos los inconvenientes de la tecnología cableada.

### **2.2.3. Herramientas para la automatización**

Existen diferentes tipos de herramientas para la automatización como:

- ANN - Artificial neural network
- DCS - Distributed Control System
- HMI - Human Machine Interface
- SCADA - Supervisory Control and Data Acquisition
- PLC - Programmable Logic Controller
- PAC - Programmable automation controller
- Instrumentación
- Control de movimiento
- Robótica (2)

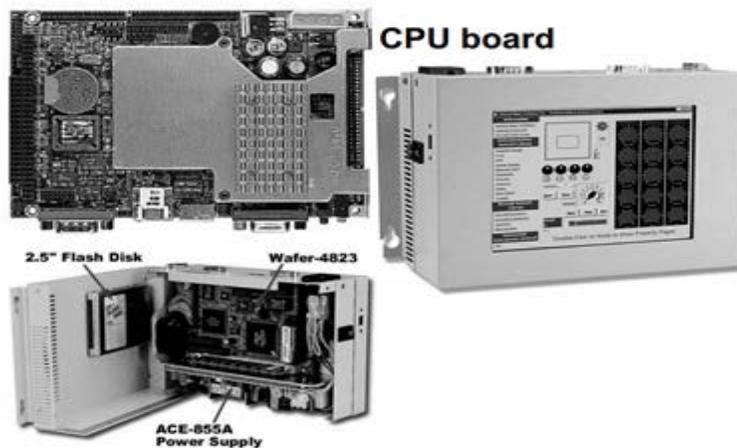
## 2.3. CONTROLADORES PROGRAMABLES

### 2.3.1. PC

La automatización basada en PC es más eficaz en situaciones en las que el proceso en cuestión necesita una mayor integración entre el control y la automatización de alto nivel de los programas de PC o la funcionalidad de otro PC, pero las PCs con sistemas operativos estándar y hardware genérico no ofrecen la confiabilidad debido a que son vulnerables como para satisfacer la demanda en control industrial.

#### 2.3.1.1. PC Industriales

PCs industriales (IPC) son equipos robustos que cumplen con las exigencias modernas de conectividad e integración, flexibilidad, fiabilidad, el mayor grado de protección IP de acuerdo a las necesidades, capacidad de expansión de tarjetas o módulos de expansión I/O para poder realizar el control de una máquina o proceso, con un sistema operativo funcional que permita operar con múltiples variables de entrada y salida (3).



**Figura II.2.** PC industrial de panel PPC-5012-ACE855

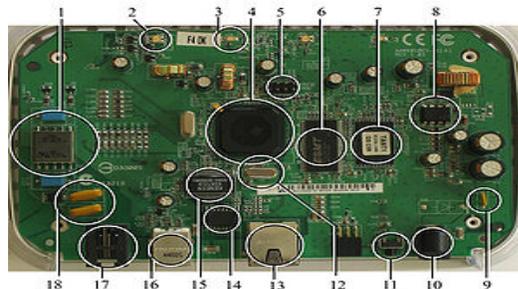
Fuente: <http://www.uv.es/>

### 2.3.1.2. PC-based Controller

Un sistema de control basado en PC está formado por la computadora personal, el software para el control, y los dispositivos d E/S, su función es controlar a través de los dispositivos de E/S las señales de campo debido al algoritmo de control que corre en la PC (4).

### 2.3.1.3. Controladores Embebidos

Un sistema embebido o empotrado es un sistema de computación diseñado para realizar una o algunas pocas funciones dedicadas a cubrir necesidades específicas. En un sistema embebido la mayoría de los componentes se encuentran incluidos en la placa base (5).

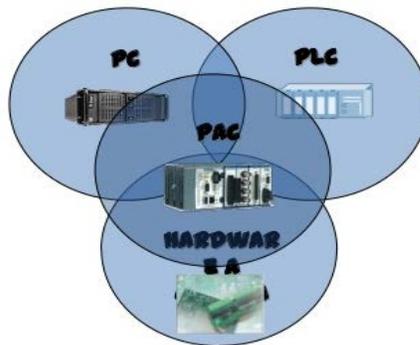


**Figura II.3.** Imagen del interior de un módem/enrutador ADSL.  
*Fuente: <http://es.wikipedia.org/>*

### 2.3.2. PAC

PAC (Controlador de Automatización Programable) es el resultado de la convergencia de la tecnologías PC+PLC+ Solución a la medida. Los PAC poseen un controlador compacto que ofrece funciones de control avanzado, conectividad de redes, interoperabilidad de dispositivos e integración de datos de la empresa.

Los PAC combinan robustez, expandible, determinismo, fiable, control, del PLC con conectividad, multitarea, capacidad de procesamiento, capacidad de almacenamiento, y funcionalidad de software de la PC por esta razón la PAC se ha convertido en una parte fundamental en la automatización cumpliendo los requisitos en aplicaciones industriales.



**Figura II.4.** PAC, convergencia entre PC, PLC y hardware.

*Fuente: <http://www.slideshare.net/>*

#### **Beneficios:**

- Alcanzar requerimientos más complejos con hardware más sencillo.
- Expandibles.
- Conectividad. Adquisición de datos en tiempo real
- Inversión gradual.
- Hace un manejo eficiente de las comunicaciones permitiendo el intercambio de datos ya que emplean protocolos estándar y tecnologías de red, tales como Ethernet, OPC, y SQL (6).

### **2.3.2.1. Compact FieldPoint**

National Instruments Compact FieldPoint es un controlador de automatización programable expandible, diseñado para aplicaciones de medición, control industrial y data logging que requieran hardware confiable y robusto. Estos sistemas incluyen típicamente diversos sensores y actuadores ubicados centralmente o distribuidos sobre grandes distancias. La arquitectura posee capacidad de operar en red, para E/S distribuidas, así como también una opción para procesar datos en un sistema operativo de tiempo real, el cual ejecuta aplicaciones desarrolladas en LabVIEW.



**Figura II.5. COMPACT FIELDPOINT**  
*Fuente: [http://www.tracnova.com/*

Los módulos de E/S industriales del Compact FieldPoint filtran, calibran y escalan señales crudas provenientes de sensores a unidades de ingeniería y realizan autodiagnósticos para hallar problemas. Las interfaces de comunicación de la red Compact FieldPoint automáticamente publican las mediciones a través de una red corporativa o mundial. Se pueden acceder puntos de E/S que se hallen cerca o a kilómetros del sitio utilizando el mismo marco de software de escritura y lectura.

Las principales ventajas que ofrece Compact FieldPoint son:

- Algoritmos embebidos y avanzados de control, análisis de datos y procesamiento de señales.
- Rendimiento de tiempo real para aplicaciones de control.
- Web browser remoto e interface FTP.
- Interface con aplicaciones empresariales.
- Data logging fácil de configurar (7).

### 2.3.3. PLC (PLC CONTROLADOR LÓGICO PROGRAMABLE)

Según la National Electrical Manufacturers Association (NEMA), un PLC es un dispositivo electrónico digital que utiliza una memoria programable para almacenar instrucciones y para implementar funciones específicas tales como funciones lógicas, secuenciales, de temporización, de conteo y aritméticas para controlar máquinas y procesos.



**Figura II.6.** PLC (Controlador Lógico Programable)  
*Fuente: <http://maleonhe.blogspot.com/2010/05/plc.html>*

Los PLCs usualmente se encuentran en aplicaciones como la robótica, maniobra de máquinas, maniobra de instalaciones, señalización y control, manejo de materiales, en líneas de transferencia automática, etc.

Dentro de las ventajas de los PLCs tenemos las siguientes:

- Permite programar fácilmente y reduce el tiempo de inactividad del proceso.
- Capacidad de entradas y salidas
- Velocidad de operación.
- Está diseñado para trabajar en condiciones severas.
- Si el controlador queda pequeño para el proceso industrial puede seguir siendo de utilidad en otras máquinas o sistemas de producción.
- Controla sistemas complejos

Como el principal inconveniente que podría presentar tenemos:

- Adiestramiento de técnicos y Costo inicial.

### *2.3.3.1. Estructura del PLC*

#### **ESTRUCTURA EXTERNA**

Arquitectura del PLC se entiende la estructura que adopta su sistema físico, esencialmente la unidad de control, el sistema de entradas y salidas y la memoria de la máquina, a fin de adaptarlo a las particularidades de la aplicación.

- **Estructura Compacta:** Suelen integrar en el mismo bloque la alimentación, entradas y salidas y/o CPU. Es posible conectar varias de estas unidades de

expansión sobre el mismo bus interno del controlador, unidas todas ellas a la misma CPU hasta que el límite de direccionamiento que ésta lo permita.

- **Estructura Modular:** este tipo de PLC se divide en módulos o partes, que realizan funciones específicas con sus propias unidades de proceso, y con sus propias interfaces de E/S. Estos módulos se encuentran conectados a la unidad central (CPU principal, maestra, supervisora, etc.) que gestiona de forma general el sistema y permite el intercambio de datos entre los elementos, procesadores o interfaces, del mismo.



**Figura II.7.** PLC Compacto y PLC Modular  
Fuente: <http://www.inatech.com.mx/>

## ESTRUCTURA INTERNA

Un PLC, es un equipo electrónico de control con un cableado interno (hardware) independiente del proceso a controlar, que se adapta a dicho proceso mediante un programa específico (software) que contiene la secuencia de operaciones a realizar. Esta secuencia de operaciones se define sobre señales de entrada y salida al proceso, cableadas directamente en los bornes de conexión del PLC.

Las señales de entrada pueden proceder a elementos digitales, o analógicos mientras las señales de salida son órdenes digitales o señales analógicas de corriente o tensión, que se envían a los elementos indicadores y actuadores del proceso.

Dentro de los elementos que conforman la arquitectura interna del PLC tenemos:

- Unidad central de proceso (CPU).
- Memorias internas.
- Interfaces de entrada.
- Interfaces de salida.
- Fuente de alimentación.

## INTERFACES DE ENTRADA Y SALIDA

Tabla II.I. Tipos y funciones de las interfaces de entrada /salida

TIPOS	CODIFICACIÓN	SENTIDO	FUNCIONES DE INTERFAZ
TODO O NADA	BINARIA 1 BIT	ENTRADAS	Adaptación de niveles de tensión. Filtrado de Perturbaciones. Aislamiento Eléctrico.
		SALIDAS	Adaptación de niveles de tensión. Ampliación de corriente. Aislamiento Eléctrico.
CONTINUAS	ANALÓGICAS (0, $\pm 10$ V) (4, 20 mA)	ENTRADAS	Adaptación y filtrado de Señal. Conversión A/D
		SALIDAS	Conversión D/A Adaptación a 0 - $\pm 10$ V o 4 - 20 mA
	DIGITALES (8, 16, 32 ... bits)	ENTRADAS	Selección de canal y multiplexado Conversión de códigos
		SALIDAS	Conversión de códigos (BIN-ASCII-7 segmentos) Amplificación de corriente
		BIDIRRECCIONALES	Conversión de código (serie- paralelo) Protocolo de dialogo (hardware y software)

Fuente: <http://148.206.53.231/UAMI10300.PDF>

### ***2.3.3.2. LENGUAJES DE PROGRAMACIÓN DE PLC'S***

Un programa es un conjunto de instrucciones, órdenes y símbolos reconocibles por el PLC, a través de su unidad de programación, que le permiten ejecutar una secuencia de control deseada. El Lenguaje de Programación en cambio, permite al usuario ingresar un programa de control en la memoria del PLC, usando una sintaxis establecida.

El usuario escribe el programa en el lenguaje de programación que mejor se adapte a su trabajo y con el que sienta poseer un mejor dominio. Es importante señalar, que algunos fabricantes no ofrecen todas las formas de representación de lenguajes de programación, por lo que el usuario deberá adaptarse a la representación disponible, además de tomar en cuenta las normas que rigen tanto para los controladores programables como para su lenguaje de programación, estas normas son la NORMA IEC 1131 y la IEC1131-3 respectivamente.

#### **NORMA IEC 1131-3**

La Comisión Electrotécnica Internacional (IEC) desarrolló el estándar IEC 1131, en un esfuerzo para estandarizar los Controladores Programables. Uno de los objetivos del Comité fue crear un conjunto común de instrucciones que podría ser usado en todos los PLCs. El estándar IEC 1131 para controladores programables consiste de cinco partes, una de las cuales hace referencia a los lenguajes de programación y es referida como la IEC 1131-3.

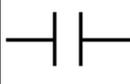
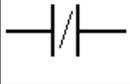
Este estándar define dos lenguajes gráficos y dos lenguajes basados en texto, para la programación de PLCs. Los lenguajes gráficos utilizan símbolos para programar las instrucciones de control, mientras los lenguajes basados en texto, usan cadenas de caracteres para programar las instrucciones.

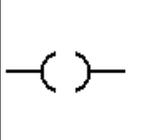
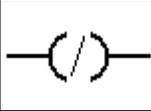
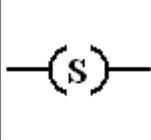
- Lenguajes Gráficos
  - Diagrama Ladder (LD)
  - Diagrama de Bloques de Funciones (FBD)
- Lenguajes Textuales
  - Lista de Instrucciones (IL)
  - Texto Estructurado (ST) (8)

## 1. LENGUAJE LADDER

El LADDER, también denominado lenguaje de contactos o de escalera, es un lenguaje de programación gráfico muy popular dentro de los Controladores Lógicos Programables (PLC), debido a que está basado en los esquemas eléctricos de control clásicos. Su principal ventaja es que los símbolos básicos están normalizados según normas NEMA y son empleados por todos los fabricantes.

**Tabla II.II. Simbología de la programación Ladder**

Símbolo	Nombre	Descripción
	Contacto NA	Se activa cuando hay un uno lógico en el elemento que representa, esto es, una entrada (para captar información del proceso a controlar), una variable interna o un bit de sistema.
	Contacto NC	Su función es similar al contacto NA anterior, pero en este caso se activa cuando hay un cero lógico, cosa que deberá de tenerse muy en cuenta a la hora de su utilización.

	Bobina NA	Se activa cuando la combinación que hay a su entrada (izquierda) da un uno lógico. Su activación equivale a decir que tiene un uno lógico. Suele representar elementos de salida, aunque a veces puede hacer el papel de variable interna.
	Bobina NC	Se activa cuando la combinación que hay a su entrada (izquierda) da un cero lógico. Su activación equivale a decir que tiene un cero lógico. Su comportamiento es complementario al de la bobina NA.
	Bobina SET	Una vez activa (puesta a 1) no se puede desactivar (puesta a 0) si no es por su correspondiente bobina en RESET. Sirve para memorizar bits y usada junto con la bina RESET dan una enorme potencia en la programación.
	Bobina RESET	Permite desactivar una bobina SET previamente activada.

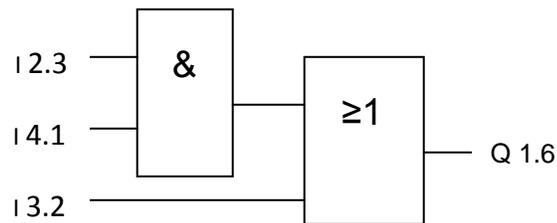
Fuente: <http://es.wikipedia.org/>

## 2. LENGUAJE BOOLEANO (Lista de Instrucciones)

El lenguaje Booleano utiliza la sintaxis del Álgebra de Boole para ingresar y explicar la lógica de control. Consiste en elaborar una lista de instrucciones haciendo uso de operadores Booleanos (AND, OR, NOT, etc.) entre otras para implementar el circuito de control.

## 3. DIAGRAMA DE FUNCIONES (FBD)

Es un lenguaje gráfico que permite al usuario programar elementos en tal forma que ellos aparecen interconectados al igual que un circuito eléctrico. Generalmente utilizan símbolos lógicos para representar al bloque de función. Las salidas lógicas no requieren incorporar una bobina de salida, porque la salida es representada por una variable asignada a la salida del bloque.

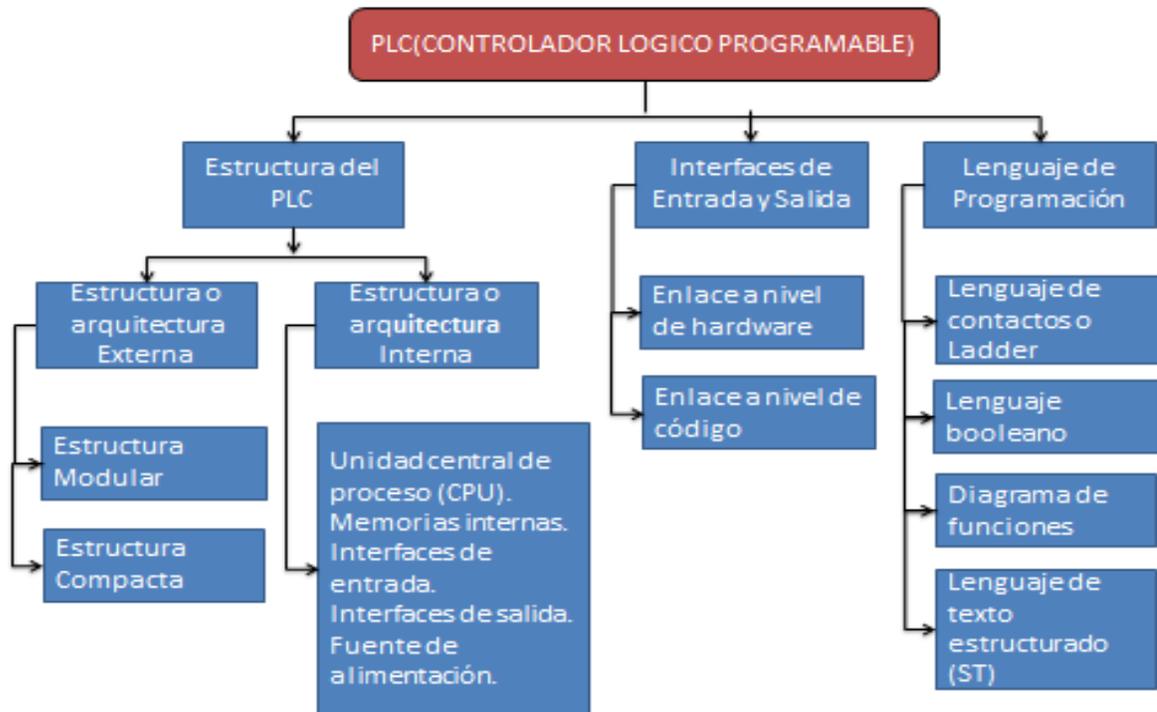


**Figura II.8.** Ejemplo de programación mediante diagrama de funciones  
*Fuente: <http://www.slideshare.net/>*

#### 4. LENGUAJE DE TEXTO ESTRUCTURADO (ST)

Es un lenguaje de alto nivel que permite la programación estructurada, es decir que muchas tareas complejas pueden ser divididas en unidades más pequeñas. Esta programación es apropiada para aplicaciones que involucran manipulación de datos, ordenamiento computacional y aplicaciones matemáticas que utilizan valores de punto flotante. ST es el mejor lenguaje para la implementación de aplicaciones de inteligencia artificial, lógica difusa, toma de decisiones, etc. (9).

```
IF Manual AND Alarm THEN
Level = Manual_Level;
Mixer = Start AND NOT Reset
ELSE IF      Other_Mode THEN
              Level = Max_level;
ELSE        Level = (Level_Indic X100)/Scale;
END IF;
```



**Figura II.9.** Diagrama Del PLC  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

## 2.4. REDES INDUSTRIALES

### 2.4.1. INTRODUCCIÓN

Debido a la llegada de los sistemas de automatización basadas en redes de campo y la tecnología digital trajo una gran variedad de soluciones propietarias de comunicación en el ambiente industrial, de allí surge la necesidad de la intercomunicación en tiempo real, debe ser capaces de resistir a un ambiente hostil, condiciones ambientales duras, e integrar una comunicación de campo y una comunicación hacia el SCADA es decir conectar distintos procesos para esto se soluciona con una red industrial que permita la integración e interoperabilidad de

equipos de control y dispositivos de campos de diferentes fabricantes hacia el sistema de producción.

La solución completa debe proveer una metodología de gestión industrial transparente y garantizar que todos los esfuerzos se dirijan a la misma meta establecida, facilitando la toma de decisión cuando haya cambios en el desempeño de los indicadores, o una desviación del planeado. Cuanto mayor y certera sea la información, mejor se puede operar una fábrica y, hacerla más productiva.

La comunicación se expandió rápidamente en el sentido horizontal en los niveles inferiores (de campo), como también en sentido vertical integrando todos los niveles jerárquicos. Según las características de la aplicación y el costo máximo a atingirse, una mezcla de distintos sistemas de comunicación parece ofrecer las condiciones ideales para las redes abiertas en procesos industriales siendo el elemento fundamental porque si falla no hay comunicación y el sistema distribuido no funciona. Las condiciones que se requiere es que todos los dispositivos conectados a la red tengan capacidad de manejar la comunicación teniendo una transmisión eficaz, rápida, periódica, asíncrona de información, eventualmente la posibilidad de mandar varios mensajes al mismo tiempo.

## 2.4.2. NIVELES DE COMUNICACIÓN INDUSTRIAL

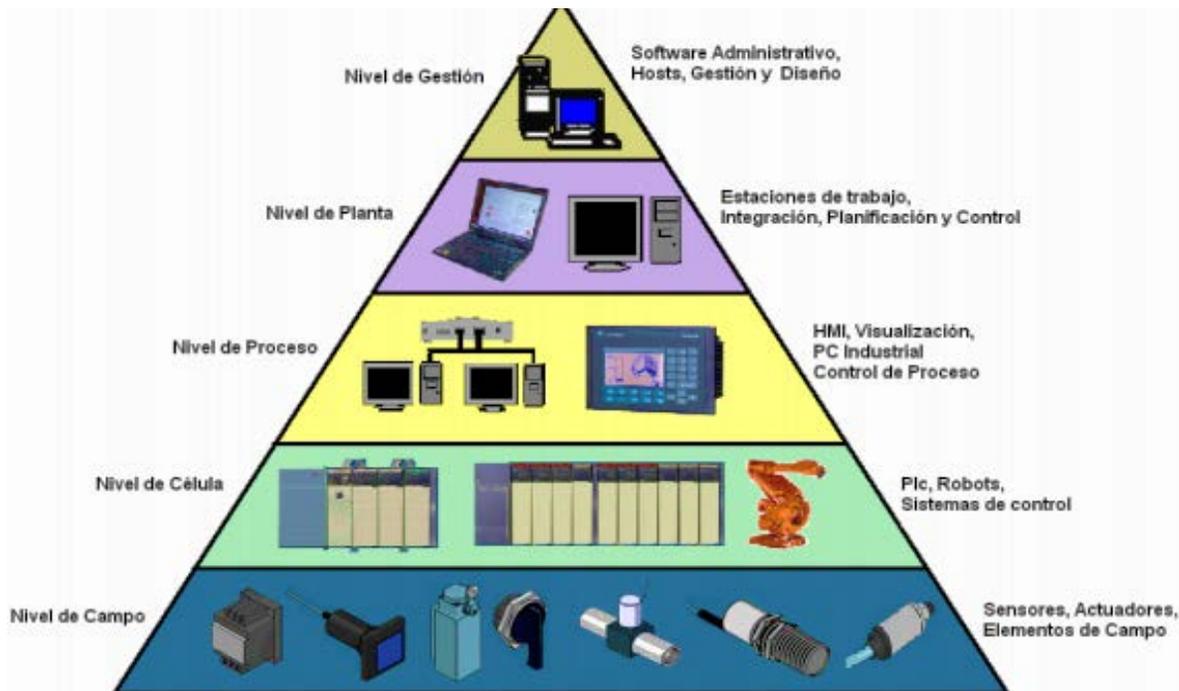


Figura II.10. Pirámide CIM

Fuente: [www.automation.siemens.com/](http://www.automation.siemens.com/)

- **Nivel de Campo (E/S):** Es el nivel más cercano al proceso de campo, aquí están las maquinas, sensores, actuadores, entradas y salidas del proceso.
- **Nivel de Célula:** Integra pequeños grupos de sistemas de control, aquí están los PLC, multiplexores de entradas y salidas, PIDs, buses de campo, controladores transmisores, etc.
- **Nivel de Proceso:** Integra células o zonas de trabajo. Se utilizan redes LAN para comunicar a los PLCs de gama alta con los computadores dedicados al diseño, programación y control.
- **Nivel de Planta:** Son estaciones de trabajo que planifican y controlan la producción.

- **Nivel de Gestión:** Integra los niveles inferiores en una estructura de empresa.

Se utilizan redes LAN y WAN.

Existen una amplia variedad de redes industriales con diferentes características y aplicaciones, algunos ejemplos que podemos mencionar son: Interbus, Profibus, CAN, CC Link, ControlNet, DeviceNet, SDS, Ethernet IP, Device/World FIP, LonWorks, Hart (10).

#### **Beneficios de una red industrial:**

- Reducción de cableado (físicamente)
- Dispositivos inteligentes (funcionalidad y ejecución)
- Control distribuido (Flexibilidad)
- Simplificación de cableado de las nuevas instalaciones
- Reducción de costo en cableado y cajas de conexión
- Aplicable a todo tipo de sistema de manufactura
- Incremento de la confiabilidad de los sistemas de producción
- Optimización de los procesos existentes

#### **2.4.3. Estructuras de comunicación de las redes industriales:**

**Master/Slave (Maestro/ Esclavo):** siempre un solo Maestro que manda y recibe información de los Esclavos

**Token Passing Access Method:** idea de pasar de forma controlada y regular una bandera de un nodo al otro. La bandera autoriza el uso de la red. Seguro y determinístico pero difícil manejar emergencias

**Control de colisiones:** acceso libre y aleatorio de cada nodo a la red pero riesgo de colisión

**Dependiendo del tipo de dispositivo podemos hablar:**

- Buses de campo: red para la automatización de procesos que se usa para interconectar células.
- Bus de control: Red para controladores y dispositivos que necesitan ser interconectados.
- Bus de dispositivos: Se utiliza para conectar bloques de E/S y dispositivos inteligentes.
- Bus de sensores: Estas redes se usan para la automatización conectando sensores y actuadores.

**Medio de transmisión de las Redes industriales:**

- Dos hilos torcidos con o sin blindaje, usualmente cable 10baseT con conector RJ-45
- Cable coaxial: buena protección contra ruido, usualmente 10base2, difícil cablear topologías complicadas
- Fibra óptica: perfecta inmunidad frente al ruido eléctrico, seguro, largo alcance, alta velocidad, usualmente 10baseF
- Radio frecuencias: red inalámbrica (wireless network)
- Uso de repetidores “activos” cuando la distancia es larga (11)

## 2.4.4. REDES INDUSTRIALES Y EL MODELO OSI

### 2.4.4.1. Modelo OSI

- 7: Aplicación:** nivel de interconexión con el usuario. Nivel de programación.
- 6: Presentación:** Maneja las reglas del protocolo de comunicación.
- 5: Sesión:** Autorización del envío del mensaje, manejo de errores de transmisión de un mensaje.
- 4: Transporte:** Separación de los mensajes en entidades más pequeñas
- 3: Red:** Preparación de las direcciones de destino y origen del mensaje.
- 2: Enlace de Datos:** Envío de poca información y manejo de errores de transmisión
- 1: Física:** Niveles de voltaje y corriente, temporización.

### 2.4.4.2. Relación con el Modelo OSI

Idealmente los buses de campo deberían cubrir los siete niveles OSI, pero en la realidad solo utilizan tres:

**Nivel Físico:** Donde se especifica el tipo de conexión, naturaleza de la señal, tipo de medio de transmisión, etc. Normalmente las especificaciones de un determinado bus admiten más de un medio físico de comunicación.



**Figura II.11.** Cable Filar AS-i  
Fuente:<http://www.automation.com/>

**Nivel de Enlace:** Especifica el protocolo de acceso al medio (MAC) y de enlace (LLC), define también los turnos de transmisión, el direccionamiento de las estaciones origen y destino de los datos y verificación de errores de las tramas.

**Nivel de Aplicación, (Usuario):** Dirigido al usuario, que le permitirá ingresar al sistema de comunicaciones. Aquí se definen las funciones de las aplicaciones que se utilizarán para el envío de mensajes entre los dispositivos del bus, la comunicación entre sistemas de automatización y dispositivos de campo. En este nivel se define el significado de los datos. Las aplicaciones suelen ser propias de cada fabricante. También permite la creación de programas de gestión y presentación, apoyándose en las funciones estándar definidas en el nivel de enlace.

#### **NIVEL FISICO,**

Los buses industriales normalmente se comunican en banda base y codificación digital NRZ (Non Return to Zero), o Manchester. La señalización consiste en la transmisión diferencial de tensión de las variables físicas observadas. En el caso de comunicaciones serie sobre conductores eléctricos, las variables utilizadas son el resultado de la diferencia de potencial o de corriente entre los valores censados y uno de referencia. Normalmente la información a transmitir corresponde a la combinación de valores binarios "1" o "0".

El medio físico por excelencia es el par trenzado, la fibra óptica se utilizará en aplicaciones que requieran alta inmunidad a las interferencias electromagnéticas. La comunicación en modo común (diferencia de potencial entre

dos conductores y masa de referencia) se utilizara en enlaces no balanceados tipo RS-232 para cortas distancias, el cual resulta susceptible a las interferencias.

### **NIVEL DE ENLACE.**

La capa de enlace es la encargada de gestionar el envío de los paquetes de datos, al cual le agrega información redundante para el control de errores, que permitirá detectar y corregir el contenido de los paquetes mal recibidos en el receptor. En los buses de campo las tramas siguen formatos derivados del protocolo HDLC o similares.

Como método de control y detección de errores se utilizan los códigos de redundancia cíclica (CRC) o la simple suma de comprobación de trama (Checksum). En los buses que utilizan comunicación asincrónica, cada palabra (byte), puede incluir un BIT de paridad que le permitirá el control de errores, además se incluye un CRC o checksum a nivel de trama.

Puesto que la mayoría de los buses industriales siguen una topología en bus, el control de acceso al medio es un tema importante, en el momento de elegir un bus industrial. Importante porque, a los requisitos de cualquier red de procesos de datos, se añaden requerimientos de tiempo real. Estos requisitos se traducen a una mayor necesidad de "determinismo". Esto obliga a consideraciones adicionales, en los cuales podemos considerar los siguientes métodos de acceso al medio:

- Maestro-esclavo.
- Técnicas de Paso de Testigos.

- Técnica de CSMA/CD o variantes como ser CSMA/CA.
- Técnicas de división de tiempo (TDMA).
- Algoritmos de BIT dominante.

### **NIVEL DE APLICACIÓN.**

Debido a que no necesitan estructuras complejas, las capas de red, transporte, sesión y presentación no son consideradas, de esta forma se pasa directamente de la capa de enlace de datos al nivel de aplicación. Pero también hay que tener en cuenta a veces que hay que realizar algún tipo de enrutamiento en la red, la gestión de envío de paquetes largos y consistentes de datos, el establecimiento de sesiones con un nodo o los cambios de presentaciones de datos, sin embargo esas características suelen incluirse como un aspecto de la especificación de la capa de aplicación y, en todo caso, se trata de reducirlas al máximo. (12).

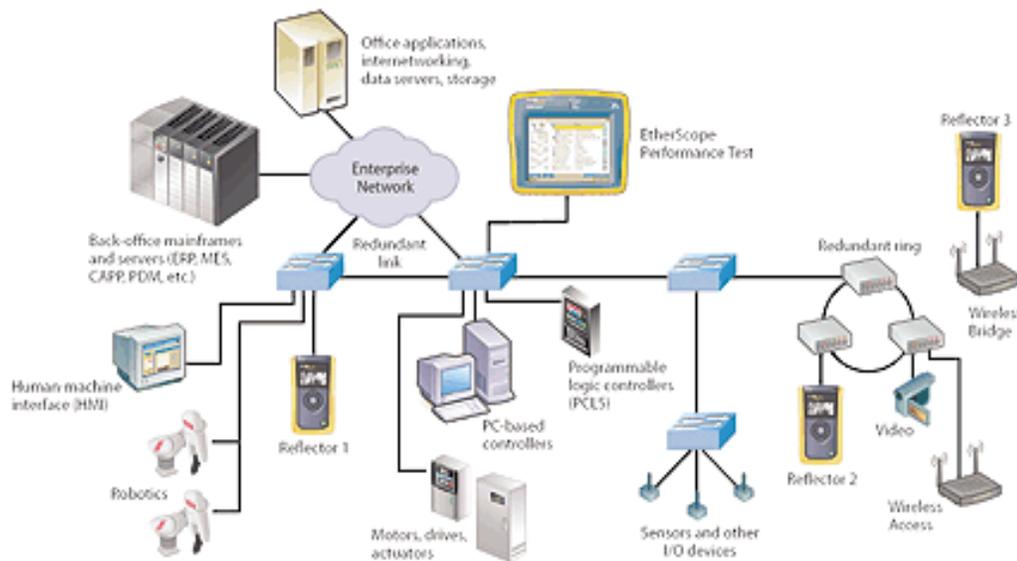
### **2.4.5. ETHERNET INDUSTRIAL**

La tendencia actual es nivelar la estructura de la red y utilizar el menor número posible de tecnologías para reducir el coste de propiedad. Ethernet se ha transformado en el estándar de la conectividad para ambientes corporativos y hogareños por su alta velocidad, bajo costo, facilidad de instalación y mantenimiento, entre otros factores. Hace algunos años, comenzó a popularizarse el concepto de Ethernet Industrial, que engloba el uso de la tecnología Ethernet para aplicaciones de control y automatización en un ambiente industrial pero para ello debe fijarse en el protocolo.

### 2.4.5.1. CARACTERÍSTICAS

#### Redundancia

Las redes Ethernet Industrial deben ser altamente confiables y seguir en funcionamiento durante duras condiciones ambientales, interrupciones accidentales de red y fallas de los equipos. La caída de una red puede ser peligrosa y cara. La confiabilidad de la red es en gran medida conseguida por el uso de Redundancia para todos los vínculos críticos. Hay cuatro esquemas de redundancia populares para Ethernet: Spanning Tree Protocol (STP), Rapid Spanning Tree Protocol (RSTP), Link Aggregation (Trunking) y topología de anillos propietaria.



**Figura II.12.** Redundancia de una red Ethernet  
*Fuente: <http://www.emb.cl/>*

#### Seguridad

Si bien la integración creciente de tecnologías de la información e Ethernet Industrial presenta el potencial para ofrecer nuevos niveles de beneficio en las operaciones

industriales, también plantea posibles vulnerabilidades. El monitoreo y el análisis de datos procedentes de sistemas de control en niveles de planta, significa que el equipo también se extiende en la otra dirección. Esto aumenta enormemente la exposición de la ampliación de la red de intrusiones y amenazas. Factores internos ofrecen diferentes riesgos. Ethernet Industrial puede usar muchos métodos para garantizar la confidencialidad e integridad de la red.

Estas medidas de seguridad de red se pueden agrupar en varias categorías, incluidas las de control de acceso y autenticación, seguridad en la conectividad y administración.

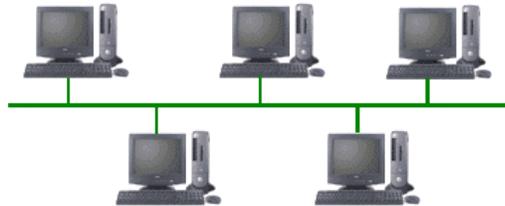
### **Monitoreo**

Los puertos espejo (Mirror Port) proporcionan a los ingenieros y técnicos herramientas de monitoreo en tiempo real para el comportamiento del sistema. El Monitoreo permite observar dentro de la operación activa de la red para tipos y cantidades de tráfico esperado. Igualmente importante es la identificación del uso inesperado de la red, para identificar fugas de la empresa a las redes de la fábrica.

El uso de puerto espejo en switches Ethernet industrial, las estadísticas y el historial pueden ser usadas para identificar las tendencias de capacidad, otorgando a los usuarios la capacidad para identificar rápidamente los problemas y a los usuarios que utilizan más el ancho de banda de un solo vistazo. Utilización de unicast, multicast, broadcast y errores pueden ser graficados hasta 18 horas para los análisis a largo plazo (13).

#### **2.4.5.2. TOPOLOGÍA DE ETHERNET INDUSTRIAL**

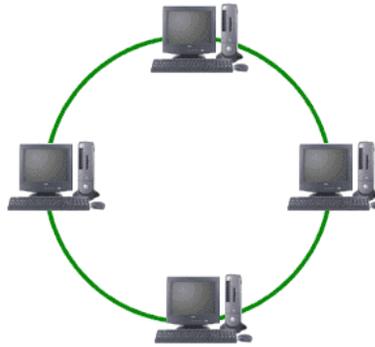
**Bus:** Las estaciones están unidas entre sí a través de unas líneas comunes compartidas por todos los nodos. Sus beneficios son flexibilidad, el fallo de una estación no provoca fallos en la red, fiabilidad, mientras que los inconvenientes pueden presentar en la rotura del cable afectaría a todos los usuarios, límites de longitud del cable.



**Figura II.13.** Topología Bus

Fuente: <http://manolin-insind.blogspot.com/>

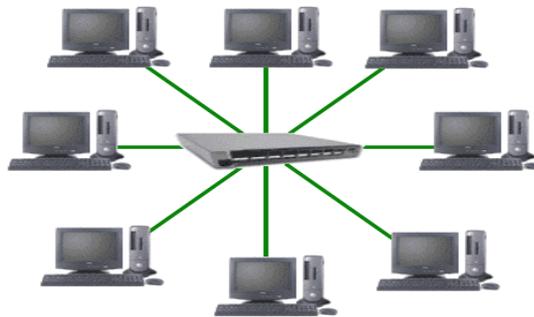
**Anillo:** Cada equipo se conecta con otros dos lazos cerrados y los datos circulan en una sola dirección de forma que cada punto recoge información del anterior, comprueba si él es el destinatario y, en caso contrario, lo retransmite al siguiente. Los beneficios son igual acceso para todos los equipos, añadir usuarios no afecta excesivamente, mientras que los inconvenientes se pueden presentar en el fallo del cable afectaría a muchos usuarios, la conexión y el cableado es costoso, y difícil añadir equipos.



**Figura II.14.** Topología Anillo

Fuente: <http://manolin-insind.blogspot.com/>

**Estrella:** Todos los nodos están conectados a un nodo central, por el cual pasan todos los datos y sirve de punto de enlace con los nodos periféricos. Entre los beneficios se encuentran la facilidad de añadir nuevas estaciones, el manejo y monitorización de la red está centralizado, la rotura de un cable solo afecta a un usuario y los inconvenientes son: mucho cableado, si falla el computador central se inutiliza la red (14).



**Figura II.15.** Topología Estrella

Fuente: <http://manolin-insind.blogspot.com/>

### ***2.4.5.3. PROTOCOLOS DE LA CAPA DE APLICACIÓN PARA ETHERNET INDUSTRIAL***

La disponibilidad de soluciones en tiempo real resulta crucial para que Ethernet sea aplicada en el sector de la automatización. Actualmente, diferentes protocolos pueden utilizarse y la elección del mismo se encuentra íntimamente ligada al estándar adoptado para las capas de los dispositivos de control, este requisito está contemplado en varios protocolos que a continuación se detallan:

- **ETHERCAT:** Significa "Ethernet para el Control de Tecnología de automatización." Se trata de un código abierto, de alto rendimiento que pretende utilizar protocolos de Ethernet en un entorno industrial.
- **FIELDBUS:** es un sistema de comunicación totalmente digital, en serie y bidireccional que conecta equipos "Fieldbus" tales como sensores, actuadores y controladores.
- **PROFINET:** Se aplica a los sistemas de automatización distribuida basados en Ethernet que integran los sistemas de bus de campo existentes, por ejemplo PROFIBUS, sin modificarlos.
- **ETHERNET POWER LINK:** El objetivo del desarrollo de Ethernet Powerlink consistió en aplicar la tecnología Ethernet estándar a la ingeniería de automatización, en condiciones de tiempo real adversas.
- **ETHERNET/IP:** Es un protocolo de comunicaciones desarrollado por Rockwell Automation, administrado actualmente por Open DeviceNet Vendors Association (ODVA) y diseñado para su uso en otros industriales de control de procesos y automatización de aplicaciones.

- **MODBUS TCP/IP:** Es una variante o extensión del protocolo Modbus que permite utilizarlo sobre la capa de transporte TCP/IP. De este modo, Modbus-TCP se puede utilizar en Internet, de hecho, este fue uno de los objetivos que motivó su desarrollo.
- **SERCOS INTERFACE:** SERCOS III combina los mecanismos en tiempo real establecidos de SERCOS y estandarizó el sistema del parámetro con las comunicaciones universales basadas en Ethernet industrial.
- **CC-LINK IE:** Es un nuevo estándar para el Ethernet industrial abierto gestionado por la CC-Link Partner Association (CLPA). Su diseño se ha realizado a partir de las exigencias de los fabricantes y de los usuarios de procesos a escala mundial (15).

Para el desarrollo de esta investigación se profundiza el estudio de los siguientes protocolos:

### ***FIELD BUS FOUNDATION***

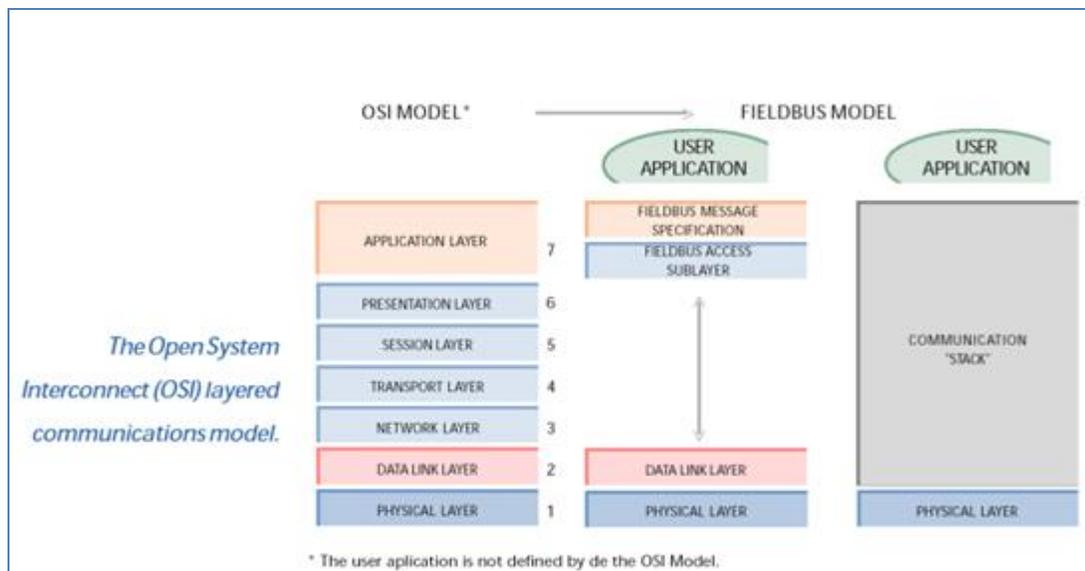
Fieldbus Foundation es un estándar abierto que se usa para la transmisión de bloques funcionales, existen bloques funcionales para representar datos del sistema y funciones del mismo. Se define principalmente tres tipos de bloques:

- **Bloques de recursos:** Describe las características del dispositivo como: el nombre, el fabricante o el número de serie. También sirve para configurar parámetros que afectan al dispositivo en conjunto.

- **Bloques de transductor:** permite la configurar los sistemas E/S de cada dispositivo. Contiene información como la calibración o tipo de sensor
- **Bloques funcionales:** Establecen las estrategias de control realizando las operaciones del sistema, cálculos numéricos, PID, adquisición de datos accionamientos de actuadores.

### FIELDBUS H1

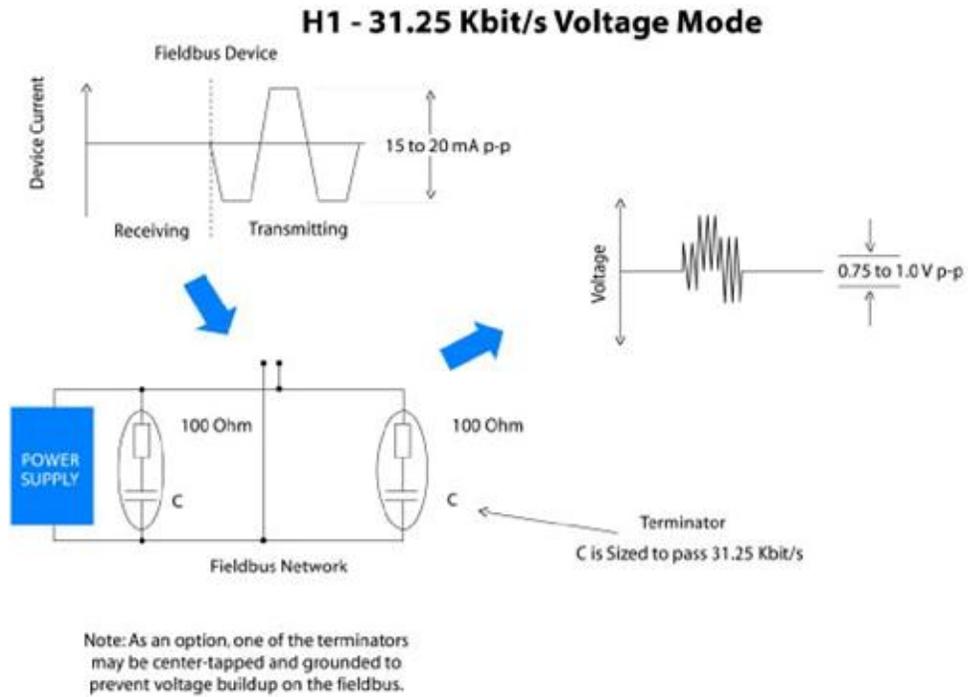
Este es un protocolo de comunicación digital en ambos sentidos que permite la conexión en red de varios equipos directamente en el campo, realizando funciones de control y monitoreo de procesos y estaciones (HMIs) a través de software de supervisión. Se basa en el estándar ISO/OSI, que contiene los siguientes estratos: Physical Layer, Communication Stack y User Application, y abarca el Fieldbus Access Sublayer (FAS), el Fieldbus Message Specification (FMS) y el modelo de Function Block más Device Descriptions.



**Figura II.16.** Estructura de estratos del Foundation Fieldbus

Fuente: <http://www.smar.com/>

El Physical Layer (medio físico) se define según los estándares internacionales IEC e ISA. Éste recibe mensajes del estrato de comunicación (Communication Stack) y los convierte en señales físicas a través del medio de transmisión Fieldbus y viceversa, incluyendo y removiendo preámbulos, y limitadores de comienzo y fin de mensajes.



**Figura II.17.** Ejemplo de señal Fieldbus en modo de tensión  
Fuente: <http://www.smar.com/>

A continuación se muestra una tabla comparativa de los diferentes buses de campo de Fieldbus.

**Tabla II.III. Comparación técnica de los tipos Fieldbus.**

	Foundation™ Fieldbus			
	H1 <sup>*1</sup>	H2 <sup>*1</sup>	H2 <sup>*1</sup>	H2 <sup>*1</sup>
Rata de transmisión [bits/s]	31.25 kB	1.0 MB	1.0 MB	2.5 MB
Comunicación	Single/ Multi-Master	Single/ Multi-Master	Single/ Multi-Master	Single/ Multi-Master
Acceso a la red	Token Passing	Token Passing	Token Passing	Token Passing
Medio de transmisión	 Par trenzado	 Par trenzado	 Par trenzado	 Par trenzado
Cantidad de Nodos máx. <sup>*9</sup>	240 por Segmento, ó 32.768 por Sistema	240 por Segmento, ó 32.768 por Sistema	240 por Segmento, ó 32.768 por Sistema	240 por Segmento, ó 32.768 por Sistema
Seguridad intrínseca?	Si	---	Si	---
Alimentación por Bus?	Si	---	Si	---
ASIC disponible? <sup>*6</sup>	Si	Planificado	planificado	planificado
Medio de transmisión Normativa	IEC 1158-2	IEC 1158-2	IEC 1158-2	IEC 1158-2
Normativa (s) aplicable (s)	ISA 850	ISA 850	ISA 850	ISA 850

Fuente: <http://www.texca.com/>

### *PROFINET (Profibus)*

PROFINet es el estándar Industrial Ethernet innovador y abierto (IEC 61158) para la automatización industrial, se desarrolló con el objetivo de favorecer un proceso de convergencia entre la automatización industrial y la plataforma de tecnología de la información de gestión corporativa y redes globales de las empresas. PROFINet se aplica a los sistemas de automatización distribuida basados en Ethernet que integran los sistemas de bus de campo existentes, por ejemplo PROFIBUS, sin modificarlos.

Básicamente hay dos tipos de redes PROFINet: PROFINet IO y PROFINet CBA. El PROFINet IO se utiliza en aplicaciones rápidas de tiempo real y el PROFINet CBA se utiliza donde el tiempo no es urgente, como en la conversión a la red PROFIBUS-DP.

Su principal foco, que revela las diferencias entre el mercado común de redes Ethernet, es la aplicación de objetos ya en uso y probados en software de tecnologías de automatización. Tras esta idea, las máquinas y fábricas pueden dividirse en módulos tecnológicos, cada uno de ellos con sus características y compromisos mecánicos, eléctrico-electrónicos y software de aplicaciones. Cada módulo entonces se encapsula según los componentes PROFINet y pueden accederse vía interfaces universales, además de interconectarse en varias aplicaciones.

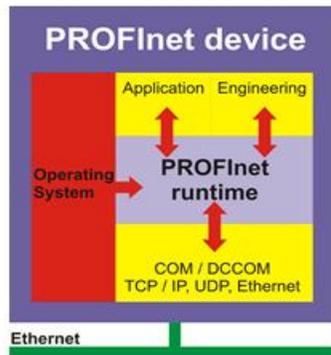


Figura II.18. Estructura de dispositivo Profinet.  
Fuente: <http://www.smar.com/>

PROFINET tiene tres modelos distintos de operación, siendo dos de ellos de tiempo real.

- Modelo TCP/IP y DCOM, para aplicaciones en las que el tiempo no es crítico.
- Soft Real Time (SRT)
- Isochronous Real Time (IRT)

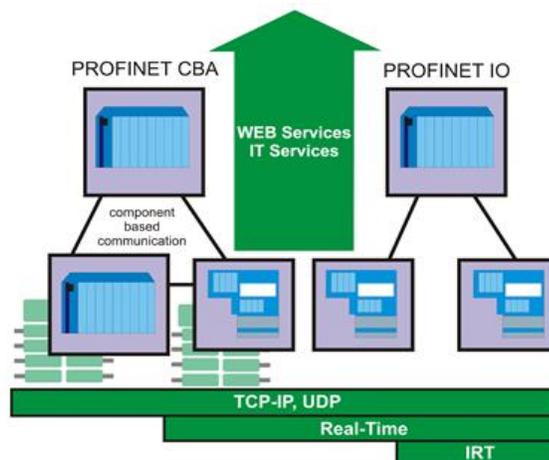


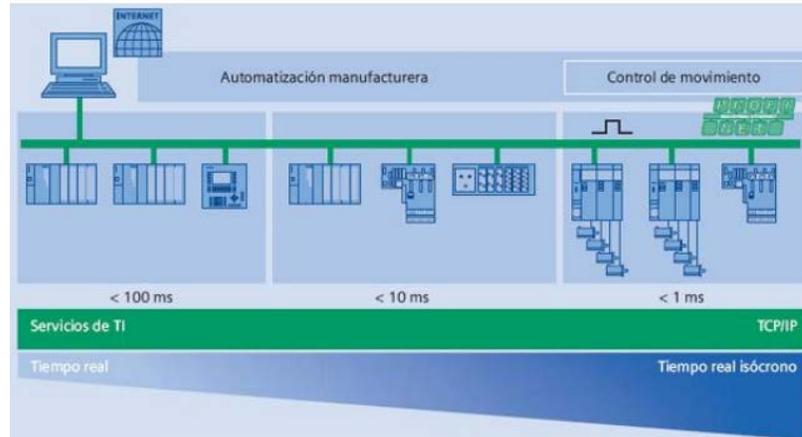
Figura II.19. Modelos de operación de Profinet.  
Fuente: <http://www.smar.com/>

1. **Modelo TCP/IP:** este modelo se basa en la arquitectura TCP/IP pura, utilizando la Ethernet en los estratos 1 y 2, el IP en el estrato 3 y el TCP o UDP en el

estrato 4. Esa arquitectura se denomina Non-real time (Non-RT), pues su tiempo de procesamiento es cercano de los 100 ms. La más importante aplicación de ese tipo de comunicación es la configuración de la red o al comunicarse con los Proxies, utilizando el PROFINET CBA. Los Proxies son conversores de protocolos.

2. **Modelo SRT:** se caracteriza por ser un canal directo entre el estrato de la Ethernet y la aplicación. Su ciclo de tiempo está en el orden de los 10 ms. El protocolo utiliza un canal de comunicación en tiempo real optimizado para las necesidades de tiempo real de los procesos de automatización. De este modo se minimizan los tiempos de ciclo y se mejora el rendimiento a la hora de actualizar los datos de proceso. Las prestaciones son comparables a las de los buses de campo, y se permiten unos tiempos de respuesta de entre 1 y 10 ms. Al mismo tiempo se reduce considerablemente la potencia de proceso necesaria en el dispositivo para la comunicación, debido a ello se pueden utilizar componentes de red estándar.
3. **Modelo IRT:** para aplicaciones de control de movimiento (ciclos de 1 ms.). La comunicación en tiempo real asistida por hardware, conocida como Isochronous Real - Time (IRT), está disponible para aplicaciones especialmente exigentes, como el control de movimiento y aplicaciones de alto rendimiento en automatización manufacturera. Con IRT se consigue un tiempo de ciclo inferior a 1 ms., con una fluctuación de menos de 1  $\mu$ s. Para ello el ciclo de

comunicaciones se divide en una parte determinista y otra abierta. Los telegramas IRT cíclicos se transmiten por el canal determinista; los telegramas RT y TCP/IP, por el canal abierto. Por lo tanto, los dos tipos de transmisión de datos coexisten sin interferirse.



**Figura II.20.** Tiempos de respuesta de Profinet.

Fuente: <http://www.proatec.com.mx/>

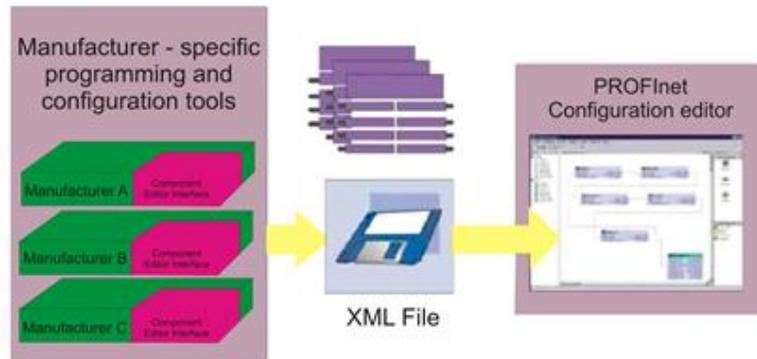
La aceptación de PROFinet depende, entre otras cosas, de si los sistemas de bus de campo existentes pueden o no ampliarse con PROFinet sin incurrir en grandes gastos. Por ejemplo el sistema de bus de campo PROFIBUS puede integrarse de dos formas distintas:

- Integración de unidades de bus de campo a través de proxies: cada unidad de campo representa un componente PROFinet independiente, cuya comunicación con otros componentes se configura mediante el editor de conexiones de PROFinet. En este caso, el Proxy representa a todas las unidades de campo de la comunicación Ethernet.

- Integración de aplicaciones de bus de campo: el segmento de bus de campo representa un componente PROFinet independiente y cuyo Proxy (por ejemplo, un controlador) incluye una interfaz PROFinet. De este modo se dispone de todas las funciones del bus de campo subordinado como si se tratase de un componente Ethernet.

Normalmente, los sistemas se componen de varios módulos tecnológicos, que funcionan de manera autónoma y se coordinan entre sí mediante un número gestionable de señales de sincronización, control de secuencia e intercambio de información. El modelo de componentes de PROFinet se basa en este tipo de módulos tecnológicos inteligentes, compuestos de una combinación de sistemas mecánicos, electrónicos y programas de usuario, formando las partes de una unidad inteligente. Externamente, la interfaz del componente se define de modo que pueda comunicarse con otros componentes pertenecientes al sistema distribuido.

Los componentes se interconectan con una aplicación a través del editor de conexiones de PROFinet, con solo pulsar el ratón en una librería determinada. Esta interconexión sustituye la dificultosa programación de las relaciones de comunicación por una sencilla configuración gráfica.

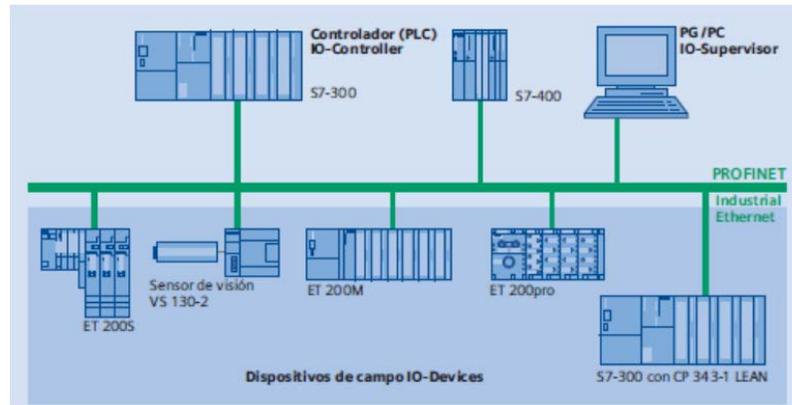


**Figura II.21.** Creación e interconexión de componentes.

*Fuente: <http://www.smar.com/>*

El modelo de componentes resulta adecuado para unidades de campo inteligente y controladores programables, y al igual que con PROFIBUS, la visualización de las entradas y salidas, permite acceder a la descripción de las unidades de campo, integrándose con los periféricos distribuidos de PROFINet. La principal función de la integración es para que el programa usuario del PLC procese los datos de entrada y salida de las unidades de campo distribuidas.

PROFINet permite las topologías de red usuales, como ser: estrella, árbol, bus y anillo. Permite aplicaciones WEB en formato HTML o XML. Independiente de las herramientas utilizadas, se puede acceder desde cualquier punto con un navegador de Internet corriente a la información del nivel de automatización, lo cual facilita considerablemente la puesta en marcha y el diagnóstico (16).



**Figura II.22.** Integración de una red PROFINET con una red Ethernet industrial  
*Fuente: <http://www.proatec.com.mx/>*

### ***ETHERNET/IP***

Otro de los protocolos utilizados en las redes Ethernet Industrial es Ethernet/IP (DeviceNet) basado en las normas IEEE 802.3, IEC 61158 y IEC 61784. Es una red abierta y compatible con la tradicional familia TCP/IP. Este protocolo permite el uso de aplicaciones de control con el protocolo de control e información CIP, utilizado como protocolo de aplicación para E/S en tiempo real.

Fundamentalmente, Ethernet/IP puede considerarse como la ampliación industrial de Ethernet TCP/IP, puesto que los mensajes CIP de la capa de aplicación se “empaquetan”, mediante encapsulación, en las tramas TCP/IP como datos de usuario.

De este modo, una aplicación puede enviar sus datos a otra aplicación a través de Ethernet; en caso necesario, la aplicación genera automáticamente un mensaje CIP y lo encapsula para convertirlo en un paquete TCP/IP. Este mensaje empaquetado se envía hasta el dispositivo de destino en el que, una vez recibido, el TCP/IP lo vuelve a enviar al protocolo de encapsulación para “desempaquetar” el mensaje original CIP y

volver a enviarlo, a través del protocolo CIP, hasta la aplicación receptora. En principio, es posible establecer este tipo de enlace de aplicación entre todos los usuarios del protocolo de aplicación CIP, incluso si proceden de distintos fabricantes o están ubicados en redes diferentes.

Esto significa que a través de TCP/IP, Ethernet/IP puede enviar “mensajes explícitos”. Aquí el receptor debe interpretar los mensajes explícitos como instrucciones, ejecutarlas y generar una respuesta. Este modo versátil de intercambio de datos se utiliza, por ejemplo, para la configuración, programación y diagnóstico de dispositivos con cantidades variables de datos. Para llevar a cabo esta transferencia de mensajes se utiliza el protocolo TCP.

No obstante, la comunicación en tiempo real impone requisitos algo distintos. En ese caso, Ethernet/IP no utiliza el protocolo TCP sino UDP vía IP (Internet Protocol).

Este protocolo resulta más compacto, por lo que es compatible con los denominados mensajes “multidifusión” (recepción simultánea por varios usuarios) y puede ser utilizado por Ethernet/IP para el envío de los denominados “mensajes implícitos”. Aquí los campos de datos ya no incluyen la información de protocolo sino sólo datos de entrada y salida en tiempo real. La aplicación receptora ya conoce cómo debe interpretar estos datos, puesto que ya lo ha negociado durante la configuración de la conexión. Es decir, que los telegramas implícitos se envían a través de una conexión virtual existente entre los usuarios y se actualizan de manera constante y cíclica en cortos intervalos de tiempo con señales de E/S y datos recientes y actualizados. En

este caso la cabecera es mínima, a fin de que los mensajes se procesen muy rápidamente y con prioridad.

En consecuencia, Ethernet/IP combina el protocolo TCP/IP y los telegramas de datos UDP/IP para el transporte de paquetes de mensajes explícitos e implícitos, lo que significa que, en este caso, tanto los datos de entrada y salida en tiempo real para las tareas de control prioritarias (UDP) como los datos de información (TCP) de una red pueden utilizarse en paralelo.

Dado que ControlNet, DeviceNet y Ethernet/IP utilizan el mismo protocolo de aplicación, también pueden acceder a perfiles de dispositivos y librerías de objetos compartidos. Gracias a estos objetos es posible la interoperabilidad plug & play de dispositivos complejos de fabricantes diferentes. Las definiciones de los objetos son compatibles con la transmisión en tiempo real de mensajes de E/S, la configuración, el diagnóstico y la adquisición de datos a través de una misma red. Así, el usuario puede establecer fácilmente enlaces de comunicación con dispositivos inteligentes, tipo unidad/actuador sin tener que recurrir a herramientas de software específicas. El resultado es una mayor rapidez en línea y una completa compatibilidad de diagnóstico.

Ethernet/IP permite combinar la transmisión de datos acíclicos (mensajes explícitos) con los datos de control transmitidos cíclicamente (mensajes implícitos), y gracias a las características fabricante/consumidor que garantiza el protocolo de control e información CIP; es compatible con los principales mecanismos de comunicación para la conexión en una red de dispositivos, desde la interrogación secuencial hasta la

activación en función del tiempo o de un evento, pasando por la multidifusión o las conexiones punto a punto para el acoplamiento de datos.

También es importante la aceptación de ControlNet y DeviceNet, ya que la combinación y complementación de estas redes produce prácticamente un único sistema universal (misma capa de aplicación), además del considerable apoyo de este grupo de fabricantes a Ethernet/IP.

Solamente EtherNet/IP con tecnología CIP Motion combina los requisitos de un control de movimiento de bucle cerrado determinista y en tiempo real con Ethernet estándar, ofreciendo un cumplimiento total de las normas de Ethernet, incluyendo IEEE 802.3 y TCP/IP. CIP Motion es una extensión de CIP (Common Industrial Protocol) que proporciona la capacidad necesaria para un movimiento sincronizado multieje de gran rendimiento, comprende una serie de perfiles de aplicación diseñados para permitir la configuración en un mecanismo de los bucles de posicionamiento, velocidad y par. Esto se combina con la tecnología CIP Sync de ODVA: sincronización de reloj de precisión conforme a IEEE-1588, encuadrado en el modelo de objeto CIP.

EtherNet/IP con CIP Motion es una solución dimensionable y completa que proporciona una interfaz y unos servicios de aplicación común para propósito general y mecanismos de control de movimiento utilizando el mismo perfil. EtherNet/IP con CIP Motion son compatibles con las topologías de Ethernet estándar como estrella y árbol.

El modelo completo de Ethernet/IP puede verse como un concepto autónomo, diferente a la estructura completa de 3 niveles utilizados por los otros protocolos, ya

que la infraestructura se la considera como una sola y de hecho puede representarse en su conjunto por un único modelo de 7 capas, por más que los niveles inferiores sigan usando otras redes o buses. Este modelo posibilita la integración desde el nivel de proceso hasta la red comercial de la empresa.

En la práctica significa que la capa de aplicación “ve” y permite configurar todos los dispositivos de la red desde un único software, para ello utiliza un protocolo especial denominado CIP (Control e Information Protocol), explicado anteriormente, el cual se encarga de organizar todos los mecanismos de red como una colección de objetos a los que llama “Perfiles de dispositivos”, sobre los cuales define propiedades, accesos y atribuciones. Estos perfiles luego pueden ser reutilizados en la creación de nuevos recursos.

### ***MODBUS TCP/IP***

Modbus TCP/IP es una variante del protocolo Modbus que permite utilizarlo sobre la capa de transporte TCP/IP. Se caracteriza por su simplicidad, bajo costo de implementación, requerimientos mínimos de hardware y por ser un protocolo abierto. La interfaz de comunicaciones Modbus se base en mensajes, el formato de estos mensajes es independiente del tipo de interfaz física utilizada, pudiéndose implementarse sobre cualquier línea de comunicaciones serie y que permita la comunicación por medio de tramas binarias o ASCII con un proceso de pregunta-respuesta. Posee las siguientes ventajas:

- Es escalable en complejidad. Un dispositivo que tiene simplemente un solo propósito, necesita solo implementar uno o dos tipos de mensaje.
- Es simple para administrar y expandir. No se requiere usar herramientas de configuración complejas cuando se añade una nueva estación a la red Modbus/TCP.
- No se necesita equipo o software propietario específico. Cualquier sistema computador o microprocesador con una pila de protocolos TCP/IP puede usar Modbus/TCP.
- Puede ser usado para comunicarse con una gran base instalada de dispositivos MODBUS, usando productos de conversión, los cuales no requieren configuración.
- Es de muy alto desempeño, limitado típicamente por la capacidad del sistema operativo del computador para comunicarse. Puede lograr altas tasas de transmisión sobre una estación única, y las redes construidas pueden lograr tiempos de respuesta en el rango de los milisegundos.
- Permite realizar la gestión de sistemas distribuidos geográficamente mediante el empleo de las tecnologías de Internet/Intranet.

En la actualidad hay cientos de dispositivos MODBUS TCP/IP disponibles en el mercado. Se emplea para intercambiar información entre dispositivos, así como monitorizarlos y gestionarlos. También se emplea para la gestión de entradas/salidas distribuidas, siendo el protocolo más popular entre los fabricantes de este tipo de componentes.

La combinación de una red física versátil y escalable como Ethernet con el estándar universal de inter-redes TCP/IP y una representación de datos independiente de fabricante, como MODBUS, proporciona una red abierta y accesible para el intercambio de datos de proceso.

Modbus/TCP simplemente encapsula una trama Modbus en un segmento TCP. TCP proporciona un servicio orientado a conexión fiable, lo que significa que toda consulta espera una respuesta. (R3)

### **ESTRUCTURA DEL PROTOCOLO**

Todas las solicitudes son enviadas vía TCP sobre el puerto registrado 502 y en forma half-duplex, sobre una conexión establecidas. Es decir, no hay beneficio en enviar solicitudes adicionales sobre una única conexión mientras una respuesta está pendiente, sin embargo, los dispositivos que deseen obtener altas tasas de transferencia puede establecer múltiples conexiones TCP al mismo destino.

El campo “dirección esclavo” de Modbus es reemplazado por un byte “identificador de unidad” el cual puede ser usado para comunicarse con dispositivos tales como: puentes y Gateway, los cuales usan una dirección IP única para soportar múltiples unidades terminales independientes.

Los mensajes de solicitud y respuesta en Modbus/TCP, poseen un prefijo o encabezado compuesto por seis bytes como se aprecia en la siguiente tabla:

**Tabla II.IV. Estructura del prefijo de Modbus/TCP**

Ref	Ref	00	00	00	Len
-----	-----	----	----	----	-----

*Fuente: <http://sedici.unlp.edu.ar/>*

El campo de referencia de transacción está representado por los dos bytes “Ref Ref”, es un número que no tiene valor en el servidor pero son copiados literalmente desde la solicitud a la respuesta a conveniencia del cliente. Este campo se utiliza para que un cliente Modbus/TCP pueda establecer simultáneamente múltiples conexiones con diferentes servidores y pueda identificar cada una de las transacciones.

El tercero y cuarto campo del prefijo representan el identificador del protocolo, un número el cual debe ser establecido en cero. El campo “len”, especifica el número de bytes que siguen. La longitud es una cantidad de dos bytes, pero el byte más alto vale 0, ya que los mensajes son más pequeños que 256.

De esta forma, un mensaje Modbus/TCP completo posee una estructura como se muestra en la siguiente tabla.

**Tabla II.V. Estructura del mensaje Modbus/TCP**

<b>POSICIÓN DEL BYTE</b>	<b>SIGNIFICADO</b>
Byte 0	Identificador de transacción. Copiado por el servidor- normalmente 0
Byte 1	Identificador de transacción. Copiado por el servidor- normalmente 0
Byte 2	Identificador de protocolo = 0
Byte 3	Identificador de protocolo = 0
Byte 4	Campo de longitud (byte alto) = 0, ya que los mensajes son menores a 256
Byte 5	Campo de longitud (byte bajo). Numero de bytes siguientes
Byte 6	Identificador de unidad, previamente “dirección esclavo”.
Byte 7	Código de función Modbus.
Byte 8 y más	Los datos necesarios

*Fuente: <http://sedici.unlp.edu.ar/>*

A continuación se observa el encapsulamiento de la trama Modbus en el protocolo TCP.

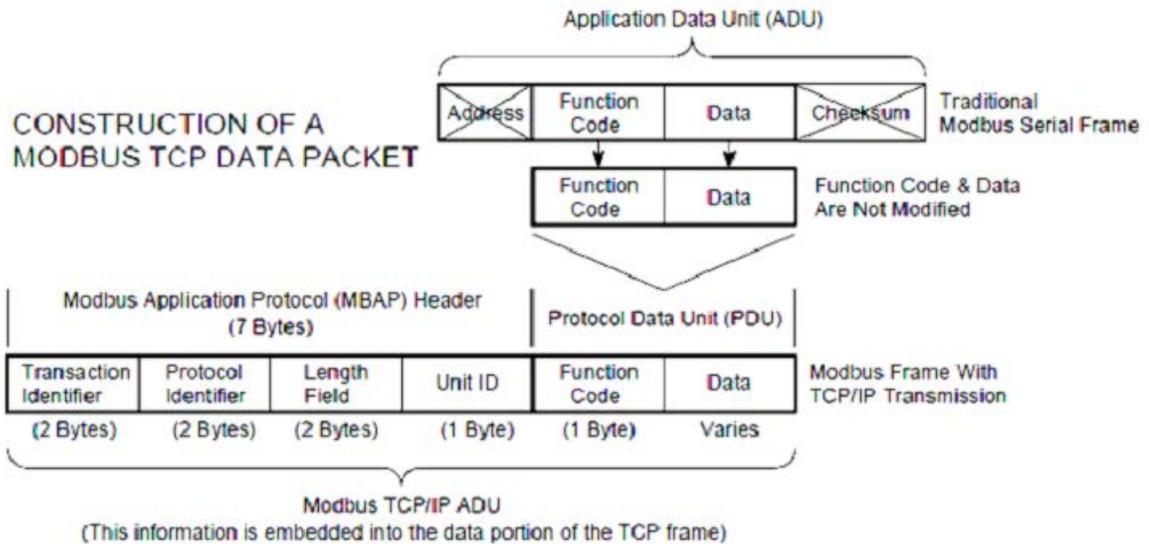


Figura II.23. Encapsulamiento de la trama Modbus en TCP.

Fuente: <http://sedici.unlp.edu.ar/>

Esta técnica de consulta/respuesta va perfectamente con la naturaleza Maestro/Esclavo de Modbus, añadido a la ventaja del determinismo que las redes Ethernet conmutadas ofrecen a los usuarios en la industria. El empleo del protocolo abierto Modbus con TCP proporciona una solución para la gestión desde unos pocos a decenas de miles de nodos.

Actualmente, Modbus se usa implementando en: TCP/IP sobre Ethernet, transmisión serial asincrónica sobre una variedad de medios (alambre: EIA/TIA-232-E, EIA-422, EIA/TIA-485-A, fibra, etc.) y Modbus con paso de testigo.

Las prestaciones dependen básicamente de la red y el hardware. Si se usa MODBUS TCP/IP sobre Internet, las prestaciones serán las correspondientes a

los tiempos de respuesta de Internet, que no siempre serán los deseables para un sistema de control. Sin embargo pueden ser suficientes para la comunicación destinada a depuración y mantenimiento, evitando así desplazamientos al lugar de la instalación. Si disponemos de una Intranet de altas prestaciones con conmutadores Ethernet de alta velocidad, la situación es totalmente diferente. En teoría, MODBUS TCP/IP, transporta datos hasta  $250/(250+70+70)$  o alrededor de un 60% de eficiencia cuando se transfieren registros en bloque, y puesto que 10 Base T proporciona unos 1.25 Mbps de datos, la velocidad de transferencia de información útil será:

$$1.25M/2*60\%= 360000 \text{ Registros por segundo}$$

En Base T la velocidad es 10 veces mayor.

Puesto que MODBUS TCP/IP es simplemente un protocolo MODBUS encapsulado en TCP, es muy sencillo comunicar dispositivos MODBUS existentes sobre MODBUS TCP/IP. Para ello se requiere una pasarela que convierta el protocolo MODBUS a MODBUS TCP/IP. (17)

#### ***2.4.5.4. DISPOSITIVOS***

##### ***PLC'S ETHERNET***

Mediante la utilización de módulos (ETN11) o CPUs ya dotados de puerto Ethernet, es posible integrar PLCs en la red Ethernet de modo que se pueda acceder a los archivos de los programas, a los datos de los parámetros, etc. La implementación de una red efectiva y segura también requiere el uso de conectores apropiados, disponibles en

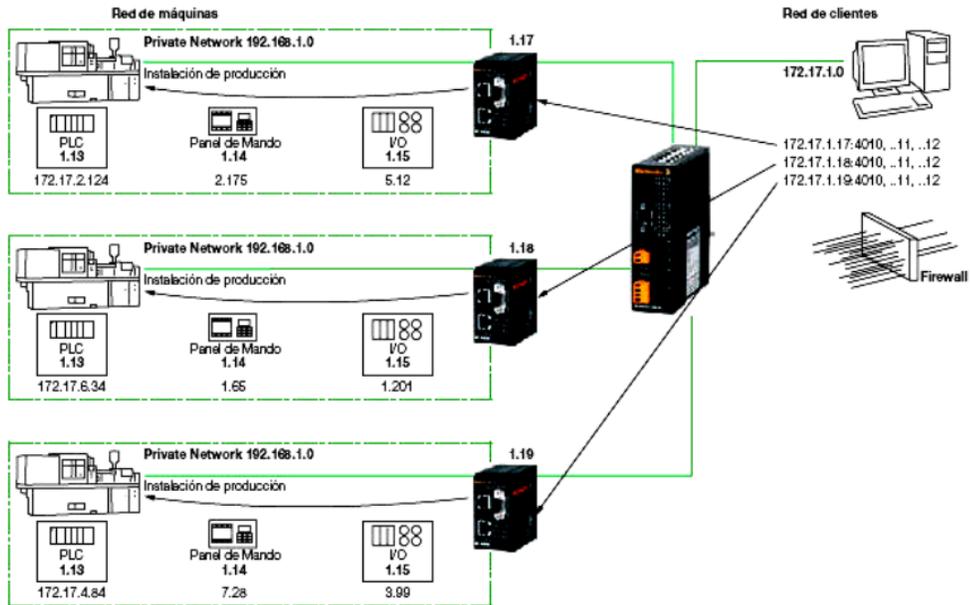
una amplia variedad y para soluciones muy flexibles. Las características principales de los puertos Ethernet integrados en las CPUs o módulos de comunicación son:

- Puerto Ethernet (100 BASE-TX)
- Comunicación TCP/IP FINS
- Funciones de servidor de archivos FTP
- Servidor Web para facilitar la configuración
- Posibilidad de enlazar hasta 255 nodos
- Transparencia de mensajes entre el controlador y otros dispositivos conectados al PLC mediante buses tipo: serie, DeviceNet, ModBus, etc.
- Configuración de las funciones Ethernet de la CPU mediante el software de programación CX-Programmer.

Las CPU CJ2M el puerto incorporado es Ethernet IP (18).

### ***ROUTERS O ENRUTADOR***

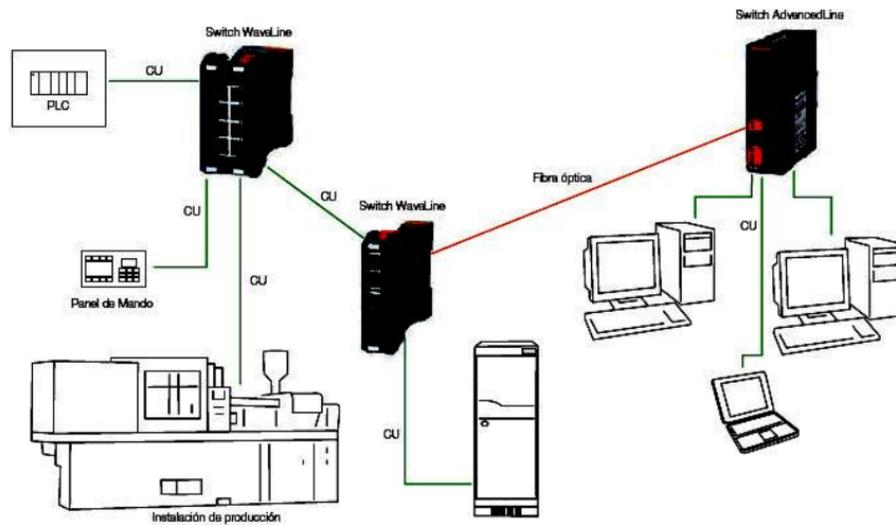
También conocido como encaminador de paquetes es un dispositivo que proporciona conectividad a nivel de red o nivel tres en el modelo OSI. Su función principal consiste en enviar los paquetes de red o encaminar paquetes de datos de una red a otra por la ruta más adecuada. Permite separa redes Ethernet por razones de seguridad de los datos o para una configuración sencilla (19).



**Figura II.24.** Utilización de un Router en una red Ethernet Industrial  
*Fuente: <http://sedici.unlp.edu.ar/>*

## SWITCH

Un conmutador o Switch es un dispositivo digital de lógica de interconexión de redes de computadores que opera en la capa 2 (nivel de enlace de datos) del modelo OSI. Su función es interconectar dos o más segmentos de red, de manera similar a los puentes, pasando datos de un segmento a otro de acuerdo con la dirección MAC de destino de las tramas en la red.



**Figura II.25.** Red Industrial con Switch  
*Fuente: <http://sedici.unlp.edu.ar/>*

#### Consideraciones para saber cuándo utilizar un Switch industrial:

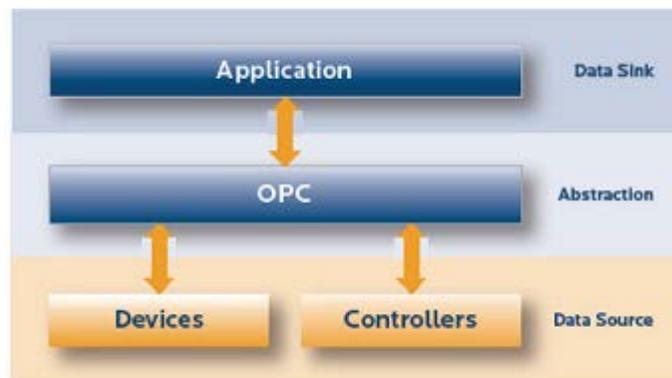
- Aislar la red de planta de la red administrativa.
- Aislar los dispositivos individuales
- Proveer un enlace de alta velocidad entre diferentes dominios de colisiones.
- En combinación con convertidores de medio a fibra óptica.
- Proveer un enlace entre dominios de colisiones alejados entre sí.
- Brindar alta inmunidad al ruido e interferencias (20).

## 2.5. OPC SERVER

Es un estándar de comunicación en el campo del control y supervisión de procesos, permite que diferentes fuentes de datos envíen datos a un mismo servidor OPC, al que a su vez podrán conectarse diferentes programas compatibles con dicho estándar. De

este modo se elimina la necesidad de que todos los programas cuenten con drivers para dialogar con múltiples fuentes de datos, basta que tengan un driver OPC.

OPC no es un protocolo, sino más bien un estándar para la conectividad de datos que se basa en una serie de especificaciones OPC gestionadas por la OPC Foundation. Cualquier software que sea compatible con estas especificaciones OPC proporciona a usuarios e integradores conectividad abierta e independiente tanto del fabricante del dispositivo como del desarrollador de la aplicación.



**Figura II.26.** Función del servidor OPC entre la fuente de datos y el cliente de datos.  
*Fuente: <http://www.google.infopl.net/>*

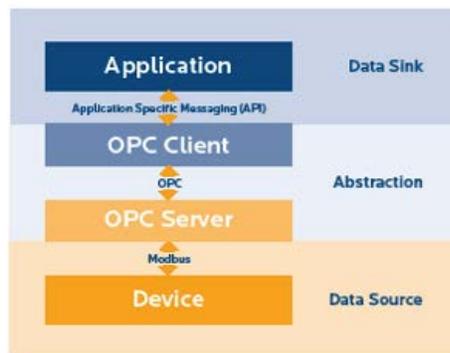
### 2.5.1. Cómo trabaja la comunicación OPC (Conceptualmente)

Se puede representar como una capa de “abstracción” intermedia que se sitúa entre la Fuente de Datos y el Cliente de Datos, permitiéndoles intercambiar datos sin saber nada el uno del otro.

### 2.5.2. Como funciona OPC (Funcionalmente)

La “abstracción de dispositivo” OPC se consigue utilizando dos componentes OPC especializados llamados Cliente OPC y Servidor OPC. Es importante resaltar que el hecho de que la Fuente de Datos y el Cliente de Datos puedan comunicar entre sí

mediante OPC no significa que sus respectivos protocolos nativos dejen de ser necesarios o hayan sido reemplazados por OPC. Al contrario, estos protocolos y/o interfaces nativos siguen existiendo, pero sólo comunican con uno de los dos componentes del software OPC. Y son los componentes OPC los que intercambian información entre sí, cerrando así el círculo. La información puede viajar de la aplicación al dispositivo sin que estos tengan que hablar directamente entre sí.



**Figura II.27.** Arquitectura Cliente/Servidor OPC.  
*Fuente: <http://www.google.infopl.net/>*

### 2.5.3. Beneficios de utilizar conectividad OPC

- Una aplicación Cliente OPC puede comunicar libremente con cualquier Servidor OPC visible en la red sin la necesidad de utilizar ningún driver específico para la Fuente de Datos.
- Las aplicaciones Cliente OPC pueden comunicar con tantos Servidores OPC como necesiten.
- Hay un Servidor OPC disponible para todos los dispositivos nuevos o antiguos que existen en el mercado.

- Las Fuentes de Datos (hardware o software) que utilizan OPC pueden ser intercambiadas o actualizadas sin la necesidad de actualizar los drivers utilizados por cada aplicación que comunique con ellas mediante OPC.
- Los usuarios pueden elegir libremente los dispositivos, controladores y aplicaciones que mejor se ajusten a sus proyectos sin preocuparse del fabricante o de si comunicarán entre si la intercomunicación se da por sentado.

#### ***2.5.4. Tipos de datos que soporta OPC***

Los tipos de datos más comunes transferidos entre dispositivos, controladores y aplicaciones en automatización se pueden encuadrar en tres categorías:

- Datos de tiempo real
- Datos históricos
- Alarmas y Eventos

Cada una de las categorías anteriores soporta una amplia gama de tipos de datos. Estos tipos de datos pueden ser enteros, coma flotante, cadenas, fechas y distintos tipos de arrays, etc.

Las tres especificaciones OPC que se corresponden con las tres categorías de datos son:

1. OPC Data Access Specification (OPC DA): utilizada para transmitir datos de tiempo real

2. OPC Historical Data Access Specification (OPC HDA): utilizada para transmitir datos históricos
3. OPC Alarms y Events Specification (OPC A-E): utilizada para transmitir información de alarmas y eventos

### **2.5.5. Servidor OPC**

Un Servidor OPC es una aplicación de software. Un driver “estandarizado” desarrollado específicamente para cumplir con una o más especificaciones OPC. La palabra “Server” en “OPC Server” no hace referencia en absoluto al ordenador donde este software se estará ejecutando. Hace referencia a la relación con el Cliente OPC.

### **2.5.6. Cliente OPC**

Un Cliente OPC es una pieza de software creada para comunicar con Servidores OPC. Utiliza mensajería definida por una especificación concreta de la OPC Foundation (21).

## **2.6. HMI**

Una interfaz Hombre -Máquina o HMI, Human Machine Interface por sus siglas en ingles es un sistema que presenta datos un operador y a través del cual este controla un determinado proceso. Este dispositivo o sistema permite una interfaz entre la persona o máquina, tiene una mayor tendencia a nivel industrial se debe principalmente a la necesidad de tener un control más preciso y agudo de las variables de producción y de contar con información relevante de los distintos procesos en tiempo real es decir monitorear y controlar de sistemas a gran distancias de los mecanismos de control.

### **2.6.1. Funciones de un HMI**

- Supervisión en su totalidad del proceso productivo
- Control de calidad de la producción
- Control de la productividad, al detectar a tiempo incidencias o bajadas de la producción.
- Mensajes de proceso y alarmas.
- Mensajes de averías y fallos.
- Mensajes de Curvas de Tendencias.
- Programación de tareas y paradas.
- Integración con máquinas para avisos de errores o mantenimientos preventivos y correctivos.
- Posibilidad de actuar desde la pantalla (22)

### **2.6.2. SCADA (Supervisory Control And Data Acquisition)**

SCADA (Supervisión, Control y Adquisición de Datos Adquisición de Datos), es una aplicación de software de control de producción, que se comunica con los dispositivos de campo y controla el proceso de forma automática desde la pantalla del ordenador. Estos sistemas actúan sobre dispositivos de la planta, y permiten controlar el proceso desde una estación remota.

### 2.6.2.1. Esquema Básico de un sistema SCADA



**Figura II.28.** Esquema de un sistema SCADA.

*Fuente: <http://repo.uta.edu.ec/>*

**Proceso objeto de control:** Es el proceso que se desea supervisar. En consecuencia, es el origen de los datos que se requiere coleccionar y distribuir.

**Adquisición de datos:** Son un conjunto de instrumentos de medición dotados de alguna interfase de comunicación que permita su interconexión.

**SCADA:** Combinación de hardware y software que permita la colección y visualización de los datos proporcionados por los instrumentos.

**Clientes:** Conjunto de aplicaciones que utilizan los datos obtenidos por el sistema SCADA (23).

### 2.6.3. BASE DE DATOS (DB)

Una base de datos de un SI (Sistema de Información) es la representación integrada de los conjuntos de entidades instancia correspondientes a las diferentes entidades tipo del SI y de sus interrelaciones.

La principal ventaja de utilizar bases de datos es que múltiples usuarios pueden acceder, visualizar, ingresar o actualizar, en concordancia con los derechos de acceso que se les hayan otorgado.

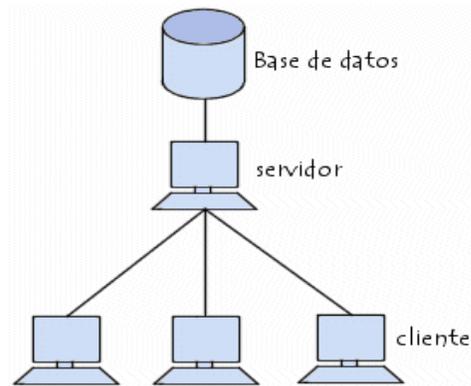


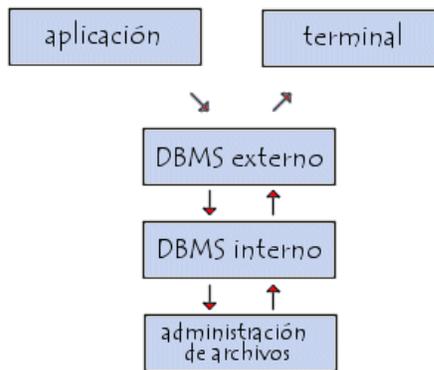
Figura II.29. Esquema y niveles de una Base de Datos.  
Fuente: <http://es.kioskea.net/>

#### 2.6.3.1. Tipos de Bases de datos

- **Base de datos Local**, puede utilizarla sólo un usuario en un equipo.
- **Base de datos distribuida**, la información se almacena en equipos remotos y se puede acceder a ella a través de una red.

### 2.6.3.2. Administración de bases de datos

Sirve para controlar tanto los datos como los usuarios. La administración de bases de datos se realiza con un sistema llamado DBMS (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) (24).



**Figura II.30.** Administración de una Base de Datos.

Fuente: <http://es.kioskea.net/>

## **NOTAS CAPÍTULO II**

- (1) [http://www.rocatek.com/forum\\_automatizacion\\_industrial.php](http://www.rocatek.com/forum_automatizacion_industrial.php)
- (2) <https://sites.google.com/site/automatizacionindustrial/>
- (3) <http://www.erhsa.com/html/electronicos/automatizacion/pc-industriales.html>
- (4) <http://www.kerwa.ucr.ac.cr/bitstream/handle/10669/537/DocumentoFinal.pdf>
- (5) [http://es.wikipedia.org/wiki/Sistema\\_embebido](http://es.wikipedia.org/wiki/Sistema_embebido)
- (6) <http://www.slideshare.net/ederdt/tema-34-icpa-controladores-e-interfases>
- (7) <http://www.tracnova.com/tracnova-ub/Comparaci%F3n%20PLC%20&%20PAC.pdf>
- (8) <http://148.206.53.231/UAMI10300.PDF>
- (9) <http://www.slideshare.net/180323675/control-industrial>
- (10) <http://repositorio.espe.edu.ec/bitstream/21000/2438/5/T-ESPE-019560-2.pdf>
- (11) [http://infoplcn.net/files/documentacion/comunicaciones/infoPLC\\_net\\_Redess\\_Industriales.pdf](http://infoplcn.net/files/documentacion/comunicaciones/infoPLC_net_Redess_Industriales.pdf)
- (12) MODELOS Y CONECTIVIDAD EN EL ÁMBITO DE PROCESOS INDUSTRIALES: Buses de campo y niveles OSI. Pág. 20-31
- (13) <http://www.emb.cl/electroindustria/articulo.mvc?xid=1034&tip=7>
- (14) <http://manolin-insind.blogspot.com/2009/06/topologia-de-red.html>
- (15) ETHERNET INDUSTRIAL: Modelos y conectividad en el ámbito de procesos industriales. Pág. 65,66.
- (16) <http://www.smar.com/espanol/articlostecnicos/article.asp?id=142>
- (17) ETHERNET INDUSTRIAL: Modelos y conectividad en el ámbito de procesos industriales. Pág. 77-80.
- (18) <http://www.etitudela.com/entrenadorcomunicaciones/downloads/2ethernet.pdf>

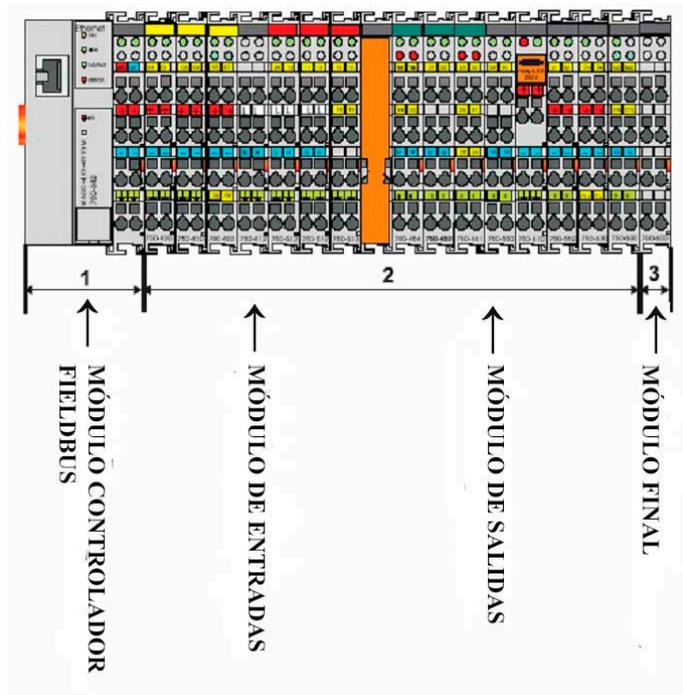
- (19) <http://www.monografias.com/trabajos72/enrutadores-inalambricos/enrutadores-inalambricos2.shtml>
- (20) <http://www.monografias.com/trabajos90/redes-interconexion/redes-interconexion.shtml>
- (21) <http://www.google.infoplcn.net/documentacion/7-comunicaciones-industriales/850-guia-para-entender-la-tecnologia-opc>
- (22) [http://www.generatetecnologias.es/sistemas\\_hmi.html](http://www.generatetecnologias.es/sistemas_hmi.html)
- (23) <http://repo.uta.edu.ec/bitstream/handle/123456789/101/t552e.pdf>
- (24) <http://es.kioskea.net/contents/66-introduccion-bases-de-datos>

## **CAPÍTULO III**

# **HARWARE Y SOFTWARE UTILIZADO EN LA IMPLEMENTACIÓN**

### **3.1. PLC WAGO 750-842**

El WAGO-I/O-SYSTEM 750 es un sistema modular para nodos Fieldbus. Está compuesto de un acoplador/controlador Fieldbus (1), conectado a módulos Fieldbus de entradas y salidas para cualquier tipo de señal (2) y un módulo de extremo o final (3).



**Figura III.31.** PLC WAGO-I/O-SYSTEM 750  
*Fuente: Manual WAGO I/O SYSTEM 750*

### 3.1.1. ETHERNET TCP/IP Programmable Fieldbus Controller (PFC)

El Controlador Programable de Buses de Entrada y Salida (FIELDBUS), combina las funciones del ETHERNET con las de un PLC. El controlador se comunica a través del bus de campo correspondiente. En el controlador de Fieldbus, todas las señales de entrada procedentes de los sensores se combinan, determina qué módulos de E/S están en el nodo y crea una imagen de proceso local de los mismos. Los datos del módulo especial y analógico se envían a través de palabras y/o bytes; datos digitales se agrupan bit a bit. La imagen de proceso local se divide en dos zonas de datos que contienen los datos recibidos y los datos que deben enviarse.

Según la norma de programación IEC 61131-3, el procesamiento de datos se produce en el PFC, aquí el controlador de Fieldbus tiene la opción de comunicarse con los sistemas de orden superior, ya sea a través de 10/100 Mbit/s (ETHERNET, utilizando una interfaz RJ-45). El programa de aplicación se crea con WAGO-I/O-PRO 32, CoDeSys de 3S (el sistema de programación estándar) sirve como la base de WAGO-I/O-PROCAA.

El controlador de Fieldbus tiene 128 KB de memoria de programa, 64 KB de memoria de datos y 8 KB de memoria remanente disponible para la programación IEC 61131-3.

Para enviar los datos de proceso a través de Ethernet, el controlador es compatible con una serie de protocolos de red. El protocolo MODBUS/TCP (UDP), se implementa para el intercambio de datos de proceso. Para la gestión y el diagnóstico del sistema, el servidor BOOTP y el HTTP protocolos están disponibles. (1)

### **3.1.2. Intercambio de datos**

Con el controlador 750-842 Fieldbus, el intercambio de los datos de proceso se lleva a cabo a través del protocolo MODBUS/UDP o el protocolo MODBUS/TCP. Este último funciona según el principio maestro/esclavo en donde el maestro controlador puede ser un PC o un PLC.

Los controladores Fieldbus WAGO-I/O-SYSTEM 750 son por lo general los dispositivos esclavos, pero por la programación según IEC 61131-3, pueden también asumir la función de maestro. Cuando el maestro solicita comunicación, ésta puede ser dirigida

a ciertos nodos por direccionamiento. Los nodos reciben la solicitud y, en función del tipo de solicitud, envía una respuesta al maestro.

Un controlador puede establecer un número determinado de conexiones simultáneas (conexiones de socket) a otros usuarios de la red:

- 1 conexión para HTTP
- 5 conexiones vía MODBUS TCP/IP
- 2 conexiones vía PFC
- 2 conexiones para WAGO-I/O-PROCAA

El número máximo de conexiones simultáneas no puede ser excedido. Las conexiones existentes deben terminar antes que las nuevas conexiones se puedan configurar. El PFC ETHERNET TCP/IP está esencialmente equipado con tres interfaces para el intercambio de datos:

1. Interface para Fieldbus (Maestro).
2. La función de PLC del PFC (CPU).
3. La interfaz de los módulos de I/O.

El intercambio de datos se lleva a cabo entre el maestro Fieldbus y los módulos de E/S, entre la función del PLC de PFC (CPU) y los módulos de E/S, y entre el maestro Fieldbus y la función PLC del PFC (CPU). Si el maestro MODBUS se utiliza como Fieldbus, el acceso se hace a los datos utilizando una función MODBUS implementadas en el controlador.

### 3.1.2.1. Área de Memoria

En la palabra del espacio de memoria 0...255, la imagen del proceso del controlador contiene los datos físicos de los módulos de bus.

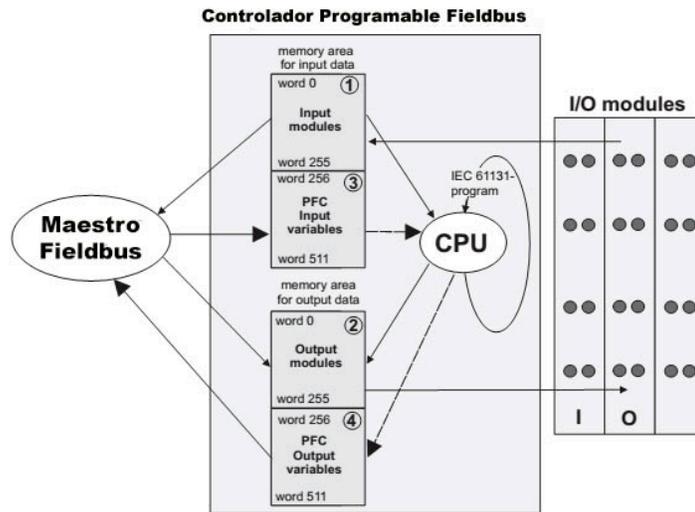


Figura III.32. Áreas de memoria e Intercambio de Datos  
Fuente: Manual WAGO I/O SYSTEM 750

- Los datos del módulo de entrada pueden ser leídos por la CPU y por el lado Fieldbus.
- La escritura en los módulos de salida es posible a partir de la CPU y desde el lado Fieldbus.

Las variables de PFC se presentan en el espacio de memoria Word 256... 511 de la imagen del proceso.

- Las variables de entrada MODBUS-PFC se escriben en el área de memoria de entrada del lado Fieldbus y se leen en la CPU para su procesamiento.

- Las variables procesadas por la CPU usando el programa de IEC-61131-3 son lugares en el área de memoria de salida, donde pueden ser leídos por el maestro. (2)

### 3.1.2.2. Intercambio de Datos entre Maestro Modbus/TCP y Módulos E/S.

Este intercambio es llevado a cabo usando funciones Modbus implementadas en el controlador mediante bit a bit o palabra por palabra en la lectura y escritura de rutinas.

Existen 4 diferentes tipos de datos de procesos en el controlador:

- Input words (Palabras de Entrada)
- Output words ( Palabras de Salida)
- Input bits (Bits de Entrada)
- Output bits (Bits de Salida)

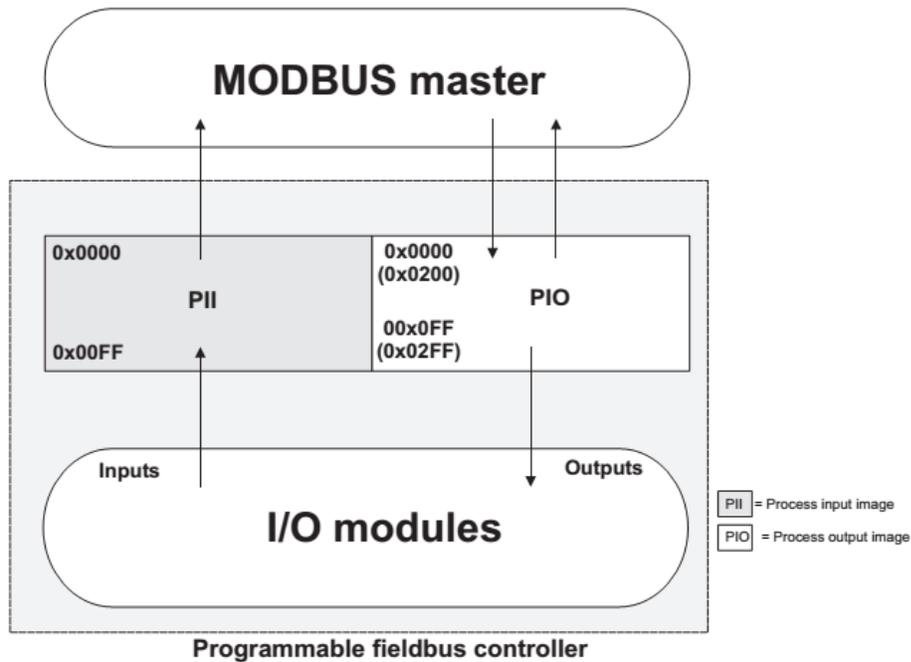
El acceso por palabra de los módulos de E/S digital se realiza de acuerdo a la siguiente tabla:

**Tabla III.VI. Asignación de entradas y salidas digitales para procesar las palabras de datos de acuerdo con el formato de Intel**

<b>Entradas/Salidas Digitales Palabra de Datos de Proceso</b>	16.	15.	14.	13.	12.	11.	10.	9.	8.	7.	6.	5.	4.	3.	2.	1.
	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>Byte</b>	High byte D1								Low byte D0							

*Fuente: Manual WAGO I/O SYSTEM 750*

La salida puede ser leída nuevamente agregando un desplazamiento de 200hex (0x0200) a la dirección MODBUS.

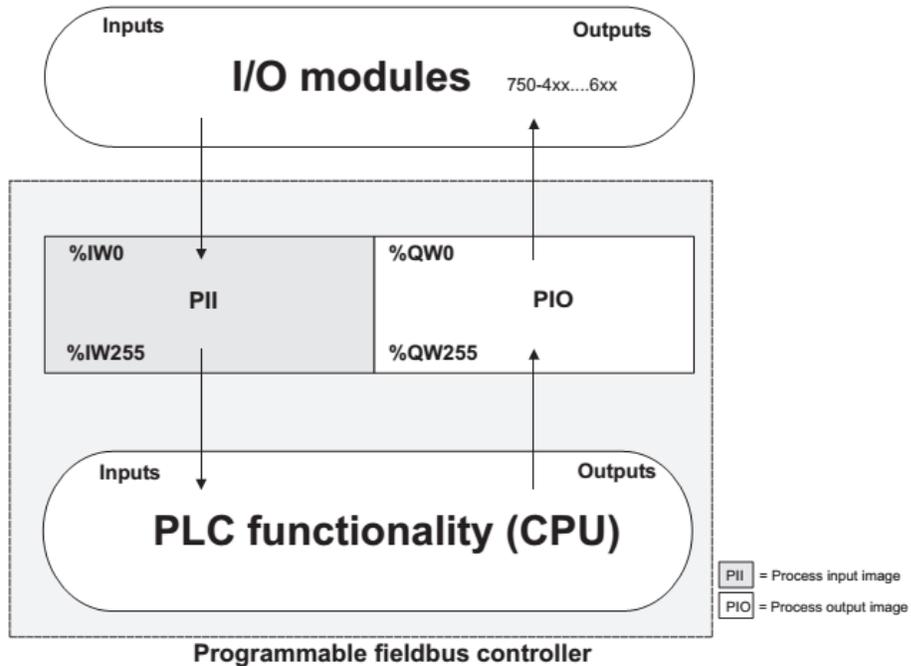


**Figura III.33.** Intercambio de datos entre el Maestro Modbus y módulos de E/S.  
*Fuente: Manual WAGO I/O SYSTEM 750*

El registro de funciones empieza en la dirección 0x1000, estas funciones pueden ser direccionadas de manera similar a los códigos de la función Modbus que son implementados (lectura/escritura).

### ***3.1.2.3. Intercambio de datos entre la función del PLC de PFC (CPU) y los módulos de E/S***

La función de PLC (CPU) utiliza un direccionamiento directo para acceder a los datos del módulo de E/S. El PFC utiliza direcciones absolutas para hacer referencia a los datos de entrada, el dato puede ser procesado internamente en el controlador utilizando el programa de IEC-61131-3. Las banderas se almacenan en un área de memoria no volátil en este proceso. Los resultados de la vinculación pueden ser escritos directamente en los datos de salida empleando direccionamiento absoluto.

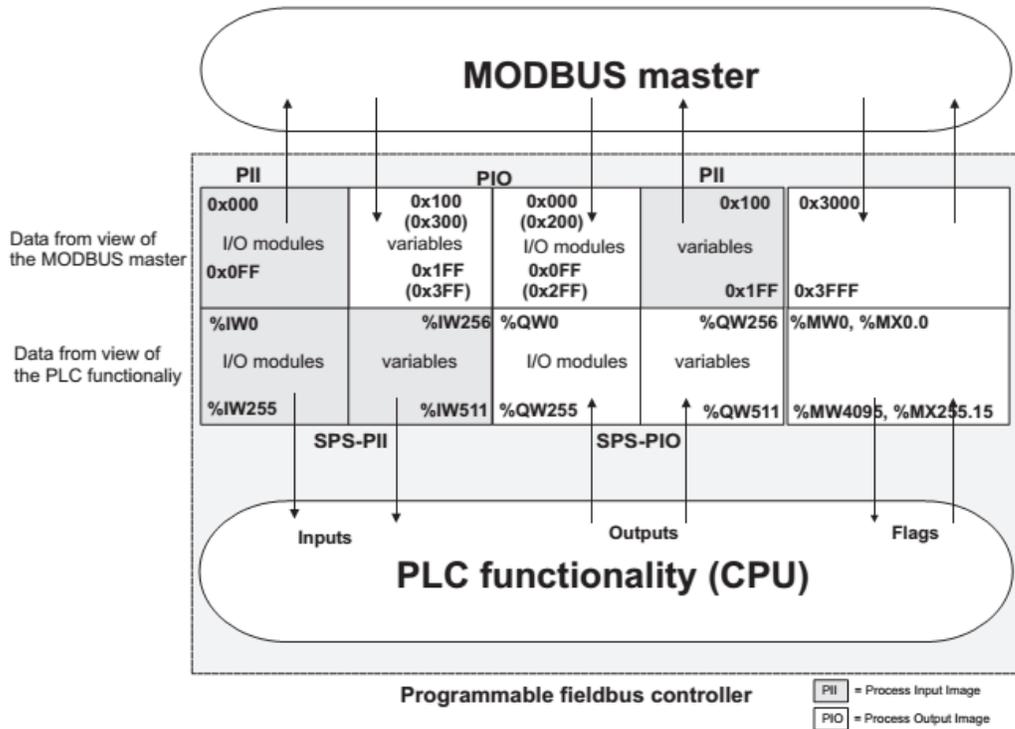


**Figura III.34.** Intercambio de datos entre la función del PLC de PFC (CPU) y los módulos de E/S.  
*Fuente: Manual WAGO I/O SYSTEM 750*

#### **3.1.2.4. Intercambio de Datos entre el Maestro y la función PLC del PFC (CPU)**

El maestro Fieldbus y la función PLC del PFC tienen diferentes perspectivas sobre los datos. Los datos variables (Variable data) generados por el maestro se envían como variables de entrada al PFC, donde son procesados. La creación de datos en el PFC son transmitidos vía Fieldbus al maestro como variables de salida.

En el PFC, el acceso a los datos variables Modbus/TCP PFC es posible a partir de dirección de palabra 256 a 511 (dirección de doble-palabra 128-255, dirección de byte 512-1023), mientras que el acceso a los datos variables de PFC es posible empezando desde la dirección de palabra 1276 a 1531 (dirección de doble-palabra 638-765, dirección de byte 2552-3063).



**Figura III.35.** Intercambio de datos entre el maestro Modbus y la función PLC del PFC.  
 Fuente: Manual WAGO I/O SYSTEM 750

### 3.1.3. COMUNICACIÓN FIELDBUS

La comunicación Fieldbus entre la aplicación principal y un controlador Fieldbus de WAGO está basada en el estándar Ethernet, que se produce a través de un protocolo de aplicación Fieldbus específico implementado. Dependiendo de la aplicación, esto puede ser, por ejemplo Modbus/TCP (UDP), Ethernet/IP, BACnet/IP, KNXnet/IP, PROFINET, SERCOS III u otro.

Además del estándar Ethernet y el protocolo de Fieldbus específico de aplicación, también son importantes otros protocolos de comunicaciones para la comunicación fiable y de transmisión de datos y otros protocolos relacionados para configurar y

diagnosticar el sistema implementado en controlador Fieldbus de WAGO basado en Ethernet. (3)

#### ***3.1.3.1. Ethernet***

En los controladores Fieldbus Ethernet desarrollados por WAGO, por lo general varios protocolos de aplicación se han empleado sobre la base de TCP/IP. Estos protocolos permiten a los usuarios crear aplicaciones con interfaces estandarizadas y transmitir datos de proceso a través de una interfaz Ethernet.

Además de una serie de protocolos de gestión y de diagnóstico, la aplicación de protocolos específicos Fieldbus se implementan para el control de los datos de los módulos, dependiendo del controlador.

Cada nodo de Fieldbus consiste en un controlador Fieldbus y una serie de módulos de E/S necesarias. El nodo de Fieldbus ETHERNET TCP/ IP de WAGO no requiere ningún componente maestro adicional que no sea un PC con una tarjeta de red. Por ello, el nodo de Fieldbus se puede conectar fácilmente a las redes locales o globales utilizando la conexión de bus de campo. Otros componentes de red, tales como concentradores, conmutadores o repetidores también pueden ser utilizados.

Además, dependiendo de los requisitos de las respectivas aplicaciones, diferentes entornos industriales, tales como la selección de protocolos, TCP/IP, el reloj interno y las configuraciones de seguridad se pueden realizar a través del sistema de gestión basado en web.

## **Modo de Transmisión en Ethernet**

Algunos controladores WAGO basados en Ethernet son compatibles con los 10 Mbit/s y 100 Mbit/s para operación full o half duplex. Para garantizar una transmisión segura y rápida, estos dos controladores y sus parejas de enlace deben configurarse con el mismo modo de transmisión.

El estándar Ethernet IEEE 802.3u define dos posibilidades para la configuración de los modos de transmisión:

1. Configuración Estática
2. Configuración Dinámica

### **1. Modo de Transmisión de Configuración Estática**

Utilizando una configuración estática, las parejas de enlace se establecen en la velocidad de transmisión estática y modo dúplex. Las siguientes configuraciones son posibles:

- 10 Mbit/s, half duplex
- 10 Mbit/s, full dúplex
- 100 Mbit/s, half dúplex
- 100 Mbit/s, full dúplex

### **2. Modo de Transmisión de Configuración Dinámica**

Esta opción se define en el estándar IEEE 802.3u. El uso de este modo, la velocidad de transmisión y el modo dúplex se negocian dinámicamente entre ambos interlocutores de comunicación.

La negociación automática permite que el dispositivo seleccione automáticamente el modo de transmisión óptimo. (4)

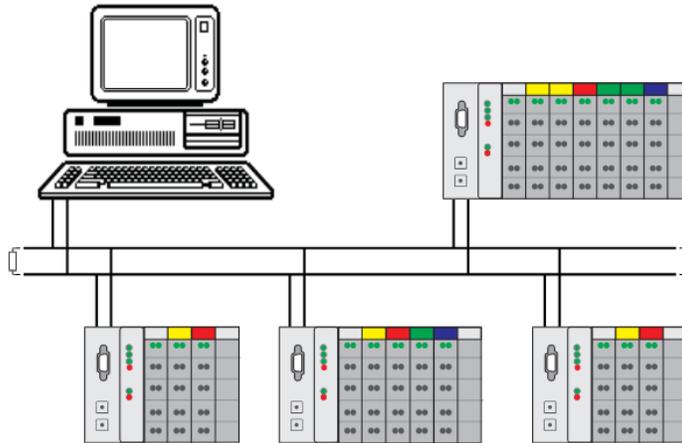
### *Arquitectura de Procesos de Datos para MODBUS/TCP*

Con algunos módulos de E/S, la estructura de los datos de proceso es específico Fieldbus. En el caso de un controlador de Fieldbus con MODBUS/TCP, la imagen del proceso utiliza una estructura de palabra (palabra con la alineación). El método de asignación interna para los datos de más de un byte se ajusta al formato de Intel.

Para la imagen del proceso PFC del controlador Fieldbus programable es la estructura de los datos del proceso de asignación idéntica.

### **Modbus en WAGO**

MODBUS es un sistema maestro/esclavo. El maestro es una unidad de control superpuesta, por ejemplo, un PC o un dispositivo PLC, mientras que el controlador del WAGO hace la función de dispositivos esclavos. Conflictos de Bus no se producen debido a que sólo un nodo está enviando. El maestro hace una consulta para la comunicación. Esta consulta se puede enviar a un nodo específico o para todos los nodos como un mensaje de difusión. Los nodos reciben la consulta y devuelven una respuesta al maestro, dependiendo del tipo de consulta.



**Figura III.36.** Ejemplo de topología Modbus con WAGO 750  
*Fuente: Manual WAGO I/O SYSTEM 750*

### Proceso de datos MODBUS/TCP

Después de la conexión, el controlador identifica todos los módulos de E/S conectados con el nodo que envía o recibe datos (ancho de datos (data width)/ancho de bits (bit width) > 0). Un nodo puede consistir en un régimen mixto de módulos analógicos y digitales.

El controlador crea una imagen de proceso local interno sobre la base de la anchura de los datos, el tipo de módulo de E/S y la posición del módulo en el nodo. Esta imagen de proceso se divide en una entrada y un rango de datos de salida.

Los datos de los módulos de E/S digitales es orientado a bits, es decir, los datos digitales son enviados poco a poco. Módulos de E/S analógicas representan el grupo de módulos orientados a bytes - los datos se envían byte a byte.

Por tanto la imagen del proceso de salida de la entrada local y, los datos de módulo de E/S se almacena en la imagen de proceso correspondiente de acuerdo con el orden en el que los módulos están conectados al controlador.

En primer lugar, todos los módulos de E/S orientados a bytes (analógica) se presentan en la imagen del proceso, los módulos de E/S orientada al bit (digital). Los bits de los módulos digitales se agrupan en bytes. Si la cantidad de información digital supera los 8 bits, el controlador inicia de forma automática con un nuevo byte. Para algunos módulos de E/S (y sus variaciones), la estructura de los datos de proceso depende del Fieldbus.

Para el controlador Fieldbus con MODBUS, la imagen de proceso se construye en ord-by word (con la alineación de palabra). El método de asignación interna para los datos de más de un byte se ajusta a los formatos de Intel.

La cabecera general de Modbus/TCP es la siguiente:

**Tabla III.VII. Cabecera de Modbus/TCP**

Byte	0	1	2	3	4	5	6	7	8... a
	Identifier (entered by receiver)		Protocol-identifier (is always 0)		Lenght flied (High byte, low byte)		Unit identifier (Slave address)	MODBUS Function code	Data

*Fuente: Manual WAGO I/O SYSTEM 750*

El acceso por el PLC para procesar los datos se realiza independientemente del sistema de Fieldbus de los controladores de WAGO Fieldbus, el acceso siempre se lleva a cabo a través de un programa de aplicación relacionada al IEC-61131-3.

### **3.1.4. Módulos de Entradas y Salidas**

#### ***3.1.4.1. Módulos de entradas digitales***

Módulos de entradas digitales proporcionan un bit de datos por canal para especificar el estado de señal para el canal correspondiente. Estos bits se asignan en el proceso de entrada de imagen.

Algunos módulos digitales tienen un bit adicional de diagnóstico por canal en el proceso de entrada de imagen. El bit de diagnóstico se utiliza para la detección de fallos que se producen (por ejemplo, roturas de cables y/o cortocircuitos).

Cuando los módulos de entrada analógicos también están presentes en el nodo, los datos digitales siempre se añaden después de que los datos analógicos en el proceso de entrada de imagen, agrupados en octetos.

#### ***3.1.4.2. Módulos de salidas digitales***

Módulos de salidas digitales utilizan un bit de datos por canal para controlar la salida del canal correspondiente. Estos bits se asignan a la imagen del proceso de salida.

Algunos módulos digitales tienen un bit adicional de diagnóstico por canal en el proceso de entrada de imagen. El bit de diagnóstico se utiliza para la detección de fallos que se producen (por ejemplo, roturas de cables y/o cortocircuitos). Para los módulos con bit de diagnóstico se establece, también los bits de datos tienen que ser evaluados.

Cuando los módulos de salida analógica también están presentes en el nodo, los datos de imagen digital siempre se añaden después de que los datos analógicos en la imagen del proceso de salida, agrupados en octetos.

#### ***3.1.4.3. Módulos de entradas analógicas***

El hardware de un módulo de entrada analógica tiene 16 bits de datos analógicos medidos por canal y 8 bits de control/estado. Sin embargo, el controlador con MODBUS/TCP no tiene acceso a los 8 bits de control/estado. Por lo tanto, el controlador con MODBUS/TCP sólo puede acceder a los 16 bits de datos por canal analógico, que se agrupan como palabras y se asignan en formato Intel en el proceso de entrada de imagen.

Cuando los módulos de entrada digital también están presentes en el nodo, los datos de entrada analógica siempre se asignan en el proceso de entrada de imagen en frente de los datos digitales.

#### ***3.1.4.4. Módulos de salidas analógicas***

El hardware de un módulo de salida analógica tiene 16 bits de datos analógicos medidos por canal y 8 bits de control/estado. Sin embargo, el controlador con MODBUS/TCP no tiene acceso a los 8 bits de control/estado. Por lo tanto, el controlador con MODBUS/TCP sólo puede acceder a los 16 bits de datos por canal analógico, que se agrupan como palabras y se asignan en formato Intel en la imagen del proceso de salida.

Cuando los módulos de salida digitales también están presentes en el nodo, los datos de salida analógicas siempre se asigna en la imagen del proceso de salida en la parte frontal de los datos digitales. (5)

#### ***3.1.4.5. Direccionamiento de los Módulos de E/S***

La disposición de los módulos de E/S en un nodo es opcional. El direccionamiento es organizado palabra por palabra y se inicia con la palabra de dirección '0', tanto para las entradas, como para las salidas.

El direccionamiento del módulo de E/S corresponde a la secuencia de su disposición detrás del controlador. El direccionamiento comienza con el controlador de bus, el cual puede asignar una o más palabras por canal. Esto es seguido por las direcciones de los módulos de E/S que se pueden asignar uno o dos bits por canal.

Si un nodo está extendido con módulos adicionales de E/S para el cual una o más palabras se asignan por canal, las direcciones de los módulos de E/S son desplazados por uno o dos bits por canal.

#### **Direcciones absolutas de las entradas, salidas y banderas**

La visualización directa de las celdas de memoria individuales (direcciones absolutas), de conformidad con la norma IEC 1131-3 se realiza mediante cadenas de caracteres especiales de acuerdo con la siguiente tabla (6).

**Tabla III. VIII. Direcciones Absolutas**

POSICION	CARACTER	DESIGNACION	COMENTARIOS
1	%	Inicia la dirección absoluta	
2	I Q M	Input (Entrada) Output (Salida) Bandera	
3	X* B W D	Bit Individual Byte (8 bits) Word (16 bits) Doble Word (32 bits)	Tamaño de dato
4		Dirección	
* El carácter "X" para los bits se puede eliminar			

Fuente: Manual WAGO I/O SYSTEM 750

Rango de direcciones para los datos de los módulos de E / S:

**Tabla III. IX. Rango de direcciones para los datos de los módulos de E / S**

Dato	Dirección									
Bit	0.0 ... 0.15		1.0 ... 1.15		.....		254.0 ... 254.15		255.0 ... 255.15	
Byte	0	1	2	3	.....	.....	508	509	510	511
Word	0		1		.....		254		255	
DWord	0				.....		127			

Fuente: Manual WAGO I/O SYSTEM 750

Rango de direcciones de datos Fieldbus:

**Tabla III. X. Rango de direcciones de datos Fieldbus.**

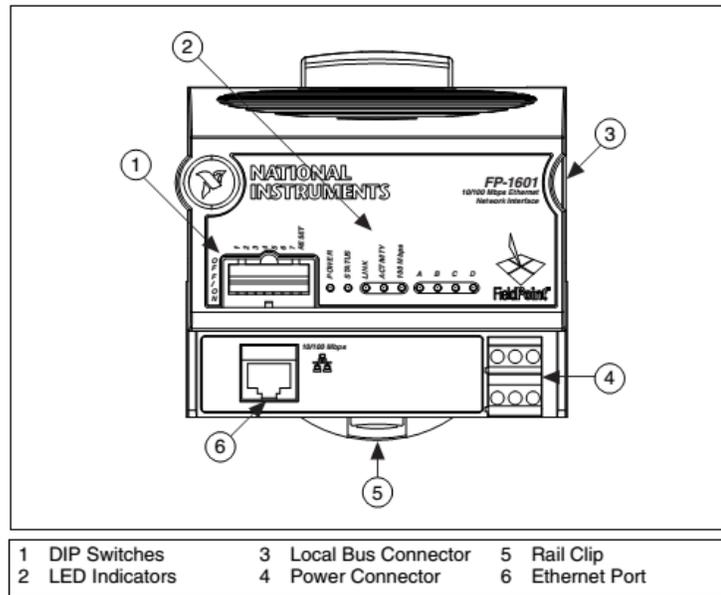
Dato	Dirección									
Bit	256.0 ... 256.15		257.0 ... 257.15		.....		510.0 ... 510.15		511.0 ... 511.15	
Byte	512	513	514	515	.....	.....	1020	1021	1022	1023
Word	256		257		.....		510		511	
DWord	128				.....		255			

Fuente: Manual WAGO I/O SYSTEM 750

### 3.2. FIELD POINT FP-1601

El módulo de red FieldPoint FP-1601 conecta una red Ethernet 10/100 a un módulo I/O FieldPoint. El FP-1601 es un dispositivo Ethernet, el número máximo de módulos

FP-1601 que se puede conectar depende de la topología de red a utilizar, se debe tomar en cuenta que cada FP-1601 puede soportar 9 módulos I/O.



**Figura III.37.** Dispositivo FieldPoint FP-1601  
*Fuente: Manual de usuario FP-1601*

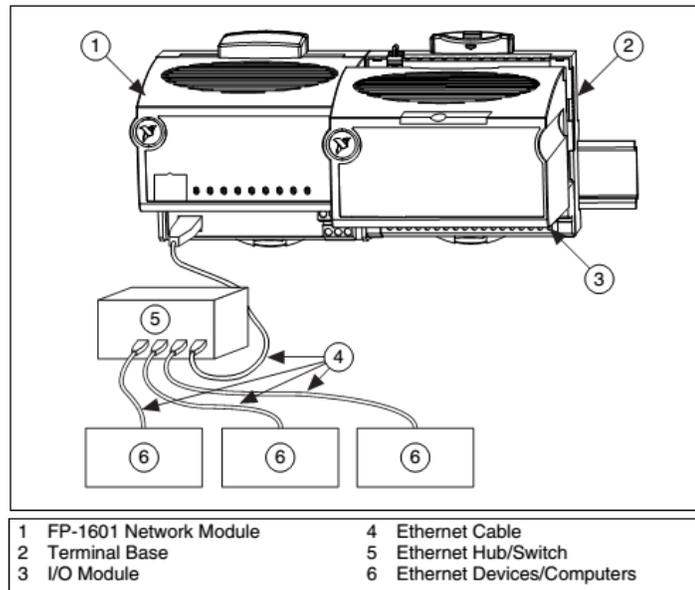
Un banco FieldPoint posee al menos un módulo de red, uno o más módulos bases y uno o más módulos I/O. Se puede acceder al banco por medio de un ilimitado número de equipos o dispositivos host habilitados para LabVIEW RT. Con Ethernet se puede utilizar un número ilimitado de bancos.

### 3.2.1. Aplicaciones

Las aplicaciones pueden comunicarse fácilmente con el FP-1601 para el intercambio de datos. La interfaz puede comunicarse con un controlador integrado o con un ordenador Windows con LabVIEW, LabWindows/CVI, Measurement Studio, Lookout, o software de su elección de aplicación OPC-cliente. El usando de comandos Optomux

estándar, también se puede comunicar con plataformas que no sean Windows, como Mac OS y Linux. Con el FP-1601 se puede crear rápidamente sistemas de automatización flexible de medición distribuida y modular.

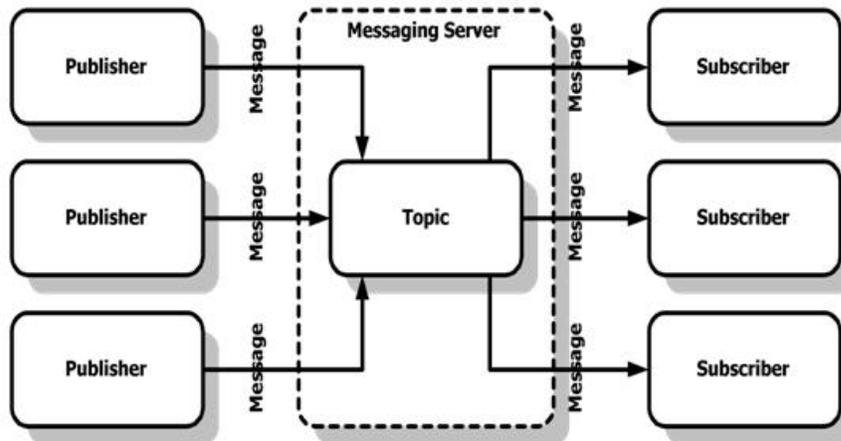
### 3.2.2. Comunicación de Red



**Figura III.38.** Conexión de FP-1601 a una red Ethernet.

*Fuente: Manual de usuario FP-1601*

Este dispositivo trabaja con una arquitectura de publish/subscribe, en donde uno o más ordenadores clientes se suscriben a los datos de E/S de los bancos FieldPoint. La interfaz de red supervisa la conexión de los módulos de E/S y publica los datos de E/S sólo cuando los valores cambian. Las señales analógicas pueden cambiar los valores dentro de los rangos seleccionables, llamadas bandas muertas. Este método orientado a eventos, junto con la compresión de datos, ayuda a evitar el tráfico Ethernet innecesarios y maximiza la eficiencia de las comunicaciones.



**Figura III.39.** Esquema de la arquitectura Publish/Subscriber.  
*Fuente: <http://streaming.uvg.edu.gt/>*

### 3.2.3. Interface de Red Ethernet para FieldPoint

El FP-1601 incluye un puerto RS-232 estándar de 9 pines y un puerto RS-485 de repetidor dúplex completo aislado. Con el puerto RS-485 del repetidor, el PC puede comunicarse a través de un puerto RS-232 con un módulo FP-1601

Una única interfaz de comunicación FP-1601 gestiona un nodo de hasta nueve bases FieldPoint y los módulos de E/S conectados. Un nodo se compone de un módulo de interfaz de red FieldPoint y cualquier combinación de módulos de E/S digitales y analógicas. El módulo de interfaz de red y bases terminales se encajan y se montan como una unidad en un riel DIN.

El FP-1601 se conecta directamente a la red Ethernet, realiza una negociación automática de red de tasas de comunicación de 10 Mb/s o 100 Mb/s. Este dispositivo cuenta con una conector RJ-45 para conexión a redes 10BaseT y 100BaseTX y usa un

protocolo basado en el estándar TCP/IP (MODBUS TCP/IP), para mantener compatibilidad con redes existentes.

El módulo de interfaz de red y las bases terminales forman un bus de datos de alta velocidad para la comunicación entre el módulo de red y los módulos de E/S. Con bases de terminales modulares, es fácil ampliar el sistema FieldPoint para satisfacer las cambiantes necesidades de la aplicación.

### **3.2.4. Software de Fieldpoint**

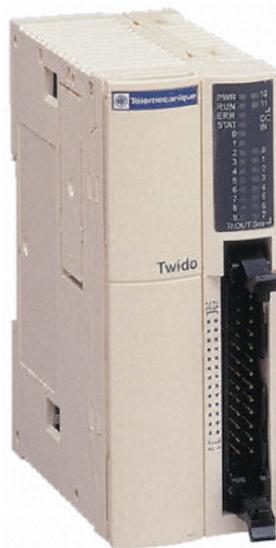
El software FieldPoint incluye una utilidad de configuración, así como software de servidor y un driver para una fácil integración en los paquetes de software de aplicación.

Estos componentes de software gestionan las comunicaciones de bajo nivel y los detalles de hardware, lo que simplifica el acceso mediante programación a los canales de E/S. La versión 3.0 del software FieldPoint se ejecuta en Windows Me/98/95, Windows 2000 o Windows NT versión 4.0 (Service Pack 3 o posterior), e incluye los siguientes componentes:

- Utilidad de configuración FieldPoint Explorer
- Labview Vis
- Funciones LabWindows/CVI
- OPC Server
- Driver Lookut (7)

### 3.3. TELEMECANIQUE TWDLMDA20DTK

El PLC TWDLMDA20DTK pertenece a la familia de TWIDO de Schneider Electric, es un controlador modular diseñado para instalaciones simples y máquinas pequeñas y compactas, cubre aplicaciones estándares con 12 entradas y 8 salidas de comunicación y con soporte de hasta 8 módulos de expansión. El software utilizado para su respectiva programación es el TwidoSoft de Telemecanique. Hay que tomar en cuenta que este PLC no ofrece conectividad directa a Ethernet, para ello se utiliza el TwidoPort, dispositivo que se detallara más adelante.



**Figura III.40.** Controlador Twido TWDLMDA20DTK  
*Fuente: Manual Twido Controladores Programables, Guía de Hardware*

#### 3.3.1. Descripción del controlador

Por defecto, todas las E/S del controlador están configuradas como E/S binarias. Sin embargo, determinadas E/S especializadas pueden asignarse a tareas específicas durante la configuración como:

- Entrada RUN/STOP
- Entradas con retención
- Contadores rápidos
- Salida de estado del controlador
- Modulación de ancho de pulsos (PWM)
- Salida del generador de pulsos (PLS)

Los controladores Twido se han programado mediante TwidoSuite que también permite que se utilicen las funciones PID y de ajuste automático PID en determinados controladores.

Las funciones principales del controlador son las siguientes:

- *Exploración:* Normal (cíclica) o periódica (constante) (de 2 a 150 ms).
- *Tiempo de Ejecución:* De 0,14  $\mu$ s a 0,9  $\mu$ s para una instrucción de lista.
- *Capacidad de memoria:*
  - *Datos:* 3000 palabras de memoria, 256 bits de memoria.
  - *Programa:* 6000 instrucciones de lista (con cartucho de 64Kb, en otro caso 3000 instrucciones de lista).
- *Puerto de programación:* EIA RS-485
- *Bloques de Función especializados:* 2 contadores rápidos, 2 contadores muy rápidos, 2 PWN/PLS.

- *Filtro de Entradas Programable:* El tiempo de filtrado de entradas puede modificarse durante la configuración. Sin filtrado, o filtrado a 3ms o 12ms. Los puntos de E/S se configuran en grupos.

### **3.3.2. Comunicación de PLC Twido**

Las bases Twido disponen de un puerto serie o de un segundo puerto opcional, que brindan un servicio en tiempo real y que se puede utilizar para cuatro tipos de comunicaciones:

- Conexión del bus AS-Interface
- Conexión al bus de campo CANopen
- Conexión de red Ethernet → solo con TWDLCAE40DRF y TWDLCDE40DRF
- Conexión por módem

Los servicios en tiempo real proporcionan funciones de distribución de datos para intercambiar datos con dispositivos de E/S, así como funciones de mensajería para comunicarse con dispositivos externos, estos servicios se controlan y configuran por medio de TwidoSuite. Cada puerto serie se utiliza para cualquiera de estos servicios, pero sólo el puerto serie 1 es válido para comunicarse con TwidoSuite.

Para poder utilizar estos servicios, existen tres protocolos disponibles en cada base:

- Conexión remota
- Modbus
- ASCII (8)

### 3.3.2.1. Características funcionales de las bases modulares

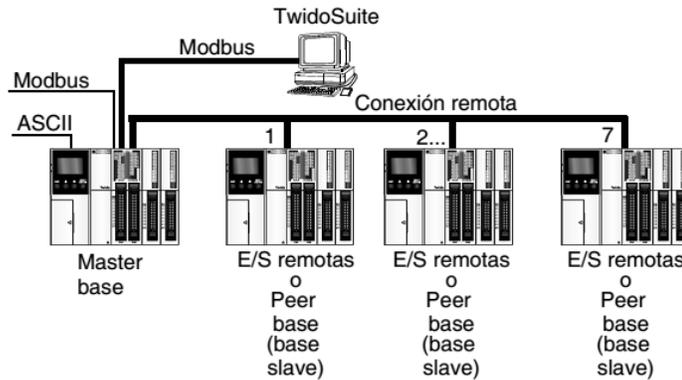
**Tabla III.XI. Características de las funciones de comunicación**

Puerto de comunicaciones	Puerto 1 (RS485)	Puerto 2 (RS232C) Módulo de ampliación de comunicaciones (TWDNOZ232D) o Módulo de ampliación del monitor de operación (TWDXCPODM) con adaptador de comunicaciones (TWDNAC232D)	Puerto 2 (RS485) Módulos de ampliación de comunicaciones (TWDNOZ485D) o (TWDNOZ485T) o Módulo de ampliación del monitor de operación (TWDXCPODM) con adaptador de comunicaciones (TWDNAC485D) o (TWDNAC485T)
Normas	RS485	RS232	RS485
Velocidad máx. en baudios	Conexión a PC: 19.200 bps Conexión remota: 38.400 bps	19.200 bps	Conexión a PC: 19.200 bps Conexión remota: 38.400 bps
Comunicación a través de Modbus (RTU master/slave)	Posible	Posible	Posible
Comunicación ASCII	Posible	Posible	Posible
Comunicación remota	7 posibles	Imposible	7 posibles
Longitud máxima del cable	Distancia máxima entre el controlador base y el remoto: 200 m	Distancia máxima entre el controlador base y el remoto: 200 m	Distancia máxima entre el controlador base y el remoto: 200 m
Aislamiento entre el circuito interno y el puerto de comunicaciones	No aislado	No aislado	No aislado
Comunicación a través de la línea telefónica	Posible Sólo se puede conectar un módem de recepción.	Imposible	Imposible

Fuente: Manual Twido Controladores Programables, Guía de Hardware

### 3.3.2.2. Arquitectura de comunicaciones con los protocolos

A continuación se muestra una gráfica de las comunicaciones posibles en el PLC. Se debe tomar en cuenta que no es posible realizar la comunicación simultánea de Modbus y Conexión Remota.



**Figura III.41.** Arquitectura de comunicaciones con los tres protocolos anteriormente mencionados.  
*Fuente: Manual Twido Controladores Programables, Guía de Hardware*

### 3.3.2.3. Protocolos de Comunicación

#### Protocolo de Control Remoto

El protocolo de conexión remota es un bus de alta velocidad "maestro/esclavo", diseñado para transferir una pequeña cantidad de datos entre el controlador principal y hasta controladores esclavos, incluso a distancia. Aplicación de E/S de datos se transfiere, en función de la configuración del controlador remoto.

#### Protocolo Modbus

El protocolo Modbus es un protocolo "maestro/esclavo", que permite al maestro solicitar respuestas de los esclavos o de tomar medidas en base a la solicitud. El maestro puede dirigirse a un esclavo individual o iniciar una difusión de mensajes para todos los esclavos. Los esclavos devuelven un mensaje (respuesta) a las preguntas que se dirigen a ellos de forma individual. No se devuelven respuestas a las solicitudes de difusión desde el maestro.

Modo maestro Modbus - El modo maestro de Modbus permite al controlador para iniciar una transmisión de peticiones Modbus, esperando una respuesta desde un esclavo Modbus.

Modo esclavo Modbus - modo esclavo Modbus permite al controlador de responder a las preguntas de un maestro Modbus. Este es el modo de comunicación predeterminado si no se configura ninguna comunicación.

### **Protocolo Modbus TCP/IP**

La siguiente información describe el protocolo de aplicación Modbus (MBAP). El protocolo de aplicación Modbus (MBAP) es un protocolo de capa-7 proporciona una comunicación punto-punto entre controladores lógicos programables (PLC) y otros nodos en una LAN.

Transacciones del protocolo Modbus son pares típicos mensajes de petición-respuesta. Un PLC puede ser a la vez cliente y servidor dependiendo de si se está consultando o respondiendo a los mensajes. Un cliente Modbus TCP / IP es equivalente a un controlador maestro Modbus en herencia Modbus, mientras que un servidor Modbus TCP / IP corresponderían a un controlador esclavo Modbus.

### **Protocolo ASCII**

El protocolo ASCII permite la comunicación entre el controlador y un sencillo dispositivo tal como una impresora.

### ***3.3.2.4. Comunicación MODBUS***

#### **Configuración de Hardware**

Un enlace Modbus se puede establecer ya sea en el puerto EIA RS232 o RS485 EIA, y se puede ejecutar en un máximo de dos puertos de comunicación a la vez. Cada uno de estos puertos se puede asignar su propia dirección Modbus, mediante el bit de sistema% S101 y palabras de sistema% SW101 y% SW102.

#### **Configuración de Software**

Para configurar el controlador para utilizar una conexión serie para enviar y recibir caracteres utilizando el protocolo Modbus, es necesario:

1. Configurar el puerto serial para Modbus utilizando el software Twido Suite.
2. Crear en una aplicación una tabla de transmisión/recepción que será utilizada por la instrucción EXCHx.

#### **Configuración del Puerto**

Un controlador Twido puede utilizar su puerto principal 1 o un puerto configurado opcionalmente 2 para utilizar el protocolo Modbus. Para configurar un puerto serie para Modbus:

1. Definir los adaptadores de comunicación adicionales o módulos configurados para la base
2. Declarar la red Modbus en el paso Describir de TwidoSuite.
3. Seleccione Puerto 1 (o puerto 2 si está instalado) para configurar en la ventana Describir.

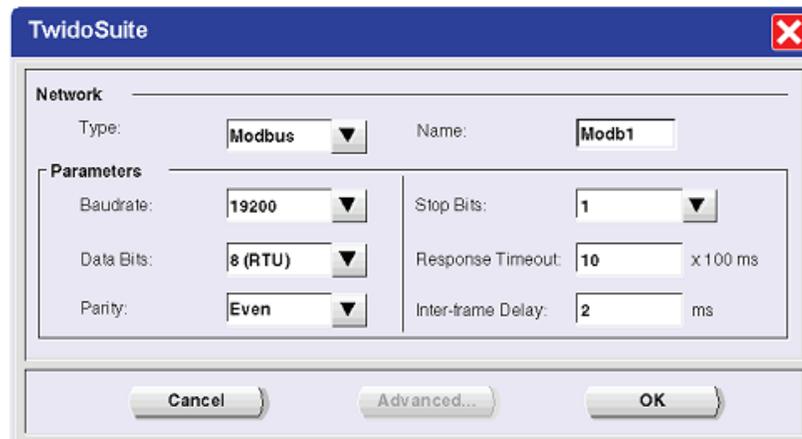
4. Para configurar el elemento Modbus, utilice cualquiera de los dos métodos:

- Haga clic en el icono Configuración de la barra de herramientas y seleccione el elemento Modbus en el gráfico de describir.
- Haga doble clic en el elemento Modbus en el gráfico de describir.

5. Para abrir el cuadro de diálogo Función asociado al hardware de conexión Modbus, utilice cualquiera de los dos métodos:

- Haga clic en el icono Configuración de la barra de herramientas y luego seleccione el enlace Modbus en el gráfico de describir.
- Haga doble clic en la conexión Modbus en el gráfico de describir.

6. Configure el cuadro de diálogo de funciones que aparece, como se explica en el siguiente gráfico:



**Figura III.42.** Configuración Modbus en Twido Suite  
*Fuente: Manual Twido Modbus Addressing*

7. Seleccionar Modbus en el Protocolo: cuadro Tipo

8. Establezca los parámetros de comunicación asociados:

- 7 bit, no paridad, 1 bit de parada.
- 8 bit, paridad par, 2 bits de parada.
- 8 bit, paridad impar, 2 bits de parada. (9)

### **3.4. PASARELA O CONVERTOR DE MODBUS SERIAL A ETHERNET**

Como se ha visto anteriormente, el PLC Twido TWDLMDA20DTK no cuenta con una conexión directa a Ethernet, por ello se necesita de un dispositivo especial capaz de convertir Modbus Serial (Twido) a Ethernet, para eso se necesita de una pasarela, dispositivo que se detalla a continuación.

#### **3.4.1. Definición y Funcionamiento de Pasarela**

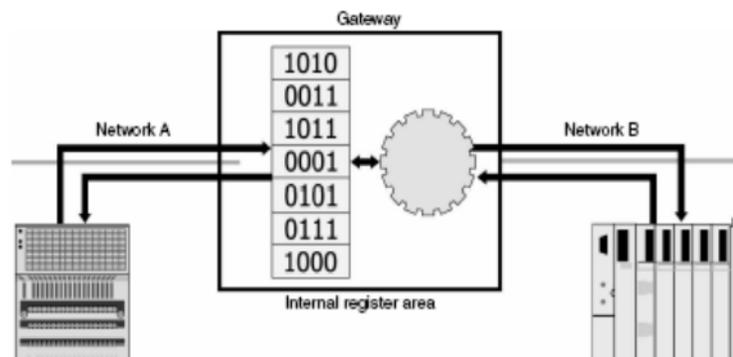
Una pasarela transfiere información entre sistemas o redes incompatibles y/o de distinto protocolo. Es decir, son dispositivos capaz de aceptar información proveniente de un canal serie y enviarlos vía TCP/IP hacia una IP y puerto determinado, y viceversa, acepta la entrada de datos vía TCP/IP y los retransmite vía serie.

Su área de trabajo son las capas superiores del modelo OSI, llegando desde el nivel de aplicación al nivel de transporte, e incluso hasta los niveles más inferiores. Aunque muchas pasarelas efectúan la adaptación de modo que también rectifique las diferencias que puedan existir, a nivel de infraestructuras, entre dos redes distintas.

Su principal función, es permitir la comunicación entre redes, para lo que se basan en la traducción o adaptación de protocolos. Para realizar esta traducción tienen

duplicada la pila de protocolos OSI, de modo que estas dos pilas sean las pilas de protocolos de los 2 protocolos a traducir, incluso aunque uno de los protocolos no sea conforme a la norma OSI. Cómo es lógico, esta traducción implica una ralentización en la transmisión de paquetes de una red a otra.

Cuando reciben los datos encapsulados de un protocolo, los van desencapsulando hasta el nivel más alto. Y a continuación irán encapsulando los datos en el otro protocolo desde el nivel más alto al nivel más bajo, para volver a dejar la información en la red, una vez que se ha completado la traducción. El nivel más bajo de la capa OSI en el que va a trabajar dependerá del tipo de pasarela, pudiendo ser principalmente el nivel de transporte o el nivel físico. (10)



**Figura III.43.** Funcionamiento de pasarela.  
*Fuente: Introducción a Ethernet Industrial pdf*

## VENTAJAS

- La respuesta de lado Ethernet es rápida (los datos se leen de la memoria compartida y no son afectados por el retardo de respuesta de los dispositivos serie).

- El tiempo de respuesta no se ve afectado por fallas en los dispositivos serie.
- Pueden usarse diferentes protocolos a cada lado de la pasarela.

### 3.4.2. Twido Port



**Figura III.44.** TwidoPort.

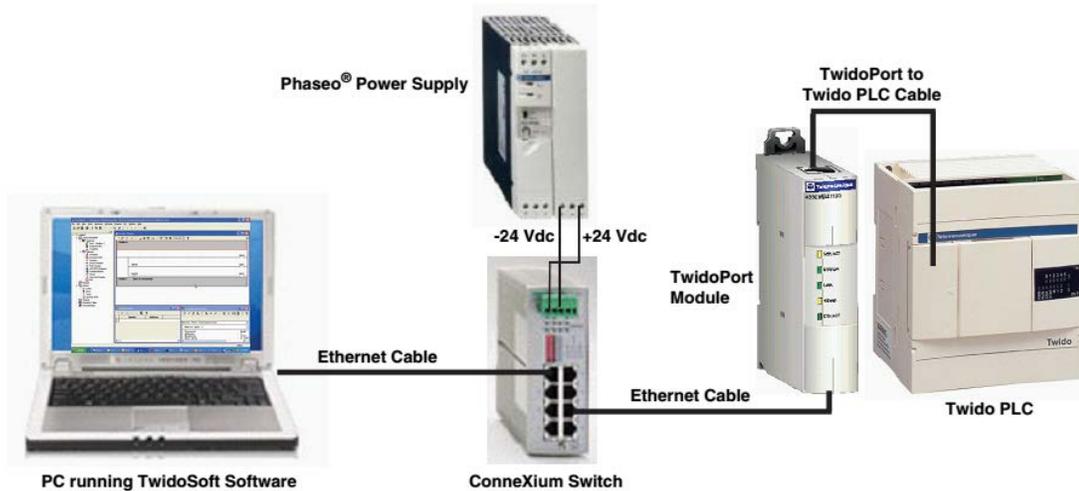
*Fuente: Manual Conexium TwidoPort*

TwidoPort añade conexiones Ethernet a la línea de productos Twido de Telemecanique. Es una pasarela entre un solo dispositivo Modbus/RTU (RS-485) Twido y la capa física de las redes Modbus/TCP en modo esclavo. TwidoPort no requiere una fuente de alimentación aparte, ya que obtiene la alimentación del controlador Twido a través de su puerto serie. Este módulo de pasarela sólo admite el modo esclavo.

TwidoPort admite hasta 8 conexiones simultáneas a Modbus/TCP. Si se intenta emplear más de 8 conexiones, se produce una disminución del rendimiento, ya que TwidoPort cierra la conexión con el tiempo de inactividad más largo para aceptar una petición de conexión nueva. Además admite autonegociación 10/100TX. Sólo se comunica en modo semi-dúplex. Admite la conmutación automática de los pares

trenzados a pares de transmisión y recepción para establecer la comunicación con el dispositivo final (MDI/MDI-X automático). (11)

### 3.4.2.1. Configuración del TwidoPort



**Figura III.45.** Conexión de TwidoPort con PLC Twido y Switch Ethernet.

*Fuente: Data Bulletin TwidoPort*

La configuración del Twido se lo realiza de la siguiente forma con la ayuda del software TwidoSoft 3.0.

1. Al emplear TwidoSoft (v. 3.0 o superior) para configurar el controlador Twido, cambiar el puerto de comunicación 1 al protocolo Modbus.
2. Hacer clic con el botón derecho del ratón en el icono Hardware situado en el control del árbol y seleccionar Agregar opción.
3. Seleccionar la opción 499TWD01100.
4. Configurar los parámetros de IP para TwidoPort.
5. Validar la configuración antes de descargarla en el controlador Twido.
6. Descargar el proyecto TwidoSoft en el controlador Twido.

7. Conectar un cable conectado a tierra al terminal de tornillo M3 en la parte inferior de TwidoPort.
8. Conectar el extremo mini-DIN del cable de conexión de TwidoPort a Twido (suministrado) en el puerto serie RS-485 del controlador Twido.
9. Conectar el extremo de conexión modular del cable de conexión de TwidoPort a Twido en el puerto serie de TwidoPort.
10. Conectar el conector RJ-45 de un cable de red Ethernet estándar en el puerto Ethernet de TwidoPort.

Se debe tomar en cuenta los siguientes parámetros:

- Se puede utilizar cualquier puerto Modbus RS-485 de Twido.
- El puerto Modbus RS-485 del controlador Twido debe configurarse a 9.600, 19.200 o 38.400 baudios para soportar la característica de velocidad de transmisión automática de TwidoPort.
- Para obtener la velocidad de transmisión automática inicial más rápida, seleccione 19200-8-N-1 con una dirección Modbus Twido de 1. (12)

### **3.5. SIEMENS S7-1200**

El controlador lógico programable (PLC) S7-1200 ofrece la flexibilidad y capacidad de controlar una gran variedad de dispositivos para las distintas tareas de automatización. La CPU incorpora un microprocesador, una fuente de alimentación integrada, así como circuitos de entrada y salida en una carcasa compacta, conformando así un potente PLC.

La CPU incorpora un puerto PROFINET para la comunicación en una red PROFINET. Los módulos de comunicación están disponibles para la comunicación en redes RS485 o RS232.



**Figura III.46.** PLC Siemens S7-1200  
*Fuente: Manual s/ Controlador programable S7-1200*

### 3.5.1. Módulos de Comunicación

La gama S7-1200 provee módulos de comunicación (CMs) que ofrecen funciones adicionales para el sistema. Hay dos módulos de comunicación: RS232 y RS485.

- La CPU soporta como máximo 3 módulos de comunicación
- Todo CM se conecta en lado izquierdo de la CPU (o en lado izquierdo de otro CM)



**Figura III.47.** Ubicación de conector para comunicación.  
*Fuente: Manual s/ Controlador programable S7-1200*

### 3.5.2. Profinet

Según la Totally Integrated Automation (TIA), Profinet es la evolución lógica del bus de campo Profibus DP y de Industrial Ethernet. Profinet como estándar de automatización, está basado en Ethernet.

Los objetivos de Profinet son:

- Ser un estándar abierto para la automatización basado en Industrial Ethernet.
- Que los componentes de Industrial Ethernet y Estándar Ethernet puedan utilizarse conjuntamente, aunque los equipos de Industrial Ethernet sean más robustos y más apropiados para el entorno industrial (temperatura, seguridad de funcionamiento, etc.).
- Usar estándares TCP/IP y de tecnologías de la información.
- Automatización con Ethernet en tiempo real.
- Integrar de forma directa sistemas con bus de campo.

Profinet especifica las funciones para la realización de una solución total de automatización desde la instalación de la red hasta el diagnóstico basado en la web.

Gracias a su estructura modular, Profinet puede ampliarse fácilmente con funciones futuras.

Por ello se ha distinguido las siguientes ventajas:

- Flexibilidad gracias al empleo de Ethernet y de los acreditados estándares IT.
- Ahorro de ingeniería y puesta en marcha gracias a la modularización.
- Protección de la inversión para equipos y aplicaciones Profibus.
- Más rápido que los actuales buses especiales en el ámbito de Motion Control.
- Amplio abanico de productos disponibles en el mercado.

#### ***3.5.2.1. Arquitectura Profinet***

Profibus International ha tenido en cuenta los siguientes aspectos para la arquitectura de Profinet:

- Comunicación entre aparatos de campo.
- Las arquitecturas Profibus existentes pueden integrarse dentro de Profinet, de este modo se protege la inversión para equipos Profibus y aplicaciones.
- Comunicación entre controladores como componentes de sistemas distribuidos.  
La estructura modular técnica es una garantía de ahorro, tanto en la ingeniería como en el mantenimiento.
- Técnica de instalación con conectores y componentes de red estandarizados.  
Así se aprovecha el potencial innovador de Ethernet y de los estándares IT.

### 3.5.3. Profinet en Siemens

La arquitectura Profinet se ha realizado de la siguiente forma:

- La comunicación entre los controladores y los aparatos de campo se realizan con Profinet IO.
- La comunicación entre los controladores como componentes de sistemas distribuidos se realiza mediante Profinet CBA (Component Based Automation).
- La técnica de instalación y los componentes de red se comercializan con la marca Simatic Net.

Los dispositivos Profinet de la familia de productos Simatic disponen de interfaces Profinet con y sin switch integrado.

Todo dispositivo Profinet puede ser identificado en la red de univoca a través de su interfaz Profinet. Para ello cada interfaz de Profinet dispone de:

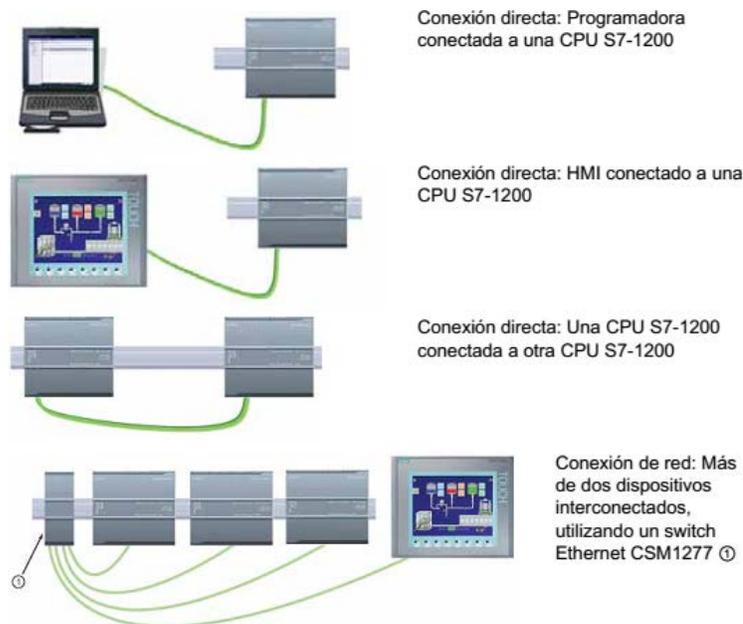
- Una dirección MAC (de fábrica).
- Una dirección IP.
- Un nombre (NameofStation).

La CPU S7-1200 incorpora un puerto PROFINET que soporta las normas Ethernet y de comunicación basada en TCP/IP. La CPU S7-1200 soporta los siguientes protocolos de aplicación:

- Transport Control Protocol (TCP)
- ISO en TCP (RFC 1006)

La CPU S7-1200 puede comunicarse con otras CPUs S7-1200, programadoras STEP 7 Basic, dispositivos HMI y dispositivos no Siemens que utilicen protocolos de comunicación TCP estándar. Hay dos formas de comunicación vía PROFINET:

- **Conexión directa:** La comunicación directa se utiliza para conectar una programadora, dispositivo HMI u otra CPU a una sola CPU.
- **Conexión de red:** La comunicación de red se utiliza si deben conectarse más de dos dispositivos (p. ej. CPUs, HMIs, programadoras y dispositivos no Siemens).



**Figura III.48.** Tipos de comunicación Profinet del S7-1200

*Fuente: Manual s/ Controlador programable S7-1200*

### **3.5.3.1. Número máximo de conexiones para el puerto PROFINET**

El puerto PROFINET de la CPU soporta las siguientes conexiones simultáneas.

- 3 conexiones para la comunicación entre dispositivos HMI y la CPU
- 1 conexión para la comunicación entre la programadora (PG) y la CPU

- 8 conexiones para la comunicación del programa del S7-1200 utilizando instrucciones del bloque T (TSEND\_C, TRCV\_C, TCON, TDISCON, TSEN, TRCV)
- 3 conexiones para la comunicación entre una CPU S7-1200 pasiva y una CPU S7 activa
  - La CPU S7 activa utiliza las instrucciones GET y PUT (S7-300 y S7-400) o ETHx\_XFER (S7-200).
  - Una conexión S7-1200 activa sólo es posible con las instrucciones del bloque T.

### ***3.5.3.2. Protocolos de Comunicación en S7-1200***

#### **Transport Control Protocol (TCP)**

TCP es un protocolo estándar descrito por RFC 793: Transmission Control Protocol. El objetivo principal de TCP es ofrecer un servicio de conexión segura y fiable entre pares de procesos. Este protocolo tiene las características siguientes:

- Protocolo de comunicación eficiente puesto que está vinculado estrechamente al hardware
- Adecuado para cantidades de datos medianas y grandes (hasta 8192 bytes)
- Ofrece numerosas prestaciones más a las aplicaciones, en particular:
  - Recuperación de errores
  - Control de flujo
  - Fiabilidad
- Protocolo orientado a la conexión

- Puede utilizarse muy flexiblemente con sistemas de terceros que soporten únicamente TCP
- Apto para routing
- Son aplicables sólo las longitudes de datos estáticas.
- Los mensajes se acusan.
- Las aplicaciones se direccionan usando números de puerto.
- La mayoría de los protocolos de aplicación (p. ej. TELNET y FTP) utilizan TCP.
- Es necesario programar la gestión de datos debido a la interfaz de programación SEND/RECEIVE.

#### **ISO on TCP (RFC 1006)**

ISO on TCP es un mecanismo que permite portar aplicaciones ISO a la red TCP/IP. Este protocolo tiene las características siguientes:

- Protocolo de comunicación eficiente vinculado estrechamente al hardware
- Adecuado para cantidades de datos medianas y grandes (hasta 8192 bytes)
- A diferencia de TCP, los mensajes tienen un indicador de fin y están orientados a los mensajes.
- Apto para routing; puede utilizarse en WAN
- Las longitudes de datos dinámicas son posibles.
- Es necesario programar la gestión de datos debido a la interfaz de programación SEND/ RECEIVE.

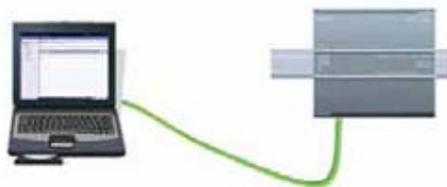
Puesto que utiliza Transport Service Access Points (TSAPs), el protocolo TCP permite establecer varias conexiones con una sola dirección IP (hasta 64K conexiones). Gracias a RFC 1006, los TSAPs identifican unívocamente estas conexiones de puntos finales de comunicación a una dirección IP.

En el área "Detalles de dirección" del diálogo "Parámetros de la conexión" se definen los TSAPs que deben utilizarse. El TSAP de una conexión en la CPU se introduce en el campo "TSAP local". El TSAP asignado a la conexión en la CPU interlocutora se introduce en el campo "TSAP del interlocutor" (13).

#### **3.5.4. Comunicación con una programadora**

Una CPU puede comunicarse con una programadora con STEP 7 Basic en una red. Al configurar la comunicación entre una CPU y una programadora debe considerarse lo siguiente:

- Configuración/instalación: Es preciso configurar el hardware.
- Para la comunicación entre dos interlocutores no se requiere un switch Ethernet. Un switch Ethernet se requiere sólo si la red comprende más de dos dispositivos.



**Figura III.49.** Conexión entre PC y S7-1200  
*Fuente: Manual S7 Controlador programable S7-1200*

### 3.5.5. Comunicación entre PLC's

Una CPU puede comunicarse con otra CPU utilizando las instrucciones TSEND\_C y TRCV\_C. Considere lo siguiente al configurar la comunicación entre dos CPUs:

- Configuración/instalación: Es preciso configurar el hardware.
- Funciones soportadas: Leer/escribir datos en una CPU interlocutora
- Para la comunicación entre dos interlocutores no se requiere un switch Ethernet. Un switch Ethernet se requiere sólo si la red comprende más de dos dispositivos.

Pasos necesarios para configurar la comunicación entre dos CPUs:

1. Establecer la conexión de hardware. Una interfaz PROFINET establece la conexión física entre dos CPUs. Puesto que la función "auto-crossover" está integrada en la CPU, es posible utilizar un cable Ethernet estándar o cruzado ("crossover") para la interfaz. Para conectar dos CPUs no se requiere un switch Ethernet.
2. Configurar los dispositivos. Es preciso configurar dos proyectos. Cada uno de ellos debe contener una CPU.
3. Configurar las conexiones de red lógicas entre dos CPUs.
4. Configurar una dirección IP en el proyecto. Utilice el mismo proceso de configuración. No obstante, es preciso configurar direcciones IP para dos CPUs (p. ej. PLC\_1 y PLC\_2).

5. Configurar los parámetros de transmisión y recepción. Las instrucciones TSEND\_C y TRCV\_C deben configurarse en ambas CPUs para habilitar la comunicación entre ellas.
6. Comprobar la red PROFINET. La configuración debe cargarse en cada una de las CPUs.

### ***3.5.5.1. Configurar los parámetros de conexión***

Toda CPU incorpora un puerto PROFINET que soporta la comunicación PROFINET estándar. Los protocolos Ethernet soportados se describen en los dos tipos de conexión siguientes:

**Tabla III.XII. Protocolos soportados en Ethernet**

<b>Protocolo</b>	<b>Nombre del protocolo</b>	<b>Uso</b>
RFC 1006	ISO on TCP	Fragmentación y reensamblado de mensajes
TCP	Transport Control Protocol	Transporte de tramas

*Fuente: Manual S7 Controlador programable S7-1200*

### ***3.3.5.2. Instrucciones de comunicación***

**Comunicación Ethernet abierta**

**Comunicación Ethernet abierta con conexión/desconexión automática (TSEND\_C y TRCV\_C)**

**Descripción de TSEND\_C**

TSEND\_C establece una conexión TCP o ISO on TCP con un interlocutor, envía datos y puede deshacer la conexión. Una vez configurada y establecida la conexión, la CPU la mantiene y la vigila automáticamente. TSEND\_C combina las funciones de TCON, TDISCON y TSEND.

El tamaño mínimo de los datos que pueden transmitirse con la instrucción TSEND\_C es un byte.

### Descripción de TRCV\_C

TRCV\_C establece una conexión TCP o ISO on TCP con una CPU interlocutora, recibe datos y puede deshacer la conexión. Una vez configurada y establecida la conexión, la CPU la mantiene y la vigila automáticamente. La instrucción TRCV\_C combina las funciones de las instrucciones TCON, TDISCON y TRCV.

El tamaño mínimo de los datos que pueden recibirse con la instrucción TRCV\_C es un byte. La instrucción TRCV\_C no soporta la transmisión de datos booleanos ni de matrices booleanas.

### Modos de recepción

TRCV\_C utiliza los mismos modos de recepción que la instrucción TRCV. La tabla siguiente muestra cómo se introducen los datos en el área de recepción.

**Tabla III.XIII. Modos de recepción del área de recepción**

<b>Variante de protocolo</b>	<b>Entrada de datos en el área de recepción</b>	<b>Parámetro "connection_type"</b>
TCP	Recepción de datos con la longitud especificada	B#16#11
ISO on TCP	Controlado por protocolo	B#16#12

*Fuente: Manual S7 Controlador programable S7-1200*

### Comunicación Ethernet mediante los protocolos TCP e ISO on TCP

Estas instrucciones del programa controlan el proceso de comunicación:

- TCON sirve para establecer una conexión.
- TSEND y TRCV permiten enviar y recibir datos.

- TDISCON deshace la conexión.

El tamaño mínimo de los datos que pueden transmitirse o recibirse con las instrucciones TSEND y TRCV es un byte. La instrucción TRCV no soporta la transmisión de datos booleanos ni de matrices booleanas.

Ambos interlocutores ejecutan la instrucción TCON para configurar y establecer la conexión. El punto final activo y el punto final pasivo de la comunicación se especifican mediante parámetros. Una vez configurada y establecida la conexión, la CPU la mantiene y la vigila automáticamente.

Si la conexión se deshace por ejemplo, debido a una interrupción de la línea o por el interlocutor remoto, el interlocutor activo intenta restablecer la conexión configurada. No es necesario volver a ejecutar TCON.

Una conexión existente se deshace y la conexión configurada se elimina cuando se ejecuta la instrucción TDISCON o cuando la CPU pasa a estado operativo STOP. Para configurar y restablecer la conexión es preciso ejecutar TCON de nuevo (14).

### 3.6. SWITCH INDUSTRIAL NO GESTIONADO (WEIDMULLER IE-SW8-WAVE)



**Figura III.50.** Switch Industrial No Gestionado IE SW8 WAVE.  
*Fuente: <http://www.weidmuller.es/>*

La falta de previsibilidad del comportamiento en el tiempo fue durante mucho tiempo utilizado como un argumento contra del uso de Ethernet en aplicaciones industriales. Sin embargo esto se producía por la experiencia basada en las primeras topologías de red en las que los usuarios se interconectan a través de los llamados hubs. Un hub reenvía todos los paquetes recibidos en un puerto a todos los demás puertos. Estas redes utilizan el método CSMA/CD (Detección de Portadora de Acceso Múltiple con Detección de Colisiones) para especificar quién puede transmitir y cuándo.

En el método CSMA la estación desea escuchar la transmisión del canal (detección de portadora) antes de transmitir los datos. Una estación puede transmitir sólo cuando el medio de transmisión ya no está siendo utilizado por otra estación. Si el medio de transmisión está en uso, la estación espera hasta que esté libre antes de enviar sus datos. No obstante debido a los retardos de propagación de señal, puede suceder que dos dispositivos transmiten al mismo tiempo. Con el fin de evitar una pérdida de datos

en caso de colisión, ambos transmisores deben ser capaces de detectar la colisión (colisión detectar) y retransmitir sus paquetes de datos después de un retraso seleccionado arbitrariamente. CSMA/CD es el estándar habitual para redes de 10 Mbit con hubs.

Tempranas topologías de red Ethernet, algunas de las cuales todavía están en uso hoy en día, usan hubs como estándar porque los complejos switches producidos en la década de 1980 y principios de 1990 eran muy caros.

El Switch Industrial no gestionado WEIDMULLER IE-SW8-WAVE permite a los usuarios configurar las infraestructuras de red para aplicaciones industriales de forma rápida y sencilla. No requiere ninguna configuración e incluso la asignación de pines de los cables de conexión es flexible gracias a su funcionalidad auto-cruzado (Auto-MDI / X). Esto significa que pueden utilizar cables directos para las conexiones entre los switches y dispositivos terminales y las conexiones entre los switches, dejando de lado el uso obligatorio de cables cruzados. Estos switches ofrecen beneficios para aplicaciones en condiciones difíciles, como por ejemplo:

- Este switch se puede utilizar en una amplia gama de temperaturas de funcionamiento para aplicaciones en salas fuera de las zonas de clima templado sin aire acondicionado.
- Separar rutas para cables de señal y de alimentación no siempre es posible en aplicaciones industriales. Los puertos de fibra óptica opcionales garantizan un funcionamiento sin problemas, incluso a largas distancias a través de campos magnéticos potentes y están disponibles en versiones multimodo o monomodo.

- Al actualizar las redes, el ancho de componente uniforme permite a las unidades existentes ser intercambiados por otros nuevos con más puertos sin tener que cambiar todos los demás componentes en el tren o tener que ampliar el gabinete de conmutación.
- Este switch proporciona a 8 puertos RJ-45 y 2 puertos Fibra Óptica opcionales en una carcasa compacta de aluminio.
- El amplio rango de suministro de voltaje garantiza un funcionamiento estable incluso si la mayoría de dispositivos terminales conectados podría ya estar apagados.
- El plug-in de conexiones de alimentación, que también incluyen una opción de fuente de alimentación redundante, permitirá una rápida conexión y desconexión de la alimentación eléctrica cuando se trabaja en la red.(15)

### **3.7. CABLE ETHERNET INDUSTRIAL**

Los conectores y cables utilizados para la conexión Ethernet deben cumplir un gran número de exigencias, debido a diversos factores existentes en la industria como la humedad, las elevadas variaciones de temperatura, las vibraciones y los golpes entre otros.

El conector RJ45 de 8 polos especificado en la IEC 60603-7 se ha establecido para el cableado Ethernet y está disponible en las versiones Cat. 5 y Cat. 6 conforme a la norma ISO/IEC 11801:2002.

### 3.7.1. Tipología de cables para Ethernet

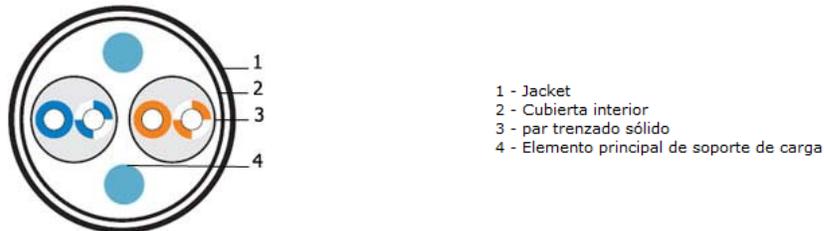
Los cables de cobre simétricos se utilizan en el cableado industrial Ethernet. Los cables de par trenzado con 2 o 4 pares trenzados son los que se utilizan normalmente, que cumplen como mínimo las exigencias de Cat. 5 conforme a la ISO/IEC 11801:2002. Los cables están equipados con una pantalla trenzada y un papel de aluminio protector (cable SF/UTP). En los cables de Cat. 6, los pares de cable también tienen una pantalla con papel de aluminio (cable s/FTP). En la versión de 4 pares trenzados, ambos tipos de cable también se pueden utilizar para Ethernet Gigabit.



**Figura III.51.** Tipos de Cable para conexión Ethernet.

Fuente: <http://www.conelectronica.com/>

#### 3.7.1.1. Cable Industrial Ethernet, 2x2x24 AWG (conductores monolíticos) UTP



**Figura III.52.** Cable UTP 2 par trenzado para conexión Ethernet.

Fuente: <http://www.hyperline.com/>

El cable sin blindaje de categoría 5 se compone de 2 pares trenzados cubiertos de PVC chaqueta trenzada con elementos portantes. El cable está diseñado para su aplicación en las redes industriales como Ethernet Industrial. Este cable cumple con ANSI/TIA/EIA-568-B.2 e ISO/IEC 11801, UL 1581 VW-1 y IEC 60332.1.

**Tabla III.XIV. Características Técnicas del cable UTP 2 Par Trenzado.**

<b>CARACTERÍSTICAS TÉCNICAS</b>	
Diámetro del conductor (núcleo)	0.51 mm (0.02 ", 24 AWG)
Diámetro del conductor aislado	0,9 mm (0,03 ")
Diámetro de la funda interior	4,4 mm (0,17 ")
Peso del cable por cada 1000 pies	13 kg (28,8 libras)
Diámetro exterior del cable	5,9 mm (0,23 ")
Temperatura de la operación	-40 ° C - 70 ° C (-40 ° F - 158 ° F)
Cableado de temperatura	-40 ° C - 60 ° C (-40 ° F - 140 ° F)
Radio de curvatura	20 diámetros exteriores (relleno), 10 diámetros exteriores (funcionamiento)
Fuerza de tensión	45 N

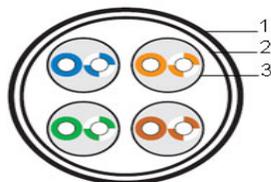
*Fuente: <http://www.hyperline.com>*

**Tabla III.XV. Características Eléctricas del cable UTP 2 Par Trenzado.**

<b>CARACTERÍSTICAS ELÉCTRICAS</b>	
Resistencia de los conductores de corriente continua máxima a 20 ° C (68 ° F)	94 Ohm / km (151,28 Ohm / km)
Min.. resistencia de aislamiento	5 GOhm / km (8,05 GOhm / milla)
Max. Asimetría óhmico	2%
Capacitiva asimetría en la frecuencia de 1 kHz	1.6 pF / m (0,49 pF / ft)
Capacidad mutua en la frecuencia de 1 kHz	46 pF / m (14,02 pF / ft)
La velocidad de propagación	68-70%
Max. voltaje de funcionamiento	220 V
Resistencia Dieléctrica	700 V / 1 min
Impedancia de onda a la frecuencia de 1-100 MHz	100 ± 15 Ohm
El retardo de propagación a la frecuencia de 1 kHz	5,7 ns / m (1,74 ns / m)
El cambio de fase en la frecuencia de 1-100 MHz	15 ns/100 m (4,57 pies ns/100)
Atenuación de acoplamiento al 1-100 MHz	40 dB

*Fuente: <http://www.hyperline.com/>*

### 3.7.1.2. Cable Industrial Ethernet, 4x2x24 AWG (conductores monolíticos) UTP



1 - Jacket  
2 - Cubierta interior  
3 - par trenzado sólido

**Figura III.53.** Cable UTP 4 par trenzado para conexión Ethernet.

*Fuente: <http://www.hyperline.com/>*

El cable sin blindaje de categoría 5 consta de 4 pares trenzados juntos retorcidos cubiertos con chaqueta de PVC doble. El cable está diseñado para redes industriales como Ethernet Industrial. El cable cumple con ANSI/TIA/EIA-568-B.2 e ISO / IEC 11801, UL 1581 VW-1 y IEC 60332.1.

**Tabla III.XVI. Características Técnicas del cable UTP 4 Par Trenzado.**

<b>CARACTERÍSTICAS TÉCNICAS</b>	
Diámetro del conductor (núcleo)	0.51 mm (0.02 ", 24 AWG)
Diámetro del conductor aislado	0,9 mm (0,03 ")
Diámetro de la funda interior	4,9 mm (0,19 ")
Peso del cable por cada 1000 pies	15 kg (33 libras)
Diámetro exterior del cable	5,9 mm (0,23 ")
Temperatura de la operación	-40 ° C - 70 ° C (-40 ° F - 158 ° F)
Cableado de temperatura	-40 ° C - 60 ° C (-40 ° F - 140 ° F)
Radio de curvatura	20 diámetros exteriores (relleno), 10 diámetros exteriores (funcionamiento)
Fuerza de tensión	90 N

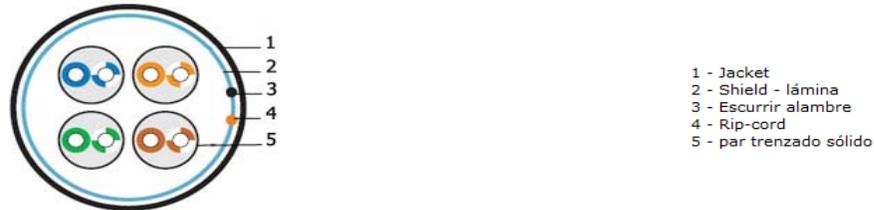
*Fuente: <http://www.hyperline.com/>*

**Tabla III.XVII. Características Eléctricas del cable UTP 4 Par Trenzado.**

<b>CARACTERÍSTICAS ELÉCTRICAS</b>	
Resistencia de los conductores de corriente continua máxima a 20 ° C (68 ° F)	94 Ohm / km (151,28 Ohm / km)
Min.. resistencia de aislamiento	5 GOhm / km (8,05 GOhm / milla)
Max. Asimetría óhmico	2%
Capacitiva asimetría en la frecuencia de 1 kHz	1,6 pF / m (0,49 pF / ft)
La velocidad de propagación	68-70%
Max. voltaje de funcionamiento	220 V
Resistencia Dieléctrica	700 V / 1 min
Impedancia de onda a la frecuencia de 1-100 MHz	100 ± 15 Ohm
El retardo de propagación a la frecuencia de 1 kHz	5,7 ns / m (1,74 ns / m)
El cambio de fase en la frecuencia de 1-100 MHz	15 ns/100 m (4,57 pies ns/100)
Atenuación de acoplamiento al 1-100 MHz	40 dB

*Fuente: <http://www.hyperline.com/>*

### 3.7.1.3. Cable diseñado para redes industriales *Industrial Ethernet, 4x2x24 AWG FTP*



**Figura III.54.** Cable FTP 4 par trenzado para conexión Ethernet.

Fuente: <http://www.hyperline.com/>

El cable apantallado de categoría 5 consta de 4 pares en aislamiento de poliolefina, el blindaje común está fijado a la cubierta de PVC. El cable incluye un conductor de drenaje y ripercord. El cable diseñado para redes industriales como Ethernet y la aplicación en condiciones severas de operación con un alto nivel de ruidos externos industrial, tiene códigos de color de los pares para aplicaciones como EtherNet/IP, soporta aplicaciones como Gigabit Ethernet, 100BaseTX, 100BaseVG AnyLAN, 155ATM, 622ATM , RS-422. El cable es compatible con conectores como conector RJ-45. El cable cumple con IEC 11801, TIA/EIA 568-B.2; NEMA WC-63.1. (16)

**Tabla III.XVIII. Características Técnicas del cable FTP 4 Par Trenzado**

<b>CARACTERÍSTICAS TÉCNICAS</b>	
Diámetro del conductor (núcleo)	0,51 mm (0,02 " , 24 AWG)
Diámetro del conductor aislado	1,08 mm (0,04 " )
El espesor de la camisa exterior	0,76 mm (0,03 " )
Espesor del aislamiento	0,24 mm (0,01 " )
Peso del cable por cada 1000 pies	14,5 kg (32 libras)
Diámetro exterior del cable	6,73 mm (0,26 " )
Temperatura de la operación	-40 ° C - 75 ° C (-40 ° F - 167 ° F)
Cableado de temperatura	-40 ° C - 65 ° C (-40 ° F - 149 ° F)
Radio de curvatura	25,4 mm (1,00 " )
Fuerza de tensión	máx. 155.68 N

Fuente: <http://www.hyperline.com/>

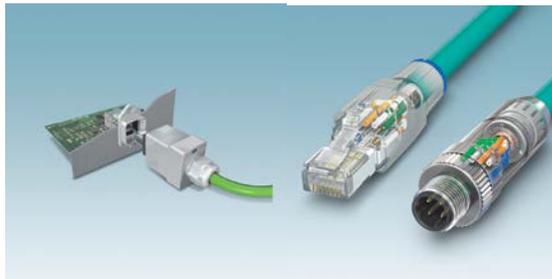
**Tabla III.XIX. Características Eléctricas del cable FTP 4 Par Trenzado.**

<b>CARACTERÍSTICAS ELÉCTRICAS</b>	
Resistencia de los conductores de corriente continua máxima a 20 ° C (68 ° F)	Ohm/100 9,38 m (2,86 pies Ohm/100)
Max. Asimetría óhmico	3%
Capacitiva asimetría en la frecuencia de 1 kHz	330 pF/100 m (101 pies pF/100)
Capacidad mutua en la frecuencia de 1 kHz	49 pF / m (14,93 pF / ft)
La velocidad de propagación	70%
El retardo de propagación a la frecuencia de 1 kHz	538 ns/100 m (163.98 ft ns/100)
Max. voltaje de funcionamiento	300 V
Cambio de fase	45 ns/100 m (13,72 pies ns/100)

*Fuente:* <http://www.hyperline.com/>

### 3.7.2. Conectores para Ethernet

Para el cableado Ethernet en IP67, el conector M12 se puede utilizar como una alternativa al RJ45. El conector M12 con codificación D ya ha sido definido para Ethernet Industrial en su versión de 4 polos conforme a la norma IEC 61076-2-101-am1 (corrección 1). Además del conector RJ45 con cierre push-pull, la PNO también ha especificado la versión M12 para el cableado Profinet. A parte de ser IP67, el conector M12, ya establecido en el campo del sensor/actuador y cableado de bus de campo, se caracteriza por tener dimensiones compactas y un gran comportamiento ante vibraciones. Sin embargo, no todas las versiones son apropiadas para el uso en redes Ethernet, donde las propiedades de transmisión deben cumplir como mínimo la Cat. 5 según la ISO/IEC 11801:2002. (17)

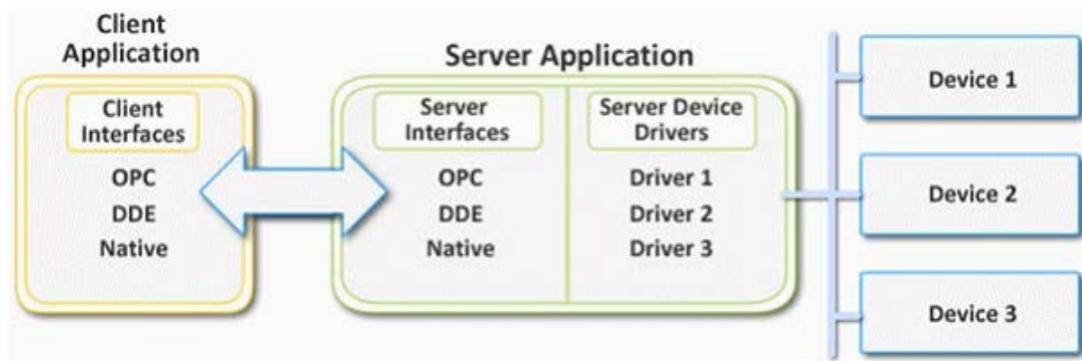


**Figura III.55.** Conectores RJ-45 y M12 para Ethernet.

*Fuente:* <http://www.conectronica.com/>

### 3.8. OPC Kepserver

El servidor OPC utilizado es el KEPServer, ya que es un servidor sencillo, casi intuitivo, de fácil adquisición, ofrece seguridad en la conexión, no ocasiona demasiados problemas de compatibilidad entre dispositivos y es compatible con cualquier plataforma de desarrollo de HMI, está diseñado para comunicaciones precisas, instalación rápida y la interoperabilidad sin precedentes entre las aplicaciones cliente, dispositivos y sistemas industriales. El diseño y única interfaz de usuario proporciona un acceso coherente de las aplicaciones basadas en estándares (como OPC) y aplicaciones no basadas en estándares con interfaces nativas.



**Figura III.56.** OPC KepServer.  
*Fuente: Manual KEPServerEx V5 Help.*

#### 3.8.1. Componentes

El servidor implementa una arquitectura cliente/servidor, dentro de sus componentes están: configuración, tiempo de ejecución (runtime), administración y registro de eventos (Event log).

## **Configuración**

La configuración es la interfaz cliente-usuario que puede modificar el proyecto en tiempo de ejecución. La configuración se puede iniciar de múltiples usuarios y eventualmente admitir la configuración de tiempo de ejecución remota.

## **Importación y Exportación CSV**

Este servidor soporta la importación y exportación de tags en Variables de Archivos Separados por Comas (CSV), cuando se usa la importación y exportación CSV, tags son creadas en la aplicación deseada.

## **Tiempo de ejecución (Runtime)**

El tiempo de ejecución se inicia como un servicio por defecto. Los clientes pueden conectarse con el tiempo de ejecución de forma remota o local.

## **Administración**

La Administración se utiliza para ver y/o modificar la configuración e iniciar las aplicaciones que pertenecen a la administración de usuarios y el servidor.

## **Registro de Eventos (Event Log)**

El servicio Registro de eventos recopila información, advertencias y sucesos de error. Estos eventos son enviados a la ventana de Registro de Eventos de la Configuración de visualización.

### **3.8.2. Modos de Proceso**

El modo del tiempo de ejecución mientras el servidor está en ejecución, sin embargo hacer esto mientras un cliente está conectado puede interrumpir la conexión por un corto periodo de tiempo. Los modos de operación son Service System e Interactive.

#### **Service System**

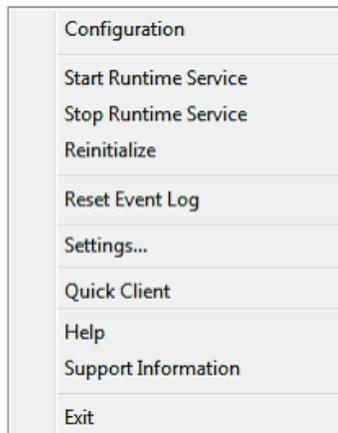
Por defecto el servidor está instalado y se ejecuta como un servicio. Cuando System Service es seleccionado, el tiempo de ejecución no requiere la intervención del usuario y se empezara cuando se inicie el sistema operativo. Esto proporciona al usuario un acceso independiente al servidor por los clientes.

#### **Interactive**

Cuando Interactive es seleccionado, el tiempo de ejecución permanece detenido hasta que el cliente intente conectarlo. El tiempo de ejecución también se detendrá si la cuenta de usuario cierra sesión en el sistema operativo. Interactive es requerido para las siguientes condiciones:

### **3.8.3. Menú de Administración**

El menú de administración es una herramienta es utilizada para ver y/o modificar la configuración de administración de usuarios y aplicaciones de servidor. El menú que debe aparecer es el siguiente:



**Figura III.57.** Menú de Administration del Kepserver.  
*Fuente: Manual KEPServerEx V5 Help.*

Descripción de las opciones del menú de administración:

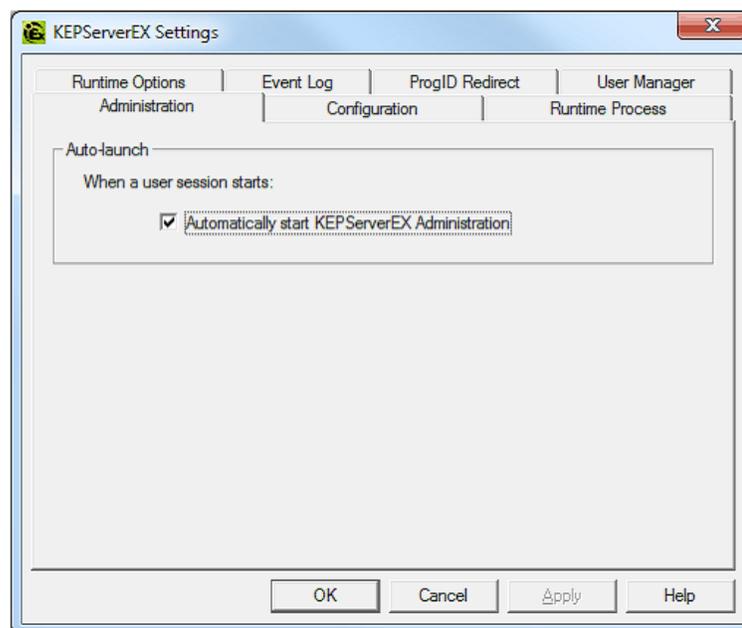
- *Configuration:* Esta opción inicia la configuración del servidor OPC.
- *Start Runtime Service:* inicializa el tiempo de ejecución del servidor y carga el proyecto de ejecución determinado.
- *Stop Runtime Service:* desconecta todos los clientes y guarda el proyecto de ejecución por defecto antes de detener el proceso de ejecución del servidor.
- *Reinitialize:* esta opción desconecta todos los clientes y restablecer el tiempo de ejecución. Se guarda automáticamente y vuelve a cargar el proyecto de ejecución por defecto sin detener el proceso de ejecución del servidor.
- *Reset event log:* resetea el registro de eventos. La fecha, la hora y el origen del reajuste se añadirán al registro de eventos en la ventana de configuración.
- *Settings:* abre el cuadro de diálogo Configuración.
- *OPC UA Configuration:* inicia el Administrador de configuración de OPC UA.
- *OPC .NET Configuration:* inicia el Administrador de configuración de OPC .NET.
- *Quick Client:* inicia el Cliente Rápido.

- *License Utility*: inicia Utilidad de licencia del servidor.
- *Help*: inicia la documentación de ayuda del servidor.
- *Support Information*: inicia un cuadro de diálogo que contiene información básica resumida sobre el servidor y los controladores actualmente instalados para su uso.
- *Exit*: cierra la Administración y lo elimina de la bandeja del sistema. Para ver de nuevo, seleccione en el menú Inicio de Windows.

### Configuración- Settings

#### Settings- Administration

La ficha Administración se utiliza para configurar las acciones de la administración en tiempo de ejecución.



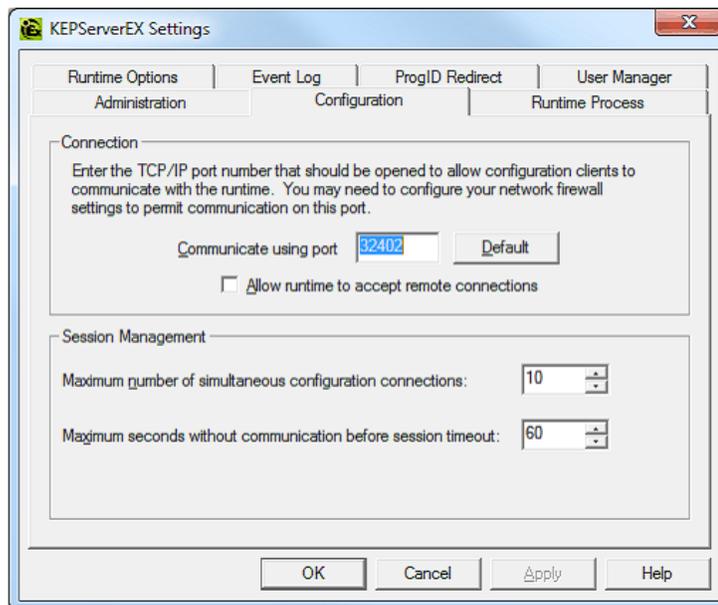
**Figura III.58.** Pestaña Settings-Administration.

*Fuente: Manual KEPServerEx V5 Help.*

- **Automatically start Administration:** Permite que la administración se inicie automáticamente. Esta aplicación permite conexiones rápidas a diversas herramientas de servidor, incluyendo la consola de Configuración, Utilidad Licensing, consola de Administrador de usuarios y controles para detener e iniciar el servicio en tiempo de ejecución.

### Settings – Configuration

La pestaña de configuración se utiliza para configurar el modo de la configuración e interactúa con el tiempo de ejecución.



**Figura III.59.** Pestaña Settings-Configuration.

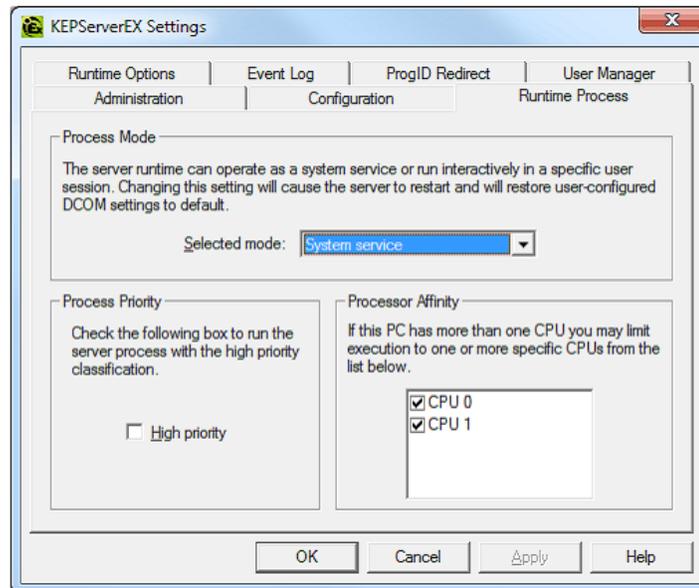
*Fuente: Manual KEPServerEx V5 Help.*

- **Communicate using port:** Este parámetro es el puerto TCP/IP que se utiliza para la comunicación entre la configuración y el tiempo de ejecución.

- **Allow runtime to accept remote connections:** Cuando se activa, el tiempo de ejecución será capaz de aceptar conexiones remotas. Por defecto está desactivada.
- **Maximum number of simultaneous configuration connections:** Especifica el número de conexiones de configuración que se pueden hacer en el tiempo de ejecución de una sola vez. El rango es de 1 a 64.
- **Maximum seconds without communication before session timeout:** Establece la cantidad de tiempo que la conexión de la consola puede permanecer inactiva antes de que el tiempo se termine. El rango es de 10 a 3600 segundos

### Settings - Runtime Process

Se utiliza para especificar el modo del tiempo de ejecución del servidor, así como la forma en que utiliza los recursos del PC.



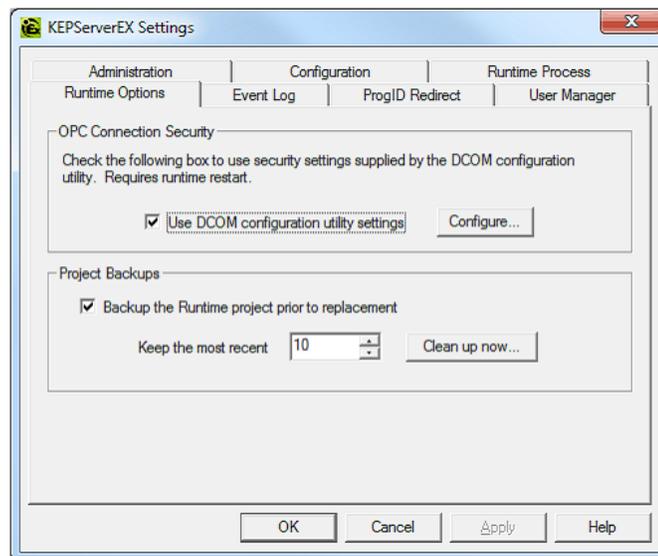
**Figura III.60.** Pestaña Settings-Runtime Process.

*Fuente: Manual KEPServerEx V5 Help.*

- **Selected Mode:** Especifica si el servidor se ejecuta como System Service o Interactive. De forma predeterminada, el servidor se instala y se ejecuta como System Service. Si cambia esta opción a todos los clientes, tanto de configuración y el proceso, al ser desconectado y el servidor se detiene y se reinicia.
- **High Priority:** Establece la prioridad del proceso del servidor a alto.
- **Process Affinity:** Especifica que servidor de CPUs puede ser ejecutado cuando este se ejecuta en PCs que contengan más de uno.

### Settings - Runtime Options

Se utiliza para cambiar la configuración en el proyecto que se está ejecutando en el tiempo de ejecución.



**Figura III.61.** Pestaña Settings-Runtime Options.

*Fuente: Manual KEPServerEx V5 Help.*

- **Use DCOM configuration utility settings:** Permite al usuario seleccionar la autenticación, así como poner en marcha y acceder a los requisitos de seguridad a través de la utilidad de configuración DCOM. Además, los usuarios pueden especificar tanto el nivel de seguridad para implementar y restringir el acceso a determinados usuarios y/o aplicaciones.
- **Backup the Runtime project prior to replacement:** Permite que el proyecto realice una copia de seguridad antes de que se sobrescriba. La ubicación de la copia de seguridad se muestra en el registro de eventos.
- **Keep the most recent:** Limita el número de copias de seguridad que serán guardadas en el disco. El rango es de 1 a 1000, por defecto es 10.
- **Clean up now:** Invoca un cuadro de diálogo de confirmación que permite a los usuarios eliminar todas las copias de seguridad en tiempo de ejecución del proyecto.

### **Settings - Event Log**

Define la comunicación y la persistencia de los ajustes del registro de eventos, registro de diagnóstico OPC y registro de diagnóstico de comunicación.

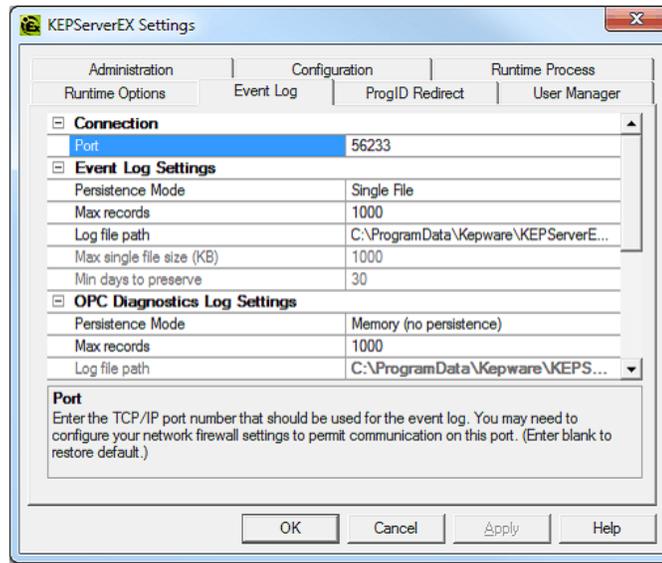


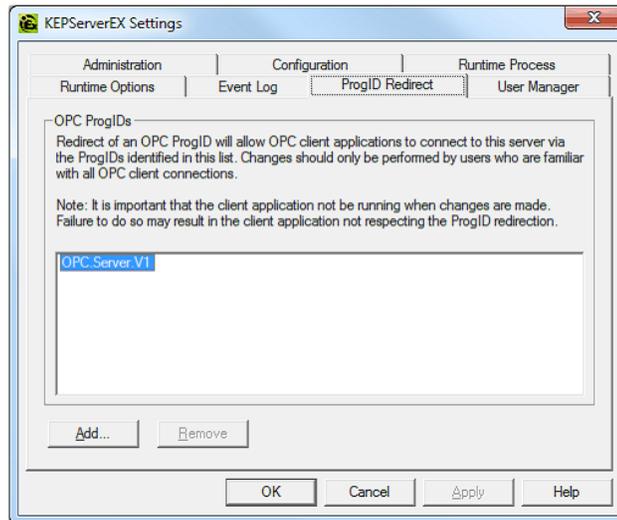
Figura III.62. Pestaña Settings-Event Log.

Fuente: Manual KEPServerEx V5 Help.

- **Port:** especifica el puerto TCP/IP que será utilizado para la comunicación entre el registro y el tiempo de ejecución.
- **Persistence Mode:** Especifica el modo de persistencia de la sesión. Las opciones incluyen :
  - Memory
  - Single File
  - Extended Data Store
- **Max records:** Especifica el número de registros que el sistema de registro mantendrá antes que los registros más antiguos empiezan a ser eliminados. El rango válido es de 100 a 30000 registros.
- **Log file path:** Especifica donde los registros van a ser almacenados.

### Settings - ProgID Redirect

Muchas aplicaciones cliente OPC se conectan a un servidor OPC a través ProgID del servidor OPC. Los usuarios que necesitan migrar o actualizar a un nuevo servidor OPC a menudo prefieren hacerlo sin cambiar su base de datos de etiquetas. Este servidor ofrece redirección ProgID para ayudar a los usuarios en estas transiciones.

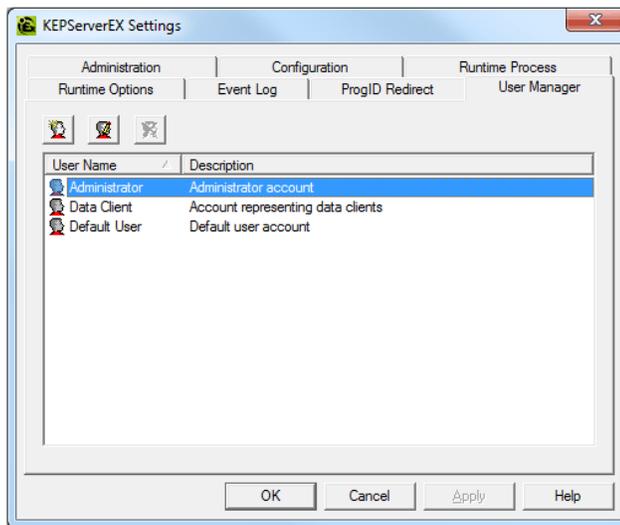


**Figura III.63.** Pestaña Settings-ProgID Redirect.

*Fuente: Manual KEPServerEx V5 Help.*

### Settings - User Manager

Este servidor incluye un Administrador de usuarios integrado que controla qué usuarios tienen acceso al tiempo de ejecución, y qué privilegios una vez que se han conectado. Cualquier acción del usuario que pueden influir o alterar el funcionamiento del servidor se registra en servidor de registro de eventos.



**Figura III.64.** Pestaña Settings-User Manager.

*Fuente: Manual KEPServerEx V5 Help.*

Cuentas de usuario:

- **Administrator:** Se utiliza para añadir nuevos usuarios al sistema o cambiar la configuración de cuentas existentes. La cuenta de administrador no se puede eliminar, pero la contraseña se puede cambiar.
- **Data Client:** Se utiliza cuando un cliente de datos externo intenta realizar una acción que se rige por el control de acceso. Es compatible con privilegios estándar pero no puede tener contraseña asignada, no se puede utilizar para el acceso anónimo a través de OPC UA, y no puede ser utilizado para crear una sesión de la configuración del servidor.
- **Default User:** se utiliza cuando no hay otra cuenta que esté activa, que es la condición normal del servidor. La cuenta de usuario predeterminado no se puede eliminar.

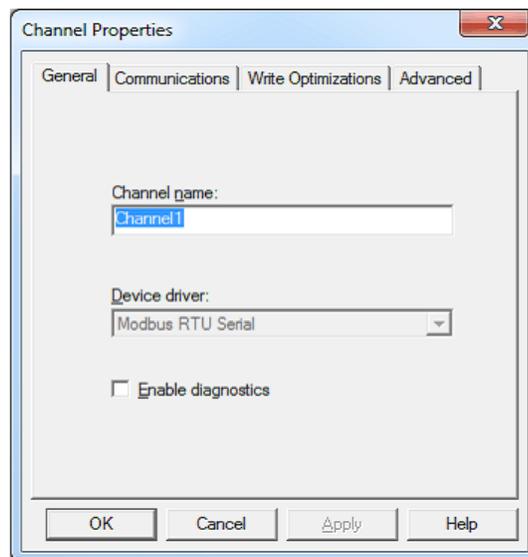
### 3.8.4. Componentes Básicos del Servidor

#### 1. Canal – Channel

Un canal representa un medio de comunicación desde el PC a uno o más dispositivos externos, puede ser utilizado para representar un puerto serie, una tarjeta instalada en el PC o un zócalo de Ethernet.

Antes de añadir dispositivos a un proyecto, los usuarios deben definir el canal que se utilizará cuando se comunica con los dispositivos. Un canal y un controlador de dispositivo están estrechamente ligados. Cada protocolo o controlador que se utilice en un proyecto de servidor se denomina un canal. Un proyecto de servidor puede consistir en muchos canales con el mismo controlador de comunicaciones o con controladores de comunicaciones únicos.

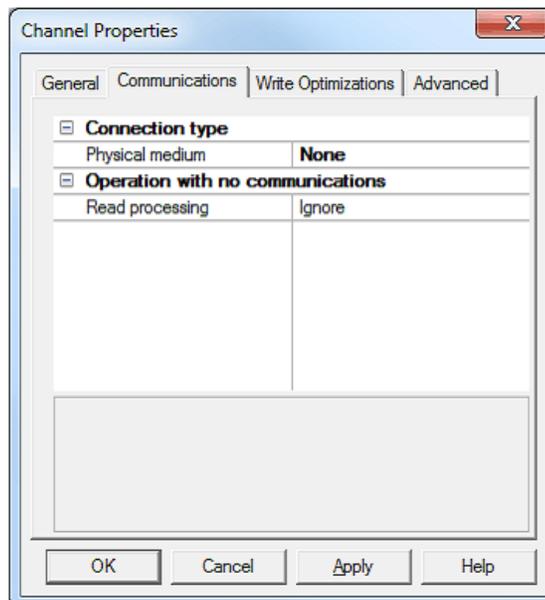
#### 1.1. Channel Properties- General:



**Figura III.65.** Propiedades de un Canal – Pestaña General.  
*Fuente: Manual KEPServerEx V5 Help.*

- **Channel Name:** especifica el nombre del canal. En la aplicación del servidor cada nombre de canal debe ser único, el nombre del canal será parte de la información del navegador OPC.
- **Device driver:** especifica el controlador del dispositivo que fue elegido en la creación del canal.
- **Enable diagnostics:** esta opción hará que la información de diagnóstico del canal esté disponible para la aplicación OPC.

### 1.2. Channel Properties- Communications:



**Figura III.66.** Propiedades de un Canal – Pestaña Comunicaciones.

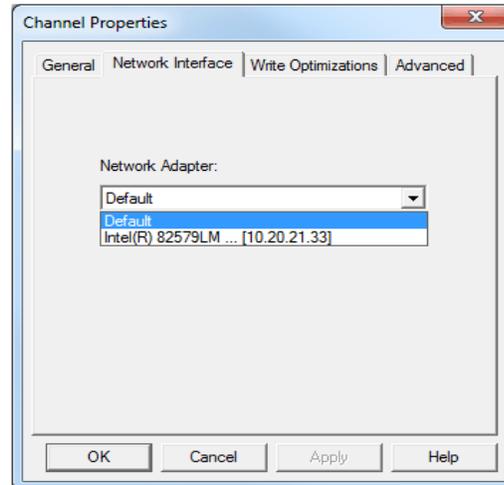
*Fuente: Manual KEPServerEx V5 Help.*

Los parámetros de esta pestaña varían en función del controlador y tipo de conexión que se seleccionan. El tipo de conexión especifica el tipo de dispositivo de hardware que se utiliza para las comunicaciones de datos. Las opciones disponibles son:

- None

- COM Port
- Modem
- Ethernet Encapsulation

### 1.3. Channel Properties - Network Interface



**Figura III.67.** Propiedades de un Canal – Pestaña Interfaz de Red.  
*Fuente: Manual KEPServerEx V5 Help.*

Con Encapsulación Ethernet, prácticamente todos los drivers disponibles en la actualidad tienen algún tipo de soporte de Ethernet. Una interfaz de red se utiliza, ya sea para un controlador de forma nativa basada en Ethernet o un controlador de serie configurado para la encapsulación de Ethernet. Este parámetro está disponible solo para controladores Ethernet.

## 1.4. Channel Properties - Write Optimizations

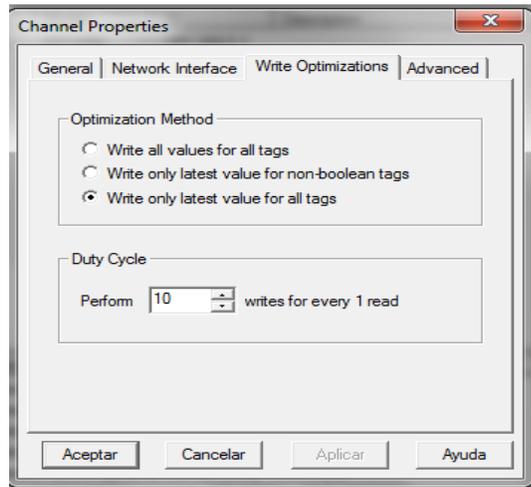


Figura III.68. Propiedades de un Canal – Pestaña Optimización de Escritura.

Fuente: Manual KEPServerEx V5 Help.

- **Optimization Method:** controla como pasan los datos de escritura al controlador de comunicaciones subyacente. También ajusta la relación a la cual la escritura será procesada y enviada al dispositivo. Cuenta con tres opciones:
  - Escribir todos los valores para todos los tags.
  - Escribir solo el último valor para tags no booleanos.
  - Escribir solo el último valor para todos los tags.
- **Duty Cycle:** es utilizado para controlar la relación entre las operaciones de Lectura y Escritura. Por defecto su valor es de 10.

## 1.5. Channel Properties – Advanced

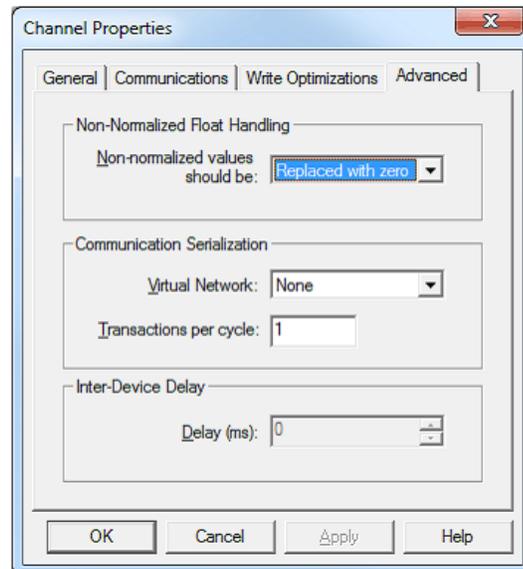


Figura III.69. Propiedades de un Canal – Pestaña Avanzado.

Fuente: Manual KEPServerEx V5 Help

- **Non-normalized values should be:** Este parámetro especifica cómo manejará un controlador no normalizado valores de punto flotante IEEE-754. Las opciones incluyen no modificado y sustituido por cero.
- **Virtual Network:** Este parámetro especifica el modo del canal de comunicación de la serialización. Las opciones son None y Network 1 - Network 50. El ajuste predeterminado es Ninguno.
- **Transactions per cycle:** Especifica el número transacciones simples de lectura/escritura bloqueado/no bloqueado que se pueden producir en el canal. El rango válido es de 1 a 99. La configuración predeterminada es 1.
- **Delay:** Especifica la cantidad de tiempo que el canal de comunicaciones retrasa el envío de una nueva petición al siguiente dispositivo después de que los datos

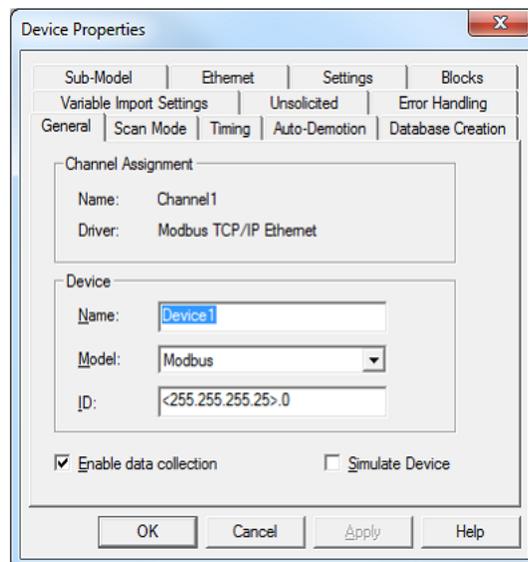
han sido recibidos desde el dispositivo actual en el mismo canal. El rango válido es de 0 a 60000 milisegundos.

## 2. Dispositivos-Device

Representan los PLC u otro hardware con el que el servidor se comunica. El controlador de dispositivo que el canal está utilizando restringe la selección del dispositivo.

### 2.1. Device Properties – General

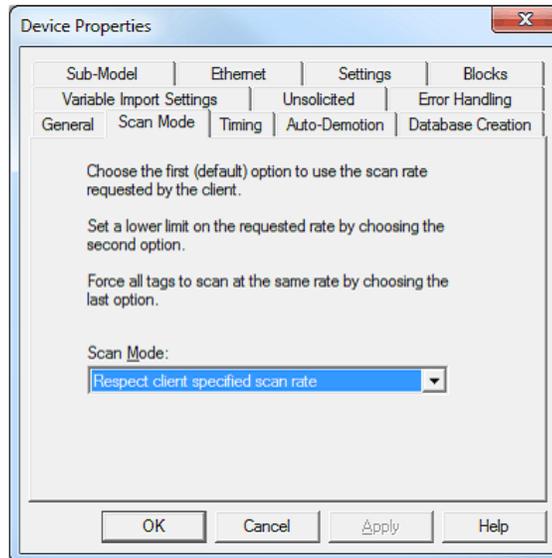
La ficha General de las Propiedades de Dispositivos tiene 2 apariencias, para Dispositivos Seriales y Dispositivos Ethernet. En este caso se mostraran las opciones para Dispositivos Ethernet.



**Figura III.70.** Propiedades de Dispositivo – Pestaña General.  
*Fuente: Manual KEPServerEx V5 Help*

- **Name:** Este parámetro especifica el nombre del dispositivo. Es un nombre definido por el usuario lógico que puede ser de hasta 256 caracteres de longitud, y se puede utilizar en múltiples canales.
- **Model:** Define el tipo específico de dispositivo que está asociada con este ID. El contenido del menú de desplegable dependerá del tipo de controlador de comunicaciones que se utiliza.
- **ID:** Especifica el nodo del controlador del dispositivo. El tipo de ID dependerá del controlador de comunicaciones que se utiliza. Para muchos controladores de comunicación, la ID es un valor numérico.
- **Enable Data Collection:** Controla el estado activo del dispositivo. Las comunicaciones no se intentará una vez que un dispositivo ha sido desactivado. Este parámetro se puede cambiar en cualquier momento a través de la selección del menú.
- **Simulate Device:** Esta opción coloca el dispositivo en modo de simulación. En este modo, el controlador no intentará comunicarse con el dispositivo físico, pero el servidor continuará enviando los datos OPC válidos. En el modo de simulación, el servidor tratará todos los datos del dispositivo como reflexivo: todo lo que se escribe en el dispositivo simulado se vuelve a leer.

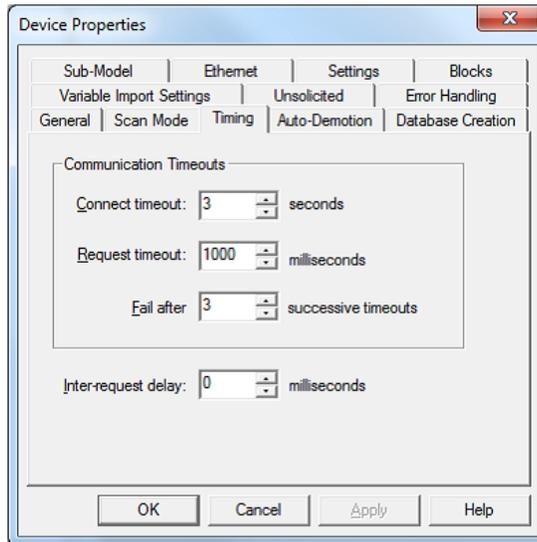
## 2.2. Device Properties – Scan Mode



**Figura III.71.** Propiedades de Dispositivo – Pestaña Scan Mode.  
*Fuente: Manual KEPServerEx V5 Help*

- **Respect client specified scan rate:** Utiliza la velocidad de escaneo solicitada por el cliente.
- **Request data no faster than x:** Especifica la velocidad de escaneo que será usada dentro del rango de 10 a 99999990 milisegundos. Por defecto 1000ms.
- **Request all data at x:** Este modo hace que todos los tags sean escaneados a un rango específico. El rango es de 10 a 99999990 ms. Por defecto 1000ms

### 2.3. Device Properties – Timing



**Figura III.72.** Propiedades de Dispositivo – Pestaña Timing.  
*Fuente: Manual KEPServerEx V5 Help*

- **Connect timeout:** Utilizado principalmente en controladores basados en Ethernet, controla el tiempo necesario para establecer una conexión de socket a dispositivo remoto. El rango válido es de 1 a 30 segundos.
- **Request timeout:** Especifica el intervalo usado por todos los controladores para determinar cuánto tiempo esperan una respuesta del dispositivo de destino completa. Tiene un rango de 50 a 99999 ms.
- **Fail After:** Especifica cuantas veces el controlador volverá a intentar la petición de comunicaciones antes de considerar que la petición ha fallado y el dispositivo está en error. El rango es de 1 a 10.
- **Inter-Request Delay:** Especifica el tiempo que el controlador esperará antes de enviar la siguiente petición al dispositivo de destino. Este retraso puede ser útil

cuando se trata de dispositivos con tiempos de respuesta lentos y en los casos en que la carga de la red es una preocupación.

## 2.4. Device Properties – Auto-Demotion

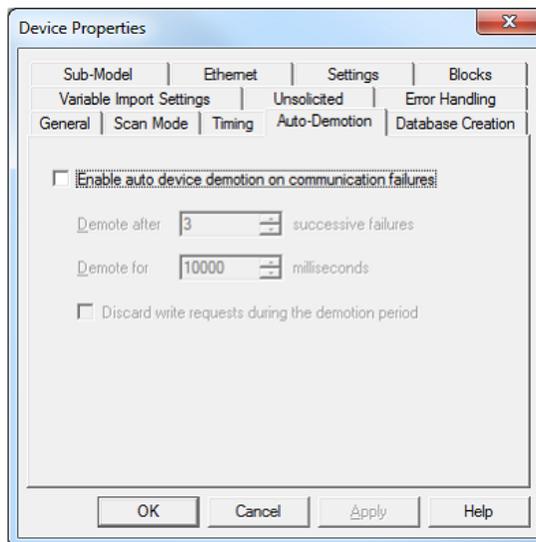
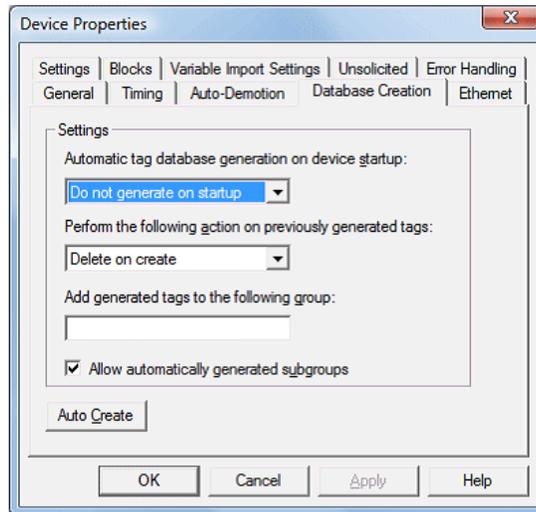


Figura III.73. Propiedades de Dispositivo – Pestaña Auto-Demotion.

Fuente: Manual KEPServerEx V5 Help

- **Enable auto device demotion on communication failures:** cuando se selecciona se active el auto-demotion.
- **Demote after \_\_\_\_ successive failures:** Indica cuantos ciclos sucesivos de tiempos de espera y reintentos se harán antes que el dispositivo quede fuera de línea. Rango de 1 a 30, predeterminado 3.
- **Demote for \_\_\_\_ milliseconds:** Indica cuánto tiempo se debe colocar el dispositivo fuera de línea cuando se ha alcanzado el parámetro anterior. Rango valido de 100 to 3600000 ms, predeterminado 10000.
- **Discard write requests during the demotion period:** Controla si hay o no solicitudes de escritura durante el periodo fuera de línea.

## 2.5. Device Properties – Database Creation



**Figura III.74.** Propiedades de Dispositivo – Pestaña Database Creation.  
*Fuente: Manual KEPServerEx V5 Help*

- **Automatic Tag Database Generation on Device Startup:** especifica cuando se generan automáticamente tags OPC. Cuenta con las siguientes opciones:
  - Do not generate on startup
  - Always generate on startup
  - Generate on first startup
- **Perform the Following Action:** Controla la forma que el servidor se encarga de los tags OPC que se han generado de forma automática y existen actualmente en el proyecto. También evita que los tags generados automáticamente que se acumulen en el servidor. Tiene las siguientes opciones:
  - Delete on create
  - Overwrite as necessary
  - Do not overwrite.
  - Do not overwrite, log error (18).

## 3.9. BASE DE DATOS

### 3.9.1. WampServer



**Figura III.75.** WampServer

Fuente: <http://www.wampserver.es>

Wampserver es un entorno de desarrollo web para Windows que instala en un solo paso todo lo necesario para ejecutar aplicaciones web creadas en PHP sobre un servidor Apache y con base de datos MySQL. El programa permite ahorrarse los pasos de configuración de PHP a menudo complejos y ofrece un panel de control desde donde se pueden realizar muchas tareas de administración y mantenimiento del servidor web.

WAMP es el acrónimo usado para describir un sistema de infraestructura de internet que usa las siguientes herramientas:

- Windows, como sistema operativo;
- Apache, como servidor web;

- MySQL, como gestor de bases de datos;
- PHP (generalmente), Perl, o Python, como lenguajes de programación.

El uso de un WAMP permite servir páginas html a internet, además de poder gestionar datos en ellas, al mismo tiempo un WAMP, proporciona lenguajes de programación para desarrollar aplicaciones web.

#### ***3.9.1.1. Características de WampServer:***

- Pre visualizar sitios web localmente
- Instalar un servidor web en Windows
- Administrar configuraciones de servidores Apache
- Crear aplicaciones web
- Gestionar bases de datos MySQL
- Utilizar lenguaje PHP
- Ejecutar archivos .php localmente
- Realizar pruebas con sitios PHP antes de subirlos a Internet

#### ***3.9.1.2. Utilidad***

WampServer permite servir paginas HTML, además de poder gestionar datos en ellas, también proporciona lenguajes de programación para desarrollar aplicaciones web.

#### ***3.9.1.3. Funcionalidades***

WampServer tiene funcionalidades que lo hacen muy completo y fácil de usar. Con un click izquierdo sobre el icono de WampServer, se podrá observar:

- Gestionar los servicios de apache y MySQL.
- Cambiar de línea/fuera de línea (dar acceso a todos o solo local)
- Instalar y cambiar de Apache, MySQL, y PHP.
- Gestión de la configuración de sus servidores.
- Acceder a sus registros.
- Acceder a sus archivos de configuración.
- Crear alias.

Mientras que con un click derecho:

- Cambiar el idioma del menú.
- Acceder a la página principal. (19)

### 3.9.2. PHP MyAdmin



**Figura III.76.** phpMyAdmin.

*Fuente: <http://codigoprogramacion.com/>*

phpMyAdmin es una herramienta escrita en PHP con la intención de manejar la administración de MySQL a través de páginas web, utilizando Internet. Actualmente puede crear y eliminar Bases de Datos, crear, eliminar y alterar tablas, borrar, editar y añadir campos, ejecutar cualquier sentencia SQL, administrar claves en campos, administrar privilegios, exportar datos en varios formatos y está disponible en 62 idiomas. Se encuentra disponible bajo la licencia GPL.

### ***3.9.2.1. Características***

- Interface sobre web intuitiva.
- Proporciona herramientas de gestión de la base de datos:  
Edición, creación, modificación y eliminación de bases de datos, tablas, vistas, campos, relaciones e índices.
- Mantenimiento de usuarios y sus privilegios.
- Mantenimiento de procedimientos almacenados.
- Importación de datos desde CSV y SQL.
- Exportación a varios formatos: CSV,SQL, XML, PDF, SO/IEC 26300 - OpenDocument Text y Spreadsheet, Word, LATEX y otros.
- Administración de múltiples servidores.
- Creación del despliegue de la base de datos en un gráfico exportado a PDF.
- Creación de consultas complejas haciendo uso QBE (Query By Example).

### ***3.9.2.2. phpMyAdmin en Wampserver***

Una vez instalado WAMP, en el menú que aparece se encuentra la herramienta PhpMyAdmin.

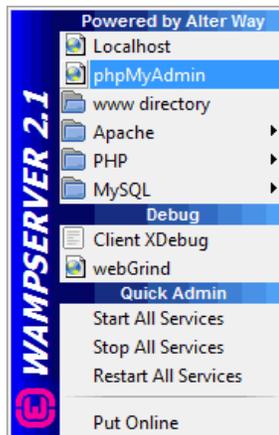


Figura III.77. Acceso a phpMyAdmin por el menu de WampServer.  
Fuente: <http://codigoprogramacion.com/>

Otra forma de acceder a phpMyAdmin, es ingresando al LocalHost de WampServer, en la sección Tools.

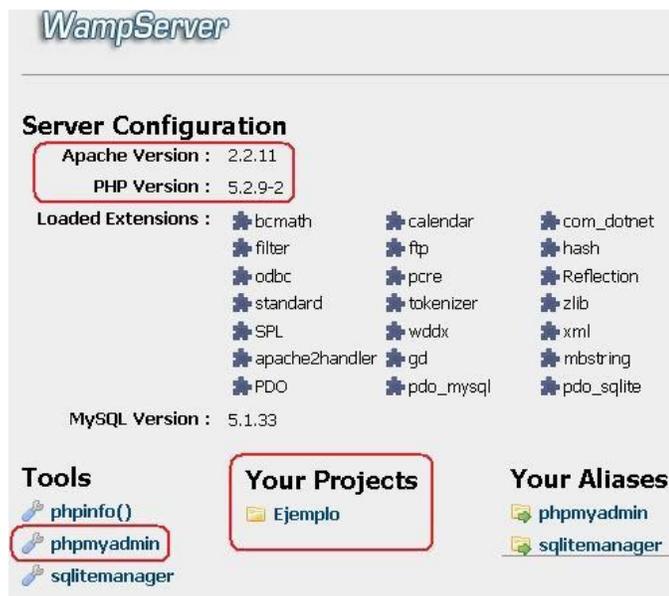


Figura III.78. Acceso a phpMyAdmin por el LocalHost.  
Fuente: <https://sites.google.com/>

Una vez ingresado se mostrara la siguiente ventana, perteneciente a phpMyAdmin.



Figura III.79. Intefaz phpMyAdmin  
Fuente: <https://sites.google.com/>

## 3.10. LABVIEW

### 3.10.1. I/O SERVER DE LABVIEW

Un I/O Server es un Plug-in de variable compartida (SVE) que permite la comunicación con los dispositivos y aplicaciones que no utilizan el protocolo de publicación-suscripción (NI-PSP) de la National Instruments NI, utilizado por variables compartidas. Servidores de E/S son los elementos de puente entre las variables compartidas en un VI de Labview y las etiquetas de datos de OPC, Modbus o EPICS. Con el Registro de Datos de Labview y el Módulo de Control y Supervisión (DSC), se puede crear en Labview I/O Server para comunicar con OPC, Modbus, y clientes EPICS.

#### 3.10.1.1. Clientes de I/O Server

##### Clientes OPC

El módulo DSC de Labview ofrece I/O Server Clientes OPC, para comunicarse con cualquier servidor de la aplicación del protocolo de la Fundación OPC-DA OPC, que es un estándar basado en COM Microsoft. Un I/O Server Clientes OPC muestra todos los OPC server instalados en el equipo y hace grupos y elementos accesibles en el

servidor. Puede crear un I/O Server Clientes OPC para acceder a los datos de un servidor OPC local o remoto.

El I/O Server Clientes OPC permite unir las etiquetas del OPC de un servidor OPC a variables compartidas. Estas variables compartidas unidas proporcionan una manera fácil para que LabVIEW pueda leer y escribir datos en las etiquetas del OPC.

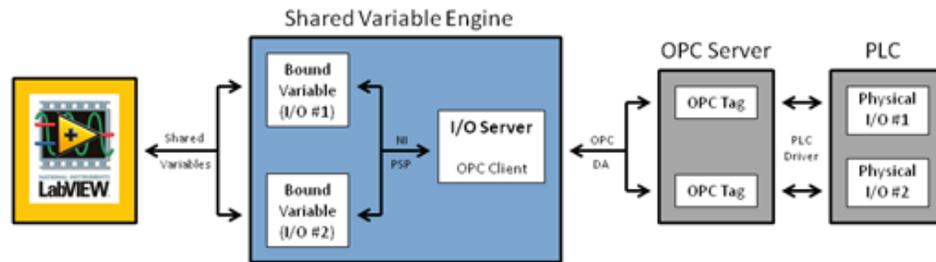


Figura III.80. Labview y red OPC con DSC.

Fuente: <http://www.ni.com/>

## Modbus

Modbus es un protocolo de mensajería de nivel de aplicación que proporciona una comunicación maestro/esclavo entre los dispositivos conectados a diferentes tipos de buses o redes. Se puede crear un I/O server Modbus o Esclavo Modbus para leer o escribir datos a los dispositivos Modbus. El DSC de LabVIEW y el Módulo Real-Time de LabVIEW proporciona al I/O server Modbus una conexión con la variable compartida. Los I/O server manejan el protocolo de bajo nivel necesario para el establecimiento de conexiones y la transmisión de datos entre un maestro y el esclavo. La variable compartida une los datos de los I/O server a variables compartidas que LabVIEW

puede fácilmente leer y escribir. No se necesita ningún software adicional para conectar LabVIEW a un dispositivo Modbus (21).

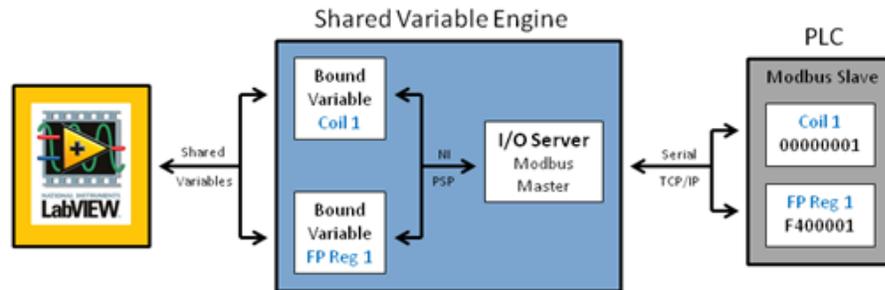


Figura III.81. Labview y red Modbus.

Fuente: <http://www.ni.com/>

### 3.10.2. Base de Datos en Labview

Para hacer posible el uso de una base de datos en Labview se debe contar con el Labview Database Connectivity Toolkit. En su nivel más básico, este toolkit utiliza instrucciones SQL para acceder, modificar y ver información en bases de datos.

Microsoft Data Access Components (MDAC) es una estructura que permite conectarse y comunicarse con la base de datos. MDAC incluye los siguientes componentes:

- ODBC (Open Database Connectivity)
- OLE DB
- ADO (ActiveX Data Objects)

#### 3.10.2.1. ODBC

Open DataBase Connectivity (ODBC) es una interfaz nativa que se accede a través de un lenguaje de programación que puede hacer llamadas en una librería nativa. En

MDAC esta interfaz se define como una DLL. Se necesita un módulo independiente o controlador para cada base de datos que se debe acceder.

### ***3.10.2.2. OLE DB***

OLE DB se compone de un conjunto de modelos de objetos componentes de Microsoft (COM), interfaces que soportan varios DBMS (Sistema de Gestión de Base de Datos). Por ejemplo, OLE DB permite la interacción con el DBMS tradicional como otros sistemas de almacenamiento de datos, como Microsoft Excel y Microsoft Access. OLE DB utiliza un proveedor (controlador) para hablar con los diferentes DBMS.

Con OLE DB se puede comunicar con cualquier DBMS que proporciona un controlador ODBC o proveedor OLE DB. OLE DB utiliza el proveedor OLE DB para ODBC como una capa de conversión entre OLE DB y ODBC si un controlador ODBC se utiliza para comunicarse con una base de datos.

Microsoft ha utilizado esta tecnología para separar la aplicación desde el almacén de datos que se necesita para acceder. Esto se hizo debido a las diferentes aplicaciones necesitan acceder a los diferentes tipos y fuentes de datos, y no necesariamente tienen que saber cómo acceder a la funcionalidad de la tecnología específica. La tecnología se divide conceptualmente en los consumidores y los proveedores. Los consumidores son las aplicaciones.

### **3.10.2.3. ADO (ActiveX Data Objects)**

Es una interfaz de programación de alto nivel para OLE DB. Se utiliza un modelo jerárquico de objetos que permite crear aplicaciones mediante programación, recuperar, actualizar y eliminar datos de fuentes compatibles con OLE DB. ADO consta de una serie de objetos y colecciones basados en COM jerárquicas, un objeto que actúa como un contenedor de muchos otros objetos. Un programador puede acceder directamente a los objetos ADO para manipular los datos, o puede enviar una consulta SQL a la base de datos a través de varios mecanismos de ADO (22).

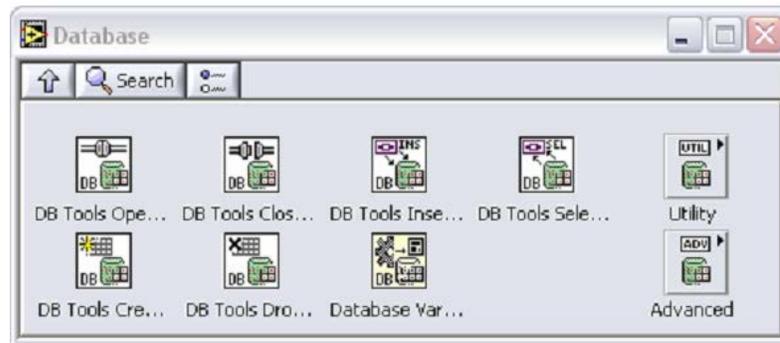
### **3.10.3. Database Connectivity Toolkit.**

Esta herramienta permite conectar rápidamente bases de datos locales o remotos, además de realizar muchas operaciones comunes de bases de datos sin necesidad de conocer el lenguaje de programación estructurado de consultas (SQL).

#### **Paletas del Database Connectivity Toolkit**

##### **Paleta General de Database Connectivity Toolkit**

En esta paleta existe VIs para abrir y cerrar las conexiones a la base de datos, insertar y eliminar tablas, insertar y seleccionar registros, y convertir variantes. Todos los datos almacenados en una base de datos se transmiten utilizando el tipo de datos Variant en LabVIEW.

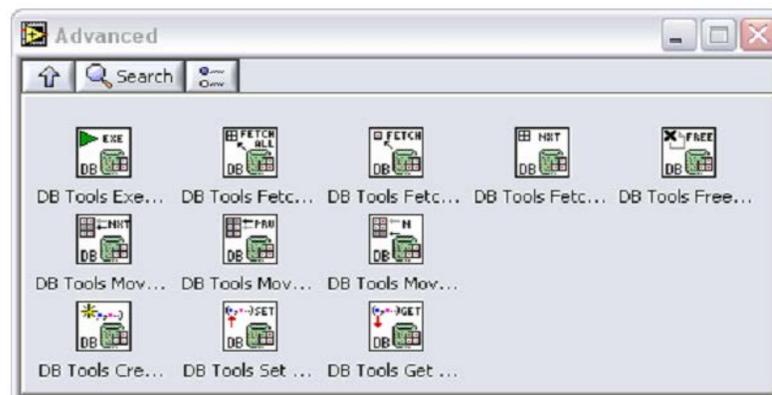


**Figura III.82.** Paleta General de Operaciones de Base de Datos.

*Fuente: Manual Labview Database Connectivity Toolkit.*

Esta paleta general cuenta con dos sub-paletas, como son:

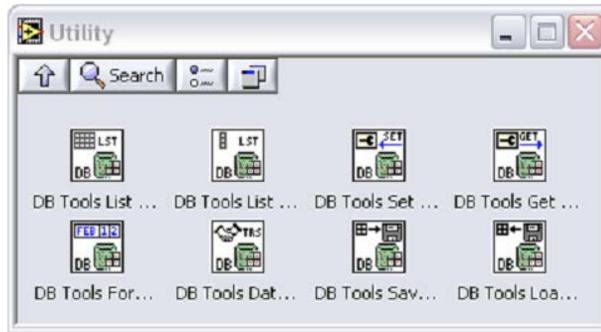
**Advanced VIs**, proporciona mayor flexibilidad en la búsqueda, modificación y visualización de datos de una base de datos, también ejecutan consultas SQL, selección y navegación a través de registros en la base de datos.



**Figura III.83.** Paleta Avanzada de Base de Datos.

*Fuente: Manual Labview Database Connectivity Toolkit.*

**Utility VIs**, se utilizan para obtener y establecer información general sobre la conexión de base de datos. En esta paleta se puede ver los nombres de las tablas de una base de datos y los campos (columnas) en una tabla, ver y configurar diferentes parámetros de conexión de la base de datos como la cadena de conexión y tiempo de espera de conexión.

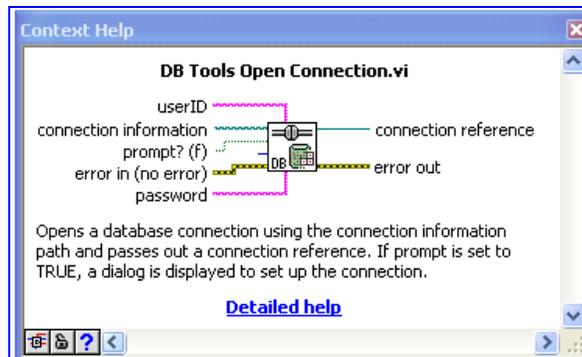


**Figura III.84.** Paleta Avanzada de Base de Datos.  
*Fuente: Manual Labview Database Connectivity Toolkit.*

A continuación se muestran algunos bloques útiles en la programación.

### DB Tools Open Connection

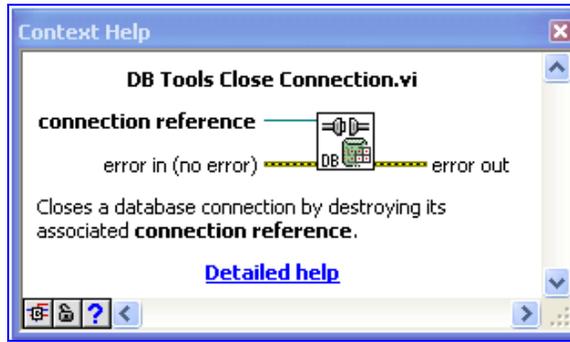
Abre una conexión de base de datos utilizando la información de conexión.



**Figura III.85.** Icono DB Tools Open Connection.  
*Fuente: Manual Labview Database Connectivity Toolkit.*

### DB Tools Close Connection

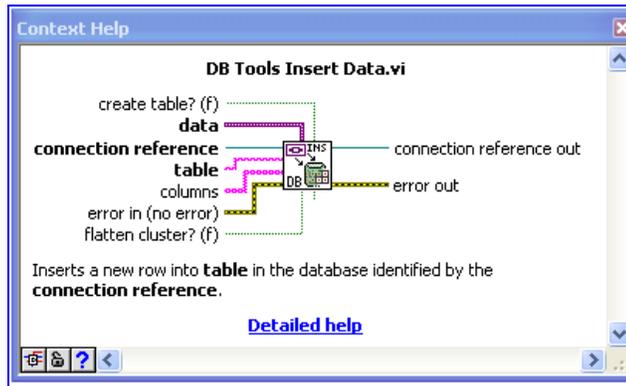
Cierra una conexión de base de datos mediante la destrucción de sus asociados respecto de referencia.



**Figura III.86.** Icono DB Tools Close Connection.  
*Fuente: Manual Labview Database Connectivity Toolkit.*

### DB Tools Insert Data

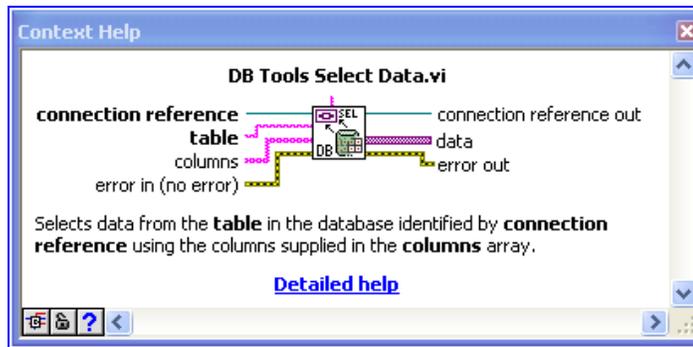
Inserta una nueva fila en la tabla de la base de datos identificada por la conexión de referencia.



**Figura III.87.** Icono DB Tools Insert Data.  
*Fuente: Manual Labview Database Connectivity Toolkit.*

### DB Tools Select Data

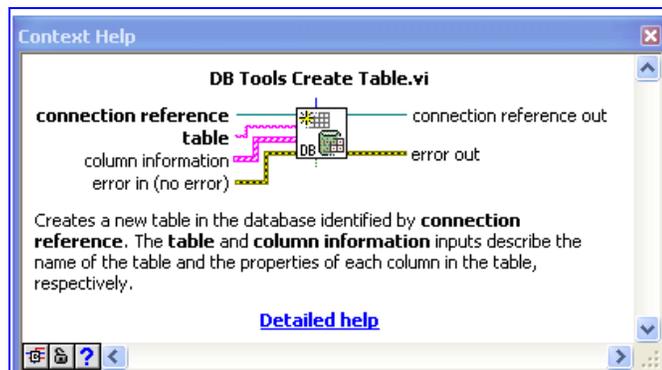
Selecciona los datos de la tabla en la base de datos definidas por el marco de referencia utilizando las columnas ofrecidas en las columnas del arreglo.



**Figura III.88.** Icono DB Tools Select Data.  
*Fuente: Manual Labview Database Connectivity Toolkit.*

### DB Tools Create Table

Crea una nueva tabla en la base de datos definidas por el marco de referencia. La tabla y la columna de información insumos describir el nombre de la tabla y las propiedades de cada columna en la tabla, respectivamente.



**Figura III.89.** Icono DB Tools Create Table.  
*Fuente: Manual Labview Database Connectivity Toolkit.*

### DB Tools Execute Query

Ejecuta una consulta SQL y muestra un conjunto de registros de referencia que deben ser liberados.

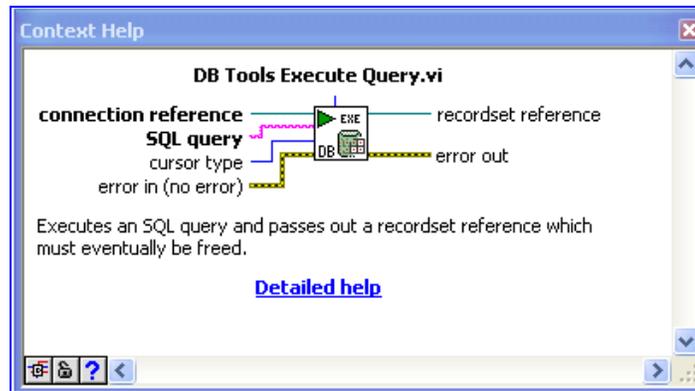


Figura III.90. Icono DB Tools Execute Query.

Fuente: Manual Labview Database Connectivity Toolkit.

### DB Tools Fetch Recordset Data

Obtiene los datos de los registros identificados por referencia. Los datos se devuelven como una matriz 2-D de variantes. Cada elemento de la matriz se puede convertir a su nativa LabVIEW utilizando el tipo de base de datos de la variante A de Datos función.

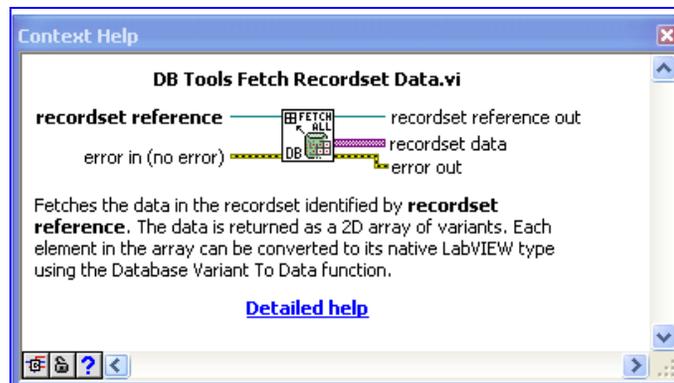


Figura III.91. Icono DB Tools Fetch Recordset Data.

Fuente: Manual Labview Database Connectivity Toolkit.

### DB Tools Fetch Element Data

Obtiene los datos ubicados en la columna índice del registro actual en el conjunto de registros identificados por los registros de referencia. La columna de índice puede ser cero el índice de posición de la columna en el conjunto de registros o el nombre de la

columna. El tipo de entrada es polimórfico y determina el tipo de datos que se devuelve.

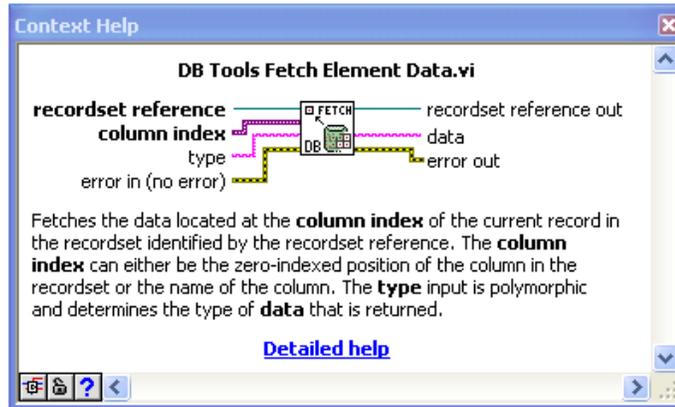


Figura III.92. Icono DB Tools Fetch Element Data.

Fuente: Manual Labview Database Connectivity Toolkit.

### DB Tools Free Object

Libera un objeto mediante la destrucción de sus asociados y de referencia pasa a cabo una referencia diferente.

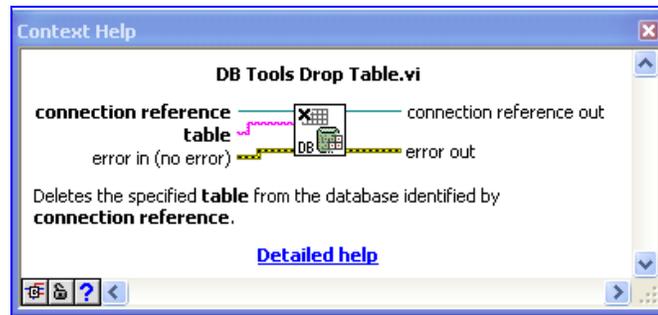


Figura III.93. Icono DB Tools Free Object.

Fuente: Manual Labview Database Connectivity Toolkit.

### DB Tools Drop Table

Borra la tabla especificada en la base de datos definidas por el marco de referencia (23).



**Figura III.94.** Icono DB Tools Drop Table.

*Fuente: Manual Labview Database Connectivity Toolkit.*

## 3.11. PROCESOS MODULARES

### 3.11.1. MESA DE INDEXACION



**Figura III.95.** Módulo Mesa de Indexación.

*Fuente: <http://www.festo-didactic.com/>*

Una mesa giratoria de indexación puede rotar, o indexar, en pequeños incrementos para torneear piezas complejas.

En este caso la mesa de indexación cuenta con las siguientes secciones:

*Mesa giratoria:* cuyo movimiento se realiza gracias a un motorreductor de corriente continua. Dispone de 6 posiciones definidas por unos tornillos de posicionado que se detectan con la ayuda de un sensor inductivo situado en la parte inferior de la mesa. Cada posición contiene un retenedor semicircular provisto de un taladro central para facilitar la detección de la pieza mediante un sensor de proximidad capacitivo.

*Un módulo de verificación:* que se encarga de reafirmar que la pieza que se encuentra en la posición deseada: Si el agujero mira hacia arriba, el émbolo del electroimán alcanza su posición final, la cual se detecta mediante un sensor de proximidad inductivo, reportando que la pieza se encuentra en la posición correcta.

*Un módulo de taladrado:* se utiliza para pulir la pieza y consta de un motor de velocidad no ajustable y que funciona a 24 V DC. Los movimientos de avance y retroceso se llevan a cabo por medio de un eje lineal con correa dentada. Un motorreductor se encarga de accionar el eje lineal y un circuito de relés para poner en marcha el motor. La detección de la posición final se realiza por medio de finales de carrera mecánicos, que cuando se activan provocan la inversión del sentido de movimiento del eje lineal.

*Una mordaza eléctrica,* que retiene la pieza para que no se gire durante el pulido de la pieza. (24)

### 3.11.2. ENVASADORA



**Figura III.96.** Módulo Envasadora.

*Fuente: Leonardo Asqui e Iván Lema (Los autores)*

Una envasadora es un sistema coordinado de preparación de productos para el transporte, la distribución, el almacenaje, la venta al detalle y uso final. (25)

En este caso el módulo de envasadora cuenta con dos etapas principales:

*Etapas de llenado:* en esta etapa tenemos el tornillo sin fin que se encarga de movilizar las latas que detecte el sensor, hasta un punto determinado donde, con la ayuda de una válvula, succiona el líquido y lo coloca en la lata, para posteriormente seguir su camino con la ayuda del tornillo sin fin.

*Etapas de sellado:* después de haber llenado de líquido las latas, el tornillo sin fin lleva la lata hasta esta etapa, en donde se tiene la tapa detenida por medio de un cilindro,

la cual se libera cuando la lata llega al lugar adecuado, y por medio de otro cilindro, se procede a tapar la lata. Una vez realizado esto, la lata ya sellada sigue su camino hasta el final.

Se debe tener en cuenta que ambas etapas pueden realizarse simultáneamente.

### 3.11.3. SEMÁFORO



**Figura III.97.** Semáforo.

*Fuente: Leonardo Asqui e Iván Lema (Los autores)*

Los semáforos son dispositivos de señales que se sitúan en intersecciones viales, pasos de peatones y otros lugares para regular el tráfico y el tránsito de peatones.

Los semáforos tienen un sistema que les permite presentar una secuencia de fases en un período de tiempo llamado ciclo. El ciclo está compuesto por la sumatoria de los tiempos de verde, amarillo y rojo.

## Componentes

El semáforo está formado por los siguientes componentes:

- **Cabeza:** Es la armadura que contiene las partes visibles del semáforo. Cada cabeza contiene un número determinado de caras orientadas en diferentes direcciones.
- **Cara:** Son las distintas luces de las cuales están formados los semáforos. En cada cara puede haber desde dos luces hasta más de tres, siendo la de tres luces las caras más usuales.
- **Lente:** Es la parte de la unidad óptica que por refracción dirige la luz proveniente de la lámpara y de su reflector en la dirección deseada. Este elemento desaparece en los nuevos semáforos de LEDs.

## Funcionamiento

El tipo más frecuente tiene tres luces de colores:

- *Rojo*, para detenerse inmediatamente. En algunos países, si el rojo está parpadeando, actúa como una señal de Alto/Pare/Stop.
- *Verde*, para avanzar, puesto que no hay obstáculos.
- *Amarillo*, para avanzar con cuidado, puesto que va a cambiar a rojo (26)

### 3.11.4. CASA AUTOMATIZADA



**Figura III.98.** Casa Automatizada

*Fuente: Leonardo Asqui e Iván Lema (Los autores)*

Una casa automatizada simultáneamente usa la electricidad, la electrónica y la informática, para crear un diseño arquitectónico propio, de tal manera que las personas que la habitan disfruten de mayores comodidades.

Las casas automatizadas además ahorran energía. Trabajan con un cerebro que es el que manda las instrucciones a cada área para que realice según la programación que se le haya asignado. Se programa para que sea capaz de llenar las necesidades y satisfacer una amplia gama de placeres personales: aumentar o disminuir la intensidad de la luz, calentar, enfriar, difundir la música, abrir y cerrar las persianas, entre otros.

Los sistemas inteligentes mantienen un diálogo a distancia, mediante el computador, el teléfono fijo, la PDA o el teléfono celular. (27)

### **NOTAS CAPÍTULO III**

- (1) ETHERNET TCP/IP Programable Fieldbus Controller. Pág. 38
- (2) ETHERNET TCP/IP Programable Fieldbus Controller. Pag 71, 72.
- (3) ETHERNET TCP/IP Programable Fieldbus Controller. Pág. 123.
- (4) ETHERNET TCP/IP Programable Fieldbus Controller. Pág. 131
- (5) ETHERNET TCP/IP Programable Fieldbus Controller. Pág. 181
- (6) Modular I/O System Modbus. Pág. 32, 58.
- (7) [http://www.ni.com/pdf/products/us/4dio541\\_543.pdf](http://www.ni.com/pdf/products/us/4dio541_543.pdf)
- (8) Manual Twido Controladores Programables, Guía de Hardware. Pág. 26-28.
- (9) Manual Twido Modbus Addressing. Pág. 114.
- (10) <http://www.ieee.org.ar/downloads/Romero-Eth-Ind.pdf>
- (11) <http://www.blogelectronica.com/los-gateways-serie-ethernet/>
- (12) Manual ConeXium TwidoPort. Pág 2.
- (13) Manual Controlador Programable S7-1200. Pág 241.
- (14) Manual Controlador Programable S7-1200. Pág 178.
- (15) [http://www.weidmuller.ru/news/pi\\_ie\\_en.pdf](http://www.weidmuller.ru/news/pi_ie_en.pdf)
- (16) <http://www.hyperline.com/catalog/cable/if4-c5e-s-io.shtml>
- (17) <http://www.conelectronica.com/Ethernet-Industrial/Ethernet-La-conexi%C3%B3n-%C3%B3ptima.html>
- (18) MANUAL KEPSERVER Accessing the administration menú, options-general. Pág 19, 59-65.
- (19) <http://www.slideshare.net/aimerodriguezrodriguez/que-es-wamp-server>
- (20) <https://sites.google.com/site/modelamientodebasesdedatos/wampserver>

- (21) <http://www.ni.com/white-paper/13865/en/>
- (22) Database Communication in LabVIEW. Pág. 9-12.
- (23) LabVIEW Database Connectivity Toolkit.
- (24) [http://e-archivo.uc3m.es/bitstream/10016/6839/1/PFC\\_Ruth\\_Chamorro\\_Casado.pdf](http://e-archivo.uc3m.es/bitstream/10016/6839/1/PFC_Ruth_Chamorro_Casado.pdf)
- (25) [http://es.wikipedia.org/wiki/Maquinaria\\_de\\_envasado](http://es.wikipedia.org/wiki/Maquinaria_de_envasado)
- (26) <http://es.wikipedia.org/wiki/Sem%C3%A1foro>
- (27) <http://www.revistasucasa.com/contenido/articles/159/1/Casas-inteligentes-Tecnologia-de-punta/Paacuteginas1.html>

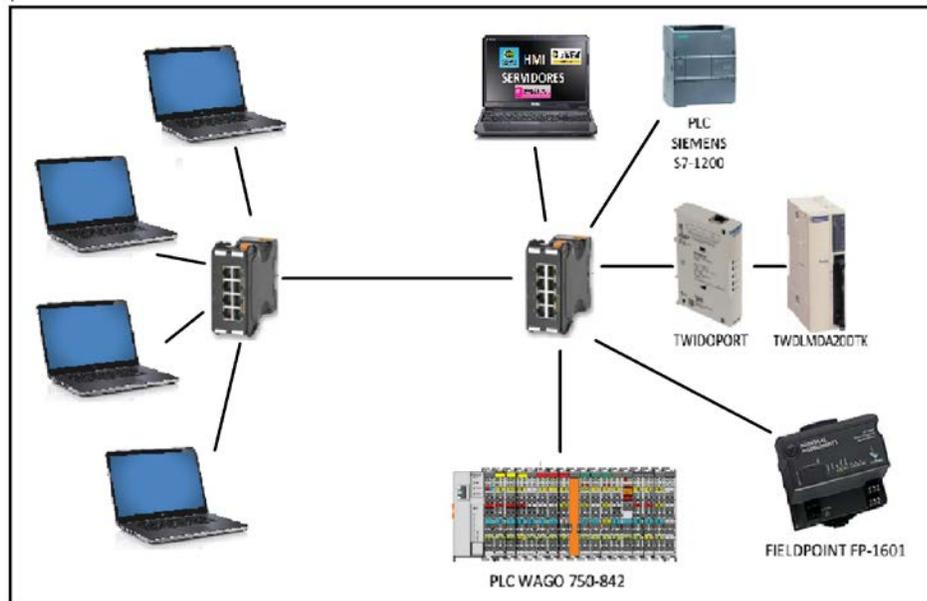
## **CAPÍTULO IV**

### **IMPLEMENTACIÓN**

#### **4.1. DISEÑO DE LA RED**

Para el diseño de red se debe tomar en cuenta los aspectos que se han visto anteriormente en cuanto a una Red Ethernet Industrial, como la redundancia, escalabilidad, etc.

Para diseñar la red se debe tomar en cuenta que se dispone de varios controladores programables de diferentes fabricantes, los cuales se conectaran por medio de la Red Ethernet, además será monitoreado y controlado por medio de un HMI con la ayuda de un OPC, que contendrá tags para cada controlador, y un OBDC, que hace posible la conexión con la Base de datos.



**Figura IV.99.** Topología de la Red Didáctica Ethernet  
*Fuente: Leonardo Asqui e Iván Lema (Los Autores)*

#### 4.1.1. Requerimientos

Se desea incorporar en la red:

- 4 controladores programables los cuales controlaran cada uno un proceso modular.
- Una computadora en el cual van correr los servidores y HMI.
- Por lo menos 4 computadoras de programadores.

Todos los dispositivos estarán en la misma red y podrán comunicarse unos con otros sin dificultad, se debe de dividir el dominio de colisiones para evitar perdida de información y demora en la transmisión de la información, se manejan direcciones estáticas, no es de suma importancia tener acceso a internet en la red LAN, Por ser una red didáctica la seguridad no es vital ya que los usuarios de esta red tendrán el

conocimiento o la orientación de personal con suficiente conocimiento para manejar los equipos pertenecientes a la red.

#### **4.1.2. Escalabilidad**

Nuestra red LAN consta de 2 partes, la una de programadores y la otra de controladores programables con su HMI, por lo que es de suma importancia la escalabilidad por la necesidad de incorporar a futuro nuevos módulos, radio enlace, wireless a la sección de controladores programables, e incorporar dispositivos terminales a la red de programadores como impresoras entre otras, es por esto que se ha dispuesto usar 2 switch uno para los controladores programables y otro para los programadores.

#### **4.1.3. Redundancia**

Nuestra red LAN no consta de enlaces redundantes, ya que esto implica controladores programables con 2 puertos Ethernet, incrementándose significativamente el costo del proyecto, el único enlace redundante es la unión de los switch de programadores y switch de controladores.

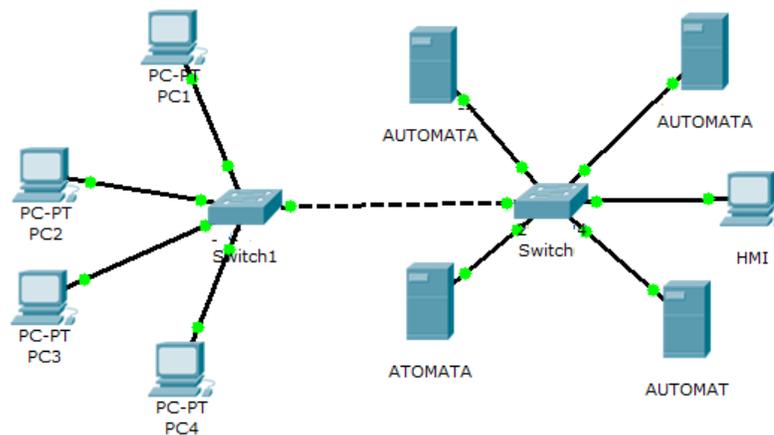
#### **4.1.4. Rendimiento**

El rendimiento de nuestra red es muy bueno ya que no existen dispositivos intermediarios que provoquen cuellos de botella como repetidores o hubs, y por las ventajas que el switch nos brinda conmutando cada enlace como un punto a punto, obteniendo de esta forma una LAN conmutada y no la tradicional CSMA/CD, es por esto que la transmisión se efectúa prácticamente a velocidad del cable.

#### 4.1.5. Disponibilidad.

Por la importancia de la transmisión de mensajes de control, ningún dispositivo (controlador) debe tener retardos en la transmisión que es lo que ocurre en las redes LAN CSMA/CD, que el dispositivo a enviar un mensajes deben escuchar el medio antes de transmitir, es por esto que nuestra red debe tener una disponibilidad alta descartando este tipo de topología, la transmisión de nuestra red debe ser necesariamente full dúplex (controlar y monitorear), esto lo logramos con la utilización del switch industrial, el cual divide el dominio de colisiones de nuestra red, permitiendo que la transmisión de los dispositivos sea full dúplex, y sin la necesidad de escuchar el medio antes de transmitir, consiguiendo con esto que cualquier dispositivo pueda hacer uso de la red en cualquier momento.

#### 4.1.6. Arquitectura.



**Figura IV.100.** Disposición de la Red Didáctica Ethernet  
*Fuente: Leonardo Asqui e Iván Lema (Los Autores)*

#### **4.1.6.1. TOPOLOGÍA**

La topología de la red está diseñada de acuerdo a las necesidades de los dispositivos que serán conectados, por lo cual se establece una topología estrella física, tomando como base el Switch no gestionado IE-SW-WAVE puesto que es compatible con el estándar IEEE 802.3, empleando una velocidad de conexión de 100Mbps, lo cual facilita el trabajo de la red Ethernet implementada.

Para configurar la velocidad de transmisión de datos se debe tener en cuenta que una velocidad de datos más alta proporcionará un ancho de red más amplio, pero también hay que recordar que a mayor velocidad de transmisión existe menor inmunidad al ruido lo que provocaría bits errados en la transmisión y mayor cantidad de tráfico.

##### ***4.1.6.1.1. Topología Física***

La topología física define el diseño del cableado para la red implementada, especifica como los elementos de la red son conectados eléctricamente unos con otros sin tomar en cuenta el tipo de dispositivo, los métodos de conectividad o las direcciones de dicha red. Como se vio anteriormente, la topología a implementarse es Estrella.

En este tipo de topología es recomendable establecer un diseño previo de la ubicación posible de los controladores que ejecutaran cada proceso, tomando en cuenta que el concentrador es el switch, y los equipos adyacentes los controladores y la Pc. Como medio de transmisión a implementarse tenemos el cable 24 AWG, UTP de cuatro pares.

#### 4.1.6.1.2. Topología Lógica

En la topología lógica se determina como están comunicados los elementos de la red y como se transmiten. En el caso de las redes Ethernet en la topología lógica de nuestra LAN se utiliza Broadcast, al inicio de la convergencia de la red ya que el switch en un primer instante tiene que crear su tabla MAC, una vez creado todos los enlaces el switch sabrá que dispositivo se encuentra conectado a que puerto, y podrá enviar sus mensajes de acuerdo a su tabla MAC. Por la característica de control y monitoreo los mensajes serán del tipo unicast y multicasts, únicas desde el controlador hasta la computadora (servidores, HMI) y multicast desde la computadora hasta el controlador.

Las direcciones de nuestra red son del tipo estáticas, ya que se asignan direcciones IP a cada elemento activo utilizado, controlador, computador, etc. los cuales deben pertenecer a la misma red. Se debe tener en cuenta que no se utiliza un servidor DHCP puesto que el TwidoPort solo soporta direccionamiento estático.

Para asignar la dirección se debe tener presente el número de host que se van a asignar, para que el bloque de direcciones sea capaz de cumplir con la demanda actual y futura.

#### 4.1.7. Direccionamiento

EL direccionamiento se lo hizo en base a la dirección 192.50.1.0/24

<b>DIRECCION DE RED</b>	<b>192.50.1.0</b>
<b>MÁSCARA DE RED</b>	255.255.255.0
<b>DIRECCION DE BROADCAST</b>	192.50.1.255/24

*Fuente: Realizado por Leonardo Asqui e Iván Lema (Los autores)*

Se decidió asignar a la dirección de red de la siguiente manera:

**Tabla IV.XXI. Direcciones de la red**

<b>DISPOSITIVOS</b>	<b>DIRECCIONES</b>
<b>Controladores</b>	192.50.1.2/24 hasta 192.50.1.80/24
<b>Radio enlaces y Wireless:</b>	192.50.1.81/24 hasta 192.50.1.100/24
<b>HMI</b>	192.50.1.101/24 hasta 192.50.1.110/24
<b>Programadores</b>	192.50.1.111/24 hasta 192.50.1.254/24

*Fuente: Realizado por Leonardo Asqui e Iván Lema (Los autores)*

**Tabla IV.XXII. Direcciones de los Procesos**

<b>PROCESO</b>	<b>DIRECCIÓN</b>
<b>Mesa Indexadora</b>	192.50.1.2
<b>Casa Automática</b>	192.50.1.3
<b>Semáforo</b>	192.50.,1,4
<b>Envasadora de líquidos</b>	192.50.1.5
<b>HMI</b>	192.50.1.101

*Fuente: Realizado por Leonardo Asqui e Iván Lema (Los autores)*

## **4.2. DISEÑO E IMPLEMENTACION ESTRUCTURAL**

En esta sección se procede a describir la estructura de elementos implementados, en este caso del modular de PLCs, de los circuitos elaborados, del tipo de cable y de la estructura de la Casa Automática.

### **4.2.1. MODULAR DE PLC'S**

La base estructural del modular está conformada por perfiles de aluminio de cuatro canales, ubicados de forma horizontal, de manera que forman un panel rectangular en el cual están dispuestos 12 perfiles.



**Figura IV.101.** Modular de PLC's.

*Fuente: Leonardo Asqui e Iván Lema (Los autores)*

## **4.2.2. CIRCUITOS DISEÑADOS**

En esta sección se mostrara los circuitos utilizados en la implementación de la tesis, en este caso se los muestra en dos grupos, puesto que se ha diseñado un circuito en general para la conexión de borneros de los módulos de procesos utilizados, y los circuitos utilizados para la implementación de la Casa Automática.

Los circuitos han sido diseñados en el ARES Professional v7.9 S1.

### ***4.2.2.1. Circuitos de Borneros***

Teniendo en cuenta que este tema tiene un enfoque didáctico, se ha implementado una placa que facilite la conexión entre los módulos y los distintos PLCs que los controlan, para evitar la tarea de conectar cada una de las entradas y salidas de los módulos a cada puerto correspondiente en el PLC, se ha optado por diseñar una Plaqueta de Borneros, esta plaqueta consta de 2 secciones la una para la conexión de

sensores (entradas de PLCs) teniéndose en esta parte tres puntos de conexión por bornera

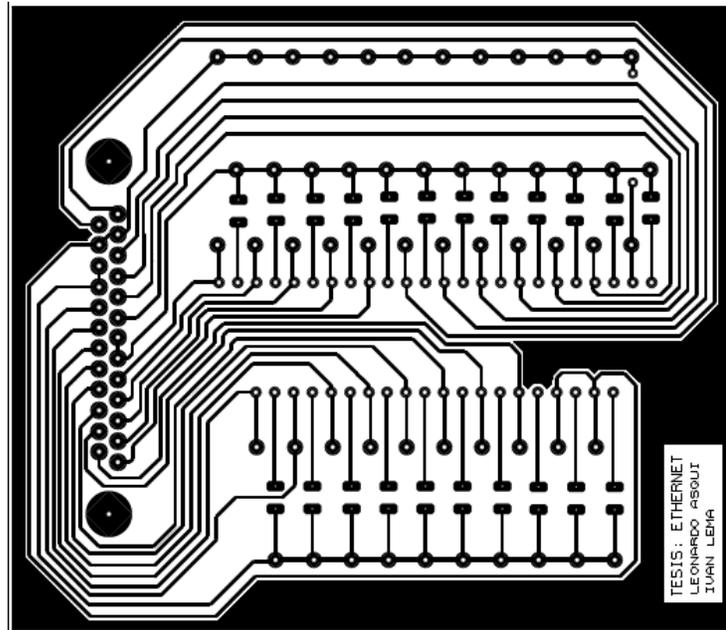
- +24v
- 0v
- Señal.

La otra sección consta de dos puntos de conexión por bornera, siendo esta sección para las salidas del PLC, el primer punto de conexión es un común y la otra es la salida de relé del PLC.

- Común.
- Salida relé.

Para indicar tanto las señales de los sensores, como la activación de las salidas de relé del PLC la plaqueta consta de leds indicadores.

La pista que se ha implementado para la plaqueta es la siguiente:

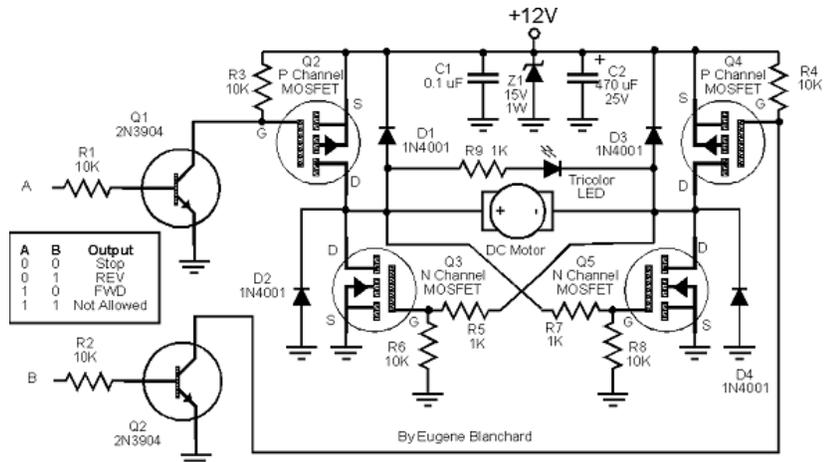


**Figura IV. 102.** Pista de la Plaqueta de Borneros  
*Fuente: Leonardo Asqui e Iván Lema (Los autores)*

#### **4.2.2.2. Circuitos de la Casa automatizada**

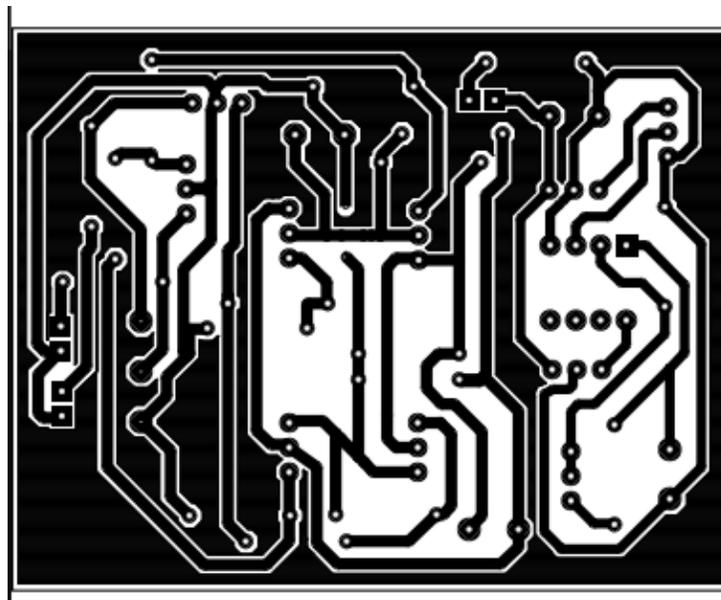
Dentro de los circuitos utilizados para la implementación de la casa automática, tenemos:

- **Puente H:** con la implementación de este circuito se tendrá la facilidad de cambiar el giro del motor que controla la puerta del garaje, puesto que debe girar en un sentido para cerrar la puerta y hacerlo inversamente para abrirla, para ello se ha utilizado transistores Mosfet, ya que mejora la eficiencia del Puente H. El Mosfet utilizado es el IRFZ44N.



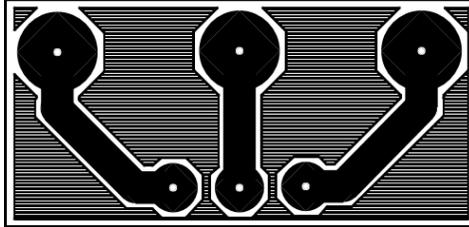
**Figura IV.103.** Circuito Base Puento H.  
*Fuente: Leonardo Asqui e Iván Lema (Los autores)*

Para una mayor facilidad a la hora de integrar los circuitos en la casa inteligente se ha optado por unir el circuito de temperatura y el puente H en una misma placa, como lo muestra la siguiente figura.



**Figura IV.104.** Placa de los circuitos Puento H y de Temperatura.  
*Fuente: Los Autores (Leonardo Asqui e Iván Lema)*

- Placa de Alimentación: Puesto que cada dispositivo requiere de alimentación, se ha diseñado esta placa con tres bananas hembra +24v, 0v y GND, de esta forma la alimentación de cada dispositivo será mucho más fácil



**Figura IV.105.** Placa de Alimentación.  
*Fuente: Los Autores (Leonardo Asqui e Iván Lema)*

### 4.2.3. CABLE DE CONEXIÓN ENTRE EL MÓDULO Y EL PLC

Puesto ya se cuenta con una placa de borneros, se implementa simultáneamente el cable de conexión entre la Placa de Borneros y el PLC. Se debe tomar en cuenta la disposición de las entradas y salidas dispuestas en las placas por un lado, y la disposición de entradas y salidas del PLC para poder tener una correcta conexión.

Para ello se ha utilizado dos cables de 12 hilos junto con un conector DB 25.

**Tabla IV.XXIII. Disposición del cable de conexión**

DB 25	
1	Entrada 4
2	Entrada 3
3	Entrada 2
4	Entrada 1
5	Entrada 0
6	-Vcc
7	-Vcc

8	Entrada 10
9	Entrada 9
10	Entrada 8
11	Entrada 7
12	+Vcc
13	Entrada 6
14	Salida 7
15	Salida 6
16	Salida 5
17	Salida 4
18	Salida 3
19	Salida 2
20	Salida 0
21	Salida 1
22	Común salidas
23	Común salidas
24	+Vcc
25	Entrada 5

*Fuente: Realizado por Leonardo Asqui e Iván Lema (Los autores)*



**Figura IV.106.** Cable de conexión.

*Fuente: Los Autores (Leonardo Asqui e Iván Lema)*

## 4.3. PROGRAMACIÓN

En esta sección se procede a explicar la programación de cada dispositivo, dependiendo del tipo de proceso al que se le ha asignado controlar, se toma en cuenta la secuencia diseñada, las entradas y salidas asignadas en cada dispositivo, el Grafcet, las ecuaciones y los tags de cada OPC.

### 4.3.1. PROGRAMACIÓN DEL SIEMENS S7-1200

El PLC Siemens S7-1200 se encarga de controlar el módulo de Envasadora.

#### 4.3.1.1. *Funcionamiento del Módulo de Envasadora.*

*Al pulsar el Botón de Inicio*

1. Se enciende el motor que hace girar el tornillo sin fin y actúa el cilindro B (B+) que retiene la tapa.

*Detecta SM2 y no SM3 (Sección llenado):*

2. El sensor SM2 detecta la lata en el inicio de la secuencia.
3. Detiene el motor y actúa el cilindro D (D+), que succiona el líquido.
4. Detecta el sensor d1, que nos indica que termino de succionar el líquido.
5. Regresa el pistón del cilindro D y se activa el Solenoide de la válvula que llena la lata (Sol+)
6. Espera un tiempo T1.
7. Sigue a la sección final.

*No detecta SM2 y Detecta Sm3 (Sección sellado):*

8. Detiene el motor del tornillo sin fin.
9. Espera un tiempo T2.
- 10.El cilindro B libera la tapa
- 11.El sensor SM1 detecta la tapa suelta.
- 12.Espera un tiempo T3 para que baje totalmente la tapa.
- 13.Activa B, que retiene la tapa consecutiva, y se activa el cilindro C, que sella la lata con la tapa (C+).
- 14.Detectan los sensores c1, b1, que hayan realizado la acción anterior esperan un tiempo T4.
- 15.El cilindro C regresa a su posición inicial.
- 16.Espera un tiempo T5.
- 17.Sigue a la sección final.

*Detecta Sm2 y Sm3 (Llenado y sellado)*

- 18.Detiene el motor del tornillo y activa D que succiona el líquido.
- 19.Espera un tiempo T6 y que detecte el sensor d1.
- 20.Regresa el pistón de D, activa la válvula que llena la lata y el cilindro B libera la tapa.
- 21.Detecta la tapa con SM1 y espera un tiempo T10.
- 22.Espera un tiempo T7.
- 23.Activa el cilindro C que sella la lata y el cilindro B que retiene l siguiente tapa.
- 24.Espera que c1, b1 detecten, además de un tiempo T11.
- 25.C regresa a su posición inicial.

26. Espera un tiempo T8.

27. Va la sección final.

*Sección final:*

Activa el motor del tornillo sin fin y desactiva el solenoide que cierra la válvula y espera a recibir la señal de los sensores SM2 y SM3

#### **4.3.1.2. Señales de Entradas y Salidas asignadas al S7-1200**

Las señales de entrada y salida deben ser etiquetadas en el PLC, la tabla siguiente muestra las direcciones asignadas a cada una de ellas.

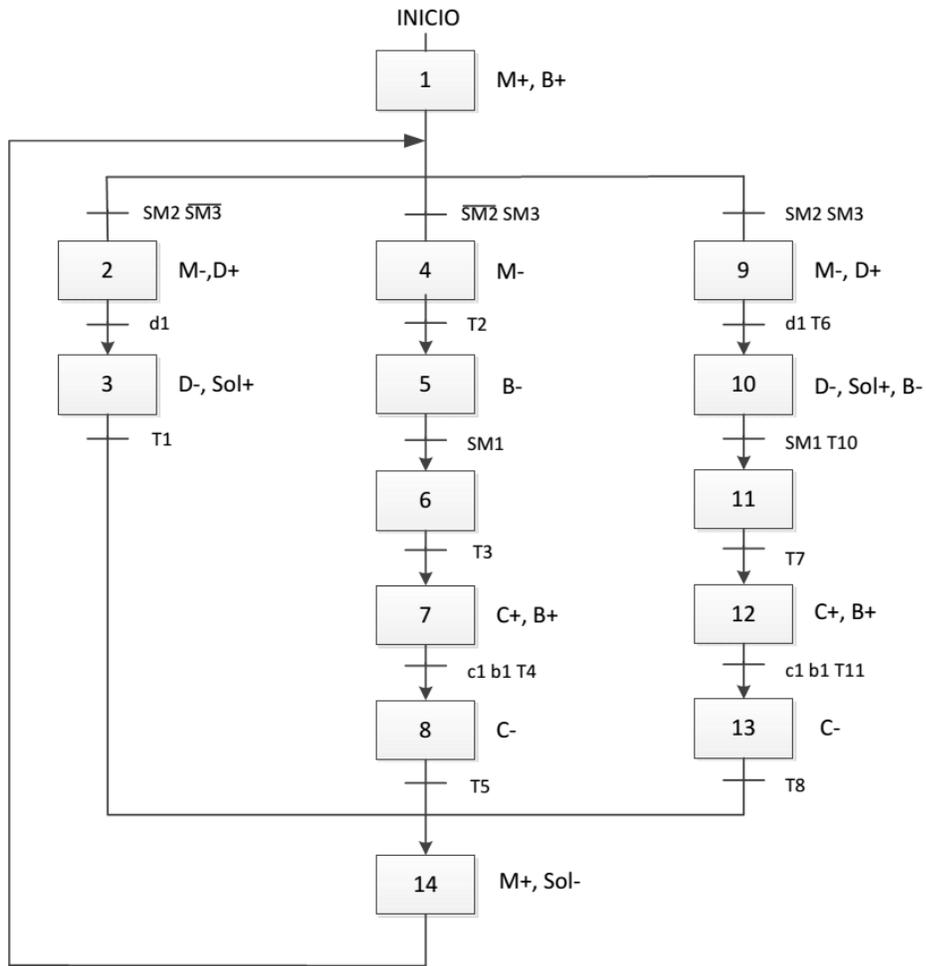
**Tabla IV.XXIV. Asignación de Entradas y Salidas del PLC**  
**ASIGNACIÓN DE DIRECCIONES AL PLC SIEMENS S7-1200**

<b>ENTRADA/SALIDA</b>	<b>DIRECCIÓN FÍSICA</b>	<b>TAG</b>	<b>DESCRIPCIÓN</b>
Entrada	%I0.0	INICIO	Pulsador NO de Inicio
Entrada	%I0.1	STOP	Pulsador NC de Paro
Entrada	%I0.2	SM3	Detecta lata en sección de tapado
Entrada	%I0.3	b1	Final de carrera cilindro B
Entrada	%I0.4	c1	Final de carrera cilindro C
Entrada	%I0.5	d1	Final de carrera cilindro D
Entrada	%I0.6	SM1	Detecta tapa de la lata
Entrada	%I0.7	SM2	Detecta lata para empezar
Salida	%Q0.0	B+	Acciona Cilindro B (Retiene tapa)
Salida	%Q0.1	C+	Acciona Cilindro C (Sella tapa)
Salida	%Q0.2	D+	Acciona Cilindro D (Succiona liquido)
Salida	%Q0.3	NC	No conectado
Salida	%Q0.4	SOL+	Solenoide de válvula que llena la lata
Salida	%Q0.5	M+	Acciona Motor

*Fuente: Realizado por Leonardo Asqui e Iván Lema (Autores).*

### 4.3.1.3. Grafcet del Módulo Envasadora

A continuación se procede a realizar el Grafcet del Módulo Envasadora, el cual tiene tres secuencias a las que se accede dependiendo de la sección en la que se encuentra la lata.



**Figura IV.107.** Grafcet del Módulo Envasadora.  
*Fuente: Leonardo Asqui e Iván Lema (Autores).*

#### 4.3.1.4. Ecuaciones

A continuación se procede a establecer las ecuaciones en base al Grafcet planteado para posteriormente realizar la programación en el PLC Siemens S7-1200.

Tabla IV.XXV. Tabla de Ecuaciones

ETAPA	ECUACIÓN
1	$M1 = INICIO + M1 \cdot \overline{M2} \cdot \overline{M3} \cdot \overline{M4}$
2	$M2 = M1 \cdot SM2 \cdot SM3 + M14 \cdot T9 \cdot SM2 \cdot \overline{SM3} + M2 \cdot \overline{M3}$
3	$M3 = M2 \cdot d1 + M3 \cdot M4$
4	$M4 = M1 \cdot \overline{SM2} \cdot SM3 + M14 \cdot T9 \cdot \overline{SM2} \cdot SM3 + M4 \cdot M5$
5	$M5 = M4 \cdot T2 + M5 \cdot M6$
6	$M6 = M5 \cdot SM1 + M6 \cdot \overline{M7}$
7	$M7 = M6 \cdot T3 + M7 \cdot \overline{M8}$
8	$M8 = M7 \cdot c1 \cdot b1 \cdot T4 + M8 \cdot \overline{M14}$
9	$M9 = M1 \cdot SM2 \cdot SM3 + M14 \cdot T9 \cdot SM2 \cdot SM3 + M9 \cdot M10$
10	$M10 = M9 \cdot T6 \cdot d1 + M10 \cdot \overline{M1}$
11	$M11 = M10 \cdot SM1 + M11 \cdot \overline{M12}$
12	$M12 = M11 \cdot T7 + M12 \cdot \overline{M13}$
13	$M13 = M12 \cdot c1 \cdot b1 \cdot T11 + M13 \cdot \overline{M14}$
14	$M14 = M3 \cdot T1 + M13 \cdot T8 + M8 \cdot T5 + M14 \cdot \overline{M2} \cdot \overline{M4} \cdot \overline{M9}$
<b>SALIDAS FÍSICAS</b>	
	$M1 + M7 + M12 = B +$ $M7 + M12 = C +$ $M2 + M9 = D +$ $M3 + M10 = Sol +$ $M1 + M14 = M +$

Fuente: Realizado por Leonardo Asqui e Iván Lema (Autores).

### 4.3.2. PROGRAMACIÓN DEL WAGO 750-842

#### 4.3.2.1. Funcionamiento de la Mesa Indexadora

Pulsar el botón de inicio.

1. El sensor óptico detecta la pieza.
2. Espera un tiempo.

Si el contador es 0 avanza a la etapa final.

*Si el contador es igual a 1:*

3. Acciona el cilindro de sujeción.
4. Acciona y baja el taladro.
5. Apaga y sube el taladro.
6. Continúa a la etapa final.

*Si el contador es igual 2 ó 3*

7. Acciona el cilindro de sujeción
8. Acciona y baja el taladro, y el avellanador al mismo tiempo.
9. Apaga y sube el taladro, y el avellanador.
10. Continúa a la etapa final

*Si el contador es mayor o igual a 4*

11. Acciona el cilindro de sujeción.
12. Acciona y baja el taladro, el avellanador, y se acciona el expulsor.
13. Apaga y sube el taladro, el avellanador y el expulsor.
14. Coloca el contador en 0.

*Etapas Finales*

15. Espera un tiempo
16. Acciona el motor de la mesa y coloca el cilindro de sujeción en la posición inicial.
17. Espera un tiempo y a que detecte el sensor inductivo.

Vuelve al paso 1 para reiniciar la secuencia.

#### 4.3.2.2. Señales de las Entradas y Salidas asignadas al Wago 750-842

A continuación se presentan las direcciones y tags de las Entradas y Salidas asignadas al PLC Wago 750-842. Se debe tomar en cuenta que al momento de asignar las direcciones Modbus en este PLC se debe iniciar desde la dirección QX256.0.

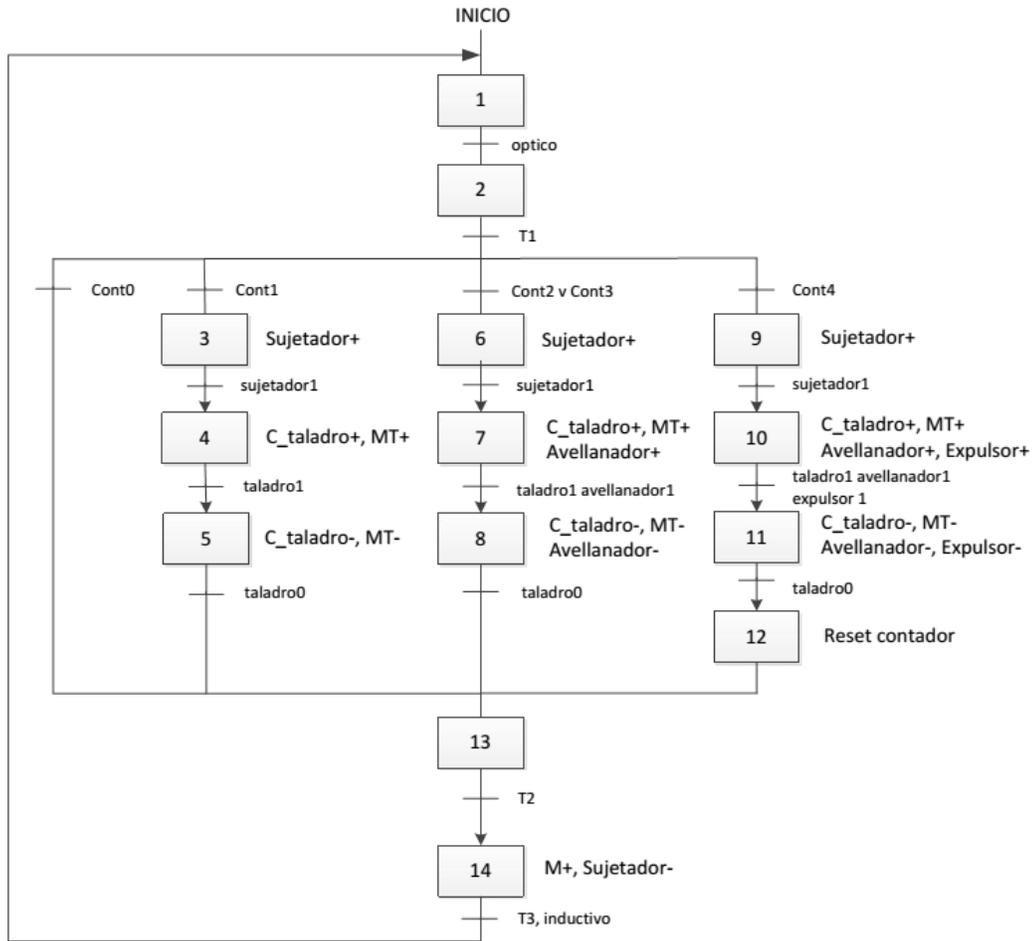
**Tabla IV.XXVI. Asignación de Entradas y Salidas del PLC WAGO**

ASIGNACIÓN DE DIRECCIONES AL PLC WAGO 750-842				
ENTRADA /SALIDA	DIRECCIÓN FÍSICA	DIRECCIÓN MODBUS	TAG	DESCRIPCIÓN
Entrada	I0.0	QX259.0	Inicio	Pulsador NO de Inicio
Entrada	I0.1	QX259.1	Inductivo	Sensor Inductivo
Entrada	I0.2	QX259.2	Óptico	Sensor Óptico
Entrada	I0.3	QX259.3	sujetador1	Final de carrera sujetador
Entrada	I0.4	QX259.4	taladro1	C. taladro posición 1
Entrada	I0.5	QX259.5	avellanador1	Final de carrera avellanador
Entrada	I0.6	QX259.6	expulsor1	Final de carrera C. expulsor
Entrada	I0.7	QX259.7	taladro0	C taladro posición 0
Entrada	I0.8	QX260.0	Paro	Pulsador NC de Paro
Salida	Q0.0	QX270.0	Expulsor+	Acciona Cilindro Expulsor
Salida	Q0.1	QX270.1	Avellanador+	Acciona Avellanador
Salida	Q0.2	QX270.2	C_taladro+	Acciona cilindro del taladro
Salida	Q0.3	QX270.3	Sujetador+	Acciona el sujetador
Salida	Q0.4	QX270.4	MT+	Acciona el Taladro
Salida	Q0.5	QX270.5	M+	Acciona Motor

*Fuente: Realizado por Leonardo Asqui e Iván Lema (Autores).*

#### 4.3.2.3. Graficet del módulo Mesa Indexadora

A continuación se muestra el Graficet para el módulo de la Mesa Indexadora, que como se observó en la secuencia cuenta con tres secuencias dependiendo del estado del contador.



**Figura IV.108.** Grafcet del Módulo Mesa de Indexación.  
*Fuente: Leonardo Asqui e Iván Lema (Autores).*

#### 4.3.2.4. Ecuaciones

A continuación se muestra la tabla con las ecuaciones resultantes del Grafcet anteriormente diseñado para el módulo de la Mesa de Indexación controlada por el PLC WAGO 750-842.

**Tabla IV.XXVII. Tabla de Ecuaciones para el PLC WAGO.**

ETAPA	ECUACIÓN
1	$M1 = INICIO + inductivo \cdot T3 \cdot M14 + M1 \cdot \overline{M2}$
2	$M2 = optico \cdot M1 + M2 \cdot \overline{M13} \cdot \overline{M3} \cdot \overline{M6} \cdot \overline{M9}$
3	$M3 = Cont1 \cdot T1 \cdot M2 + M3 \cdot \overline{M4}$

4	$M4 = \text{sujetador1} \cdot M3 + M4 \cdot \overline{M5}$
5	$M5 = \text{taladro1} \cdot M4 + M5 \cdot \overline{M13}$
6	$M6 = (\text{Cont2} + \text{Cont3}) \cdot T1 \cdot M2 + M6 \cdot \overline{M7}$
7	$M7 = \text{sujetador1} \cdot M6 + M7 \cdot \overline{M8}$
8	$M8 = \text{taladro1} \cdot \text{avellanador1} \cdot M7 + M8 \cdot \overline{M13}$
9	$M9 = \text{Cont4} \cdot T1 \cdot M2 + M9 \cdot \overline{M10}$
10	$M10 = \text{sujetador1} \cdot M9 + M10 \cdot \overline{M11}$
11	$M11 = \text{taladro1} \cdot \text{avellanador1} \cdot \text{expulsor1} \cdot M10 + M11 \cdot \overline{M12}$
12	$M12 = \text{taladro0} \cdot M11 + M12 \cdot \overline{M13}$
13	$M13 = T1 \cdot \text{Cont0} \cdot M2 + (M5 + M8) \cdot \text{taladro0} + M12 + M13 \cdot \overline{M14}$
14	$M14 = T2 \cdot M13 + M14 \cdot \overline{M1}$
<b>SALIDAS FÍSICAS</b>	
	$M3 + M6 + M9 = \text{Sujetador} +$ $M4 + M7 + M10 = \text{C\_taladro} +, \text{MT} +$ $M7 + M10 = \text{Avellanador} +$ $M10 = \text{Expulsor} +$ $M14 = \text{M} +$

*Fuente: Realizado por Leonardo Asqui e Iván Lema (Autores).*

### 4.3.3. PROGRAMACIÓN DEL PLC TWIDO

El PLC Twido TWDLMDA20DTK controlara en funcionamiento de un semáforo de 4 vías

#### 4.3.3.1. Secuencia del Semáforo

Pulsar el botón inicio:

1. Se encenderán la luces rojas 1,3, verdes 2,4; se pagaran las luces rojas 2,4, amarillas 1,3.
2. Después de un tiempo seguirá encendido la luces rojas 1,3; se apagara las luces verdes 2,4 y se encenderán las luces amarillas 2,4.
3. Después de otro tiempo se apagaran las luces rojas 1,3, amarillas 2,4 y se encenderán las luces verdes 1,3, rojas 2,4.

4. Luego de otro tiempo seguirá encendido las luces rojas 2,4; se apagaran las luces verdes 1,3; y se encenderán las luces amarillas 1,3.

Luego reiniciara con el paso 1.

Pulsar el botón de apagado hará que se apaguen todas las luces del semáforo.

Pulsar el botón de emergencia hará que se prendan solamente las luces rojas.

Con el botón tiempo se podrá cambiar los tiempos de las luces de los semáforos.

#### ***4.3.3.2. Señales de las Entradas y Salidas asignadas al Twido***

Las señales de entradas/salidas físicas asignadas al PLC.

**Tabla IV.XXVIII. Asignación de Entradas y Salidas del PLC TWIDO**

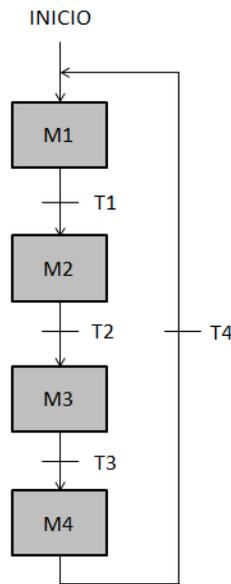
<b>ENTRADA /SALIDA</b>	<b>DIRECCIÓN FÍSICA</b>	<b>TAG</b>	<b>DESCRIPCIÓN</b>
Entrada	I0.0	Inicio	Pulsador NO de Inicio
Entrada	I0.1	Paro	Pulsador NO de paro
Entrada	I0.2	Emergencia	Pulsador NO de emergencia
Entrada	I0.3	N/A	No asignado
Entrada	I0.4	N/A	No asignado
Entrada	I0.5	N/A	No asignado
Entrada	I0.6	N/A	No asignado
Entrada	I0.7	N/A	No asignado
Entrada	I0.8	N/A	No asignado
Entrada	I0.9	N/A	No asignado
Entrada	I0.10	N/A	No asignado
Entrada	I0.11	N/A	No asignado
Salida	Q0.0	ROJO_1	Acciona los relés para activar las luces rojo 1 y 3
Salida	Q0.1	AMARILLO_1	Acciona los relés para activar las luces amarillo 1 y 3

Salida	Q0.2	N/A	No asignado
Salida	Q0.3	ROJO_2	Acciona los relés para activar las luces rojo 2 y 4
Salida	Q0.4	AMARILLO_2	Acciona los relés para activar las luces amarillo 2 y 4
Salida	Q0.5	VERDE_2	Acciona los relés para activar las luces verde 2 y 4
Salida	Q0.6	N/A	No Asignado
Salida	Q0.7	VERDE_1	Acciona los relés para activar las luces verde 1 y 3

**Fuente:** Realizado por Leonardo Asqui e Iván Lema (Autores).

#### 4.3.3.3. Grafcet del Semáforo

A continuación se muestra el Grafcet para el funcionamiento del semáforo, que como se observó en la secuencia cuenta con cuatro tiempos para las secuencias de las luces.



**Figura IV.109.** Grafcet del semáforo.

*Fuente:* Leonardo Asqui e Iván Lema (Autores).

#### 4.3.3.4. Ecuaciones

Tabla IV.XXIX. Tabla de Ecuaciones para el PLC TWIDO.

ETAPA	ECUACIÓN
1	$M1 = \text{INICIO} + T4 \cdot M4 + M1 \cdot \overline{M2}$
2	$M2 = M1 \cdot T1 + M2 \cdot \overline{M3}$
3	$M3 = M2 \cdot T2 + M3 \cdot \overline{M4}$
4	$M4 = M3 \cdot T3 + M4 \cdot \overline{M1}$
5	$T1 = M1 \cdot \text{TM0.P}$
6	$T2 = M2 \cdot \text{TM1.P}$
7	$T3 = M3 \cdot \text{TM2.P}$
8	$T4 = M4 \cdot \text{TM3.P}$
9	$M19(S) = M1$
10	$M19(R) = M3$
11	$M20(S) = M3$
12	$M20(R) = M1$
<b>SALIDAS FÍSICAS</b>	
	M19 = ROJO_1 M4 = AMARILLO_1 M3 = VERDE_1 M20 = ROJO_2 M2 = AMARILLO_2 M1 = VERDE_2

*Fuente: Realizado por Leonardo Asqui e Iván Lema (Autores).*

#### 4.3.4. PROGRAMACIÓN DEL FIELDPOINT FP-1601

El PAC FP-1601 controlara el funcionamiento de la casa automatizada.

##### 4.3.4.1. Funcionamiento de la Casa Automatizada

El funcionamiento de la casa automatizada se divide en 3 partes independientes que son el funcionamiento de puerta del garaje, el accionamiento de las luces de los cuartos, y la activación de las alarmas.

Primera parte: (Funcionamiento de garaje)

Se pulsa el botón inicio:

1. Primero se activara el la luz de Parar y no dejara que se encienda el motor para abrir o cerrar la puerta.
2. Ahorra si se podrá seleccionar la acción de cerrar o abrir el garaje.
3. Si se escoge Abrir:  
Se activara si el sensor de garaje abierto no está activado;  
Se abrirá hasta que este se active el sensor de garaje abierto, se pulse parar, o se pulse cerrar.
4. Si se escoge Cerrar:  
Se activara si el sensor de garaje cerrado no está activado.  
Se cerrara hasta que se active el sensor de garaje cerrado, se pulse parar, o se pulse abrir.
5. Se parara cualquier acción de abrir o cerrar si se pulsa parar.
6. Después se puede repetir cualquier acción del paso 3 al 5.

Segunda parte: (Funcionamiento de las Luces)

Cada luz de la casa funcionara de forma independiente pero con la misma lógica.

Se pulsa el botón de inicio:

1. Al momento de presionar por primera vez el botón inicio las luces estarán apagadas.
2. Si se pulsa el botón de la luz se encenderá el foco.
3. Si se pulsa otra vez el botón se apagara el foco.

Se vuelve a repetir la acción del 2 al 3.

Tercera parte: (Funcionamiento de la alarma)

La alarma se podrá activar y desactivar si el usuario y la contraseña son correctos, para activar o desactivar la alarma tendrá 15 segundos para realizar esta acción.

Se pulsa el botón inicio:

1. Al momento de presionar por primera vez la alarma estará desactivada.
2. Si se pulsa el botón de la alarma se activara la alarma.
3. Si se pulsa otra vez la alarma se desactivará la alarma.
4. Si esta activada la alarma y los sensores se han activado sonara la sirena de la alarma. Se podrá desactivar la alarma accediendo a la alarma.
5. Si la alarma esta desactivada y los sensores se han activado no sonara la sirena de la alarma.

#### ***4.3.4.2. Señales de las Entradas y Salidas asignadas al FieldPoint***

Las señales de entradas/salidas físicas asignadas al PAC.

**Tabla IV.XXX. Asignación de Entradas y Salidas del PAC FIELD POINT FP-1601**

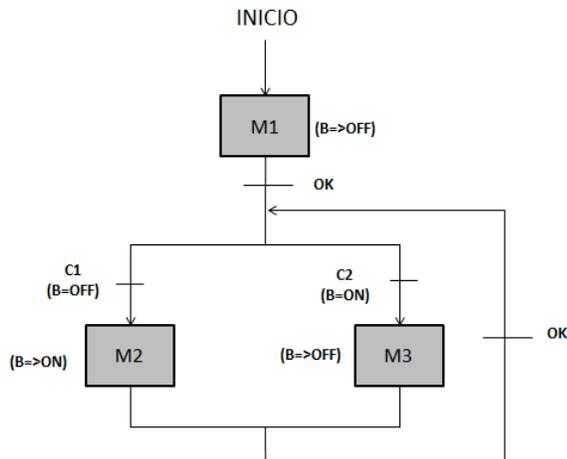
<b>MODULO</b>	<b>ENTRADA /SALIDA</b>	<b>DIRECCIÓN FÍSICA</b>	<b>TAG</b>	<b>DESCRIPCIÓN</b>
FP-AI-100	Entrada	Channel0	Ent_Principal	Sensor de la entrada principal
FP-AI-100	Entrada	Channel1	Garaje_open	Sensor del garaje abierto
FP-AI-100	Entrada	Channel2	Garaje_close	Sensor del garaje cerrado
FP-AI-100	Entrada	Channel3	N/A	No asignado
FP-AI-100	Entrada	Channel4	Ent_Secund	Sensor de la entrada secund.
FP-AI-100	Entrada	Channel5	Ventana	Sensor de la ventana
FP-AI-100	Entrada	Channel6	N/A	No asignado
FP-AI-100	Entrada	Channel7	N/A	No asignado
FP-RLY-420	Salida	Channel0	Cerrar	Cierra el garaje

FP-RLY-420	Salida	Channel1	Abrir	Abre el Garaje.
FP-RLY-420	Salida	Channel2	Foco_1	Activa y desactiva el foco 1
FP-RLY-420	Salida	Channel3	Foco_2	Activa y desactiva el foco 2
FP-RLY-420	Salida	Channel4	Foco_3	Activa y desactiva el foco 3
FP-RLY-420	Salida	Channel5	Foco_4	Activa y desactiva el foco 4
FP-RLY-420	Salida	Channel6	Foco_5	Activa y desactiva el foco 5
FP-RLY-420	Salida	Channel7	Sir_alarma	Acciona la sirena de la alarma

**Fuente:** Realizado por Leonardo Asqui e Iván Lema (Autores).

#### 4.3.4.3. Grafcet de la Casa Automatizada

El grafcet de la casa automatizada será dividido en tres partes pero la programación no se basa en Ladder ya que será programada en el software LabVIEW para su funcionamiento pero se basó en el grafcet para la programación.



**Figura IV.110.** Grafcet de la Casa Automatizada. Activación o desactivación de luces.

*Fuente:* Leonardo Asqui e Iván Lema (Autores).

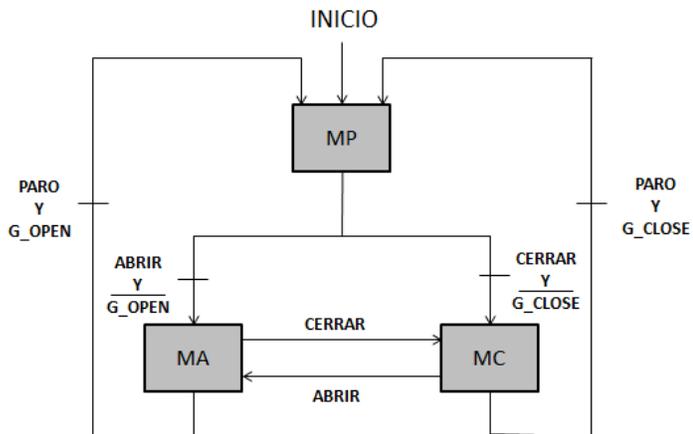


Figura IV.111. Grafcet de la Casa Automatizada. Apertura o cierre de garaje.

Fuente: Leonardo Asqui e Iván Lema (Autores).

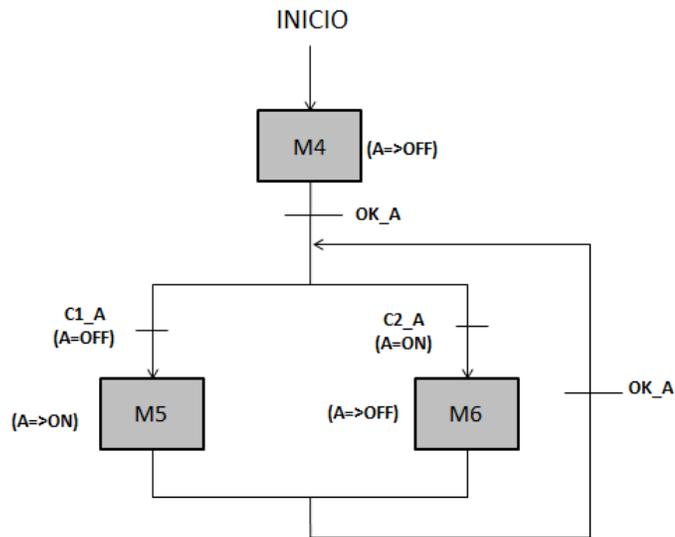


Figura IV.111. Grafcet de la Casa Automatizada. Activación o desactivación de alarma.

Fuente: Leonardo Asqui e Iván Lema (Autores).

#### 4.3.4.4. Ecuaciones

Tabla IV.XXXI. Tabla de Ecuaciones para el FP 1601 para el funcionamiento del garaje.

ETAP A	ECUACIÓN
1	$\overline{MP} = \overline{INICIO} + \overline{MA} \cdot (\overline{PARO} + \overline{G\_OPEN}) + \overline{MB} \cdot (\overline{PARO} + \overline{G\_ABRIR}) + \overline{MP} \cdot \overline{MA} \cdot$
2	$\overline{MA} = (\overline{MP} + \overline{MC}) \cdot \overline{ABRIR} \cdot \overline{G\_ABRIR} + \overline{MA} \cdot \overline{MP} \cdot \overline{MC}$
3	$\overline{MC} = (\overline{MP} + \overline{MA}) \cdot \overline{CERRAR} \cdot \overline{G\_CERRAR} + \overline{MC} \cdot \overline{MP} \cdot \overline{MA}$
<b>SALIDAS FÍSICAS</b>	
	$\overline{MA} \cdot \overline{MP} = \text{MOTOR\_DERECHA}$ $\overline{MC} \cdot \overline{MP} = \text{MOTOR\_IZQUIERDA}$ $\overline{MP} = \text{LUZ DE PARO}$ $\overline{MA} \cdot \overline{MP} = \text{LUZ\_ABRIR}$ $\overline{MC} \cdot \overline{MP} = \text{LUZ\_CERRAR}$

*Fuente:* Realizado por Leonardo Asqui e Iván Lema (Autores).

Las mismas ecuaciones funcionan para cada foco, para que funcionen de manera independiente.

Tabla IV.XXXII. Tabla de Ecuaciones para el FP-1601 para el funcionamiento de las luces.

ETAPA	ECUACIÓN
1	$\overline{M1} = \overline{INICIO} + \overline{M1} \cdot \overline{M2} \cdot \overline{M3}$
2	$\overline{M2} = (\overline{C1} \cdot \overline{M1} + \overline{M3}) \cdot \overline{OK} + \overline{M2} \cdot \overline{M3}$
3	$\overline{M3} = (\overline{C2} \cdot \overline{M1} + \overline{M2}) \cdot \overline{OK} + \overline{M3} \cdot \overline{M2}$
4	$\overline{MX(S)} = \overline{M2}$
5	$\overline{MX(R)} = \overline{M3}$
<b>SALIDAS FÍSICAS</b>	
	$\overline{MX} = \text{FOCO\_X}$

*Fuente:* Realizado por Leonardo Asqui e Iván Lema (Autores).

El graficet de los focos con la alarma es similar; difieren en la salida de la alarma con las salidas de los focos.

Tabla IV.XXXIII. Tabla de Ecuaciones para el FP-1601 para el funcionamiento de las luces.

ETAPA	ECUACIÓN
1	$M4 = \text{INICIO} + M4 \cdot M5 \cdot \overline{M6}$
2	$M5 = (\overline{C1\_A} \cdot M4 + M6) \cdot \text{OK\_A} + M5 \cdot \overline{M6}$
3	$M6 = (\overline{C2\_A} \cdot M4 + M5) \cdot \text{OK\_A} + M6 \cdot M5$
4	$M7(S) = M5 \cdot (\text{IPP}_1 + \text{IPP}_2 + \text{VENTANA})$
5	$M4(R) = M6$
<b>SALIDAS FÍSICAS</b>	
	$M4 = \text{SIR\_ALARMA}$

Fuente: Realizado por Leonardo Asqui e Iván Lema (Autores).

## 4.4. BASE DE DATOS

### 4.4.1. DISEÑO

Para el diseño de la base de datos de forma inicial se ha propuesto la implementación de cuatro tablas, PLC, ALARMA, USUARIO y EVENTO; las cuales cuentan con relaciones múltiples entre sí, como se muestra en la siguiente figura.

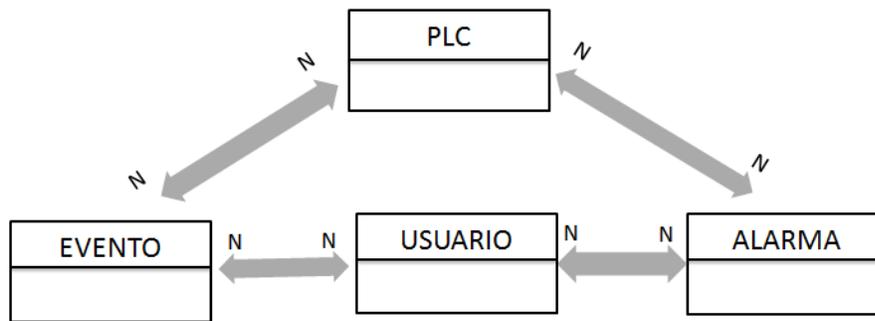
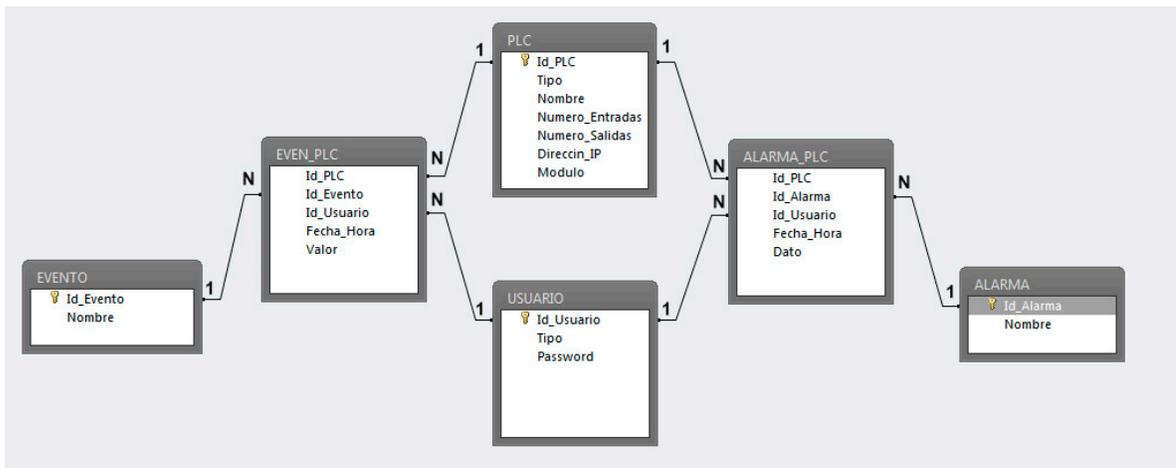


Figura IV.112. Modelado de la Base de Datos  
Fuente: Leonardo Asqui e Iván Lema (Los Autores)

Pero según las consideraciones para el diseño de las bases de datos, las tablas a ser creadas no pueden tener una relación múltiple, varios a varios; de modo que para

evitar este problema se deben crear Tablas Auxiliares, en donde la relación de varios a varios se descomponga en dos relaciones de uno a varios.

Las tablas auxiliares son EVEN\_PLC y ALARMA\_PLC, las cuales se relacionan las tablas EVENTO, PLC y USUARIO; y ALARMA, PLC y USUARIO, respectivamente, con una relación 1-N para evitar inconsistencias.

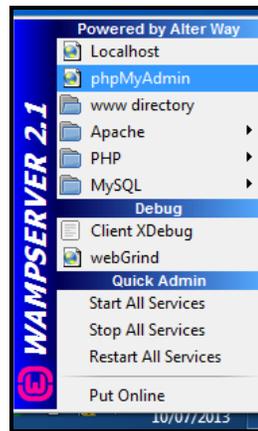


**Figura IV.113.** Diseño de la Base de Datos  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

## 4.4.2. CONFIGURACIÓN

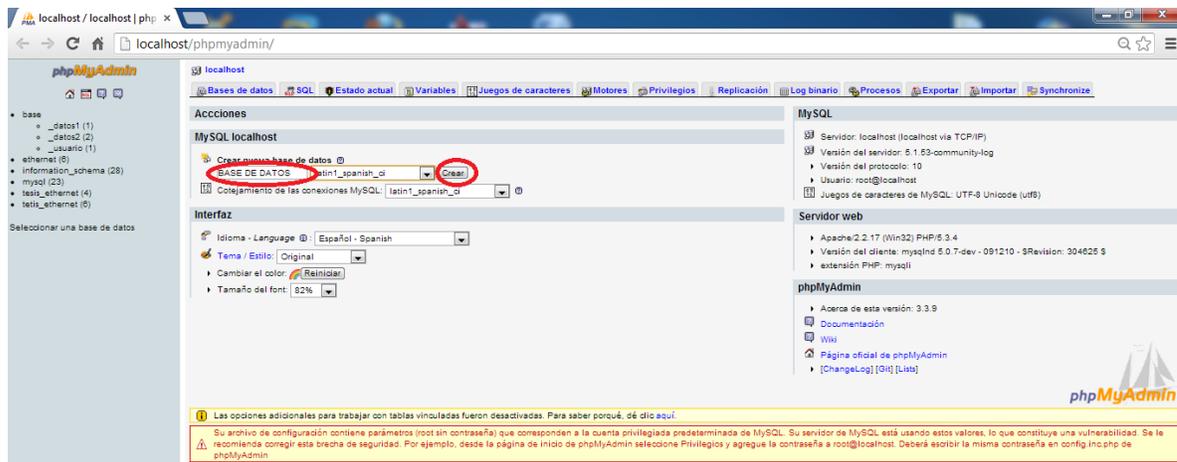
### 4.4.2.1. CREACIÓN BASE DE DATOS EN MySQL PHP

Una vez instalado correctamente WampServer, lo ejecutamos y entramos en phpMyAdmin



**Figura IV.114.** Menú WampServer  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

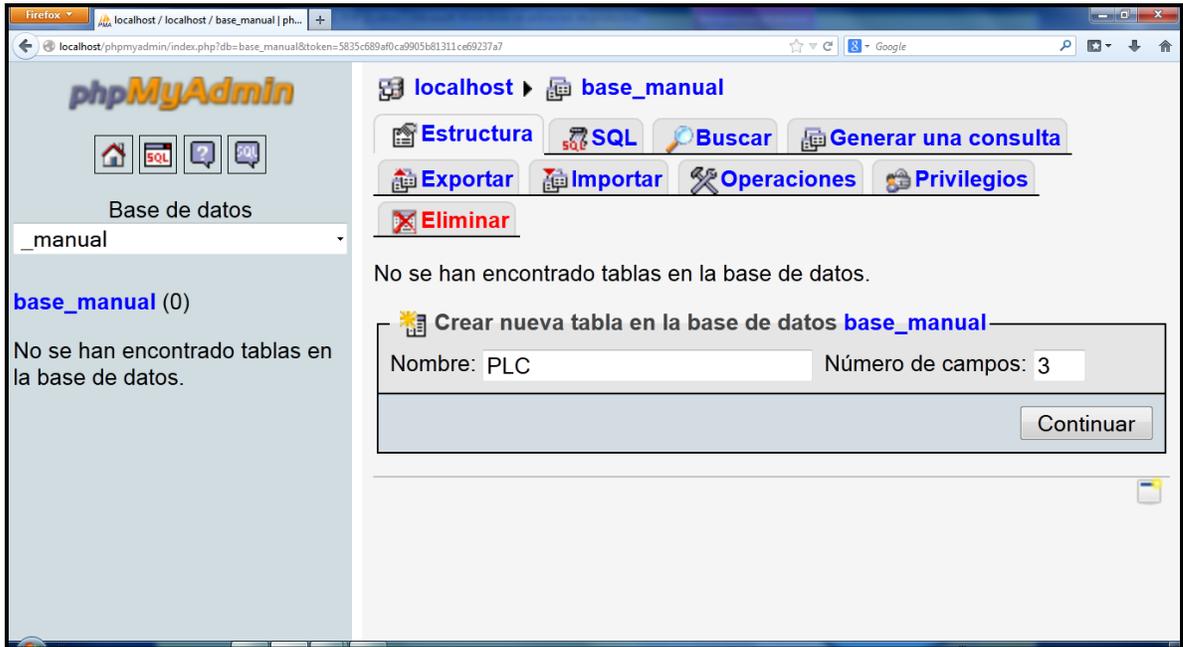
Seguidamente nos mostrara la interfaz de configuración de base de datos, procedemos a crear una nueva base de datos, según la Figura IV.113. Diseño de la Base de Datos, ingresamos el nombre de la base de datos y damos en crear.



**Figura IV.115.** Creación de una base de datos  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

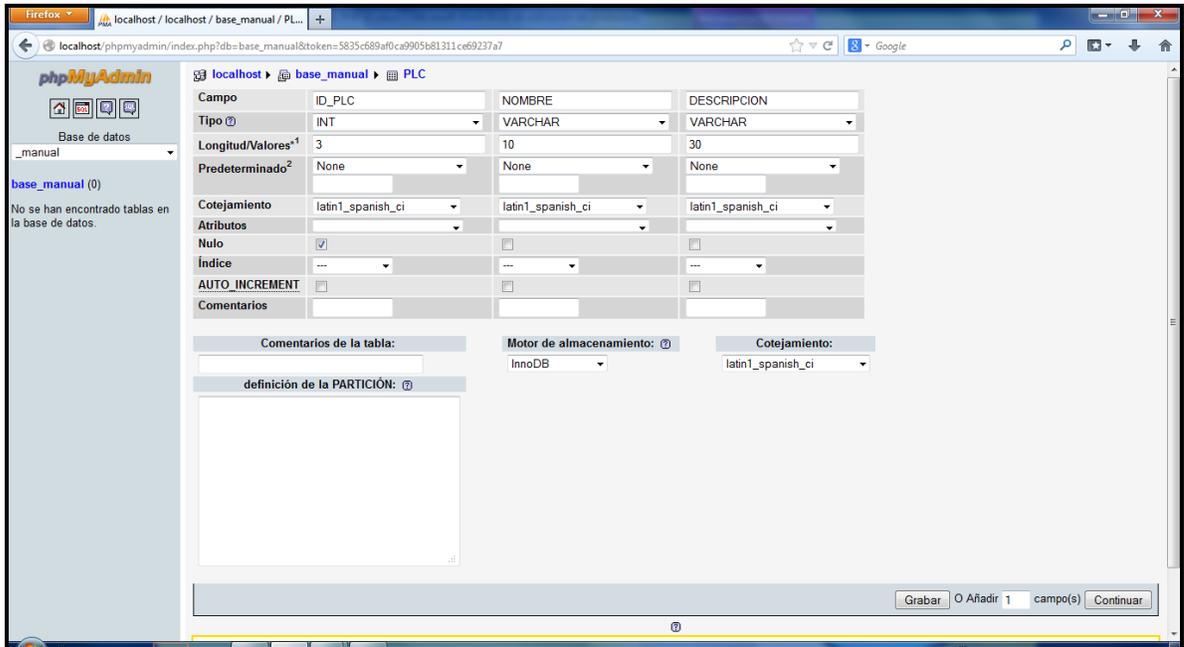
## CREACION DE UNA TABLA

Ahora creamos una tabla en la base de datos llamada “PLC”, en donde ingresamos el nombre de la tabla y número de campos, seguidamente pulsamos continuar



**Figura IV.116.** Creación de una tabla  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

Ingresamos los datos de los campos, nombre, tipo, longitud máxima, si es clave primaria, y si se le asigna autoincremento, como motor de almacenamiento elegimos InnoDB.



**Figura IV.117.** Configuración de campos de una tabla.  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

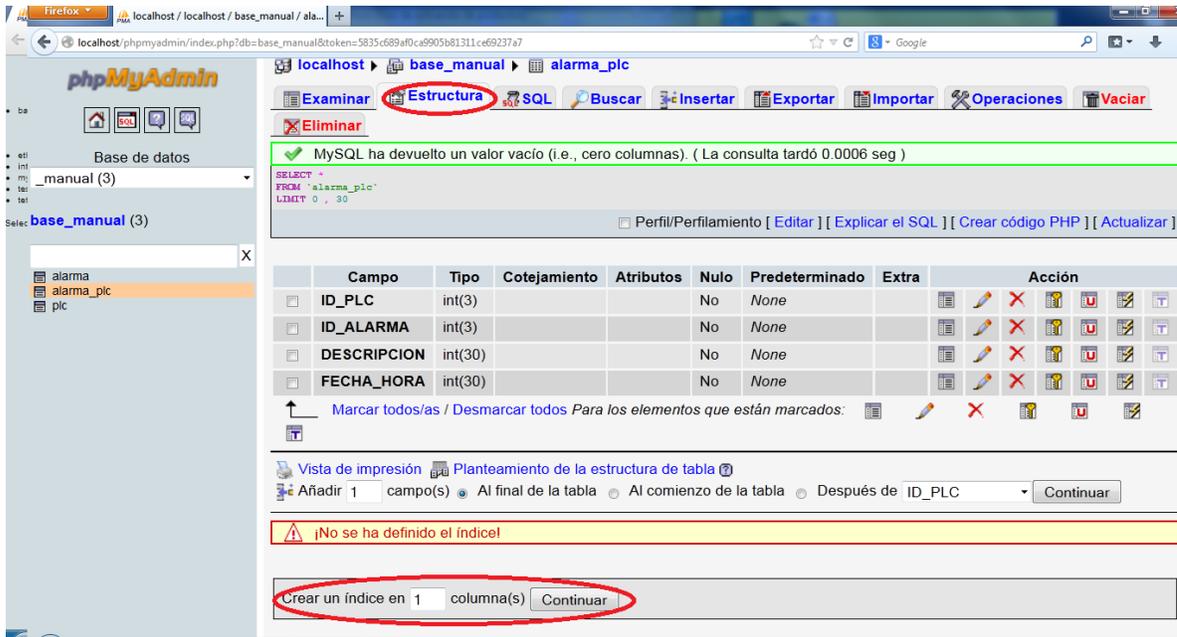
De esta forma creamos el resto de tablas de nuestra base de datos:

- Alarma.
- Alarma\_plc.
- Evento.
- Even\_plc..
- Usuario.

## CREACION DE UN INDEX

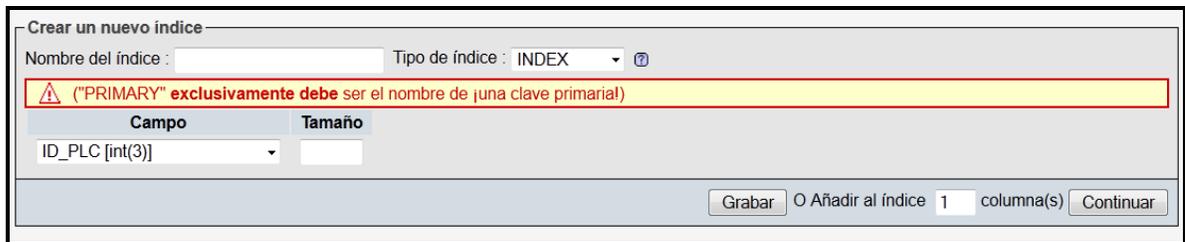
El index se crea para relacionar las tablas, cuando se tiene una relación 1-N y 1-1. De acuerdo a nuestra diagrama E-R en la tabla alarma\_plc se tiene la relación N es por esto que el index se crea en esta tabla.

Para crear un index, vamos a Estructura y crear un index ponemos 1, damos click en continuar.



**Figura IV.118.** Creación de un INDEX  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

En la ventana que se abre escogemos el campo con el cual deseamos hacer el index, en este caso ID\_PLC, damos en grabar.



**Figura IV.119.** Campo de INDEX  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

## CREACIÓN DE UNA RELACIÓN

Creamos las relaciones en la tabla evento con la tabla plc y alarma, previamente se debió haber configurado correctamente los index, para crear la relacione damos click en la tabla evento y vamos a estructura → vista de relaciones, la relación se hace por medio de las claves primarias quedando como se muestra.



**Figura IV.120.** Creación de una relación  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

## 4.5. HMI

### 4.5.1. OPC

Como se ha visto anteriormente, para una conexión exitosa entre los controladores y el HMI se necesita de un OPC Server, en este caso tenemos el Kepserver.

Para poder trabajar de mejor manera se ha procedido a crear un Canal por cada controlador, con lo que tenemos cuatro channels, cada uno con las direcciones y requerimientos específicos de cada controlador.

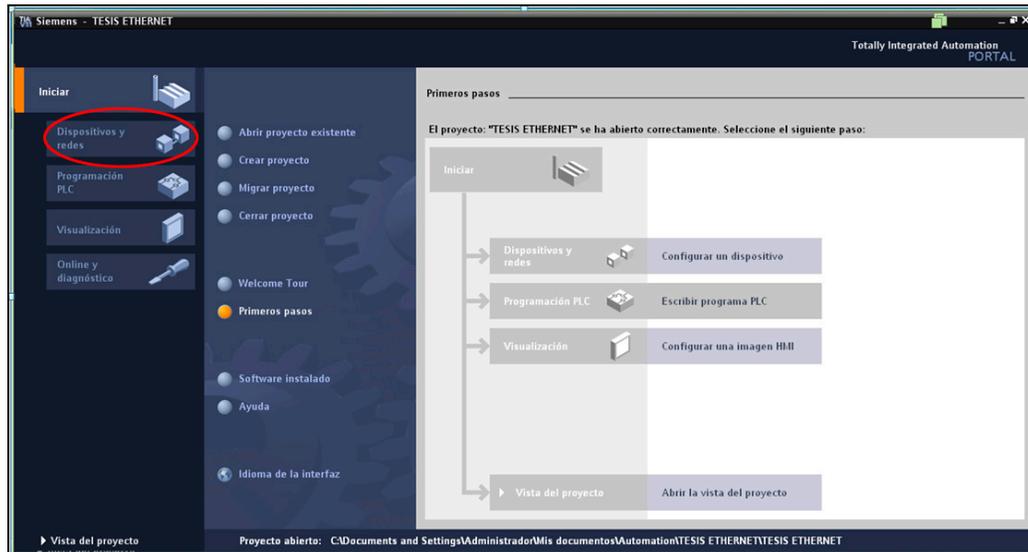
#### 4.5.1.1. Configuración S7-1200

Para poder trabajar correctamente con el OPC en el PLC Siemens, primero se debe realizar el reseteo y configuración del PLC, seguido de la creación de direcciones en el Bloque de Datos del PLC y finalmente se crearan los tags de OPC.

### RESETEAR Y CONFIGURAR LA DIRECCION IP DEL PLC

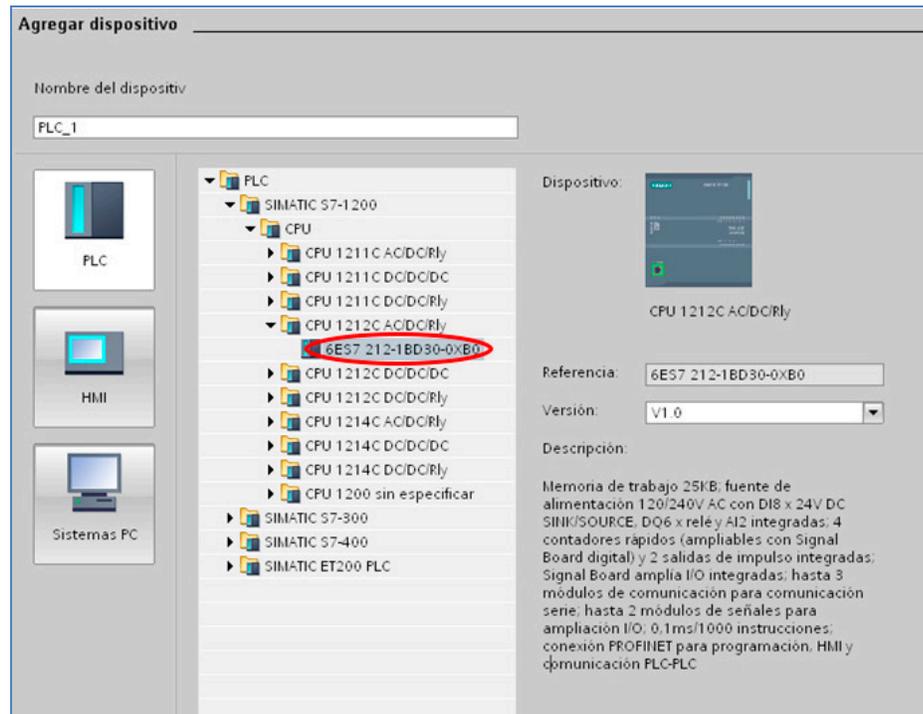
Para configurar la dirección IP del PLC Siemens se debe tomar en cuenta que el PLC debe estar conectado a la PC, para transferir de forma satisfactoria los nuevos parámetros a utilizarse.

1. Ejecutar el TIA Portal y Crear un Nuevo Proyecto
2. Ahora procedemos a Configurar los Dispositivos, damos click en la opción ***“Dispositivos y Redes”*** y elegimos la opción ***“Agregar Dispositivo”***.



**Figura IV.121.** Selección de Dispositivos y Redes. En TIA Portal  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

3. Buscamos el dispositivo que estamos utilizando, en este caso **CPU 1212C AC/DC/Rly**. Debemos tener muy en cuenta la versión del Firmware, en este caso es **Versión 1.0**.

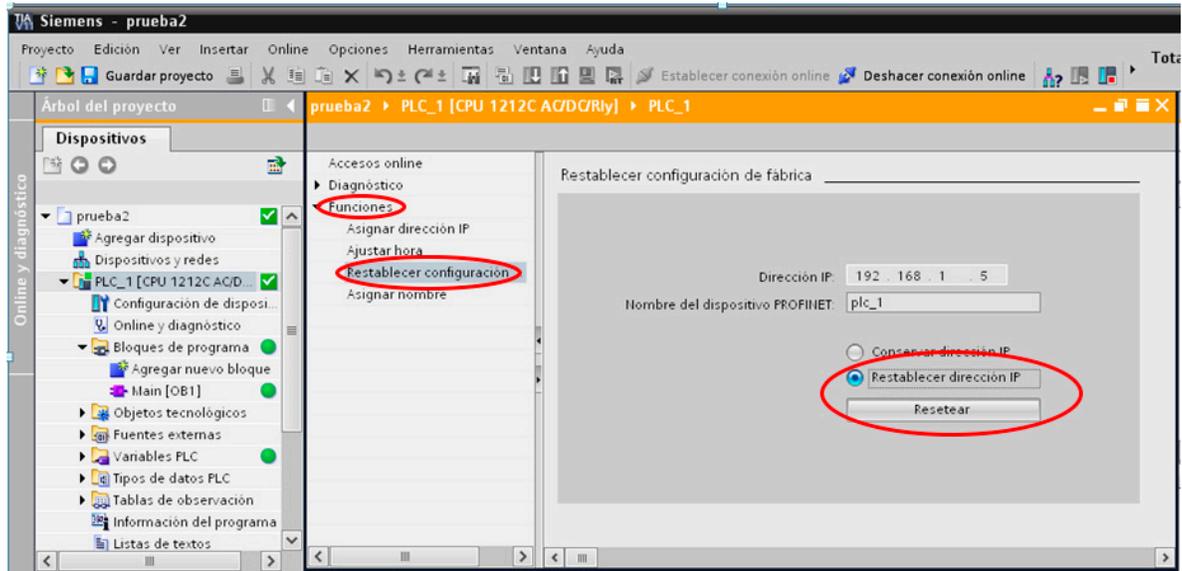


**Figura IV.122.** Selección del dispositivo empleado en la implementación. En TIA Portal  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

Una vez configurado el dispositivo en el TIA Portal, procedemos a **Reestablecer la configuración de fábrica del PLC**.

4. En el **Árbol de Proyecto** seleccionamos el PLC que estamos utilizando, a continuación vamos a la barra de iconos del programa, y damos click en **Conexión Online**. El color de la barra de título se volverá naranja indicando el estado online.

5. En estado online vamos seleccionamos el PLC → **Funciones** → **Reestablecer configuración**. Seleccionamos la opción **“Reestablecer dirección IP”**, y damos click en **“Resetear”**.



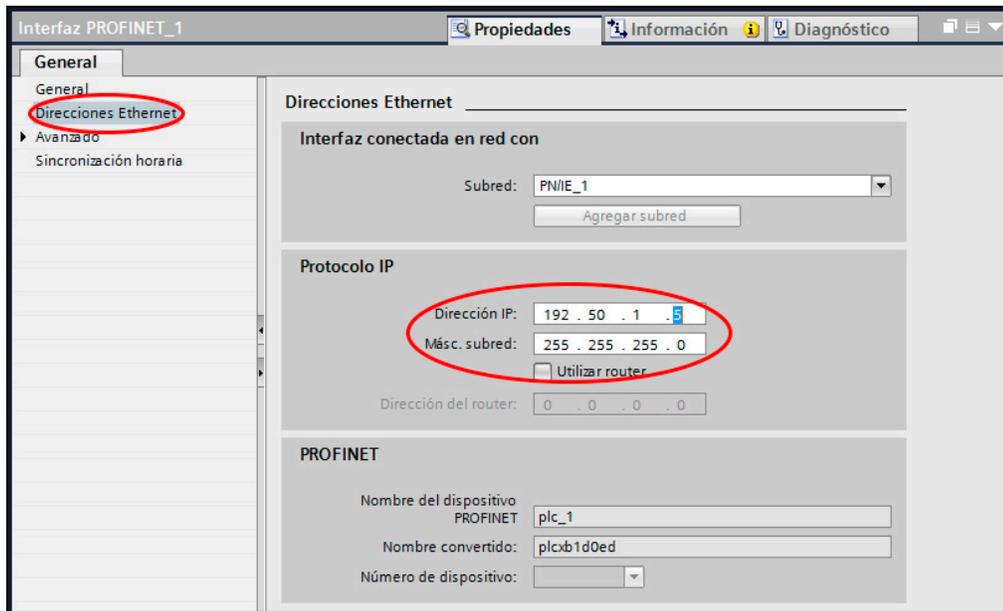
**Figura IV.123.** Reseteo del PLC. En TIA Portal  
Fuente: Leonardo Asqui e Iván Lema (Autores)

6. En la ventana de confirmación que aparece damos click en **“Si”**. De igual forma en la ventana siguiente que nos pide para el dispositivo.

Ahora vamos a colocar una **nueva dirección IP** en el PLC.

7. Seleccionamos el PLC, al mostrar el dispositivo, damos doble click en **el Puerto Ethernet** de la figura.

8. A continuación aparece la ventana en donde se localiza la opción **“Direcciones Ethernet”** y establecemos la nueva dirección IP del PLC.



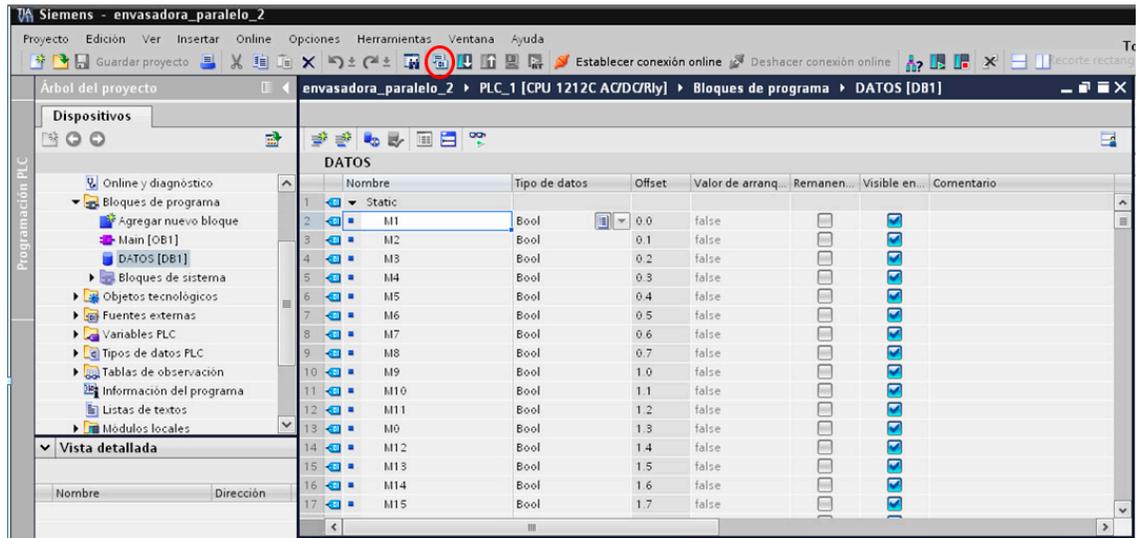
**Figura IV.124.** Asignación de la dirección IP en el PLC. En TIA Portal  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

9. Ahora se procede a cargar el dispositivo con la nueva dirección IP.

## CREACION DE DIRECCIONES PARA CONEXIÓN CON OPC

1. Ejecutar el *TIA PORTAL*
2. Acceder a la *“Vista de Proyecto”*, en el *“Árbol del Proyecto”*, elegir el PLC que utilizamos, la opción *“Variables de PLC”*, y *“Tabla de Variables Estándar”*.
3. Ahora vamos a *Bloques de Programa* en el *Árbol de Proyectos*, y damos doble click en *Agregar nuevo Bloque*.
4. Seleccionamos el *“Bloque de Datos”*, ponemos el Nombre, en la opción Número elegimos *“Manual”*, en Acceso a Bloques elegimos *“Estándar-compatible con S7-300/400”* y damos click en *“Aceptar”*.

5. En el bloque de datos se ingresan las memorias, entradas y salidas que se utilizan en la programación. A continuación se procede a **Compilar** para que aparezcan los numerales **Offset** que nos servirán posteriormente.



**Figura IV.125.** Bloque de Datos con las variables a utilizar ya compilado. En TIA Portal  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

6. Ahora se procede a crear una **“Nueva Tabla de Observación”**, opción que se encuentra en el árbol de proyecto. En esta tabla de observación se agregan los datos del Bloque de Datos, llamándolos primero por el nombre del bloque y por su nombre específico.

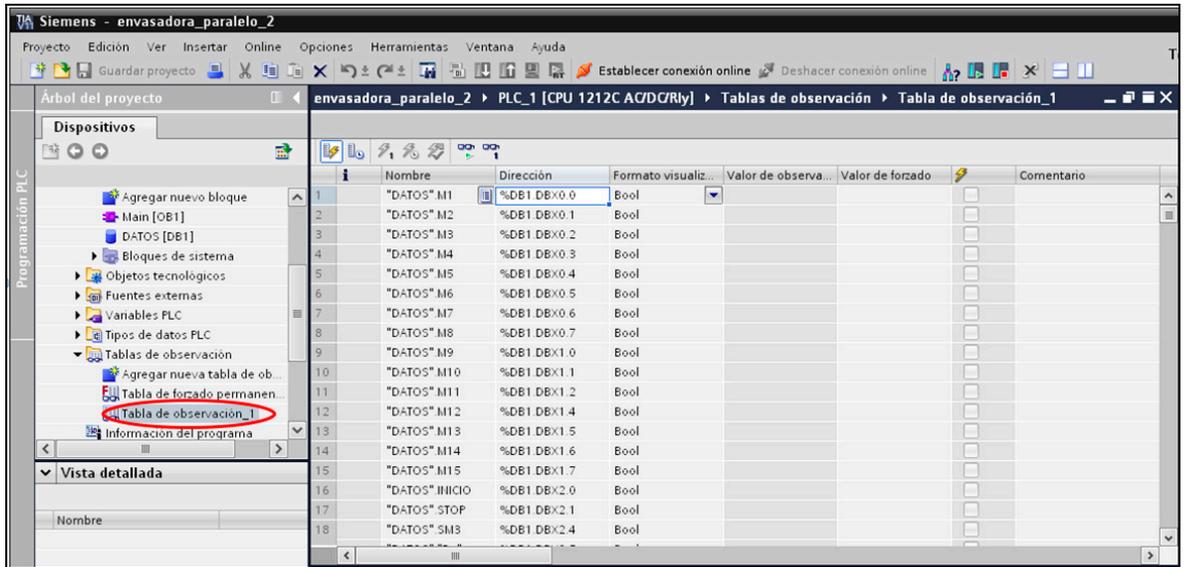


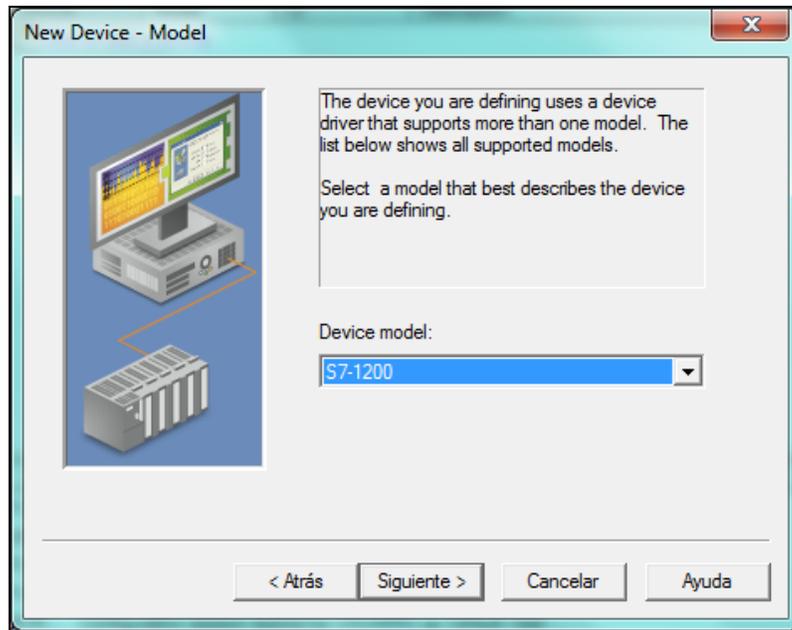
Figura IV.126. Tabla de Datos con las variables del Bloque de Datos. En TIA Portal

Fuente: Leonardo Asqui e Iván Lema (Autores)

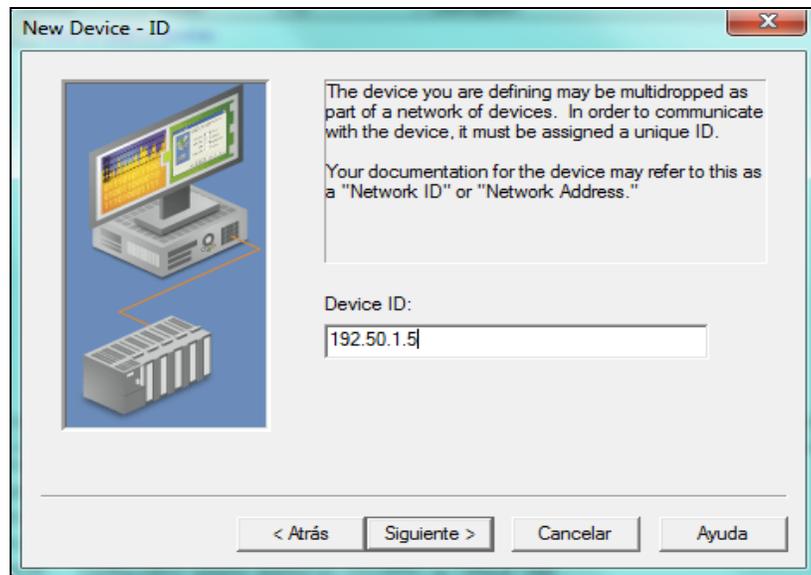
*Nota:* Estas direcciones creadas en la tabla de observación serán posteriormente utilizadas en la creación de Tags en el OPC Kepsver.

## CREACION Y CONFIGURACIÓN DEL OPC PARA SIEMENS

1. Ejecutamos Kepsver y escoger **“Nuevo”**. A continuación hacer click en **“Click to add a channel”**, y configurar el nuevo canal, empezando por el nombre, el driver del dispositivo (en este caso Siemens TCP/IP Ethernet), y la tarjeta de red que se utiliza.
2. Una vez creado el canal, se procede a crear el dispositivo. Click en **“Click to add to device”**. En las ventanas que aparecen, se asigna el nombre del dispositivo, el modelo que se utiliza, la dirección IP que posee, dentro de los detalles más importantes.



**Figura IV.127.** Elección del modelo del dispositivo. En Keepserver  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*



**Figura IV.128.** Asignación de la dirección IP del dispositivo. En Keepserver  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

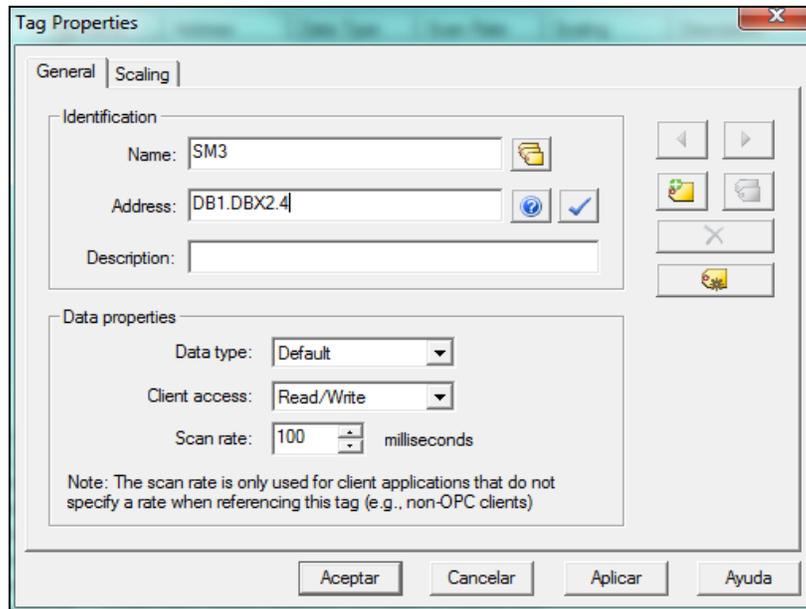
Una vez creado el canal y el dispositivo de Siemens, se crean las tags que se utilizarán posteriormente en el HMI.

3. Para crear las tags hacemos click en ***“Click to add a static tag”*** en la ventana principal del KepServer. En el caso del PLC Siemens, la dirección a utilizar para las tags se obtiene a partir de la **Tabla de Observación** creada en el programa TIA Portal.

	Nombre	Dirección	Formato visualiz...	Valor de observa...	Valor de forzado		Comentario
1	"DATOS".M1	%DB1.DBX0.0	Bool			<input type="checkbox"/>	
2	"DATOS".M2	%DB1.DBX0.1	Bool			<input type="checkbox"/>	
3	"DATOS".M3	%DB1.DBX0.2	Bool			<input type="checkbox"/>	
4	"DATOS".M4	%DB1.DBX0.3	Bool			<input type="checkbox"/>	
5	"DATOS".M5	%DB1.DBX0.4	Bool			<input type="checkbox"/>	
6	"DATOS".M6	%DB1.DBX0.5	Bool			<input type="checkbox"/>	
7	"DATOS".M7	%DB1.DBX0.6	Bool			<input type="checkbox"/>	
8	"DATOS".M8	%DB1.DBX0.7	Bool			<input type="checkbox"/>	
9	"DATOS".M9	%DB1.DBX1.0	Bool			<input type="checkbox"/>	
10	"DATOS".M10	%DB1.DBX1.1	Bool			<input type="checkbox"/>	
11	"DATOS".M11	%DB1.DBX1.2	Bool			<input type="checkbox"/>	
12	"DATOS".M12	%DB1.DBX1.4	Bool			<input type="checkbox"/>	
13	"DATOS".M13	%DB1.DBX1.5	Bool			<input type="checkbox"/>	
14	"DATOS".M14	%DB1.DBX1.6	Bool			<input type="checkbox"/>	
15	"DATOS".M15	%DB1.DBX1.7	Bool			<input type="checkbox"/>	
16	"DATOS".INICIO	%DB1.DBX2.0	Bool			<input type="checkbox"/>	
17	"DATOS".STOP	%DB1.DBX2.1	Bool			<input type="checkbox"/>	
18	"DATOS".SM3	%DB1.DBX2.4	Bool			<input type="checkbox"/>	

**Figura IV.129.** Direcciones de la Tabla de Observación utilizadas para las tags de OPC. En TIA Portal  
Fuente: Leonardo Asqui e Iván Lema (Autores)

4. Escribir el nombre en ***“Name”***, dirección en ***“Address”***, la descripción en ***“Description”***. Y click en ***“Aplicar”*** y ***“Aceptar”***.



**Figura IV.130.** Creación de las tags de OPC en Kepserv.   
Fuente: Leonardo Asqui e Iván Lema (Autores)

5. Para crear las demás tags, se elige la opción “**New Tag**”, haciendo click derecho en el espacio de las tags.

#### 4.5.1.2. Configuración Wago 750-842

Antes de crear las tags de OPC se debe tener muy en cuenta el direccionamiento con este PLC, para ello se debe revisar la siguiente tabla:

**Tabla IV.XXXIV. Áreas de Direccionamiento IEC-61131-3**

Área de Dirección	Acceso MODBUS	Acceso PLC	Descripción
phys.inputs	Read	read	Entradas físicas (%IW0...%IW255 und %IW512...%IW1275)
phys.outputs	read/write	read/write	Salidas físicas (%QW0...%QW255 und %QW512...%QW1275)
MODBUS/TCP PFC-IN variables	read/write	read	Variable Entrada Volátil PLC (%IW256...%IW511)
MODBUS/TCP	Read	read/write	Variable Salida Volátil PLC

PFC-OUT variables			(%QW256...%QW511)
Ethernet/IP PFC-IN variables	-	read	Variable Entrada Volátil PLC (%IW1276...%IW1531)
Ethernet/IP PFC-OUT variables	-	read/write	Variable Entrada Volátil PLC (%QW1276...%QW1531)
Configuration register	read/write	-	
Firmware register	read/write	-	
Retain variables	read/write	read/write	Memoria Remanente (%MWO...%MW12288)

Fuente: Manual WAGO Ethernet Driver Help

Como se puede apreciar, lo más importante de esta tabla son las direcciones. Aquí las direcciones físicas tienen el rango de palabra de 0 a 255.

- **Entradas:** La entrada 1 %IX0.0 corresponde a 100001 en Kepserver
- **Salidas:** La salida 1 %QX0.0 corresponde a 000001 en Kepserver
- Las direcciones **Modbus** corresponden las palabras desde 256 hasta 511

El bit 0 de la palabra 256 corresponde a 4097 en Kepserver siendo:

- Salida %QX256.0 corresponde la dirección 104097

## CONFIGURACIÓN DE DIRECCIÓN IP EN WAGO

### 1. Configuración del BOOTP SERVER

- Ejecutamos el Boot server en modo administrador.
- Damos click en "*Edit Bootptab*"



**Figura IV. 131.** Pantalla principal del Boot Server  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

Aparecerá el siguiente block de notas

```
# bootptab.txt: database for bootp server
#
# Blank lines and lines beginning with '#' are ignored.
#
# Legend:
#   first field -- hostname (not indented)
#   bf -- bootfile
#   bs -- bootfile size in 512-octet blocks
#   cs -- cookie servers
#   df -- dump file name
#   dn -- domain name
#   ds -- domain name servers
#   ef -- extension file
#   gw -- gateways
#   ha -- hardware address
#   hd -- home directory for bootfiles
#   hn -- host name set for client
#   ht -- hardware type
#   im -- impress servers
#   ip -- host IP address
#   lg -- log servers
#   lp -- LPR servers
#   ns -- IEN-116 name servers
#   ra -- reply address
#   rl -- resource location protocol servers
#   rp -- root path
#   sa -- boot server address
#   sm -- subnet mask
#   sw -- swap server
#   tc -- template host (points to similar host entry)
```

```
# td -- TFTP directory
# to -- time offset (seconds)
# ts -- time servers
# vm -- vendor magic number
# Tn -- generic option tag n
#
# Be careful about including backslashes where they're
# needed. Weird (bad)
# things can happen when a backslash is omitted where one is
# intended.
# Also, note that generic option data must be either a string
# or a
# sequence of bytes where each byte is a two-digit hex value.
#
# Example of entry with no gateway
WORKGROUP:ht=1:ha=0030DE0013EE:ip=192.50.1.2:
sm=255.255.255.0:
# Example of entry with gateway
# The gateway address must be inserted in hexadecimal
# after the T3 parameter
#
hamburg:ht=1:ha=0030DE000003:ip=10.1.254.203:T3=0A.01.FE.01:
```

Donde modificaremos una línea

**WORKGROUP:ht=1:ha=0030DE0013EE:ip=192.50.1.2:**

**sm=255.255.255.0:**

Dónde:

WORKGROUP: Es el grupo de trabajo.

Ht: Tipo de hardware.

Ha: La dirección MAC del controlador.

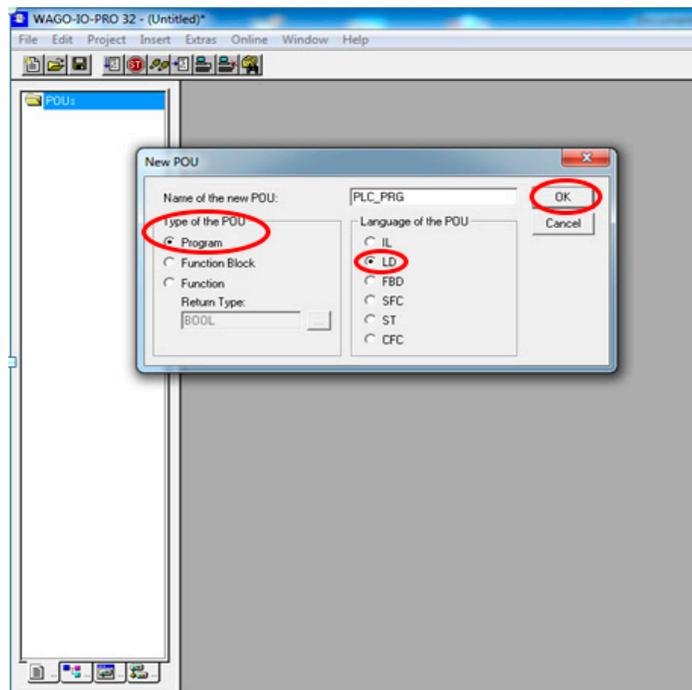
Ip: La dirección Ip que va a llevar el controlador.

Sm: la máscara de subred.

- Guardar y salir.
- Damos click en Start.

- Apagamos el PLC por unos minutos y lo arrancamos.
- Nos salen diálogos de la conexión que se ha establecido y damos click en Stop y salimos.

2. Se ejecuta el programa WAGO IO PRO 32, en donde se crea un Nuevo Programa en **File** → **New**, en la ventana de configuración se elige el tipo de lenguaje (para este caso LD, ladder).



**Figura IV. 132.** Creación del nuevo programa en WAGO IO PRO 32.  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

3. A continuación se debe escoger el PLC que se usara en la red, **CONTROLADOR ETHERNET WAGO 750-842**, para esto nos dirigimos a la pestaña **Extras**→**PLC selection**.

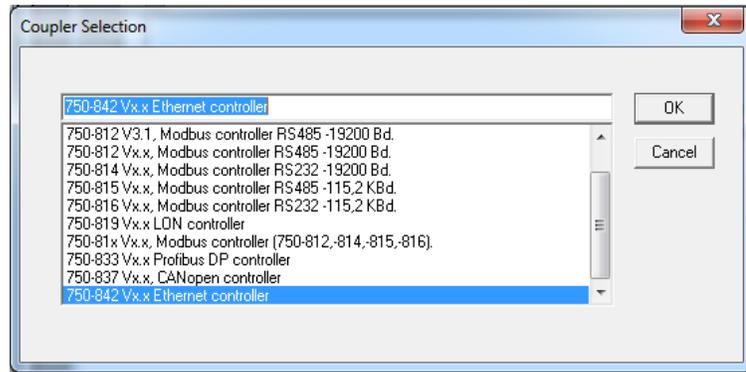


Figura IV. 133. Elección del PLC WAGO 750-842.

Fuente: Leonardo Asqui e Iván Lema (Autores)

4. Para configurar los parámetros de comunicación, vamos a la pestaña **Online** → **Communication Parameters**, en donde se configura un Nuevo Canal. Aquí se le da un nombre y se elige un dispositivo (en este caso **Ethernet\_TCP\_IP**).

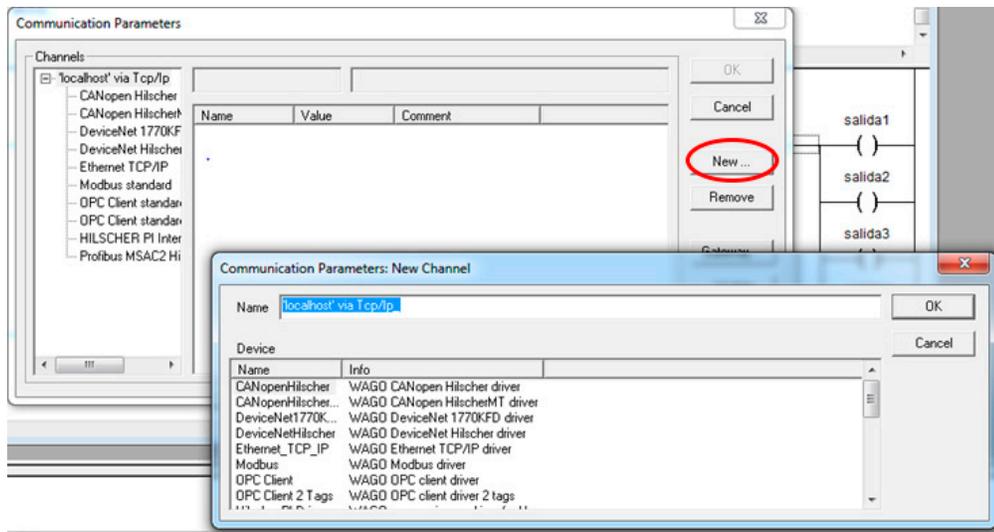


Figura IV. 134. Creación del Nuevo Canal de comunicación.

Fuente: Leonardo Asqui e Iván Lema (Autores)

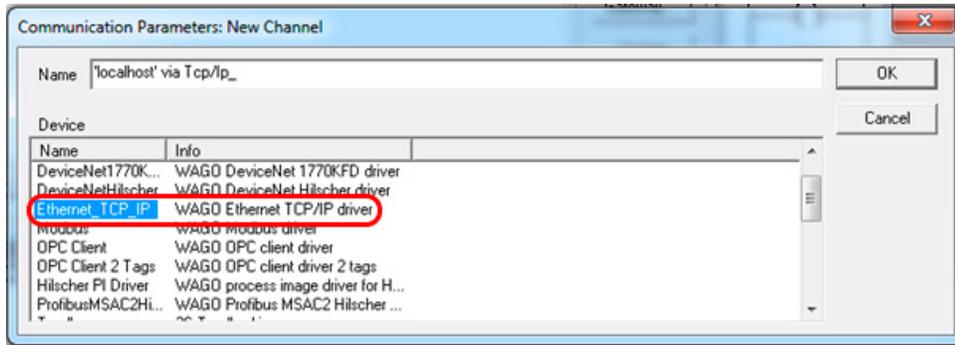


Figura IV. 135. Elección del canal Ethernet\_TCP\_IP.  
Fuente: Leonardo Asqui e Iván Lema (Autores)

5. A continuación se debe ingresar la dirección IP del controlador, la cual debe ser la misma que se ingresó en Bootp Server.

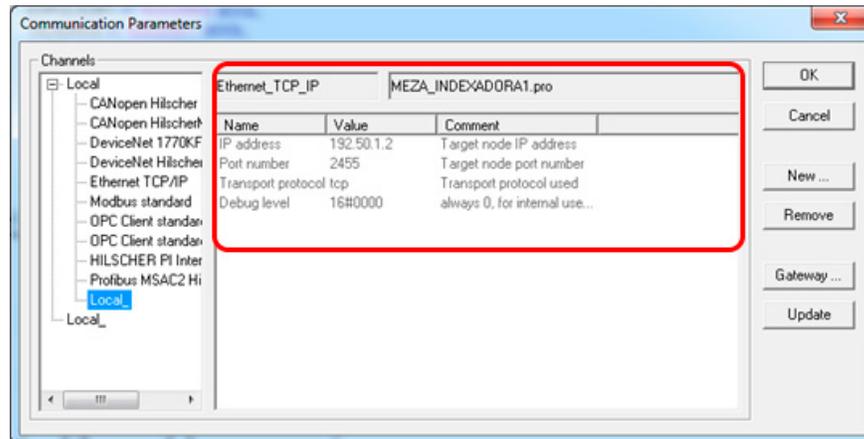
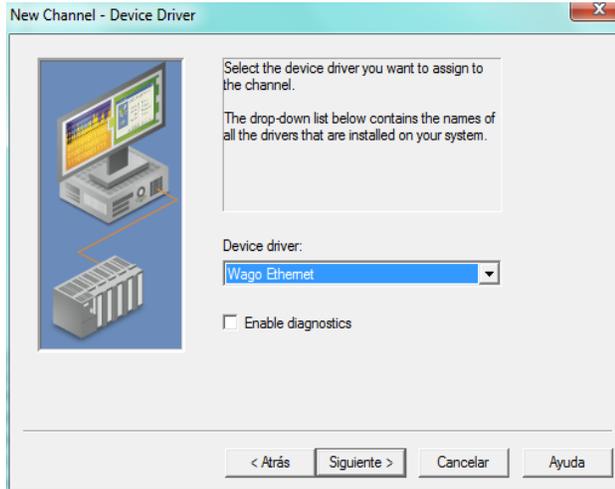


Figura IV. 136. Ventana de Parámetros de Comunicación con la dirección IP del Wago.  
Fuente: Leonardo Asqui e Iván Lema (Autores)

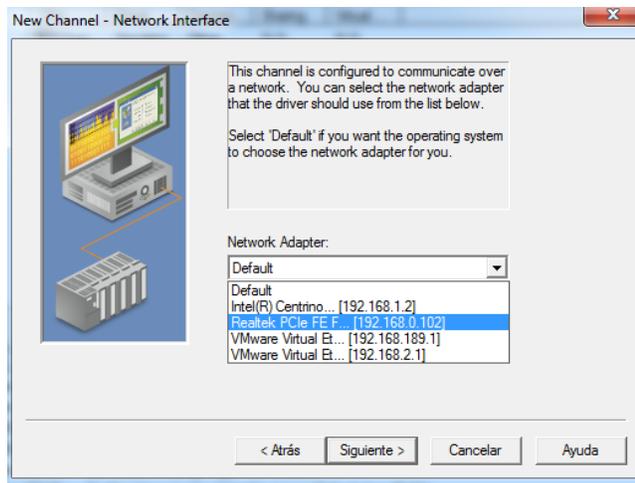
## CONFIGURACIÓN DEL OPC DE WAGO

Ahora se procede a crear un nuevo canal, tomando en cuenta que esta vez lo haremos para el PLC Wago, para ello debemos tener en cuenta los parámetros que anteriormente se han declarado en el PLC.

1. En el área de canales del Kepserver, damos click derecho y elegimos la opción “**New Channel**”. Se configura el nombre del canal, el driver (Wago Ethernet) y la tarjeta de red, como pasos fundamentales.

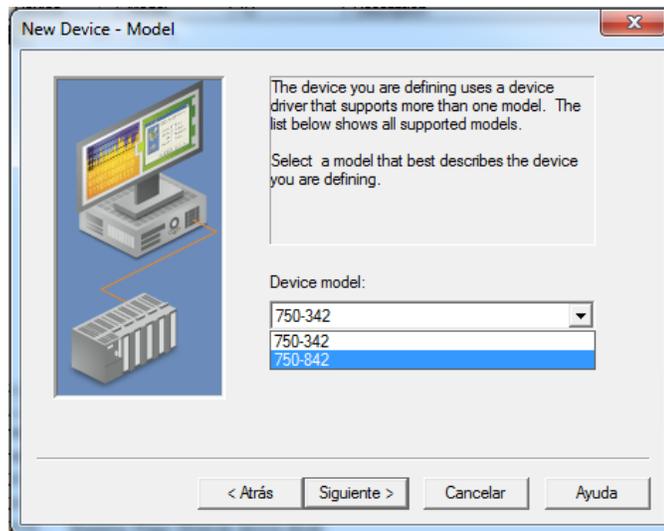


**Figura IV.137.** Elección del Driver del dispositivo para el canal.  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

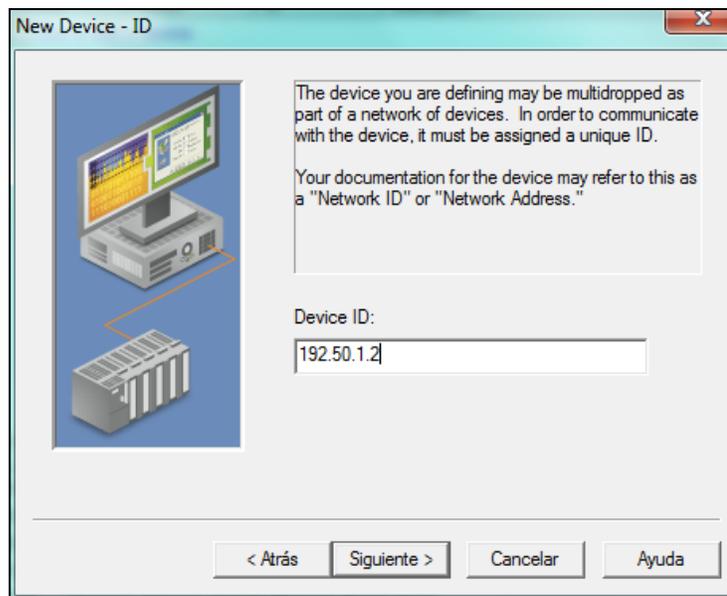


**Figura IV. 138.** Elección de la tarjeta de red.  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

2. Para crear un **Nuevo Dispositivo**, damos un nombre al dispositivo, elegimos el modelo que estamos utilizando, y la dirección IP del mismo.

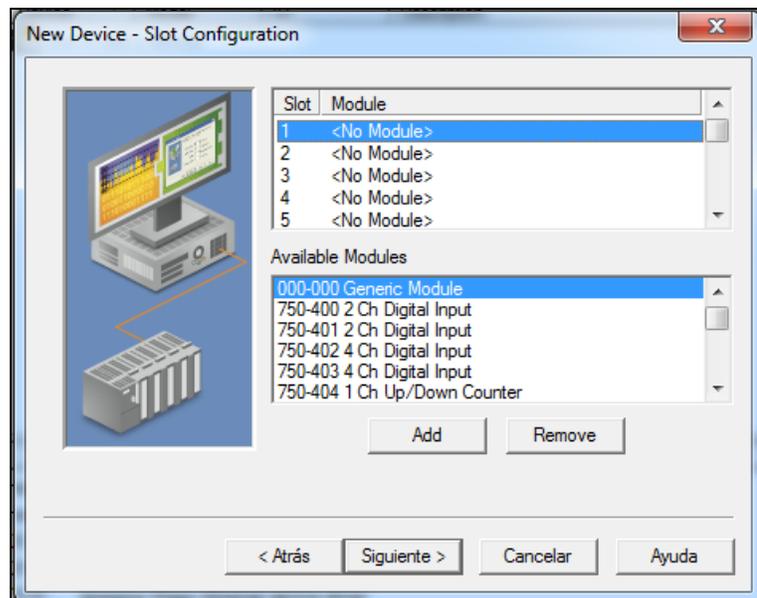


**Figura IV. 139.** Elección del modelo que se utiliza.  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*



**Figura IV. 140.** Asignación de la dirección IP del Wago.  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

3. En la **pantalla Slot Configuration** No escogemos ningún módulo de entrada salida, ya que en nuestro caso solo nos interesa leer y escribir en las direcciones Modbus.



**Figura IV. 141.** Ventana de configuración de módulos adicionales para Wago.  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

Ahora se crearan los tags de Wago, teniendo muy en cuenta el direccionamiento anteriormente visto y el programa cargado en el controlador.

### **Configuración de las Salidas de WAGO a Kepserver**

1. Como se observa en la siguiente figura, la variable **M** (%MX260.6) corresponde a una memoria interna del controlador que se usa en el Ladder, la misma se asigna a la **salida 6** (%QX0.5), y a la variable Modbus **MM** (%QX270.5), esta última es la que se va a leer desde el HMI vía Modbus.



**Figura IV. 142.** Configuración de las salidas en Wago IO Pro 32.  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

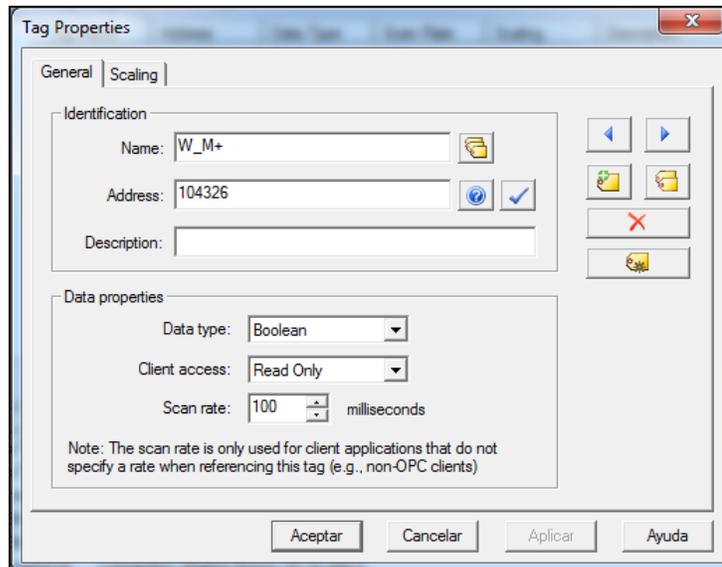
2. Desde Kepserver se puede acceder a la **memoria Modbus (%QX270.5)** con la **dirección 104326 (dirección Modbus)**, esta dirección se calcula con la siguiente formula:

$$No. bit = (palabra * 16) + No. bit en la palabra(1)$$

$$4326=(270*16)+6$$

Como hablamos anteriormente por la salida se le antepone el 10.

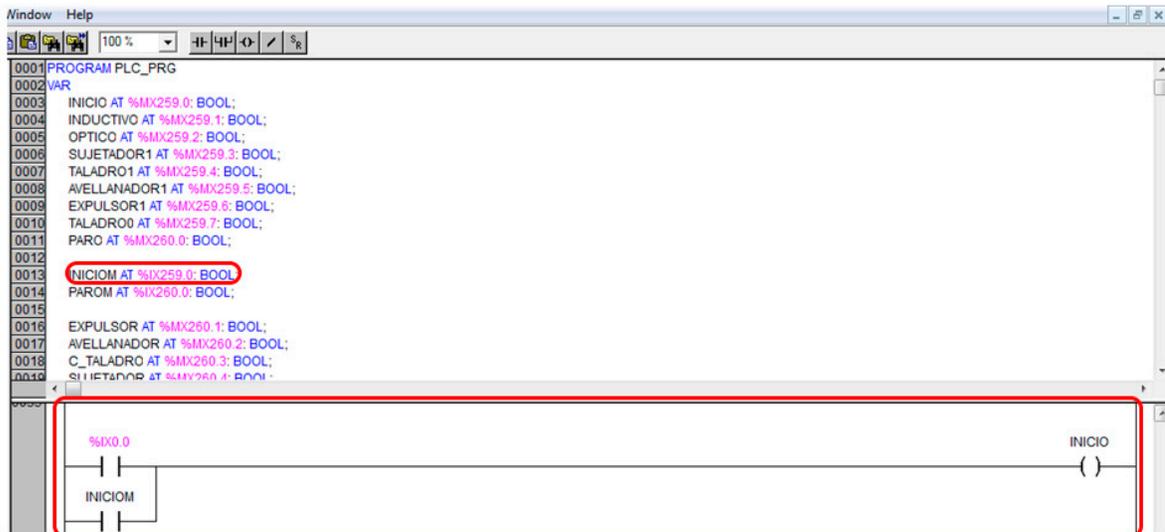
3. Con lo que finalmente para crear el tag en Kepserver de la salida para WAGO se tiene la siguiente asignación:



**Figura IV. 143.** Creación del tag de salida en Kepserver.  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

### Configuración de las Entradas de WAGO a Kepserver

1. Como se muestra a continuación, la variable **INICIOM** (%IX259.0) correspondiente a una memoria Modbus del controlador que junto con la entrada física (%IX0.0) comanda a la variable **INICIO** (%MX259.0) utilizada en el Ladder.

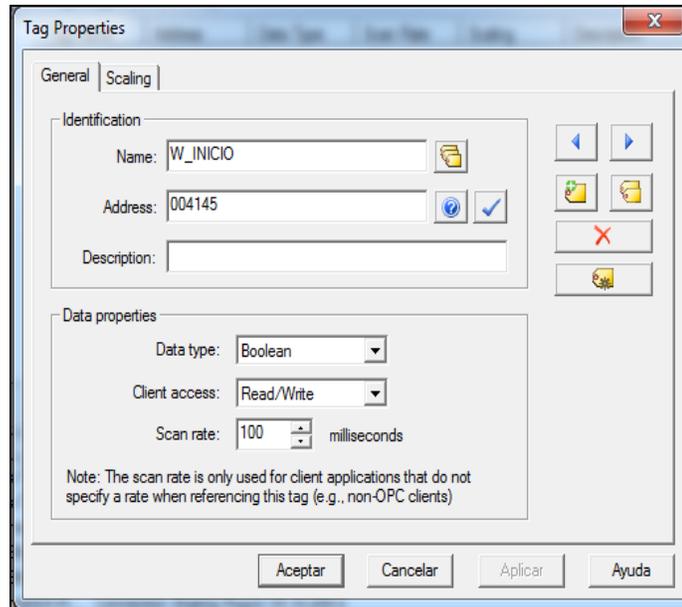


**Figura IV. 144.** Configuración de las entradas en Wago IO Pro 32.  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

2. Con la formula anteriormente vista, se calcula la dirección Modbus que se asigna en el tag de Kepserver para la Variable Inicio.

$$4145 = (259 * 16) + 1$$

3. Con la dirección ya obtenida, se crea el tag Inicio en Kepserver.



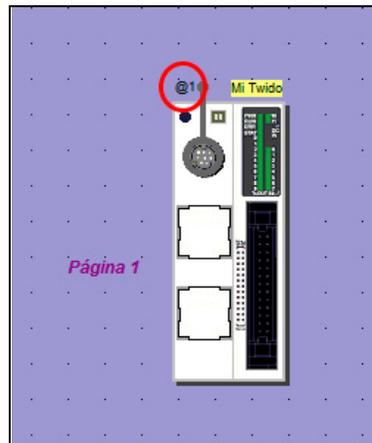
**Figura IV. 145.** Creación del tag de Inicio en Kepserver.  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

#### **4.5.1.3. Configuración TWDLMDA20DTK**

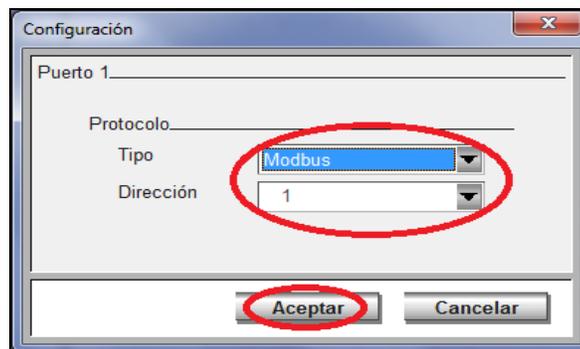
### **CONFIGURACIÓN DE LA PASARELA ETHERNET**

1. Para que exista comunicación entre la pasarela y el PLC Telemecanique, lo primero que tenemos que hacer es actualizar el firmware de TwidoSuite para la configuración del TwidoPort. La versión a la que debe actualizar el firmware de TwidoSuite es 3.4 o superior, se lo puede encontrar en la página de Schneider.

2. Ejecutar el programa **TwidoSuite**, elegir el **“Modo Programación”** y crear un nuevo proyecto, aquí pondremos el Nombre del mismo y el directorio donde se guardará.
3. Ahora se elige la pestaña **“Describir”**, donde seleccionamos el controlador que se utilizará (TWDLMDA20DTK), y se lo lleva al espacio de trabajo. Aquí se da doble click en **“@”** para configurar el puerto.

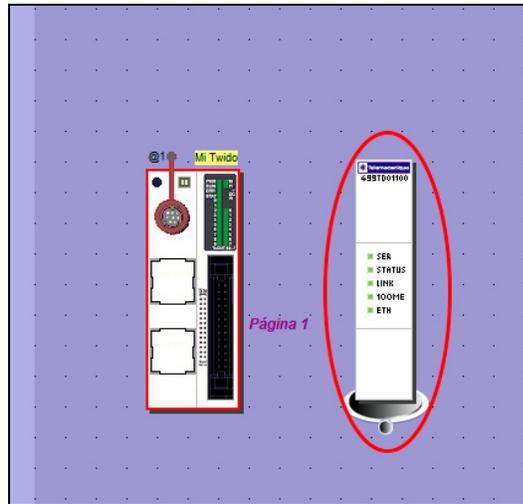


**Figura IV.146.** Tipo de controlador Twido que se utiliza TWDLMDA20DTK.  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*



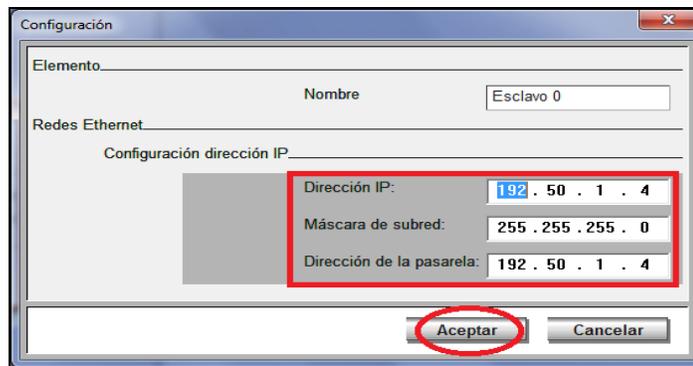
**Figura IV. 147.** Configuración del puerto de comunicación para el twido.  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

4. A continuación se selecciona la Pasarela Ethernet, en el catálogo de Twido. **Elementos Redes→Elementos Modbus→Interfase Ethernet**, y llevar hasta el espacio de trabajo.



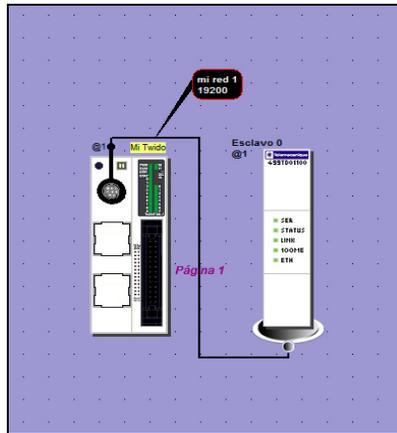
**Figura IV. 148.** Pasarela Ethernet en el espacio de trabajo.  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

5. Se da doble click en la pasarela para configurarla. Se cambia la dirección IP Estática, la Máscara de Subred y la Dirección de la pasarela a la red que se utiliza.

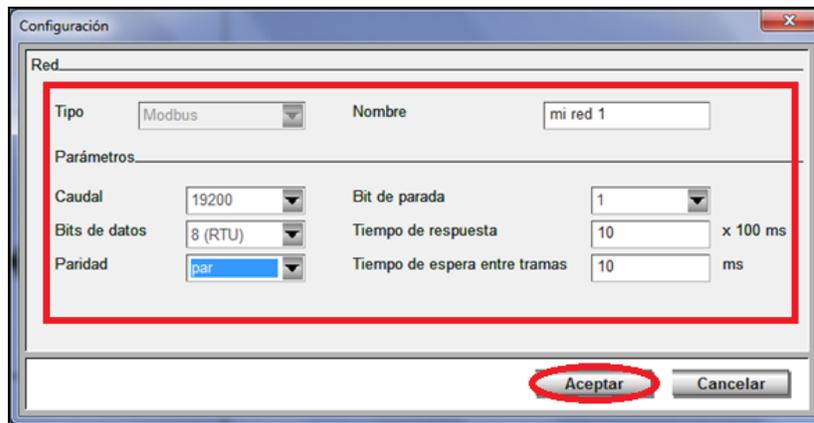


**Figura IV. 149.** Configuración de la dirección de la Pasarela.  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

6. En el espacio de trabajo, se arrastra desde el controlador al Twidoport, y se configura la red a los parámetros ocupados en la comunicación.

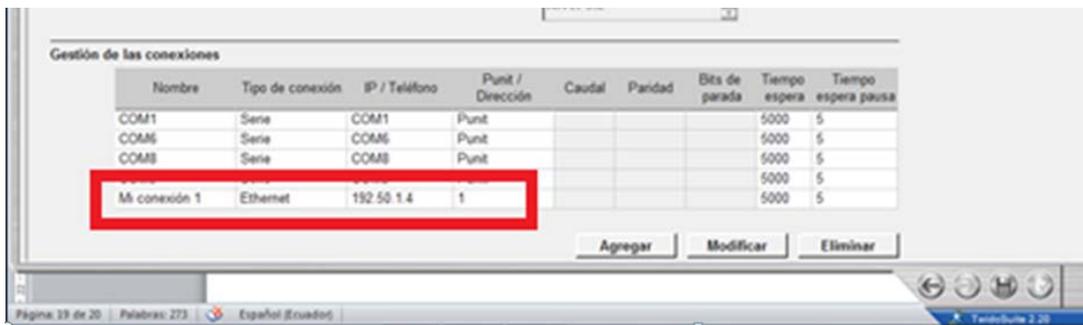


**Figura IV. 150.** Conexión entre Twido y la Pasarela en el espacio de trabajo.  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*



**Figura IV. 151.** Configuración de la red.  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

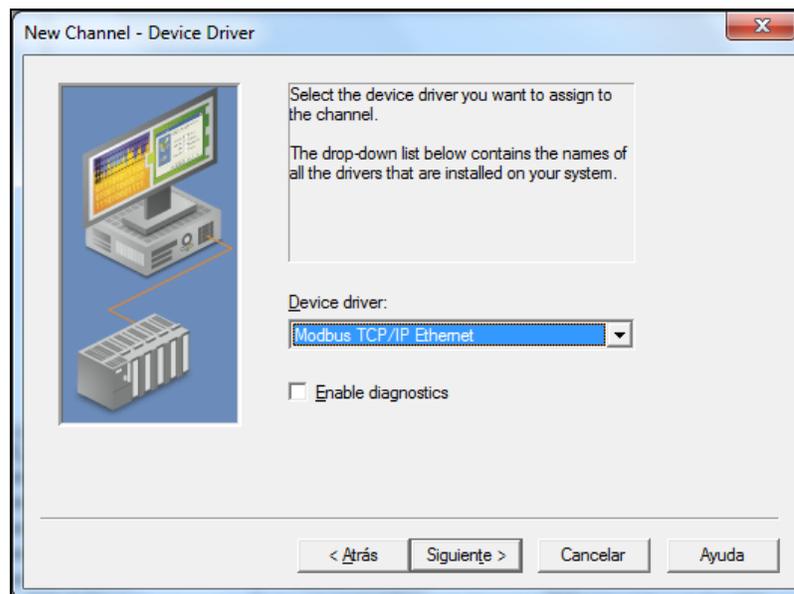
7. A continuación se selecciona **Preferencias** y se espera el análisis de puertos. Para agregar una nueva conexión, se da click en Agregar y se configura la nueva conexión, Nombre, Tipo de Conexión (Ethernet), IP/teléfono (Dirección de la pasarela) y Punit/Dirección (Dirección Modbus) y se guarda la nueva configuración en el controlador.



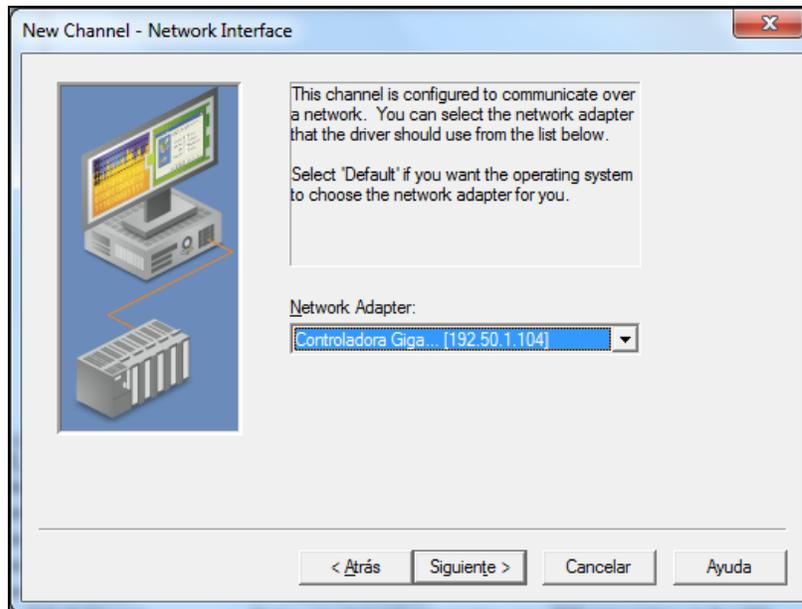
**Figura IV. 152.** Creación de la nueva conexión.  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

## CONFIGURACIÓN DEL OPC PARA TWIDO

1. Se crea un nuevo canal, como se ha visto anteriormente se agrega un nombre de canal, el driver (*Modbus TCP/IP Ethernet*) y la tarjeta de red.



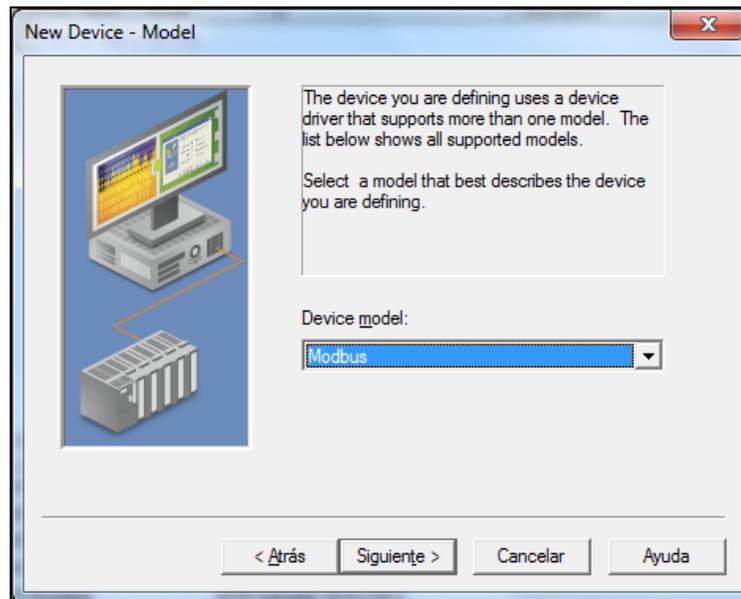
**Figura IV. 153.** Selección del driver para Twido.  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*



**Figura IV. 154.** Asignación de la tarjeta d red para Twido.

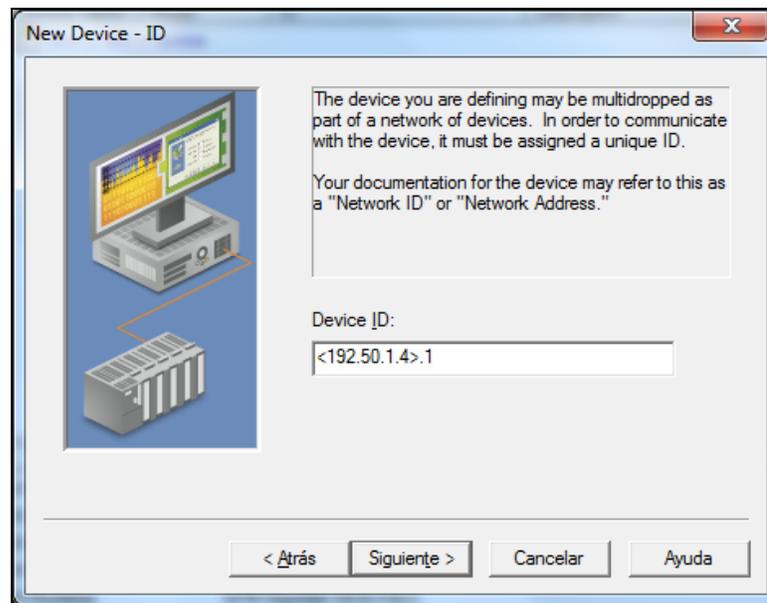
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

2. Para agregar el dispositivo a utilizar se siguen los pasos ya vistos, hasta la ventana Model, en donde se elige la opción Modbus para este caso, y el la ventana Device ID se coloca la dirección del Twido.



**Figura IV. 155.** Elección del modelo del dispositivo para Twido.

*Fuente: Leonardo Asqui e Iván Lema (Autores)*



**Figura IV. 156.** Asignación de la dirección IP del Twido.  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

Para la creación de los tags de Twido en Kepserver se debe tener en cuenta el direccionamiento Modbus, ya que los PLC Twido trabajan con este protocolo, aquí las variables o direcciones de memoria que ocupamos en el PLC se pueden leer o escribir dependiendo el caso, las **Mx para output o inputs coils**, **MWx son los Holding Registers**. En este caso no se ocupa ningún registro interno.

Así también se asignan las Mx a la dirección 00000x+1

Ejm:

%M0 a la dirección 000001

%M1 a la dirección 000002

%M12 a la dirección 000013

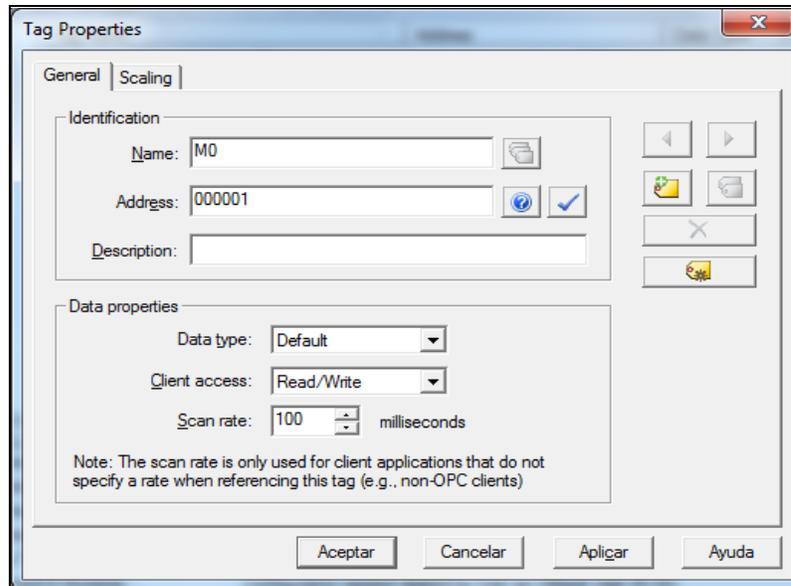
Igual que los holding registers

%MWO a la dirección 400001

%MW1 a la dirección 400002

%MW12 a la dirección 400013

3. Tomando en cuenta este direccionamiento, en Kepserver se crean los nuevos tags de Twido, para entradas y salidas.

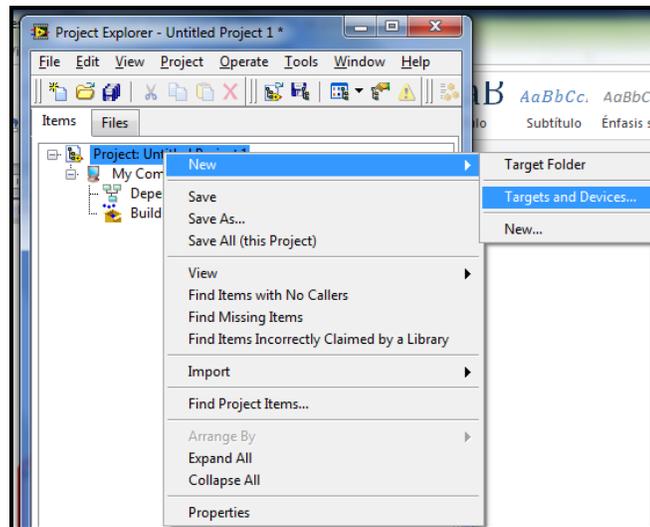


**Figura IV. 157.** Creación de los tags del Twido.  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

#### **4.5.1.4. Configuración FieldPoint 16-01**

Para la utilización del FP-1601, se lo hará mediante Labview, ya que permite manejar directamente con el OPC mediante la asignación de tags y el uso de sus propias librerías.

1. Ejecutamos LabVIEW, y abrimos o creamos un proyecto.
2. Dar click derecho en **“My Project”**, seleccionar **“New”**, y damos click en **“Targets and Devices...”**



**Figura IV. 158.** Creación de un nuevo proyecto.

*Fuente: Leonardo Asqui e Iván Lema (Autores)*

3. En este caso debemos tener conectado el FieldPoint y tener debidamente asignada la dirección IP.

Seleccionar en: *“Existing Target or device”*, luego *“Discover an existing targets and devices”*, después dar click en: *“FieldPoint Network Modules”*, seleccionar el FieldPoint con la dirección IP que ha sido configurado y dar click en *“OK”*.

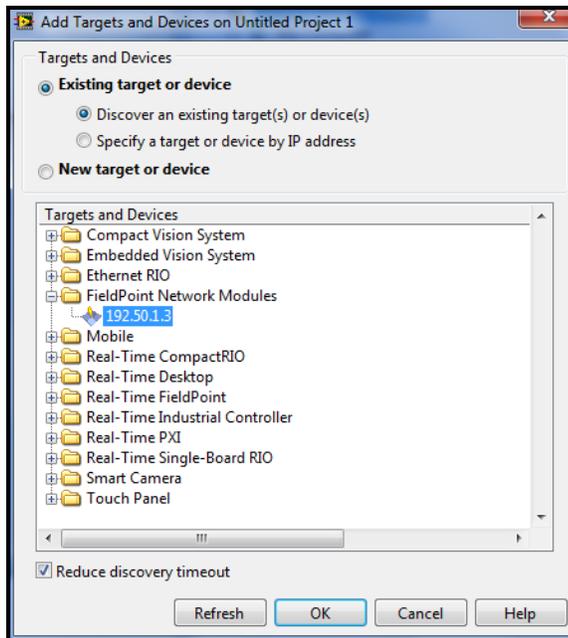


Figura IV. 159. Selección del dispositivo con la red a utilizar.  
Fuente: Leonardo Asqui e Iván Lema (Autores)

4. Se podrá observar el FieldPoint con dispositivos conectados al FP en el proyecto.

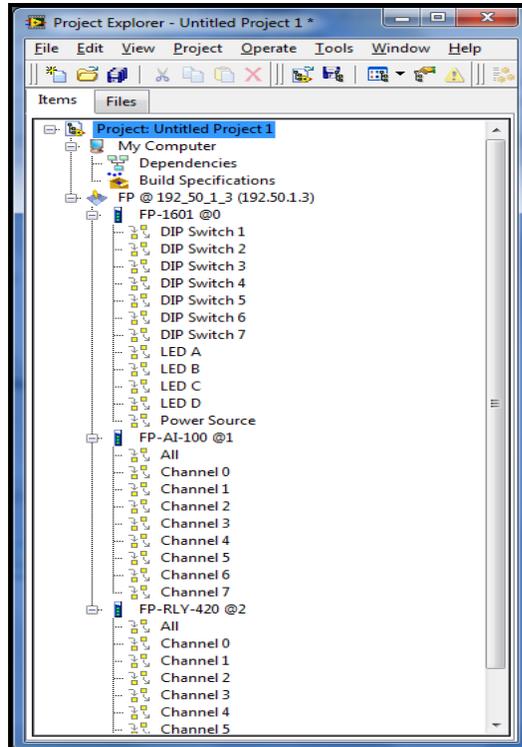
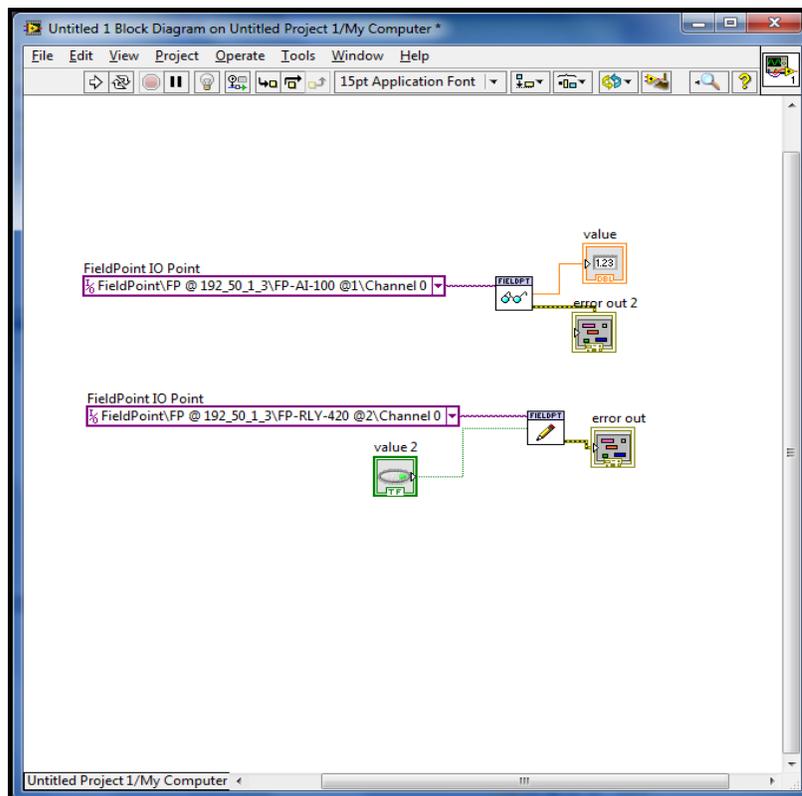


Figura IV. 160. Vista del FieldPoint y sus dispositivos.  
Fuente: Leonardo Asqui e Iván Lema (Autores)

5. Para poder programar y controlar las salidas de los dispositivos asociadas al FieldPoint; vamos a ocupar LabVIEW y las tags asociadas al proyecto.
6. Abrir o crear un VI en el proyecto.
7. Arrastrar los canales de los dispositivos asociados al FP que se van a ocupar al diagrama de bloque.

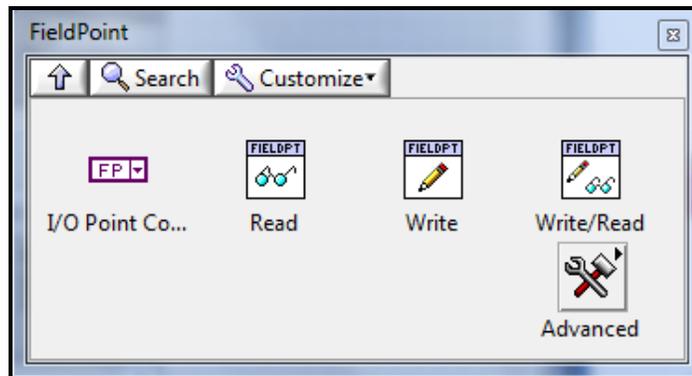
Se podrá observar que para el dispositivo de entradas analógicas se va a crear con un bloque de lectura; y para el dispositivo de salidas por relé se va a crear con un bloque de escritura el que se podrá activar mediante una señal booleana.



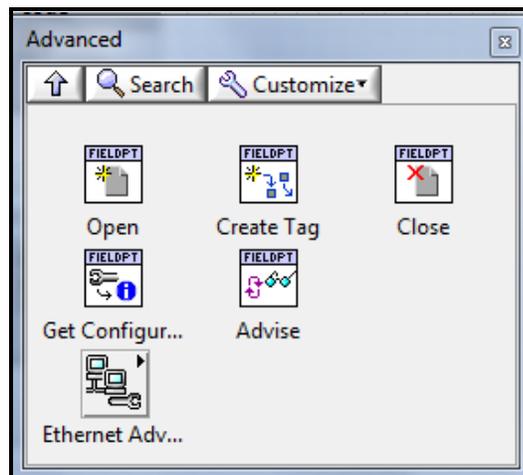
**Figura IV. 161.** Bloques del dispositivo de entradas analógicas en Labview.  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

Existen librerías o bloques de función para la utilización del FieldPoint en el diagrama de bloques de LabVIEW, para acceder a estos bloques debemos:

Podremos utilizar cualquier función de la paleta del FieldPoint.



**Figura IV. 162.** Paleta de FieldPoint.  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*



**Figura IV. 163.** Paleta de FieldPoint (Advanced).  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

## 4.5.2. LABVIEW

El HMI se ha implementado en Labview 2011, puesto que es un software de fácil manejo y ofrece varias opciones para crear una interfaz accesible y de fácil entendimiento tanto para el programador como para el usuario; también se utilizó esta versión porque el FP-1601 no tiene soporte para un Labview más actual.

Por motivo de estudio nos centraremos en la utilización del I/O Server y la utilización de las librerías de la Base de Datos; pero antes explicaremos el funcionamiento del HMI.

### 4.5.2.1. *Modo de operación del HMI*

Está dividido en varias pantallas donde va indicar el funcionamiento de cada proceso.

***Envasadora de líquidos:*** Consta de controles remotos para el encendido y el apagado del proceso, un botón de emergencia, e indicadores del funcionamiento de la envasadora de cada sensor y activación de los actuadores, como un botón para activar la comunicación con el PLC S7-1200, y un contador para decir el número de latas envasadas y con tapas.

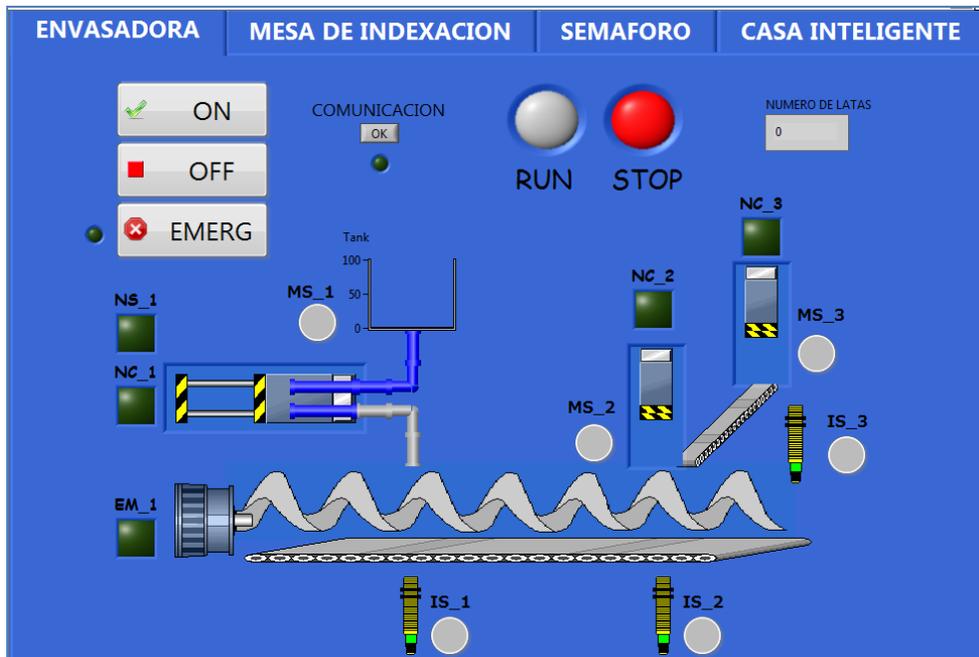


Figura IV. 164. HMI Módulo Envasadora (Panel Frontal)  
Fuente: Leonardo Asqui e Iván Lema (Autores)

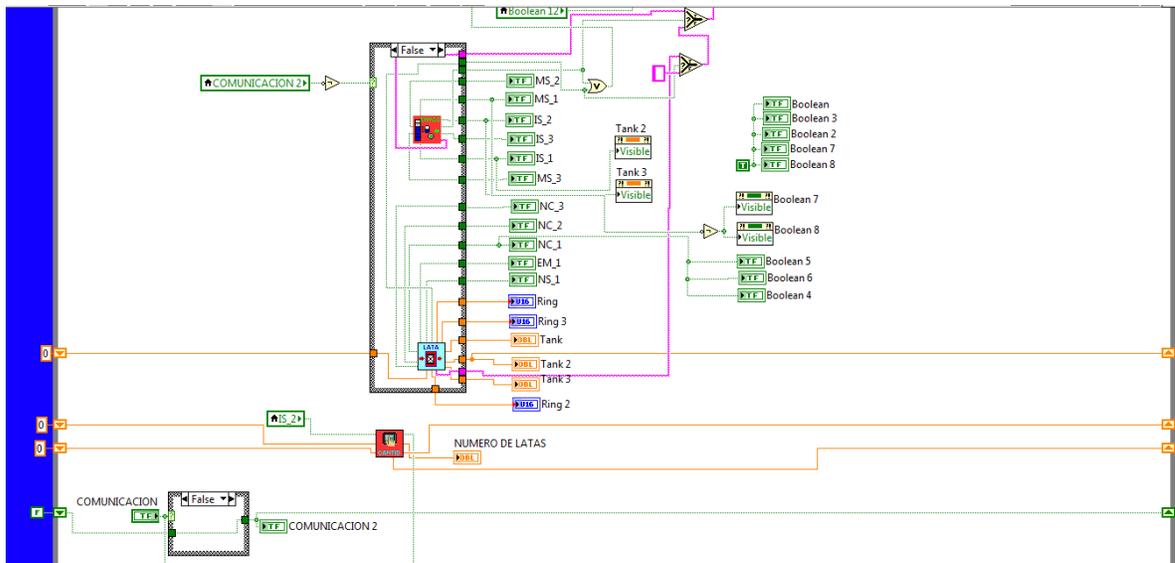
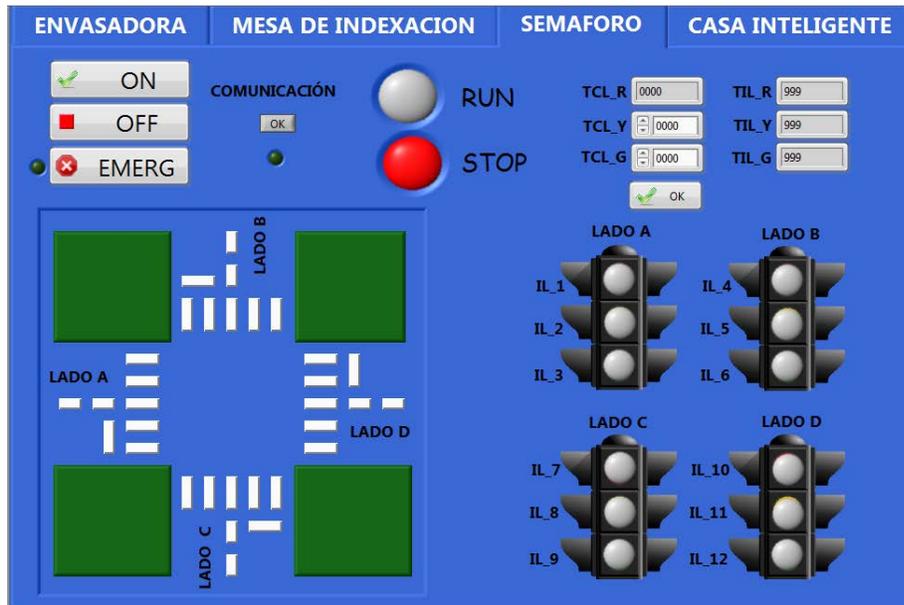


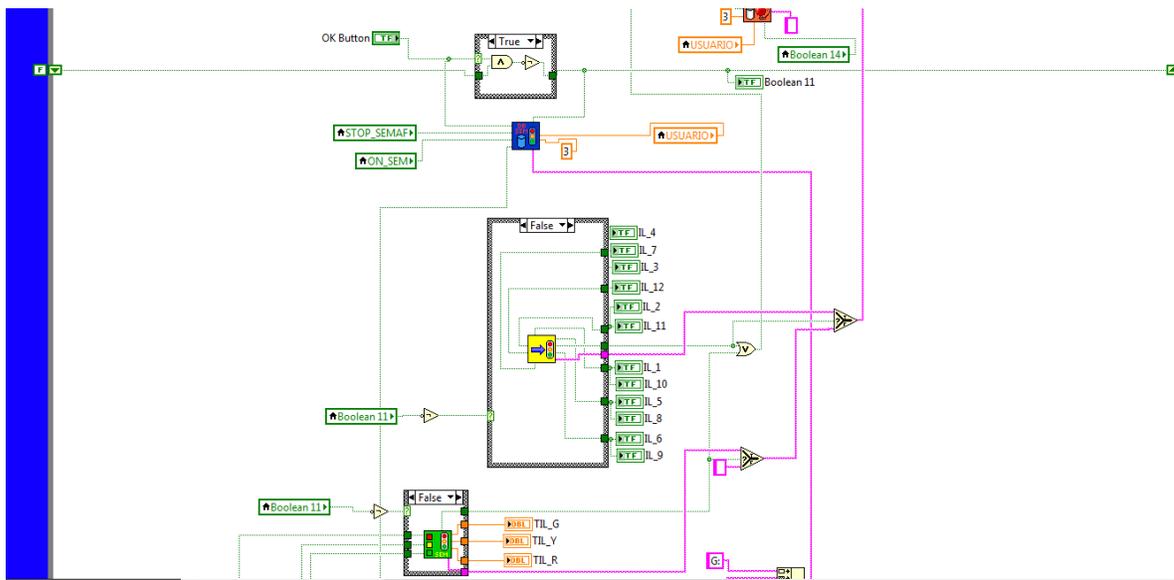
Figura IV. 165. Diagrama de Bloques Módulo Envasadora  
Fuente: Leonardo Asqui e Iván Lema (Autores)



**Semáforo:** Al igual que los otros procesos tiene los mismos botones de acceso remoto, y una visualización del funcionamiento, y la posibilidad de cambiar los tiempos de las luces del semáforo.



**Figura IV. 168.** HMI Semáforo (Panel Frontal)  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*



**Figura IV. 169.** Diagrama de Bloques Semáforo  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

**Casa Automatizada:** Al igual tenemos un acceso al funcionamiento y panel de visualización, así como un control de las luces, un control de usuarios para la activación de las alarmas, así como un control del garaje.



Figura IV. 170 HMI Casa Automatizada (Panel Frontal)

Fuente: Leonardo Asqui e Iván Lema (Autores)

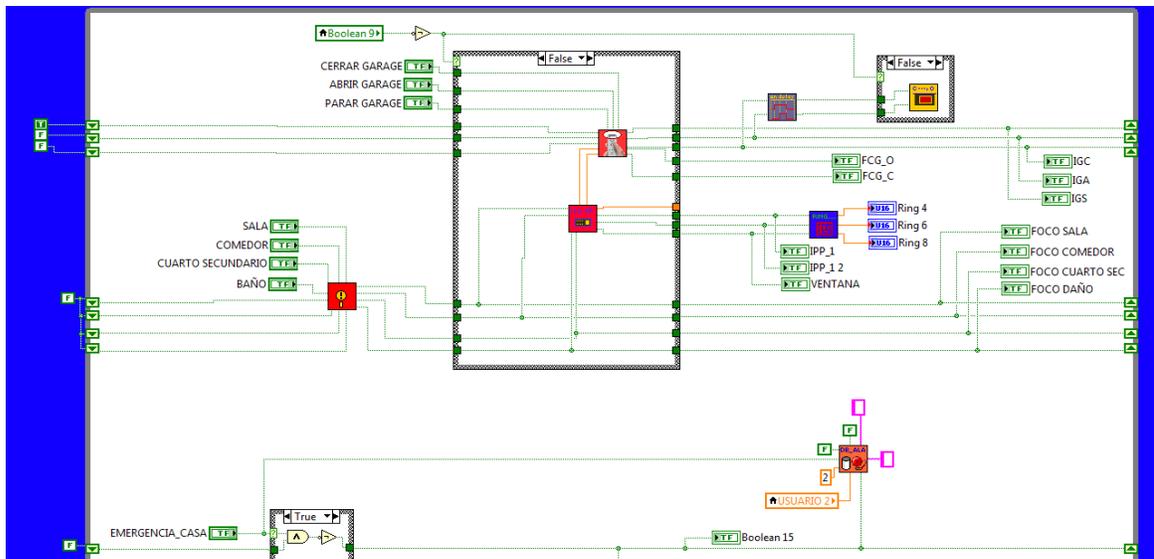
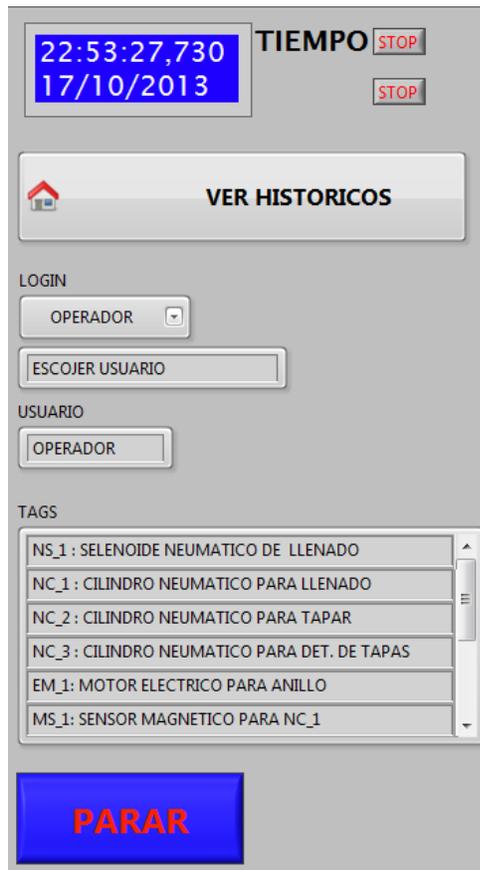


Figura IV. 171. Diagrama de Bloques Casa Automatizada

Fuente: Leonardo Asqui e Iván Lema (Autores)

Ahora indicaremos el funcionamiento de otra pantalla.



**Figura IV. 172.** Pantalla adyacente de Tiempo, Históricos, Usuarios y Tags.

*Fuente: Leonardo Asqui e Iván Lema (Autores)*

Tenemos en esta pantalla un indicador de la hora y la fecha, un pulsador para acceder a los históricos, un manejo de usuarios, un indicador de tags para poder observar que hace cada tag que está en cada uno de los diferentes procesos y el botón de para que detendrá todo el HMI.

#### 4.5.2.2. *Enlace del OPC con LabVIEW*

Ahora indicaremos como utilizar el I/O Server de LabVIEW.

Se creará una interfaz en LabVIEW para las etiquetas (tags) OPC llamadas I/O Server. El I/O Server automáticamente actualiza LabVIEW con los valores de las etiquetas actuales en el rango que usted especifica.

1. Abrimos un proyecto en Labview
2. Click derecho *“My Computer”*
3. Seleccionar *“New”*
4. Click en *“I/O Server”*

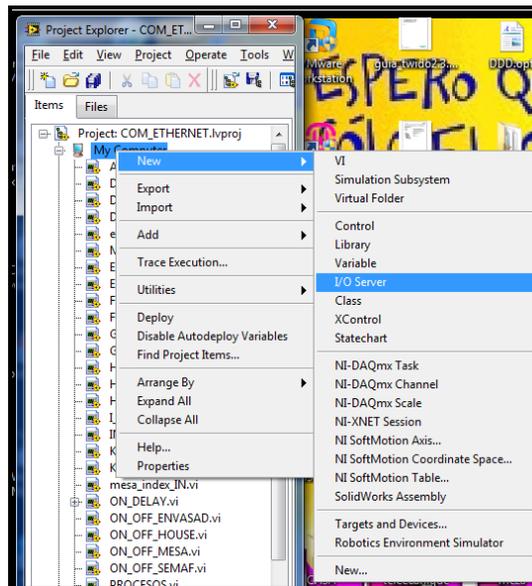
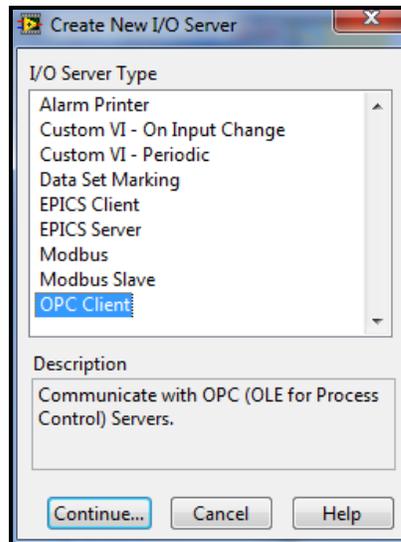


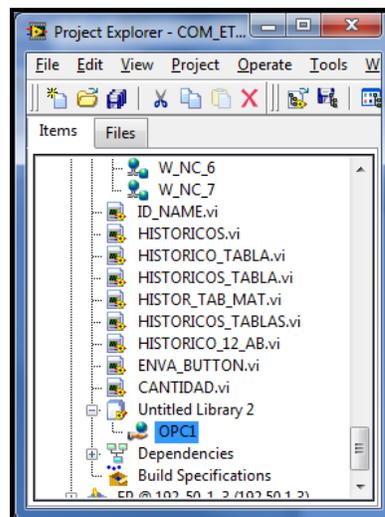
Figura IV. 173. Creación de I/O Server en Labview.  
Fuente: Leonardo Asqui e Iván Lema (Autores)

5. Seleccionar *“OPC Client”* y click en *“Continue...”*



**Figura IV. 174.** Selección OPC Client.  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

6. Seleccionamos el OPC Server que estamos utilizando y click en **“OK”**
7. Aparecerá un mensaje el que nos dirá que tiene que estar corriendo el OPC Server, si es así tenemos que hacer click **“OK”**
8. Ahora se podrá observar las nuevas variables del I/O Server en el proyecto.

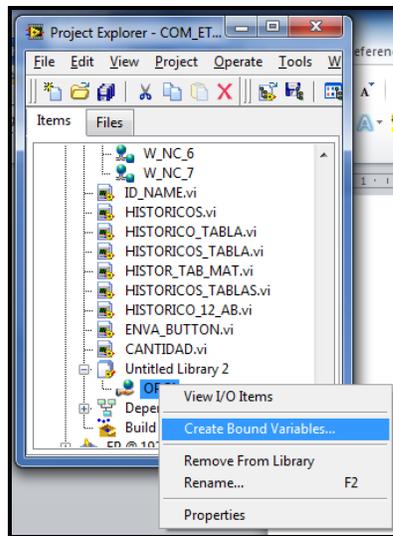


**Figura IV. 175.** Variables I/O Server utilizadas en el proyecto.  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

***Crear variables compartidas que conectan las etiquetas opc al I/O Server:***

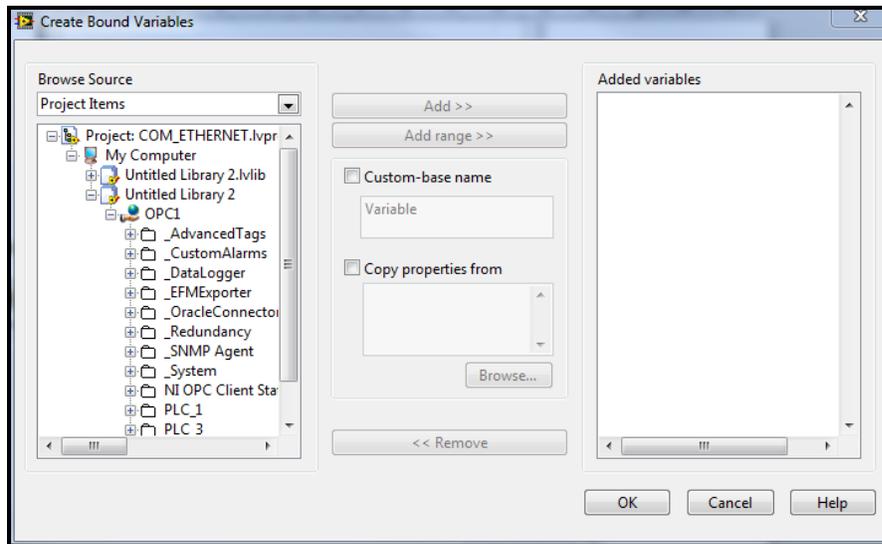
Se crean las variables compartidas, las cuales están limitadas en las etiquetas del OPC, dándole el acceso nativo en LabVIEW a los datos del PLC. Con la variable compartida, usted puede compartir datos a través de los usos de LabVIEW en una sola computadora o a través de la red.

1. Click derecho en el I/O Server creado y click en ***“Create Bound Variables”***



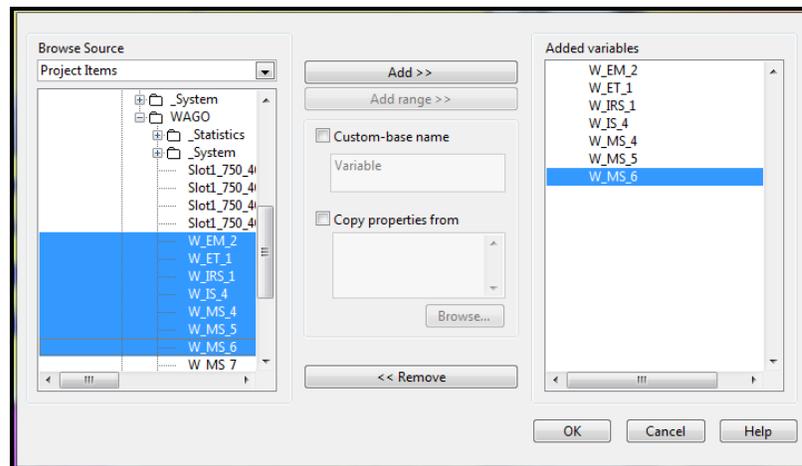
**Figura IV. 176.** Creación de una nueva variable  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

2. Seleccionar librería que se ha creado conjuntamente con el OPC, seleccionar el nombre del canal del OPC, y el dispositivo del OPC



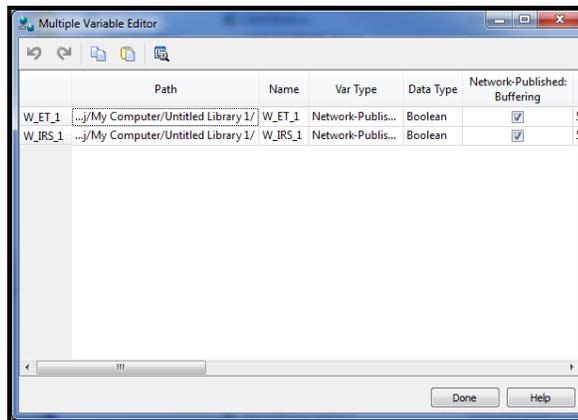
**Figura IV. 177.** Selección del canal y dispositivo utilizados.  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

3. Seleccionar todas las tags del OPC que vamos a ocupar, hacemos click en **“Add>>”**, observamos como las variables han sido agregadas y click en **“OK”**.



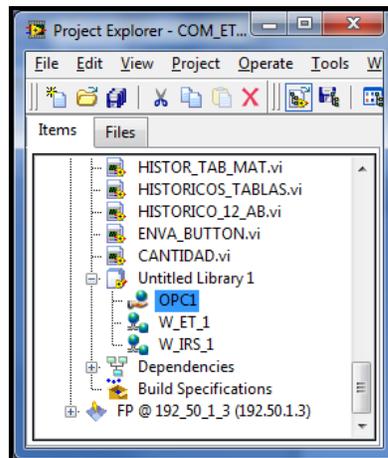
**Figura IV. 178.** Adición de las nuevas variables.  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

4. Podemos observar cómo se han creado todas las variables y hacer click en **“Done”**.



**Figura IV. 179.** Nuevas variables adicionadas.  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

5. Podemos observar cómo se han creado las nuevas variables en el proyecto.



**Figura IV. 180.** Nuevas variables mostradas en el proyecto.  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

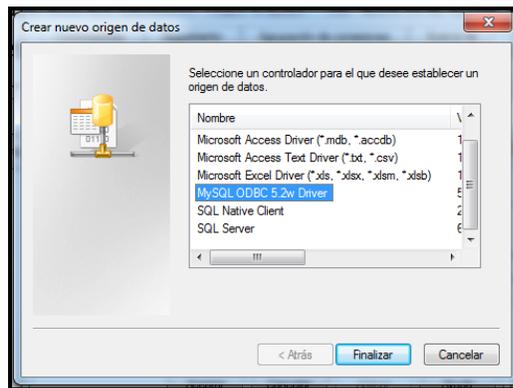
#### 4.5.2.3. Enlace de la Base de Datos con LabVIEW

Para el enlace de la Base de datos primeramente tenemos que descargar el conector e instalarlo, en esta tesis, para la base de datos fue implementada con MySQL por lo que tenemos que instalar el ODBC para MySQL.

Después de haber hecho esto procedemos a realizar los siguientes pasos, considerando que en LabVIEW existe un conflicto con el ODBC para 32 bits y de 64 bits

ya que LabVIEW tiene un problema con la Arquitectura. Para mayor información puede revisar el MANUAL DE LABVIEW en la sección de Enlace de la base de datos con LabVIEW.

1. Ingresar al ***“Panel de Control”, “Sistema y seguridad”, “Herramientas Administrativas” y “Origenes de datos ODBC”***.
2. Hacer click en ***“Agregar”*** en ***“DNS de usuario”***.
3. Seleccionar ***“MySQL ODBC 5.2w Driver”*** y dar click en ***“Finalizar”***.



**Figura IV. 181.** Conexión de ODBC con Labview.  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

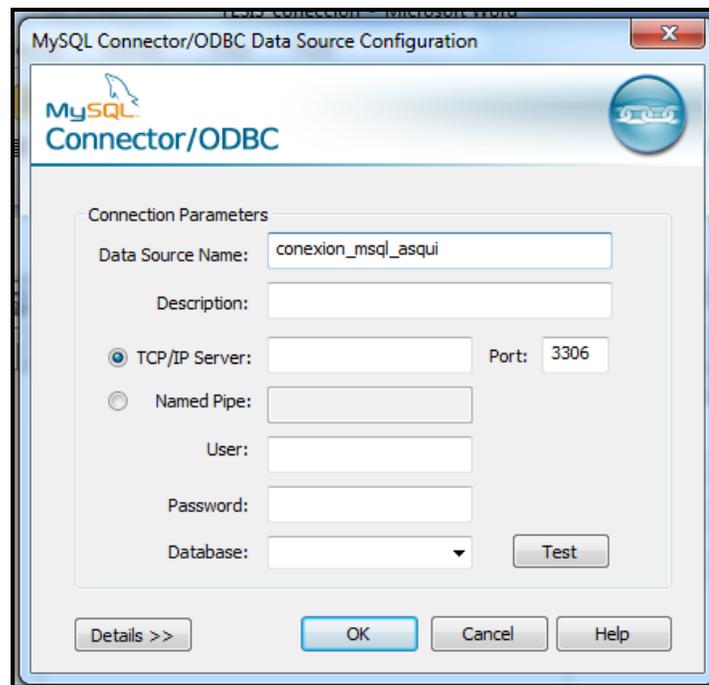
4. Llenar los datos como se muestra en la figura.

*Data Source Name:* El nombre que queremos poner a la conexión.

*TCP/IP Server:* Ponemos “localhost” ya que la base de datos estará en nuestra computadora caso contrario pondríamos una dirección ip, la dirección ip del servidor remoto.

*User:* Nombre del Usuario

*Password:* La contraseña del Usuario.



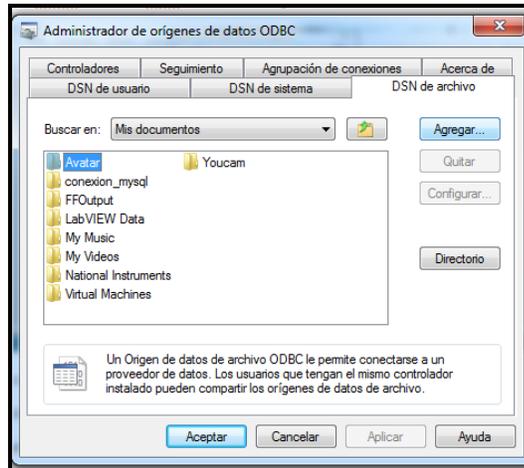
**Figura IV. 182.** Configuración de parámetros para la conexión.  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

5. Dar click en ***“Test”***. Para probar la conexión.
6. Dar click en ***“Ok”***
7. Dar click en ***“Aceptar”***

### ***CREACION DE UN ARCHIVO \*.DNS***

Antes de crear un archivo DNS debíamos haber generado una base de datos, para crear una base de datos podemos leer el manual de creación de una base de datos con WAMPSEVER, así como haber creado la conexión con MySQL y este activado. Debemos tomar en cuenta que la creación del archivo DNS se debe crear con el obdcad32.exe de acuerdo al sistema operativo.

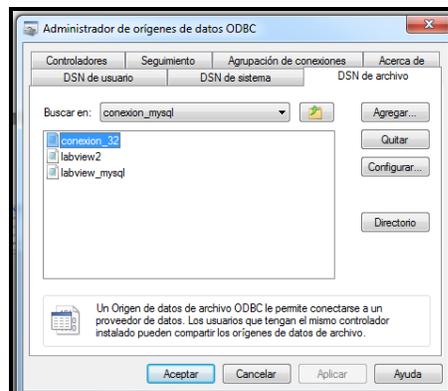
1. Abrir el obdcad32.exe dar click en ***“DNS ARCHIVO”***, luego escoger ***“Agregar”***.



**Figura IV. 183.** Creación del archivo DNS.

*Fuente: Leonardo Asqui e Iván Lema (Autores)*

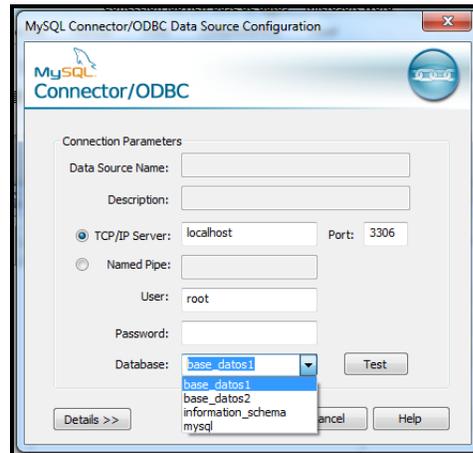
2. Seleccionar ***“My SQL ODBC 5.2w Driver”*** y click en ***“Siguiente”***.
3. Dar click en ***“Examinar”***.
4. Escoger la carpeta para guardar el archivo .dns
5. Dar un nombre y click en ***“Guardar”***.
6. Luego dar click en ***“Siguiente”***.
7. Dar click en ***“Finalizar”***
8. Buscar la ubicación de la conexión creada.
9. Seleccionar la conexión del archivo DNS y dar click en ***“Configurar”***



**Figura IV. 184.** Configuración del archivo DNS.

*Fuente: Leonardo Asqui e Iván Lema (Autores)*

10. Escoger la base de datos con la que se va hacer la conexión y completar los demás datos. Para poder establecer la conexión con la base de datos debe estar bien escrita TCP/IP Server, y estar activo la base de datos.



**Figura IV. 185.** Configuración final para la conexión.  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

11. Realizar un **“Test”** para probar la conexión.
12. Dar click en **“Aceptar”**

### **CONEXIÓN A LABVIEW VIA .UDL**

1. Abrir LabVIEW.
2. Dar click en **“Tools”** y **“Create Data Link...”**

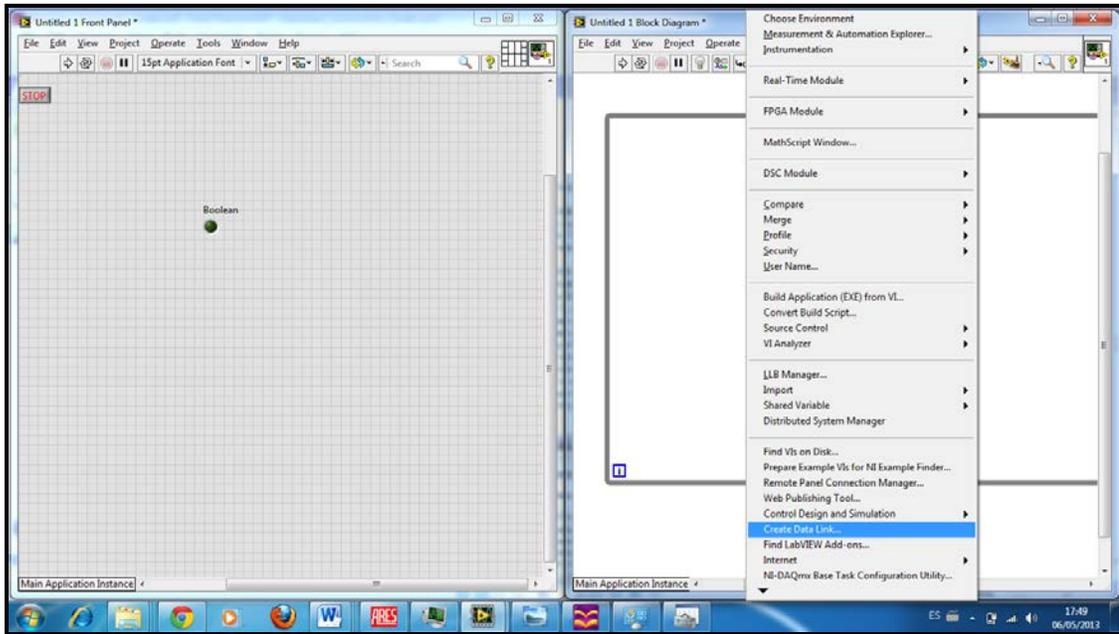


Figura IV. 186. Create Data Link para conexión por UDL.  
Fuente: Leonardo Asqui e Iván Lema (Autores)

3. Escoger **“Proveedor”**, Seleccionar **“Microsoft OLE DB Provider for ODBC Drivers”**. Y click **“Siguiente”**.

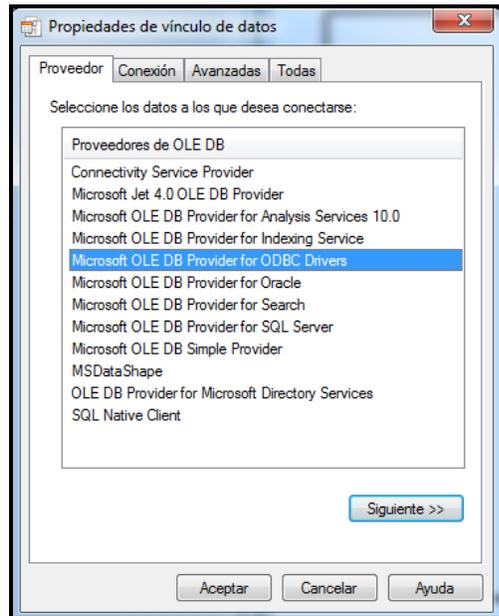


Figura IV. 187. Selección del proveedor para ODBC.  
Fuente: Leonardo Asqui e Iván Lema (Autores)

4. Seleccionar la conexión creada con el ODBC y poner el nombre del usuario.
5. Probar la conexión, para observar si es satisfactoria y dar click en ***“Aceptar”***.
6. Escoger el directorio donde se va a guardar la conexión, damos click en ***“Aceptar”***.
7. Después podemos observar si la conexión fue creada satisfactoriamente.

## **NOTAS CAPÍTULO IV**

- (1) WAGO-I/O-SYSTEM 750ETHERNET TCP/IP Programmable Fieldbus Controller. Pág.  
90

## **CAPÍTULO V**

### **RESULTADOS**

#### **5.1. INTRODUCCION.**

En este capítulo se presentara los resultados de la encuesta realizada a los alumnos salientes de la materia de redes industriales, con el fin de comprobar la hipótesis planteada.

Se pretende determinar la satisfacción obtenida en cuanto a la mejora del control y monitoreo de los distintos procesos modulares, utilizándose para ello la ponderación por convenio.

#### **5.2. ANALISIS DE LA ENCUESTA.**

Tomando en cuenta que la encuesta se realizara a los estudiantes de la materia de redes industriales, se ha decidido realizarla a una muestra, según la siguiente ecuación:

$$\eta = \frac{N(z^2)(s^2)}{(N-1)e^2 + z^2s^2} \quad (1)$$

Dónde:

- $\eta$ : Tamaño de la muestra.
- N: Tamaño de la población.
- z: Nivel de confianza.
- s: Varianza.  $S = pxq$
- e: Error muestral.

Para determinar el tamaño de la muestra se utilizara:

- $z=1.96$ , para un nivel de confianza de 95%.
- $S=0.5$ .
- $e= 15\%$
- donde la población es  $N = 36$

Entonces tenemos:

$$\eta = \frac{N(z^2)(s^2)}{(N-1)e^2 + z^2s^2}$$
$$\eta = \frac{36(1.96^2)(0.5^2)}{(36-1)0.15^2 + 1.96^20.5^2}$$

$$\eta = 19.7$$

Como podemos ver se va realizar la encuesta a 20 personas de la población, con un 95% de confianza y con un error muestral del 15%, significando esto que si los resultados de la encuesta dan un total de 30 personas satisfechas en la mejora del control y monitoreo en la red, podrían estar realmente satisfechos unos 25 a 35 personas.

A continuación se muestra los resultados obtenidos de la encuesta, utilizaremos la ponderación por convenio para obtener un nivel de satisfacción (Is.), según la fórmula:

$$I_s = \frac{P_a \times A\% + P_b \times B\% + P_c \times C\% + \dots + P_z \times Z\%}{P_z \times 100} \quad (2)$$

Dónde:

- Pa, Pb, Pc, ..., Pz: ponderaciones de las respuestas a, b, c, ..., z
- A%, B%, C%, ..., Z%: Porcentajes de personas que optaron por la respuesta a, b, c, ..., z respectivamente
- Pz: ponderación más alta.

A cada respuesta se le asignó una ponderación de esta forma la respuesta variara entre 0 y 1 siendo:

- 0: El 100% de la población insatisfecha
- 1: El 100% de la población satisfecha

Encuesta:

1. ¿Cree Ud. que la programación de varios Controladores programables al estar todos en la red Ethernet es?

Tabla V.XXXV. Tabla de resultados y ponderaciones pregunta 1

ESCALA	PERSONAS	PORCENTAJE %	PONDERACION
Nada fácil	3	15	0
Complejo	4	20	1
Fácil	11	55	2
Muy fácil	2	10	3

Fuente: Realizado por Leonardo Asqui e Iván Lema (Autores).

$$I_s = \frac{15 \times 0 + 20 \times 1 + 55 \times 2 + 1 \times 3}{300} = 0.53$$

El índice de satisfacción para esta pregunta es 0.53 dándonos a notar que la programación de controladores en la red no es ni muy fácil ni muy compleja.

2. ¿Qué tan fácil es el diseño de interfaces de usuario (HMI) y creación de tags en un OPC server al estar todos los dispositivos en la red Ethernet?

**Tabla V.XXXVI. Tabla de resultados y ponderaciones pregunta 2**

ESCALA	PERSONAS	PORCENTAJE %	PONDERACION
Nada fácil	1	5	0
Fácil	5	25	1
Muy fácil	14	70	2

*Fuente: Realizado por Leonardo Asqui e Iván Lema (Autores).*

$$I_s = \frac{5 \times 0 + 25 \times 1 + 70 \times 2}{200} = 0.825$$

Como podemos ver la satisfacción para esta pregunta es buena, en donde nos indica que la creación de HMIs y tags en un OPC Server en la red es muy fácil.

3. ¿Cree Ud. que agregar PLCs, computadoras a la red Ethernet y configurar sus parámetros de red como dirección IP, mascara de subred entre otros es?

**Tabla V.XXXVII. Tabla de resultados y ponderaciones pregunta 3**

ESCALA	PERSONAS	PORCENTAJE %	PONDERACION
Nada fácil	3	15	0
Complejo	6	30	1
Fácil	8	40	2
Muy fácil	3	15	3

*Fuente: Realizado por Leonardo Asqui e Iván Lema (Autores).*

$$I_s = \frac{15 \times 0 + 30 \times 1 + 40 \times 2 + 15 \times 3}{300} = 0.51$$

De igual forma el agregado de dispositivos a la red la población la considera ni complicado ni muy fácil manteniéndose en un nivel medio de complejidad.

¿Monitorear y controlar al mismo tiempo varios procesos modulares pertenecientes a la red Ethernet mediante un HMI, que controlarlos de forma separada es?

**Tabla V.XXXVIII. Tabla de resultados y ponderaciones pregunta 4**

ESCALA	PERSONAS	PORCENTAJE %	PONDERACION
Mucho mejor	12	60	2
Mejor	8	40	1
Peor	0	0	0

*Fuente: Realizado por Leonardo Asqui e Iván Lema (Autores).*

$$I_s = \frac{0 \times 0 + 40 \times 1 + 60 \times 2}{200} = 0.8$$

Notamos que para el control y monitoreo de procesos modulares en la red tenemos un nivel de satisfacción de 0.8, en donde la población considera mucho mejor el control y monitoreo dentro de la red.

4. ¿La recolección, análisis y almacenamiento de datos en la red Ethernet es?

**Tabla V.XXXIX. Tabla de resultados y ponderaciones pregunta 5**

ESCALA	PERSONAS	PORCENTAJE %	PONDERACION
Nada fácil	1	5	0
Complejo	1	5	1
Fácil	6	30	2
Muy fácil	12	60	3

*Fuente: Realizado por Leonardo Asqui e Iván Lema (Autores).*

$$I_s = \frac{5 \times 0 + 5 \times 1 + 30 \times 2 + 60 \times 3}{300} = 0.81$$

Como podemos ver la población se encuentra satisfecha con el almacenamiento y análisis de los datos dentro de la red dando un índice de satisfacción de 0.81.

5. ¿Al estar controlando varios procesos modulares al mismo tiempo con qué frecuencia el sistema no responde?

**Tabla V.XL. Tabla de resultados y ponderaciones pregunta 6**

ESCALA	PERSONAS	PORCENTAJE %	PONDERACION
Nunca	15	75	2
Casi nunca	5	25	1
Siempre	0	0	0

*Fuente: Realizado por Leonardo Asqui e Iván Lema (Autores).*

$$I_s = \frac{0 \times 0 + 25 \times 1 + 75 \times 2}{200} = 0.875$$

Notamos que el sistema funciona de forma adecuada considerándose una buena confiabilidad y disponibilidad ya que el sistema en su mayor parte se encuentra disponible y confiable para el usuario al no colgarse o no responder.

6. ¿La comunicación con un controlador perteneciente a la red Ethernet es?

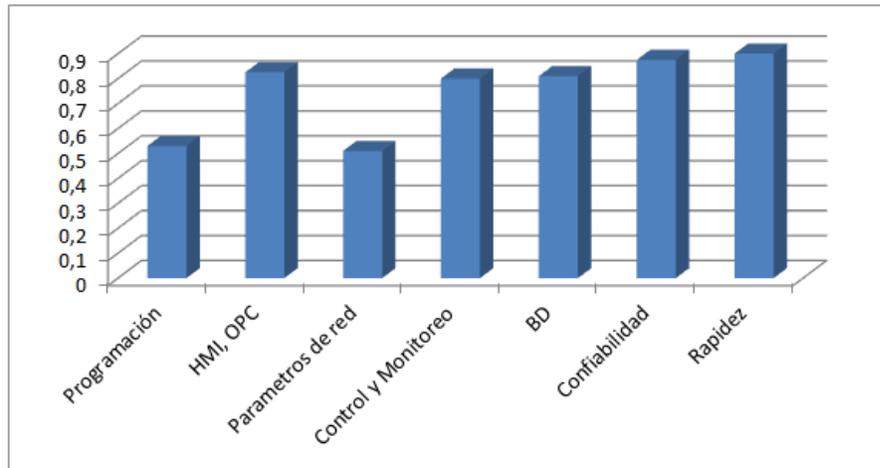
**Tabla V.XLI. Tabla de resultados y ponderaciones pregunta 7**

ESCALA	PERSONAS	PORCENTAJE %	PONDERACION
Muy rápido	17	85	2
Rápido	2	10	1
Lento	1	5	0

*Fuente: Realizado por Leonardo Asqui e Iván Lema (Autores).*

$$I_s = \frac{5 \times 0 + 10 \times 1 + 85 \times 2}{200} = 0.9$$

La población encuentra a la comunicación con un controlador muy rápido, mostrando un índice de satisfacción alto con respecto a la rapidez de la comunicación.



**Figura IV. 188.** Resultados del nivel de satisfacción de las preguntas de la encuesta  
*Fuente: Leonardo Asqui e Iván Lema (Autores)*

También a cada pregunta se le asignó un peso dependiendo de qué tan importante es para mejorar el control y monitoreo de los procesos modulares, los pesos a las preguntas son:

**Tabla V.XLII. Pesos de las variables de las preguntas de la encuesta**

VARIABLE DE LA PREGUNTA	PESO %
1.- Programación de controladores	5
2.-Diseño de HMI, y creación de tags En OPC Server	10
3.-Configuración de parámetros de red	10
4.- Controlar y monitorear varios procesos	20
5.-Almacenamiento de datos	5
6.-Confiabilidad	25
7.-Rapidez	25

*Fuente: Realizado por Leonardo Asqui e Iván Lema (Autores).*

De acuerdo a la fórmula de ponderación por convenio con asignación de pesos tenemos:

$$I_s = \mu A * PA\% + \mu B * PB\% + \dots + \mu Z * PZ\% \quad (3)$$

DONDE:

- $I_s$ : índice de satisfacción.
- $\mu x$ : índice de cada pregunta.
- $Px\%$ : peso de cada pregunta

$$I_s = 0.53 \times 0.05 + 0.82 \times 0.1 + 0.51 \times 0.1 + 0.8 \times 0.2 + 0.81 \times 0.05 + 0.87 \times 0.25 + 0.9 \times 0.25$$

$$I_s = 0.8025$$

El resultado general de la encuesta de acuerdo a los pesos de las variables mostradas en la tabla V.XL.II nos da un índice de satisfacción para la mejora del control y monitoreo de los procesos modulares de 0.8025. Ósea un 80.25%.

## **NOTAS CAPÍTULO V**

- (1) [http://es.wikipedia.org/wiki/Tama%C3%B1o\\_de\\_la\\_muestra](http://es.wikipedia.org/wiki/Tama%C3%B1o_de_la_muestra)
- (2) ESTRATEGIAS DE PONDERACIÓN DE LA RESPUESTA EN ENCUESTAS DE SATISFACCIÓN DE USUARIOS DE SERVICIOS. Pág. 7
- (3) ESTRATEGIAS DE PONDERACIÓN DE LA RESPUESTA EN ENCUESTAS DE SATISFACCIÓN DE USUARIOS DE SERVICIOS. Pág. 5

## CONCLUSIONES

La finalidad de este trabajo es mostrar como la incorporación de varios procesos modulares comandados por controladores de diferentes fabricantes, mejora el control y monitoreo de los mismos, para lo cual se ha investigado la forma de conectarlos en red tomando en cuenta que los protocolos de comunicación son distintos para cada controlador; aunque todos se comunican bajo el stack de protocolos TCP/IP no es posible la comunicación directa entre ellos teniéndose que incorporar a la red un servidor OPC, en donde cualquier dispositivo o aplicación basado en OPC puede modificar cualquier variable en el Servidor, de esta forma EL HMI creado en Labview puede leer, escribir las variables existentes en OPC para el control y monitoreo de los procesos modulares. La base de datos fue creada en PHP MySQL; tenemos acceso a la base de datos desde la aplicación por medio de un servidor ODBC, este hace que cualquier aplicación tenga acceso a los datos sin importar su gestor de base de datos, de esta forma usando Labview podemos ingresar nuevos datos y usarlos posteriormente para los históricos. Como podemos ver ODBC Y OPC funcionan de forma similar, ODBC permite el acceso a datos de una base de datos, mientras OPC permite el acceso a datos de controladores, gracias a estos dos servidores tenemos la facilidad de que cualquier dispositivo o aplicación tenga acceso a la base de datos y variables de proceso en todo momento mejorando de este modo el control y monitoreo los procesos existente en la red.

Como la velocidad, disponibilidad, confiabilidad y escalabilidad son factores indispensables en una red de control, se tomó la decisión de usar una topología en estrella, donde toda la información pasara a través de un concentrador, este concentrador debe ser capaz de manejar gran flujo de datos y evitar al máximo las colisiones de las tramas, a de más de proporcionar facilidad de incorporar nuevos módulos a futuro, dicho concentrador será un switch industrial que prestara todas estas facilidades mencionados anteriormente, dividirá el dominio de colisiones evitando perdida de mensajes, aumentara la disponibilidad de la red ya que los dispositivos no tendrán que escuchar el medio antes de transmitir, reducirá la latencia dentro de la red aumentando la velocidad de transmisión, por todas estas ventajas mencionadas, el rendimiento de la red mejora en gran medida siendo la transmisión prácticamente a velocidad del cable.

## RECOMENDACIONES

- Recordamos verificar los manuales y notas de fabricantes de los controladores antes de proceder a realizar algún cambio en su configuración
- Cablear de forma adecuada y segura la alimentación eléctrica de los dispositivos de la red
- Realizar una comprobación de entradas-salidas para verificar el cableado de los sensores-actuadores, antes de proceder a programarlos.
- Realizar una topología gráfica bien documentada de la red antes de empezar la configuración de dispositivos.
- Realizar una comprobación de comunicación, al configurar los controladores con una nueva dirección IP con el comando ping.
- Direccionar correctamente los tags en el OPC server, al momento de establecer las direcciones.
- Para la creación del HMI, realizarlo todo sobre un proyecto y anexar las variables del OPC al proyecto, para evitar que el sistema no responda al desconectarse un controlador.

## RESUMEN

Se diseñó e implementó una red de comunicación Ethernet didáctica entre Controladores Lógicos Programables (PLCs) de distintos fabricantes para el control y monitoreo de procesos modulares mediante una Interfaz Hombre-Máquina (HMI) y una Base de Datos (DB) en el Laboratorio de la Escuela de Ingeniería Electrónica en Control y Redes Industriales de la Escuela Superior Politécnica de Chimborazo

Con el método deductivo se partió de tecnologías Industriales generales como Ethernet Industrial, OPC (OLE para procesos de Control) y OBDC (Open Database Connectivity) con la finalidad de mejorar el control y monitoreo de los procesos modulares.

Los PLCs usados fueron S7-1200, TWDLMDA20DTK con una pasarela Ethernet, y WAGO 750-842; además de un Controlador de Automatización Programable (PAC) FP-1601. En la comunicación se ocupó el protocolo Ethernet Industrial de cada controlador como MODBUS TCP/IP, PROFINET. El enlace HMI-PLCs se realizó mediante un OPC. El HMI se programó en Labview y la DB en MySQL. Los procesos modulares fueron: mesa de indexación, envasadora de líquidos, semáforo, y una casa automatizada.

Como resultados se obtuvo que la Red Ethernet didáctica mejoro el control y monitoreo en un 80% en comparación a una conexión independiente de cada uno de los procesos modulares.

Se concluyó que para la comunicación de PLCs de diferentes fabricantes se lo puede realizar mediante un OPC permitiendo la comunicación de hardware y software mejorando de esta forma el monitoreo y control de los procesos modulares sin perder su autonomía.

Se recomienda configurar debidamente los parámetros de red para una óptima comunicación

## SUMMARY

It was designed and implemented a didactic Ethernet communication network between Programmable Logic Controllers (PLC's) from different manufacturers for the control and monitoring of processes by modular man-machine interface (HMI) and a database (DB) in the Laboratory of the School of Electronic Engineering Industrial Networking and control of the Polytechnic School of Chimborazo.

Whit the deductive method, it started from general industrial technologies as Industrial Ethernet, OPC (OLE for process control) and ODBC (Open Database connectivity) in order to improve the control and monitoring of the modular processes.

The PLC used were S7-1200 whit an Ethernet TWDLMDA20DTK and WAGO 750-842, plus a programmable Automation Controller (PAC) FP-1601. In the communication, Industrial Ethernet protocol used each controller as MODBUS TCP/IP, PROFINET. The HMI-PLC link was carried out by OPC. The HMI was programmed in Labview and MySQL DB. Modular processes were table indexing, packaging of liquid, light, and automated house.

As a result it was found that the improved didactic Ethernet network control and monitoring by 80% compared to an independent connection from each of Modular processes

It was concluded that for the communication of PLC's from different manufacturers can achieved using an OPC allowing communication hardware and software thereby improving the monitoring and control of modular processes without losing their autonomy

It is recommended to properly configure the network parameters for optimal communication

## GLOSARIO

**Grafcet:** (GRAPhe Functional Commande Etape Transition) es un grafo o diagrama funcional normalizado, que permite hacer un modelo del proceso a automatizar, contemplando entradas, acciones a realizar, y los procesos intermedios que provocan estas acciones. El Grafcet se compone de etapas o estados, transiciones y uniones entre etapas y transiciones.

**Labview:** (acrónimo de Laboratory Virtual Instrumentation Engineering Workbench) es una plataforma y entorno de desarrollo para diseñar sistemas, con un lenguaje de programación visual gráfico. Recomendado para sistemas hardware y software de pruebas, control y diseño, simulado o real y embebido, pues acelera la productividad. El lenguaje que usa se llama lenguaje Grafico.

**OPC Server:** (OLE for Process Control) es un estándar de comunicación en el campo del control y supervisión de procesos industriales, basado en una tecnología Microsoft. La comunicación OPC se realiza a través de una arquitectura Cliente-servidor. El servidor OPC es la fuente de datos (como un dispositivo hardware a nivel de planta) y cualquier aplicación basada en OPC puede acceder a dicho servidor para leer/escribir cualquier variable que ofrezca el servidor.

**Tag:** marca o etiqueta asignada a un espacio de memoria utilizada en programación para referirse a él.

**Profibus:** Profibus es un estándar de comunicaciones para bus de campo. Deriva de las palabras PROcess FieId BUS.

**ControlNet:** es un protocolo de red abierto para aplicaciones de automatismos industriales, también es conocido como bus de campo.

**HART:** (acrónimo del inglés Highway Addressable Remote Transducer) es un protocolo abierto de uso común en los sistemas de control, que se emplea para la configuración remota y supervisión de datos con instrumentos de campo.

**Red LAN:** Una red de área local, red local o LAN (del inglés Local Area Network) es la interconexión de uno o varios dispositivos. Antiguamente su extensión estaba limitada físicamente a un edificio o a un entorno de 200 metros, que con repetidores podía llegar a la distancia de un campo de 1 kilómetro, sin embargo, hoy en día y gracias a la mejora de la potencia de redes inalámbricas y el aumento de la privatización de satélites, es común observar complejos de edificios separados a más distancia que mantienen una red de área local estable.

**Modelo TCP/IP:** El modelo TCP/IP, describe un conjunto de guías generales de diseño e implementación de protocolos de red específicos para permitir que un equipo pueda comunicarse en una red

**SCADA:** Supervisory Control And Data Acquisition (Supervisión, Control y Adquisición de Datos) es un software para ordenadores que permite controlar y supervisar procesos industriales a distancia.

**ODBC** (Open DataBase Connectivity) es un estándar de acceso a las bases de datos desarrollado por SQL Access Group en 1992. El objetivo de ODBC es hacer

posible el acceder a cualquier dato desde cualquier aplicación, sin importar qué sistema de gestión de bases de datos (DBMS) almacene los datos.

**PFC:** Controlador Programable Fieldbus

**CSMA/CD:** (del inglés Carrier Sense Multiple Access with Collision Detection) o, en español, acceso múltiple con escucha de portadora y detección de colisiones, es un protocolo de acceso al medio compartido.

## BIBLIOGRAFIA

1. **ACEDO SANCHEZ, J.**, Instrumentación y Control Avanzado de Procesos., 2a ed., Madrid-España., Díaz de Santos., 2006., Pp. 137,138; 153; 31-32; 447.
2. **ALONSO, N.**, Redes de Comunicaciones Industriales, 1a ed., Madrid-España., Universidad Nacional de Educación a Distancia, 2013., Pp. 24-34, 151-178, 389-413.
3. **DORDOIGNE, J.**, Redes Informática: Nociones Fundamentales., 3a ed., Barcelona-España, Ediciones ENI, 2011., Pp.192, 193, 197, 198, 218, 219.
4. **GARCÍA MORENO, E.**, Automatización de Procesos Industriales: Robótica Automática., 1a ed., Valencia-España., Universidad Politécnica de Valencia. Servicio de publicación., 2009., p. 90, 91,92, 97,98
5. **GUERRERO, V.; Y OTROS.**, Comunicaciones Industriales.,1a ed., México D.F.-México., Alfaomega Grupo Editor., 2010., Pp. 254-278, 298-304.

# BIBLIOGRAFÍA DE INTERNET

## 1. AUTOMATIZACIÓN INDUSTRIAL

<http://books.google.com.ec/>

<http://www.automatizacion-industrial.es/>

[http://www.rocatek.com/forum\\_automatizacion\\_industrial.php](http://www.rocatek.com/forum_automatizacion_industrial.php)

[http://infoplcn.net/files/documentacion/mercado\\_automatizacionf](http://infoplcn.net/files/documentacion/mercado_automatizacionf)

<http://es.scribd.com/doc/29338450/AUTOMATIZACION-INDUSTRIAL>

<http://www.ni.com/data-acquisition/pc-based-automation/esa/>

<http://www.inves.es/>

2012-12-23

## 2. AUTOMATIZACIÓN INDUSTRIAL

<http://www.automation.siemens.com>

[http://www.uv.es/rosado/courses/sid/Capitulo5\\_PCind.pdf](http://www.uv.es/rosado/courses/sid/Capitulo5_PCind.pdf)

<http://www.webelectronica.com.ar/news15/nota09.htm>

[http://es.wikipedia.org/wiki/Sistema\\_embebido](http://es.wikipedia.org/wiki/Sistema_embebido)

<http://www.mes-sigma.net/Literatura/Tecnologia%20PAC.pdf>

<http://www.ni.com/white-paper/3755/es>

2012-12-13

## 3. BASE DE DATOS

<https://sites.google.com/site/modelamientodebasesdedatos/wampserver>

2013-02-27

**4. BASE DE DATOS**

<http://es.scribd.com/doc/23633288/>  
2013-02-27

**5. BASE DE DATOS**

<http://docs.phpmyadmin.net/en/latest/>  
2013-02-28

**6. BASE DE DATOS**

<http://www.slideshare.net/aimerodriguezrodriguez/que-es-wamp-server>  
2013-02-01

**7. ETHERNET INDUSTRIAL**

<http://infoplcn.net/files/documentacion/comunicaciones/>  
[http://es.wikipedia.org/wiki/Red\\_industrial](http://es.wikipedia.org/wiki/Red_industrial)  
<https://es.wikipedia.org/wiki/Router>  
2012-12-12

**8. ETHERNET INDUSTRIAL**

<http://www.monografias.com/trabajos90/>  
<http://www.uco.es/grupos/eatco/automatica/ihtm/descargar/scada.pdf>  
<http://es.wikipedia.org/wiki/SCADA>  
[http://www.disa.bi.ehu.es/spanish/ftp/material\\_asignaturas/](http://www.disa.bi.ehu.es/spanish/ftp/material_asignaturas/)  
<http://www.uoc.edu/masters/oficiales/img/913.pdf>  
2012-12-14

**9. I/O SERVER LABVIEW**

<http://www.ni.com/white-paper/13865/en/>  
2013-02-15

**10. I/O SERVER LABVIEW**

<http://digital.ni.com/public.nsf/>  
2013-02-18

**11. MANUAL FIELDPOINT**

[http://www.ni.com/pdf/products/us/4dio538\\_540.pdf](http://www.ni.com/pdf/products/us/4dio538_540.pdf)  
<http://www.ni.com/pdf/manuals/370707a.pdf>  
2013-02-15

**12. MANUAL LABVIEW DATABASE CONNECTIVITY TOOLKIT**

<http://home.hit.no/~hansha/documents/labview>  
[http://www.ni.com/pdf/labview/us/database\\_connectivity\\_toolkit.pdf](http://www.ni.com/pdf/labview/us/database_connectivity_toolkit.pdf)  
2013-03-15

**13. MANUAL SIEMENS S7-1200 – STEP 7**

<http://www.swe.siemens.com/>  
<http://cache.automation.siemens.com>  
2013-01-22

**14. MANUAL TELEMECANIQUE TWIDO**

<http://www.global-download.schneider-electric.com/8525757800/>  
2013-01-25

**15.MANUAL TELEMECANIQUE TWIDO**

<http://www2.schneider-electric.com/>

<http://ebookbrowse.com/twido-hardware-reference.guide-v2-5.pdf>

2013-02-12

**16.MANUAL WAGO**

<http://docs-europe.electrocomponents.com/webdocs/0d0d>

2012-12-20

**17.MANUAL WAGO**

<http://www.wago.com/wagoweb/documentation/750/>

<http://www.wago.com/infomaterial/pdf/51266141.pdf>

2013-01-30

**18. OPC SERVER – KEPSERVER OPC**

[http://www.kepware.com/Products/kepserverex\\_features.asp](http://www.kepware.com/Products/kepserverex_features.asp)

<http://www.infoplcn.net/>

<http://ebookbrowse.com/>

2013-03-11

**19.PASARELA DE MODBUS SERIAL A ETHERNET**

<http://static.schneider-electric.us/>

2013-02-12

**20.PASARELA DE MODBUS SERIAL A ETHERNET**

<http://www.4shared.com/office/erNh0c9/Romro-Eth-Ind.html>

2013-02-13

## **21. PLC & PAC**

<http://www.infoplcn.net/>

<http://www.schneider-electric.com.co/>

<http://www.tracnova.com/>

<http://www.emb.cl/electroindustria/>

2012-12-20

## **22.PROCESOS MODULARES**

<http://e-archivo.uc3m.es/bitstream/>

<http://dSPACE.upS.edu.ec/bitstream/>

2013-04-11

## **23.PROCESOS MODULARES**

[http://es.wikipedia.org/wiki/Maquinaria\\_de\\_Envasado](http://es.wikipedia.org/wiki/Maquinaria_de_Envasado)

2013-04-04

## **24.PROCESOS MODULARES**

<http://es.wikipedia.org/wiki/Sem%C3%A1foro>

2013-04-06

## **25.PROCESOS MODULARES**

<http://www.revistasucasa.com/contenido/articles/>

2013-04-27

## **26. REDES INDUSTRIALES**

<http://www.smar.com/>

<http://infoplcn.net/>

<http://juce.galeon.com/>

2012-12-02

**27. REDES INDUSTRIALES**

[http://jpadilla.docentes.upbbga.edu.co/redes\\_industriales/](http://jpadilla.docentes.upbbga.edu.co/redes_industriales/)

<http://www.schneiderelectric.es/>

2012-12-11

**28. SWITCH INDUSTRIAL NO GESTIONADO**

[http://www.sas-dv.ru/fls/application/pdf/cat9\\_b\\_en.pdf](http://www.sas-dv.ru/fls/application/pdf/cat9_b_en.pdf)

2013-03-01

**29. SWITCH INDUSTRIAL NO GESTIONADO**

<http://ebhdirect.com/customer/ebhoso/>

2013-03-02

# ANEXOS

## ***MANUAL DE LABVIEW***

### **TEMARIO:**

#### ***Tema:***

#### *I/O Server*

- *Conecte LabVIEW al PLC utilizando un servidor I/O*
- *Crear variables compartidas que conectan las etiquetas OPC al I/O Server*
- *Asignación de una variable para una Entrada o Salida de LabVIEW*

#### *Base de datos*

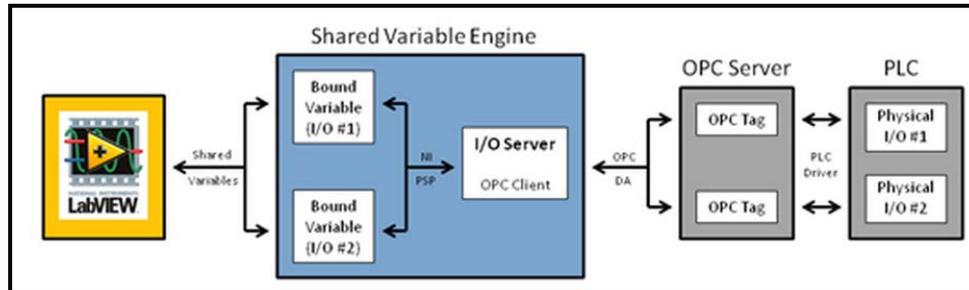
- *Descarga e Instalación del ODBC de MySQL.*
- *Creación del conector de MySQL*
- *Creación de un archivo \*.dns*
- *Conexión a LabVIEW vía .udl*
- *Librerías de Database de LabVIEW*

Este manual estará centrado en la configuración del I/O Server y como enlazar una base de datos con LabVIEW.

## **I/O SERVER**

El módulo de LabVIEW Datalogging and Supervisory Control (DSC) ofrece servidores OPC Cliente de I/O para comunicarse con cualquier servidor de la aplicación del protocolo de la Fundación OPC-DA OPC, que es un estándar basado en COM Microsoft. Un servidor OPC Client I/O muestra todos los servidores OPC instalados en el equipo y

hace que los grupos de acceso y elementos en el servidor. Puede crear un servidor de I / O del cliente OPC para acceder a los datos de un servidor OPC local o remoto.



Servidores OPC Cliente de I/O permiten el motor de variable compartida para unir las etiquetas del OPC de un servidor OPC a variables compartidas. Estas variables compartidas unidas proporcionan un fácil utilizar el modo de LabVIEW para leer y escribir datos en las etiquetas del OPC. Servidores OPC son piezas separadas de software fuera de LabVIEW, National Instruments y proporciona la NI OPC Server como comprar separadamente.

Antes de crear el I/O Server se debe crear el OPC para enlazar con LabVIEW.

- **CONECTE LABVIEW AL PLC UTILIZANDO UN SERVIDOR I/O:**

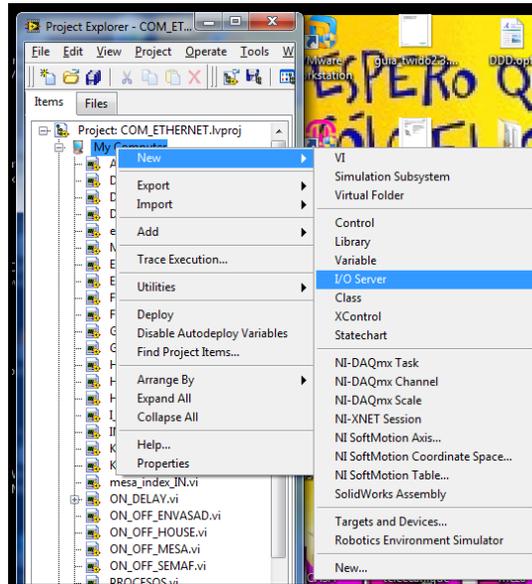
Se creará una interfaz en LabVIEW para las etiquetas (tags) OPC llamadas I/O Server. El I/O Server automáticamente actualiza LabVIEW con los valores de las etiquetas actuales en el rango que usted especifica.

9. Abrimos un proyecto en Labview

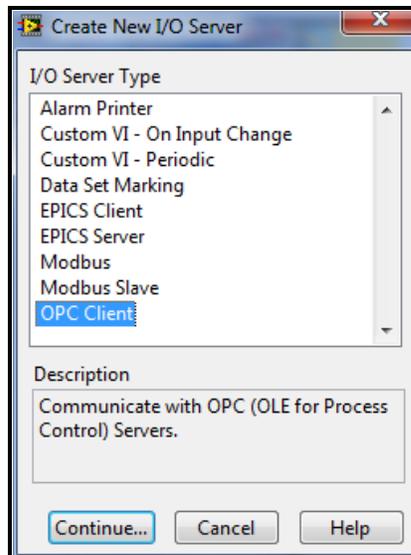
10. Click derecho *"My Computer"*

11. Seleccionar *"New"*

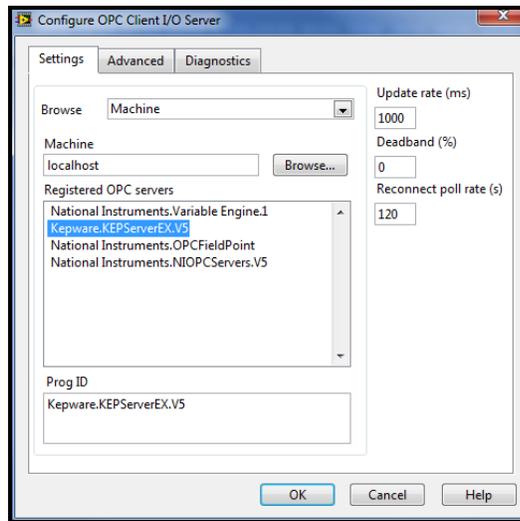
12. Click en “*I/O Server*”



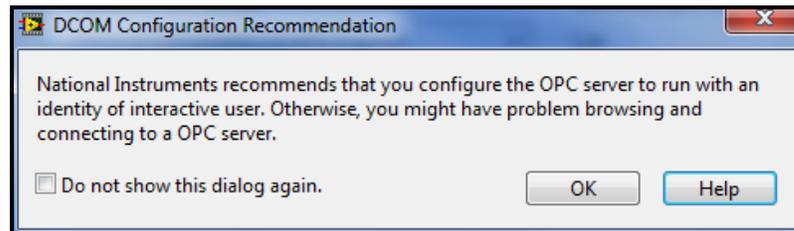
13. Seleccionar “*OPC Client*” y click en “*Continue...*”



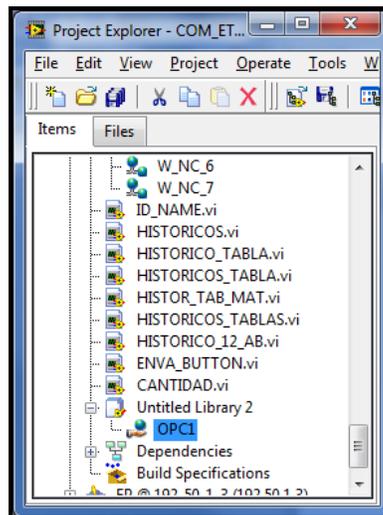
14. Seleccionamos el OPC Server que estamos utilizando y click en “*OK*”



15. Aparecerá un mensaje el que nos dirá que tiene que estar corriendo el OPC Server, si es así tenemos que hacer click **“OK”**



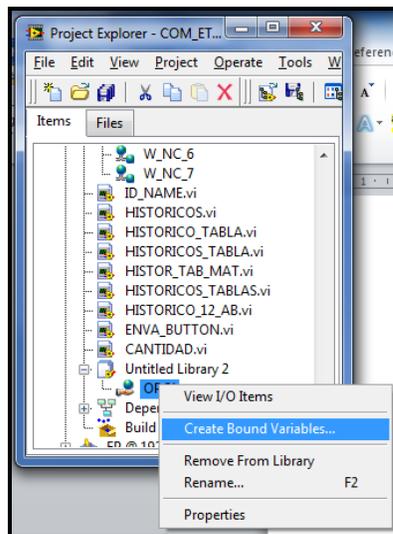
16. Ahora se podrá observar las nuevas variables del I/O Server en el proyecto.



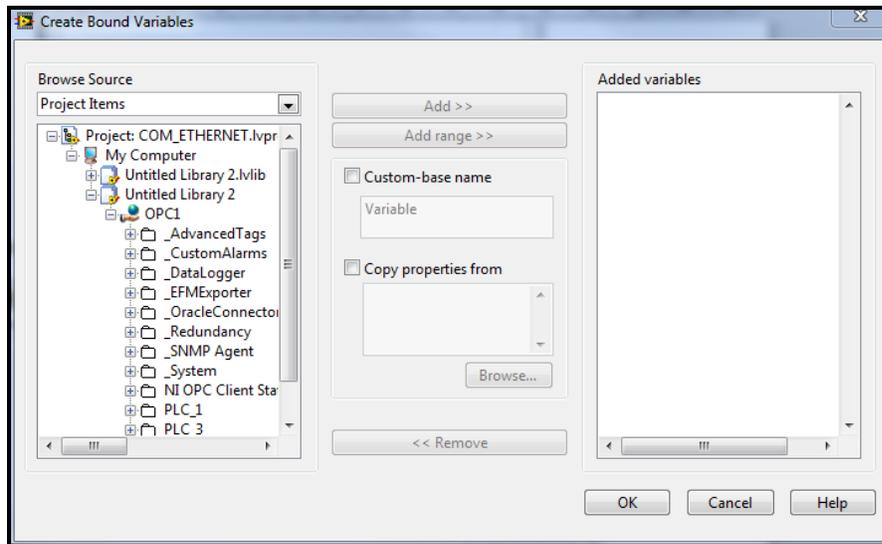
- **CREAR VARIABLES COMPARTIDAS QUE CONECTAN LAS ETIQUETAS OPC AL I/O SERVER:**

Se crean las variables compartidas, las cuales están limitadas en las etiquetas del OPC, dándole el acceso nativo en LabVIEW a los datos del PLC. Con la variable compartida, usted puede compartir datos a través de los usos de LabVIEW en una sola computadora o a través de la red.

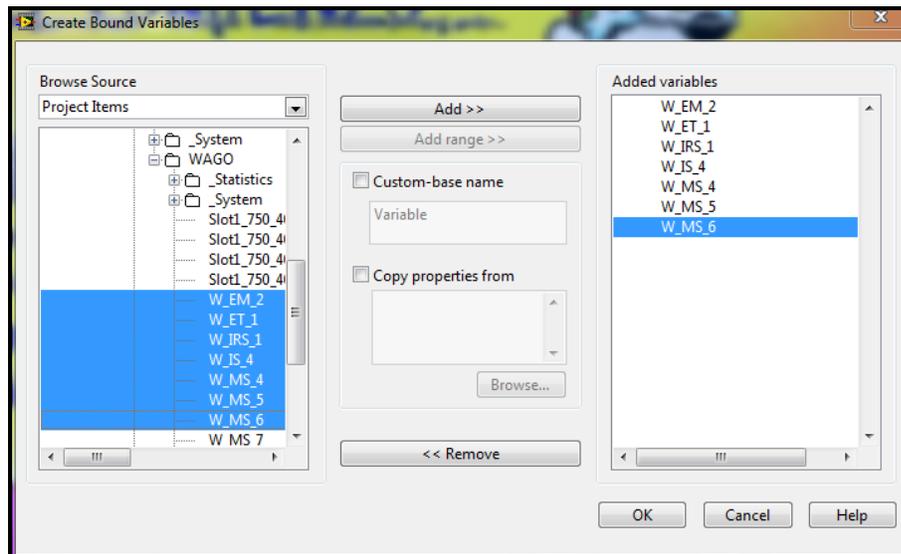
6. Click derecho en el I/O Server creado y click en ***“Create Bound Variables”***



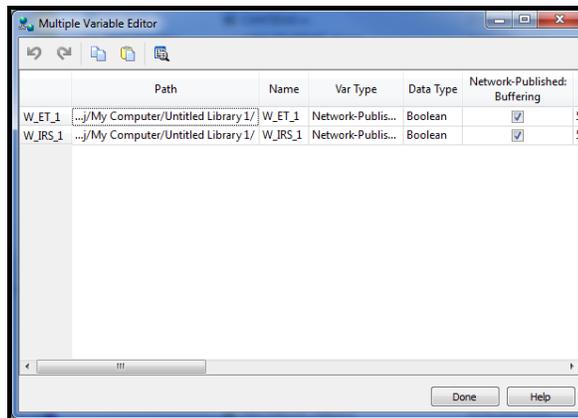
7. Seleccionar librería que se a creado conjuntamente con el OPC, seleccionar el nombre del canal del OPC, y el dispositivo del OPC



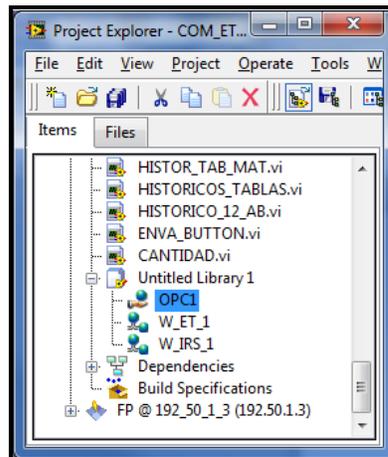
8. Seleccionar todas las tags del OPC que vamos a ocupar, hacemos click en ***“Add>>”***, observamos como las variables han sido agregadas y click en ***“OK”***.



9. Podemos observar cómo se han creado todas las variables y hacer click en ***“Done”***.

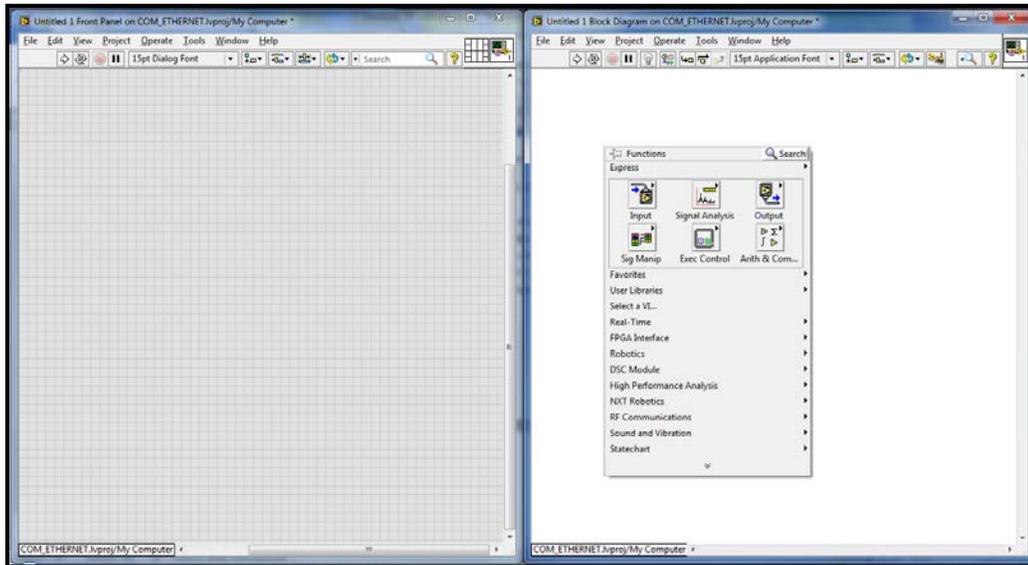


10. Podemos observar cómo se han creado las nuevas variables en el proyecto.

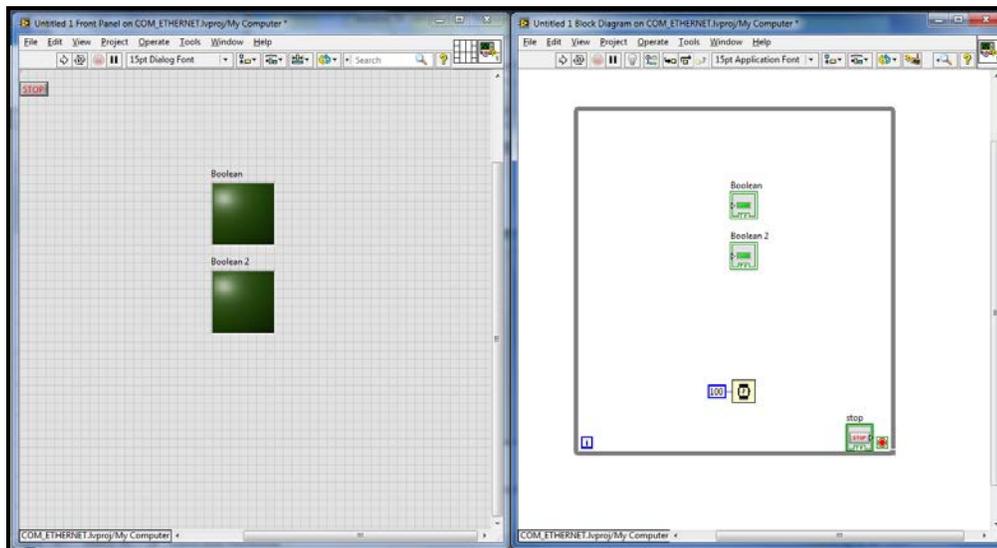


- **ASIGNACIÓN DE UNA VARIABLE PARA UNA ENTRADA O SALIDA DE LABVIEW**

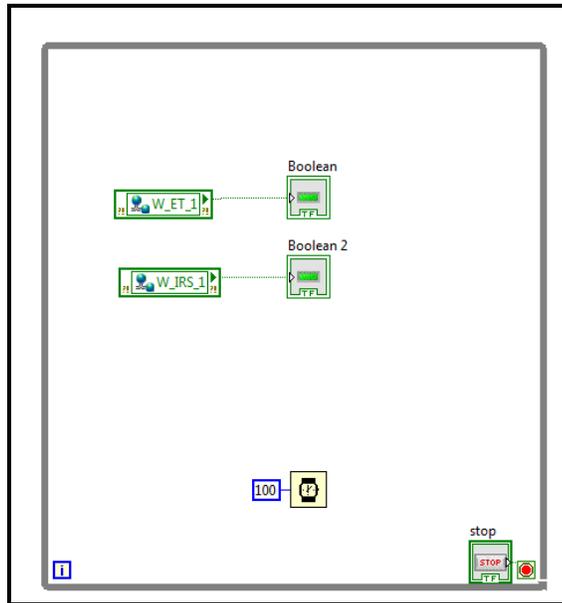
1. Abrimos un nuevo VI en LabVIEW. Lo podemos hacer dando Click derecho en **“My Computer”**, **“New”**, y **“VI”**.
2. Ahora observamos cómo se han creado dos paneles. Los Paneles se llaman Panel Frontal y el otro Diagrama de bloques. En el panel frontal podemos crear una interfaz gráfica, y en el diagrama de bloques podemos enlazar cualquier indicador o control que se cree en el panel frontal para que realice algún programa, ya que en el diagrama de bloques existe funciones para realizar una programación.



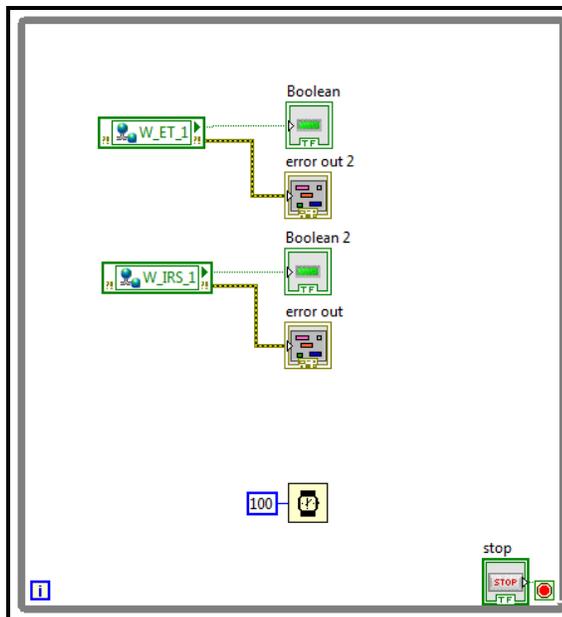
3. Agregamos en este caso dos indicadores el panel frontal y creamos un while en el diagrama de bloques.



4. Ahorra arrastrar las variables creadas en el proyecto, las variables I/O Server y enlazarlas con los indicadores.



5. Se recomienda crear un indicador del error, para cuando haya un problema de enlace con el OPC este no tienda a colgar el programa.



## BASE DE DATOS

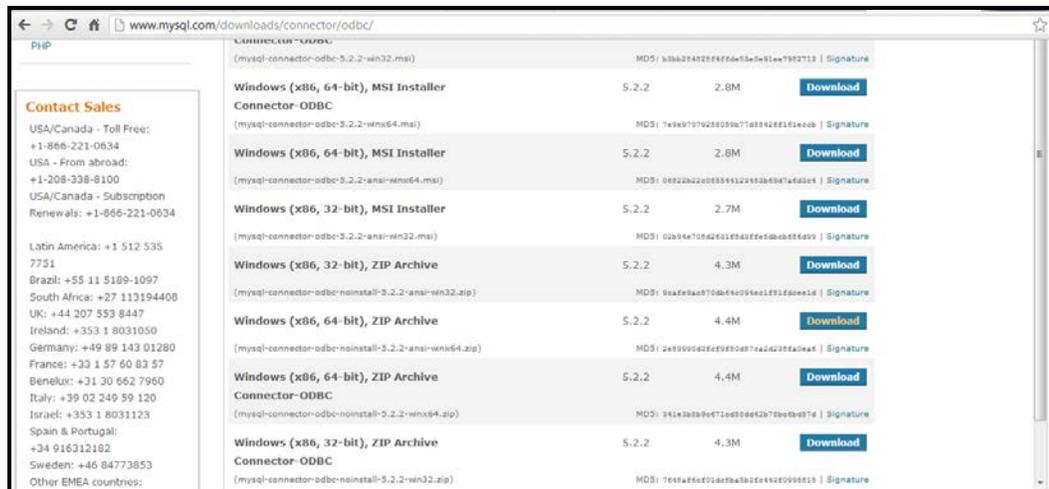
Nos centraremos en enlace de la base de datos con LabVIEW y una pequeña introducción a las librerías de la base de datos. Para el enlace le dividiremos en dos partes para la conexión en LabVIEW con un sistema operativo de 32 bits y otra con sistema operativo de 64 bits. También este manual está realizado con una base datos realizado con MySQL.

- **DESCARGA E INSTALACIÓN DEL ODBC DE MYSQL.**

Para que exista una conexión con la Base de Datos debe existir un conector, que permitirá que exista un enlace, y el conector se gestionara a través de un ODBC en este caso de MySQL.

1. Descargar el ODBC de MySQL en la página:

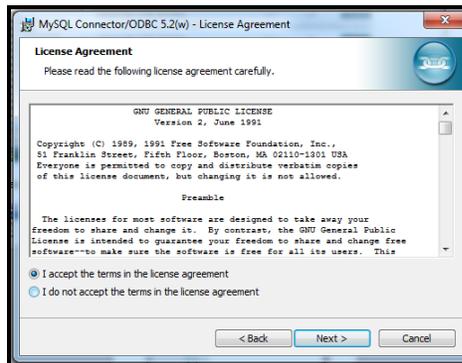
<http://www.mysql.com/downloads/connector/odbc/>



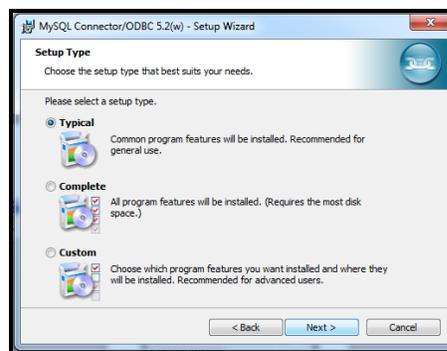
2. Ejecutar el Instalador del ODBC, y click en ***“Next”***.



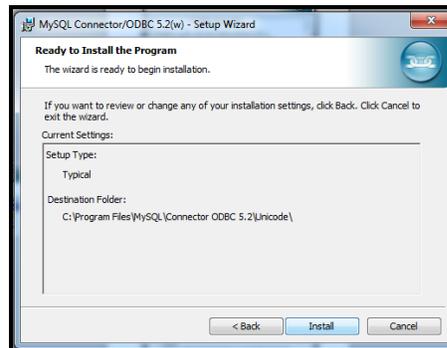
3. Aceptar los términos de la licencia y click en ***“Next”***.



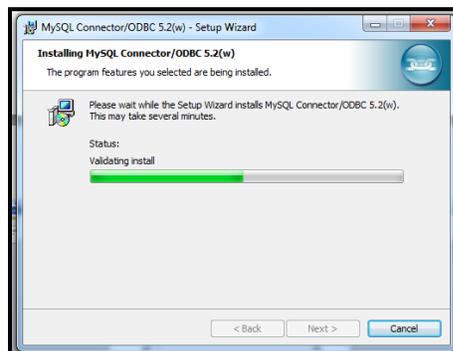
4. Para nuestra conexión solo necesitaremos una instalación Típica. Seleccionar ***“Typical”*** y click en ***“Next”***



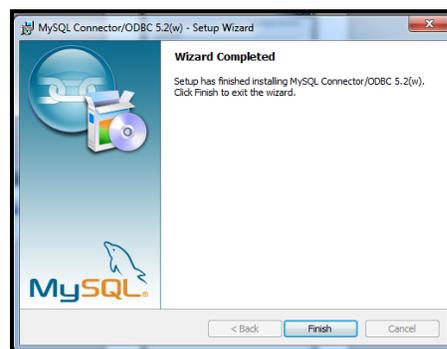
5. Seleccionar ***“Install”***, para ya instalar el ODBC.



6. Esperamos que se complete la instalación.



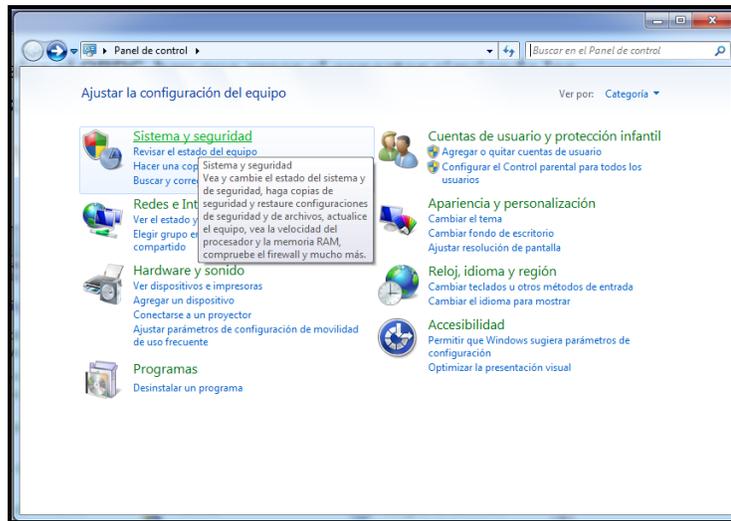
7. Y finalmente click en ***“Finish”***.



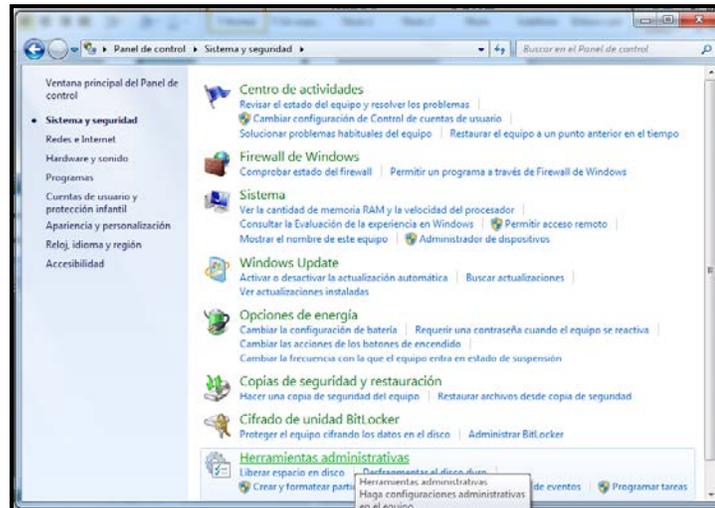
- ***CREACION DEL CONECTOR DE MYSQL***

- A. EN SISTEMA OPERATIVO DE 32 BITS***

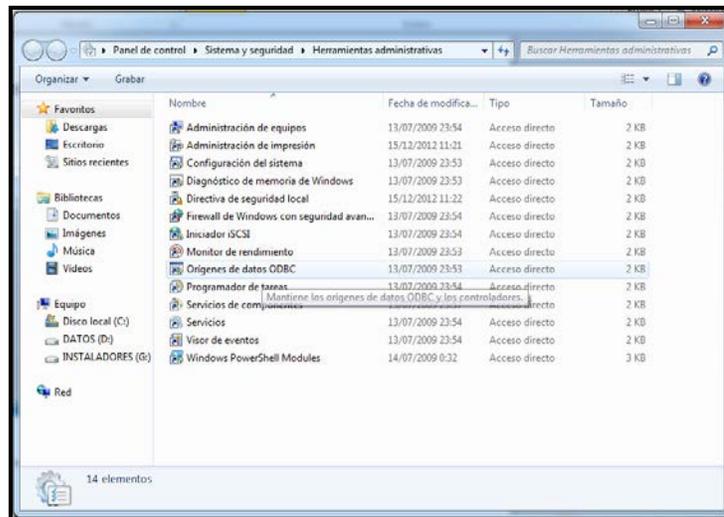
8. Ingresar al **“Panel de Control”**, y luego **“Sistema y seguridad”**.



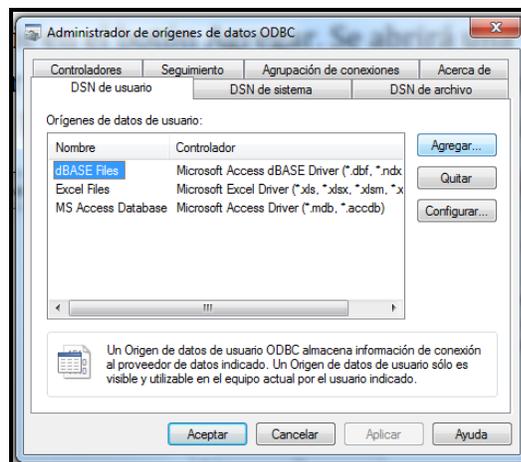
9. Click en **“Herramientas Administrativas”**



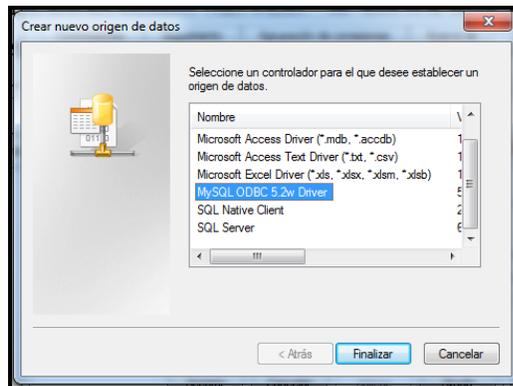
10. Hacer click **“Orígenes de datos ODBC”**.



11. Hacer click en **“Agregar”** en **“DNS de usuario”**.



12. Seleccionar **“MySQL OBDC 5.2w Driver”** y dar click en **“Finalizar”**.



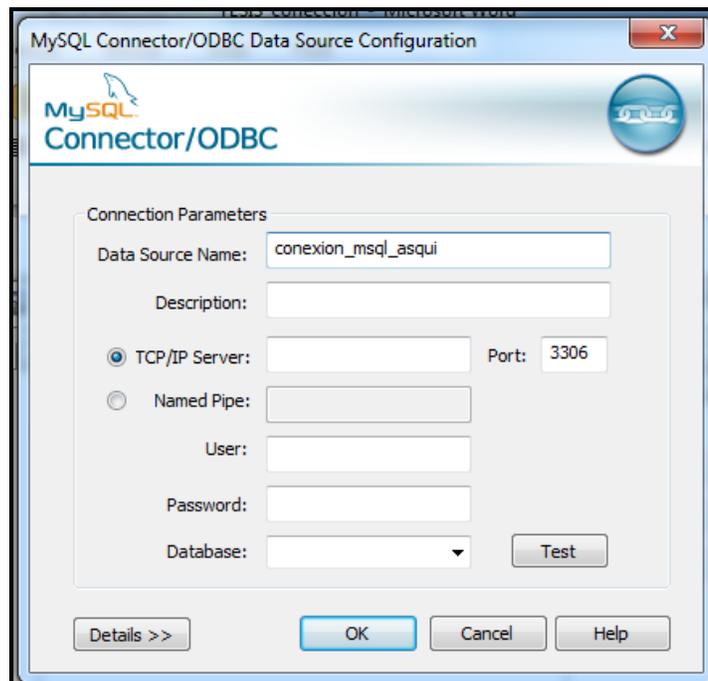
13. Llenar los datos como se muestra en la figura.

*Data Source Name:* El nombre que queremos poner a la conexión.

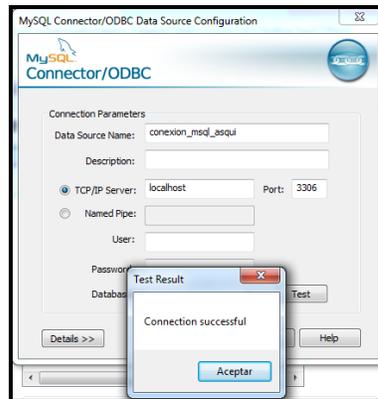
*TCP/IP Server:* Ponemos "localhost" ya que la base de datos estará en nuestra computadora caso contrario pondríamos una dirección ip, la dirección ip del servidor remoto.

*User:* Nombre del Usuario

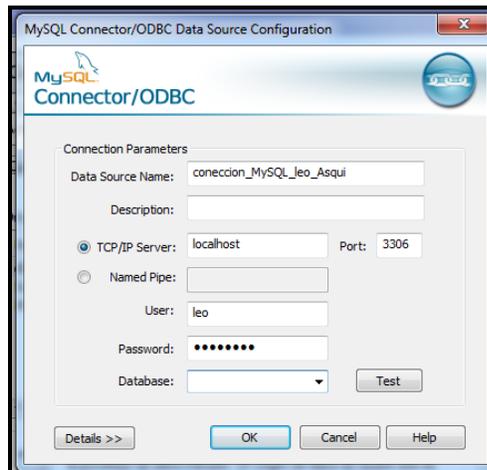
*Password:* La contraseña del Usuario.



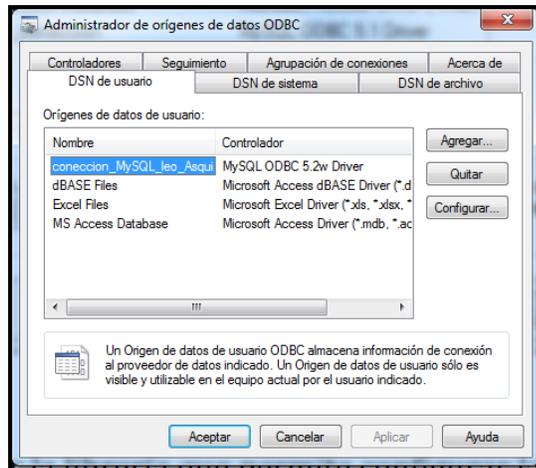
14. Dar click en **“Test”**. Para probar la conexión.



15. Dar click en **“Ok”**



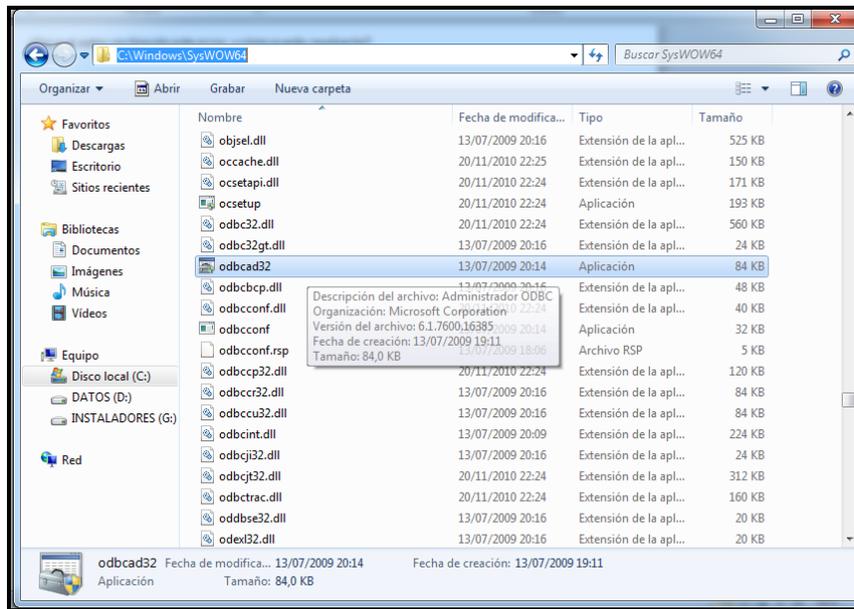
16. Dar click en ***“Aceptar”***



### ***B. EN SISTEMA OPERATIVO DE 64 BITS***

Existe un error que se produce porque la versión de 64-bit de odbcad32.exe, el programa de orígenes de datos ODBC Administrador, se utiliza para configurar el nombre de origen de datos (DSN) a la base de datos cuando la versión de 32 bits del programa Administrador de orígenes de datos ODBC debe haberse utilizado. La aplicación de 32-bit LabVIEW está tratando de utilizar el DSN creado por la aplicación de 64-bit odbcad32.exe para conectarse a la base de datos, y esto no es posible arquitectónicamente. Este es el desajuste de arquitectura mencionado en el mensaje de error. La aplicación de 32-bit sólo puede utilizar LabVIEW DSN creados con la versión de 32-bit de odbcad32.exe. Este tema se discute en el artículo de Microsoft Developer Network [Managing Data Sources](#)





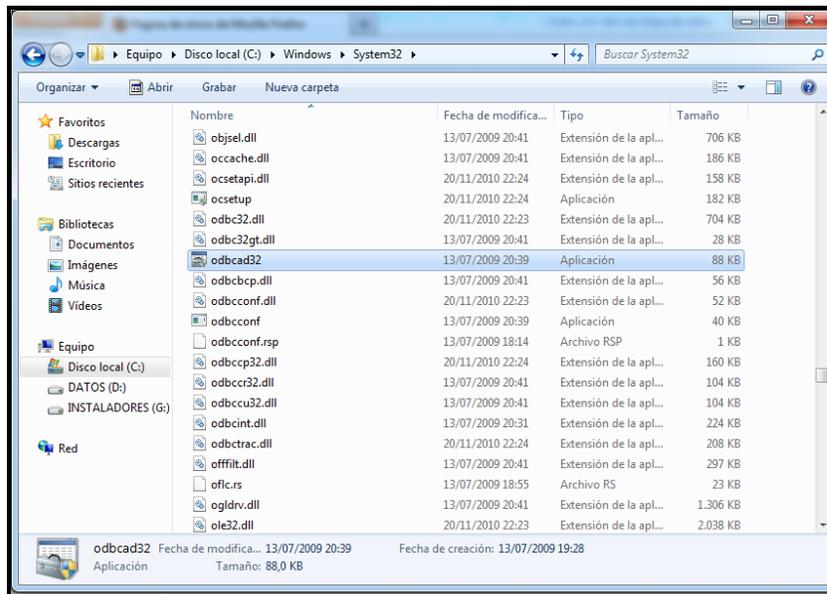
Hay dos opciones para corregir el error:

- Fije la configuración de ODBC. Para ello, abra la versión de 64-bit de odbcad32.exe, elimine la configuración de base de datos, a continuación, abra la versión de 32-bit como administrador y configurar la base de datos que hay.
- Cambiar su código de LabVIEW para utilizar la UDL con la cadena de conexión exacta. Usted puede aprender más acerca de esto en KnowledgeBase 3GJAJKEQ: How Do I Set Up a Microsoft Data Link File (UDL) Through LabVIEW?

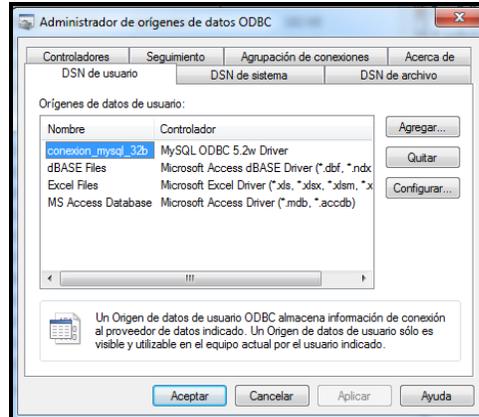
## PROCEDIMIENTO

Si se instaló y se creó primero la conexión con el ODBC de 32 bits que es la de 64 bits siga los dos primeros pasos, sino es así continúe con el paso 3.

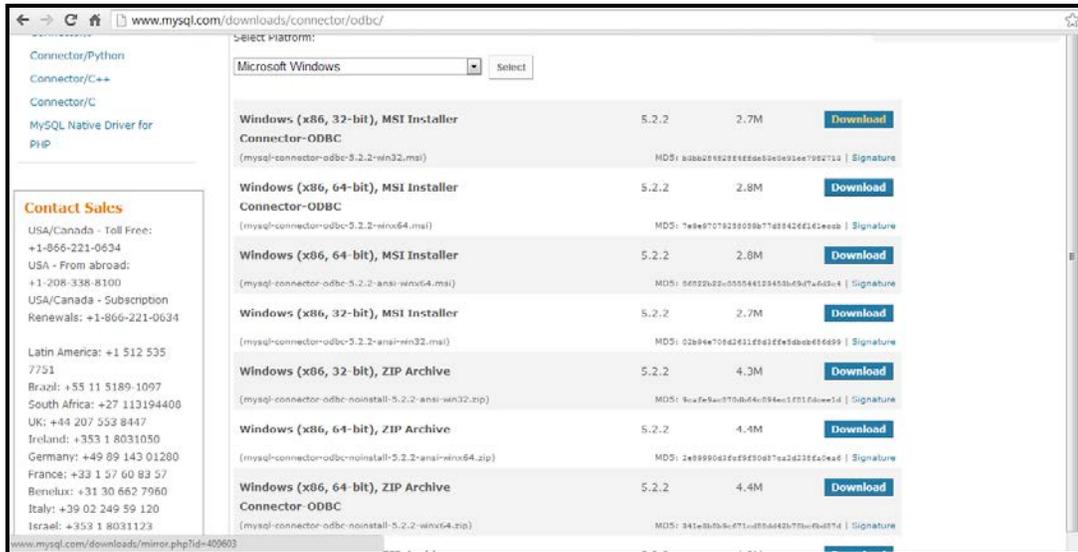
1. Abrir el odbcad32.exe que es la versión de 64 bits.



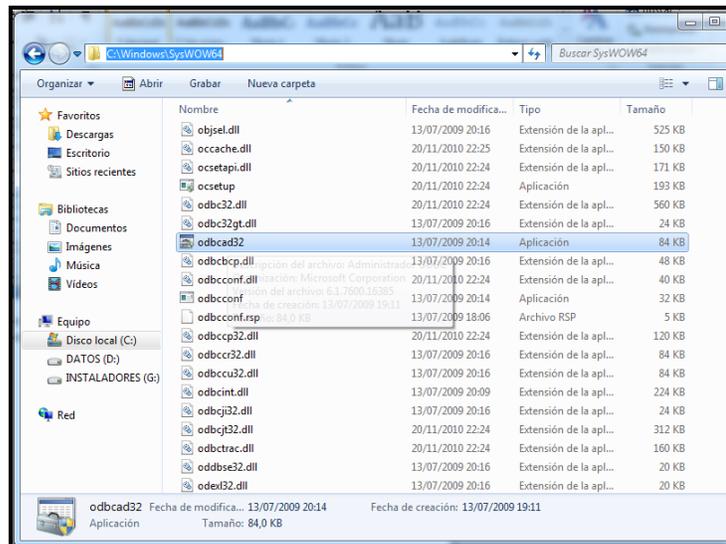
2. Quite la conexión que se agregó. Seleccionar la conexión y luego click en **“Quitar”** y después en **“Aceptar”**.



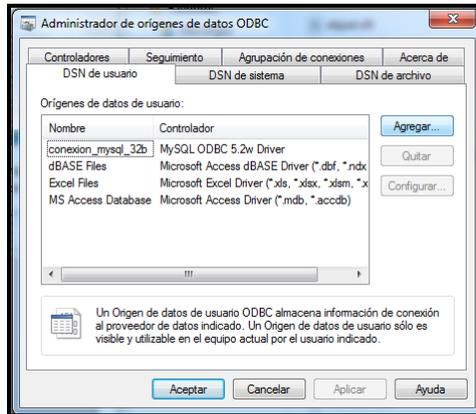
3. Descargar de [www.mysql.com](http://www.mysql.com) la versión de conexión de 32 bits y no la de 64 no hace falta desinstalar el driver de conexión de 64 bits y después instalar.



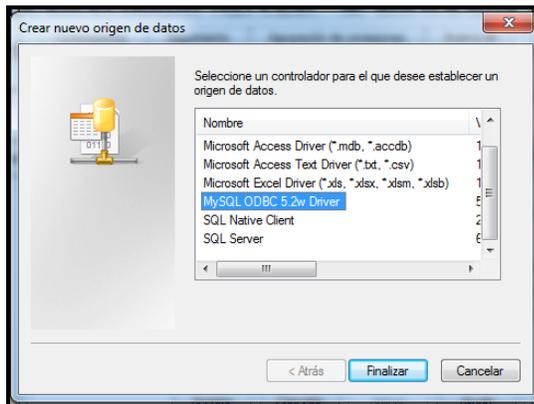
4. Abrir el odbcad32.exe de 32 bits como administrador.



5. Dar click en "Agregar" en DNS de Usuario.



6. Seleccionar **“MySQL ODBC 5.2w Driver”** y dar click en **“Finalizar”**.



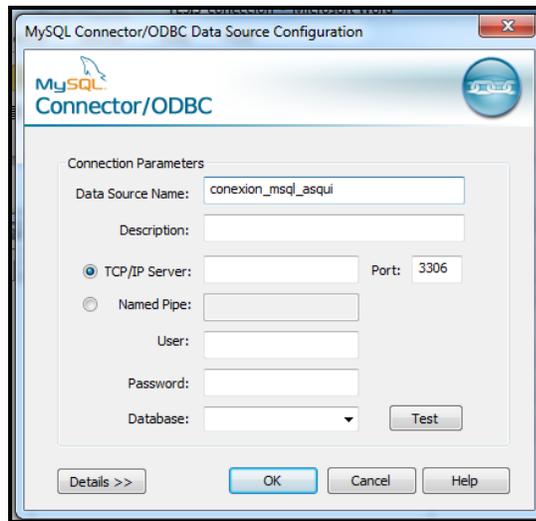
7. Llenar los datos como se muestra en la figura.

**Data Source Name:** El nombre que queremos poner a la conexión.

**TCP/IP Server:** Ponemos “localhost” ya que la base de datos estará en nuestra computadora caso contrario pondríamos una dirección ip, la dirección ip del servidor remoto.

**User:** Nombre del Usuario

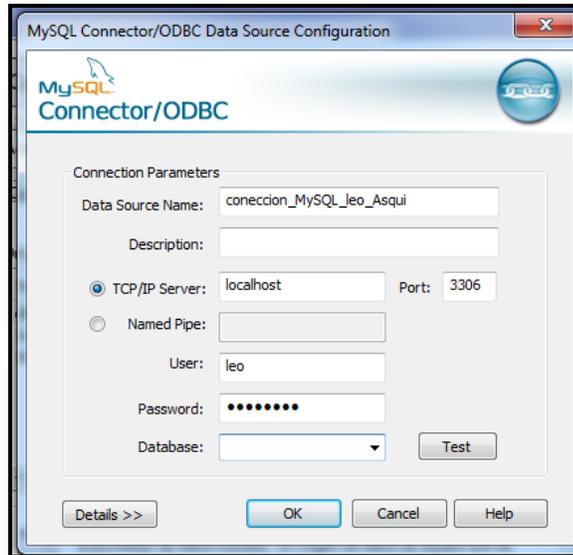
**Password:** La contraseña del Usuario.



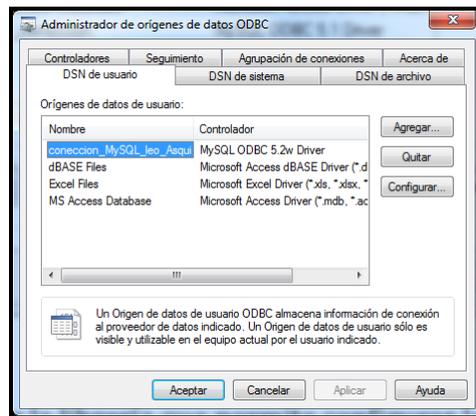
8. Dar click en ***“Test”***. Para probar la conexión.



9. Dar click en *“Ok”*



10. Dar click en *“Aceptar”*

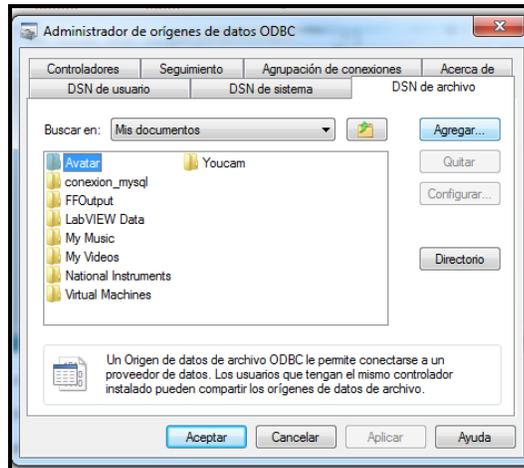


- **CREACION DE UN ARCHIVO \*.DNS**

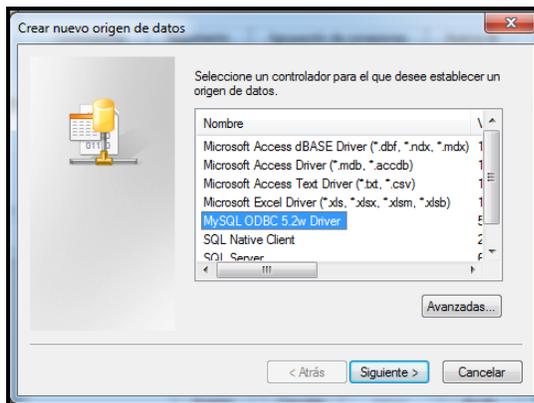
Antes de crear un archivo DNS debíamos haber generado una base de datos, para crear una base de datos podemos leer el manual de creación de una base de datos con WAMPSEVER, así como haber creado la conexión con MySQL y este activado.

Debemos tomar en cuenta que la creación del archivo DNS se debe crear con el obdcad32.exe de acuerdo al sistema operativo.

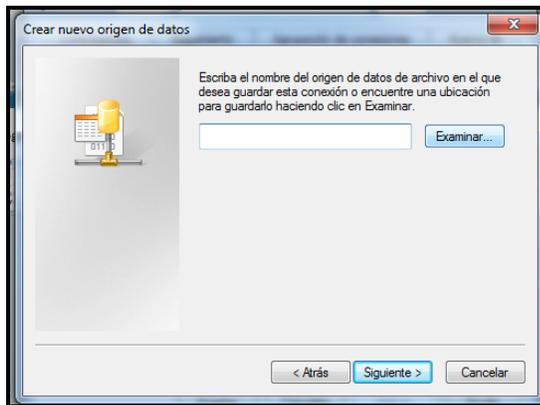
13. Abrir el obdcad32.exe dar click en ***“DNS ARCHIVO”***, luego escoger ***“Agregar”***.



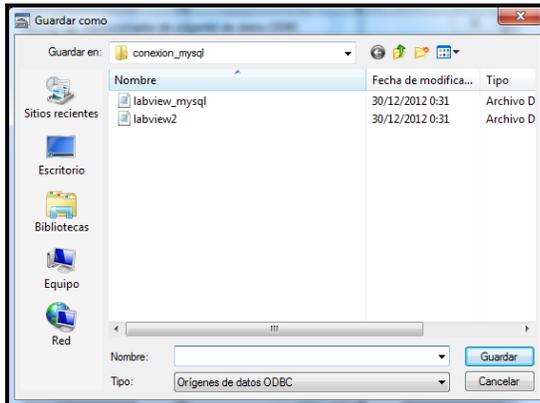
14. Seleccionar ***“My SQL ODBC 5.2w Driver”*** y click en ***“Siguiente”***.



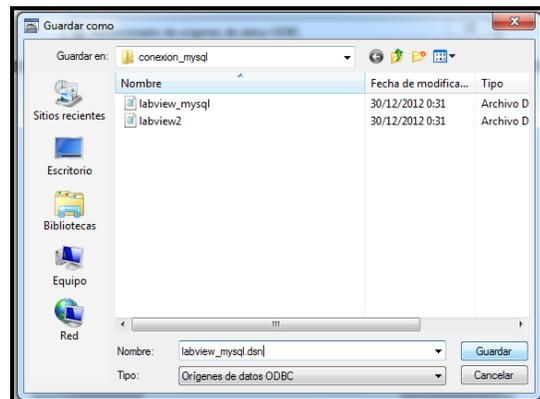
15. Dar click en ***“Examinar”***.



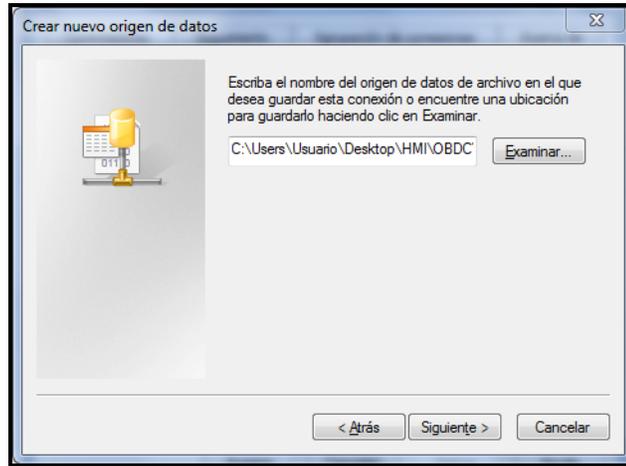
16. Escoger la carpeta para guardar el archivo .dns



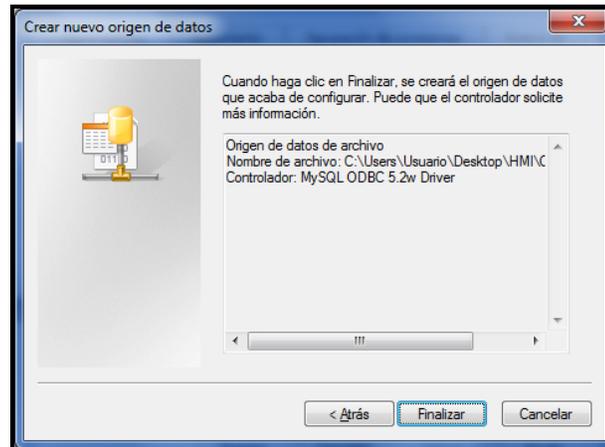
17. Dar un nombre y click en "Guardar".



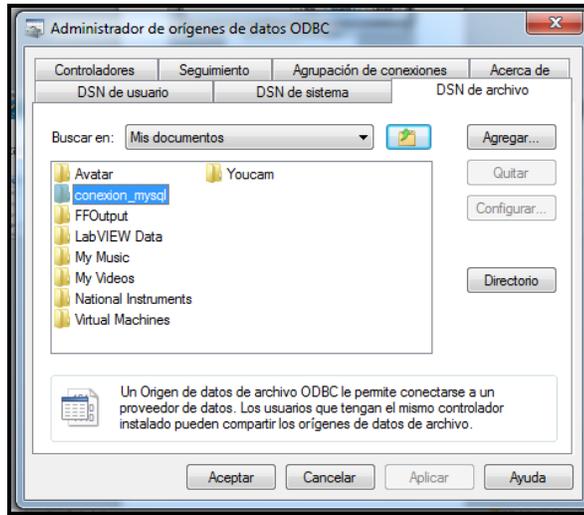
18. Luego dar click en ***“Siguiete”***.



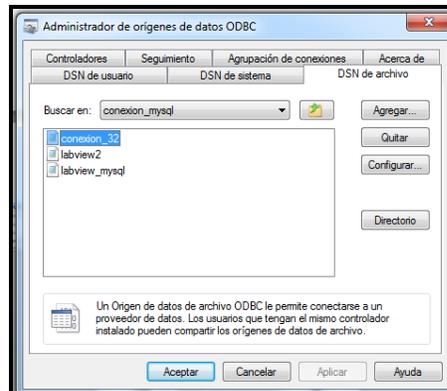
19. Dar click en ***“Finalizar”***



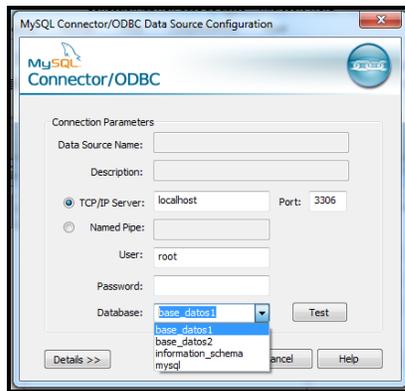
20. Buscar la ubicación de la conexión creada.



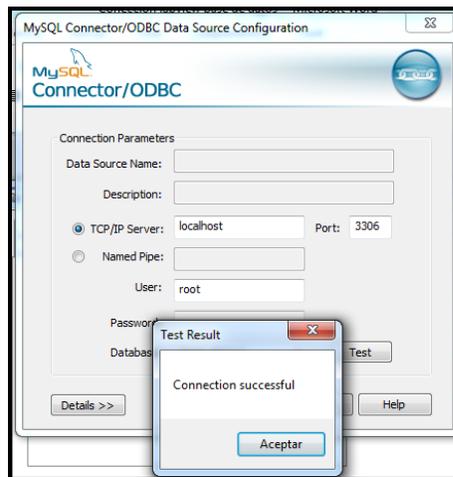
21. Seleccionar la conexión del archivo DNS y dar click en ***“Configurar”***



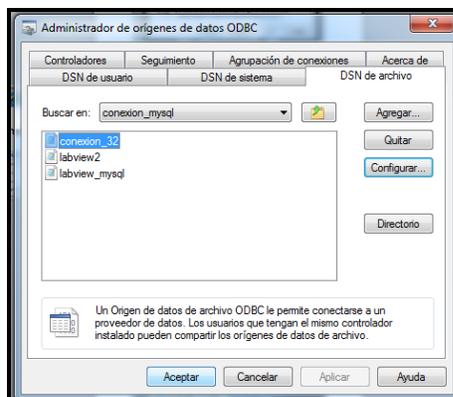
22. Escoger la base de datos con la que se va hacer la conexión y completar los demás datos. Para poder establecer la conexión con la base de datos debe estar bien escrita TCP/IP Server, y estar activo la base de datos.



23. Realizar un **“Test”** para probar la conexión.



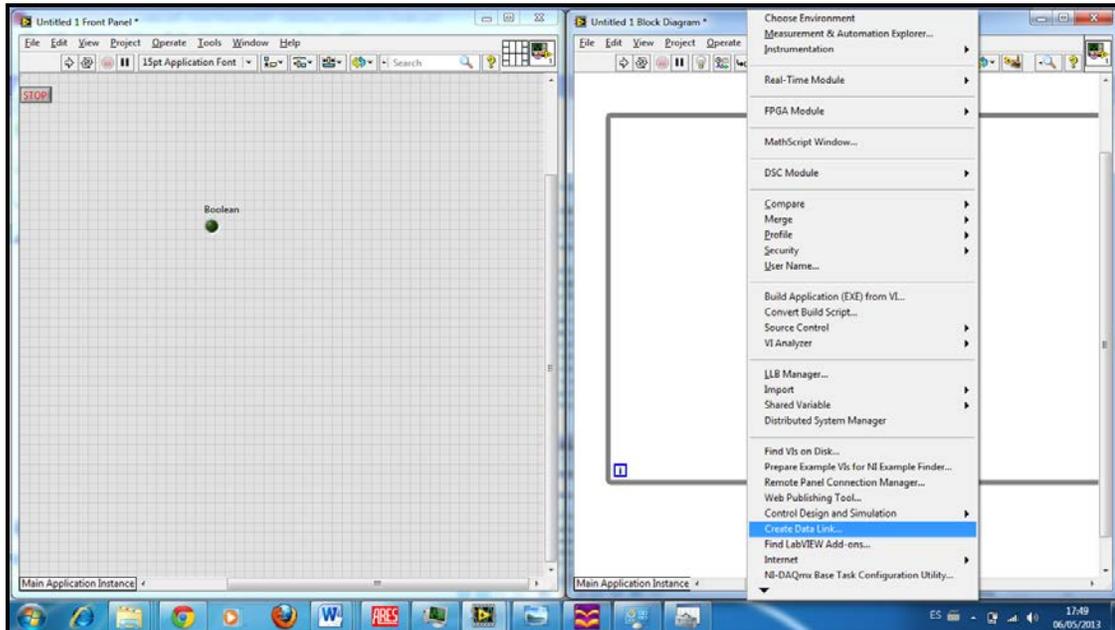
24. Dar click en **“Aceptar”**



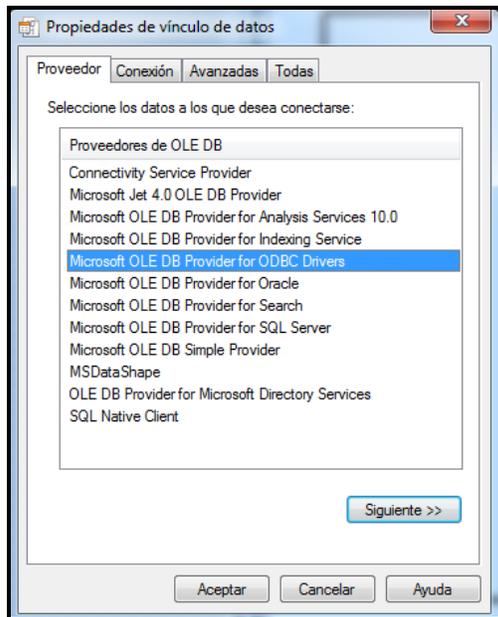
- **CONEXIÓN A LABVIEW VIA .UDL**

8. Abrir LabVIEW.

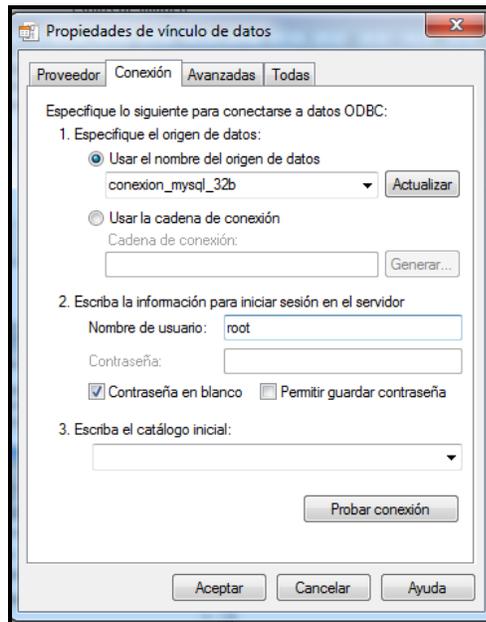
9. Dar click en **“Tools”** y **“Create Data Link...”**



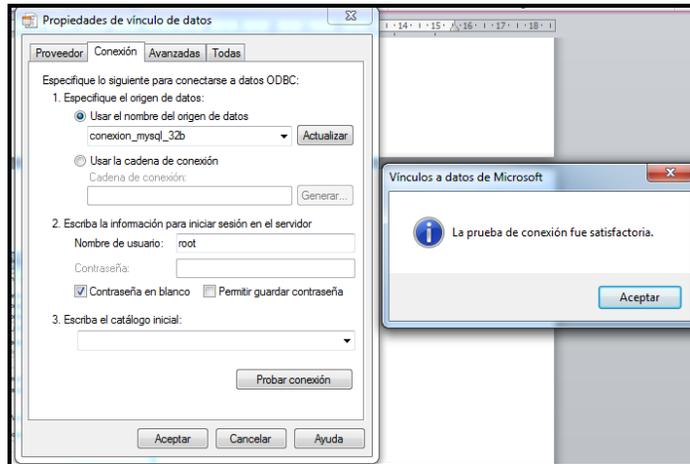
10. Escoger **“Proveedor”**, Seleccionar **“Microsoft OLE DB Provider for ODBC Drivers”**. Y click **“Siguiente”**.



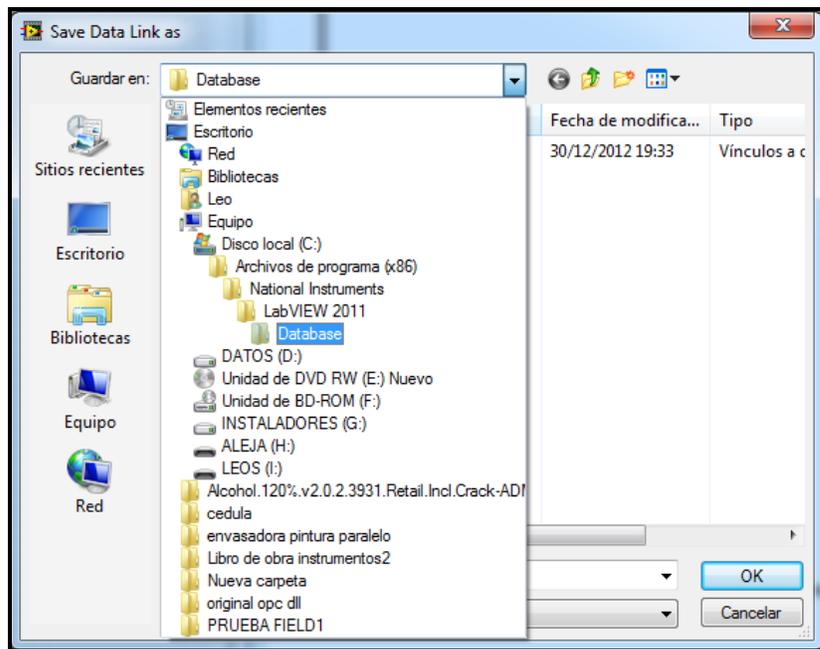
11. Seleccionar la conexión creada con el ODBC y poner el nombre del usuario.



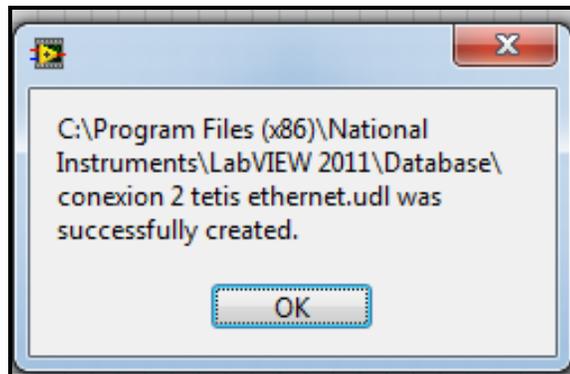
12. Probar la conexión, para observar si es satisfactoria y dar click en **"Aceptar"**.



13. Escoger el directorio donde se va a guardar la conexión, damos click en **“Aceptar”**.



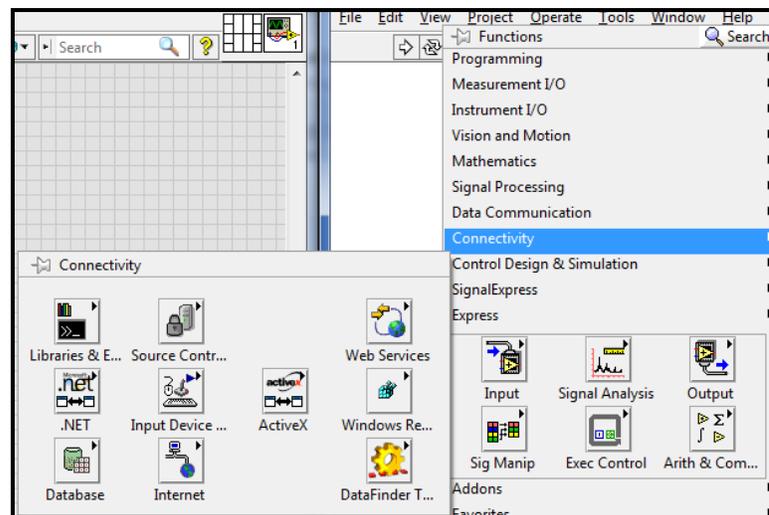
14. Después podemos observar si la conexión fue creada satisfactoriamente.



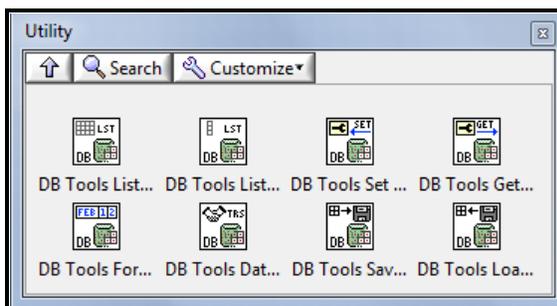
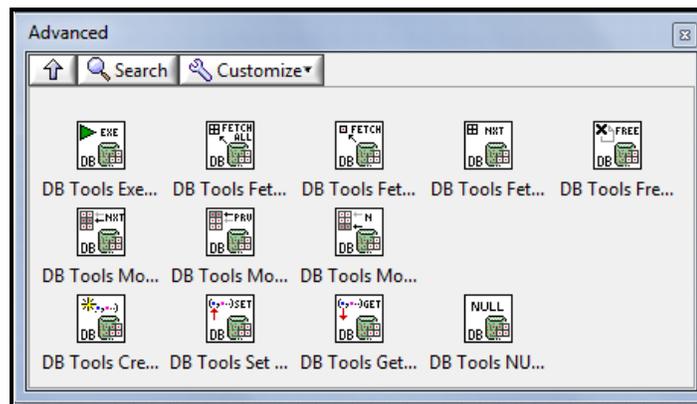
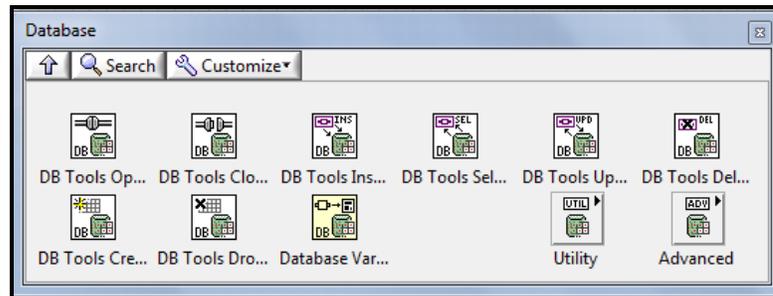
- **LIBRERÍAS DE DATABASE DE LAVBIEW**

Las funciones de Database de LabVIEW se las encuentra:

1. Dar click derecho en el bloque de funciones, seleccionar **Conectivity** y luego **Database**.



2. Podremos observar las funciones de Database.



3. Ahora mostraremos un programa sencillo para poder guardar y otro para leer en la base de datos:

