



**ESCUELA POLITÉCNICA DEL EJÉRCITO SEDE  
LATACUNGA**

**CARRERA DE ELECTRÓNICA E INSTRUMENTACIÓN**

**“Proyecto de Grado previo la obtención del Título de Ingeniería Electrónica en  
Electrónica e Instrumentación”**

**“Diseño e Implementación del Sistema SCADA para el  
Laboratorio de Control de Procesos e Instrumentación de la ESPE  
sede Latacunga”**

**Julio Francisco Acosta Núñez**

**Diego Sebastián Vallejo Santiana**

**Latacunga – Ecuador**

**2005**

## INDICE

PÁG.

### INTRODUCCIÓN

#### CAPÍTULO I : INTRODUCCIÓN

1.1	SISTEMAS SCADA	1
1.1.1	PRESTACIONES	2
1.1.2	REQUISITOS	3
1.1.3	MÓDULOS DE UN SCADA	3
1.2	CONCEPTOS ASOCIADOS A SISTEMAS SCADA	4
1.2.1	TIEMPO REAL	5
1.2.2	HARDWARE EN SISTEMAS DE SUPERVISIÓN: PLC Y PC	5
1.2.3	TARJETAS DE EXPANSIÓN	6
1.2.4	LA ARQUITECTURA ABIERTA	7

#### CAPÍTULO II : EL SOFTWARE LABVIEW DSC

2.1	INTRODUCCIÓN	8
2.1.2	EL PANEL FRONTAL Y EL DIAGRAMA DE BLOQUES	8
2.1.3	LAB VIEW COMO ENTORNO BÁSICO DE INSTRUMENTACIÓN VIRTUAL	9
2.2	MÓDULO LABVIEW DATALOGGIN AND SUPERVISORY CONTROL (DSC)	16
2.2.1	INTRODUCCIÓN	16
2.2.2	DESARROLLANDO UNA APLICACIÓN	20
2.3	SERVIDORES	21
2.3.1	TIPOS DE SERVIDORES A SELECCIONAR	22
2.3.2	SERVIDORES IAK	23

2.3.3	PROCESO DE INSTALACIÓN Y CONFIGURACIÓN DE LOS SERVIDORES	23
2.3.4	REGISTRO DE SERVIDORES	24
2.3.5	CARGA DEL UTILITARIO DE CONFIGURACIÓN DE SERVIDORES	24
2.3.6	REVISIÓN DE LA CONFIGURACIÓN DEL SERVIDOR	25
2.3.7	USO LABVIEW DSC COMO UN SERVIDOR OPC	25
2.3.8	INGRESO AL SERVIDOR OPC USANDO LABVIEW DSC COMO CLIENTE OPC	26
2.3.9	USO DE LABVIEW DSC CON SERVIDORES DEE	27
2.4	USO DE TAGS PARA MANEJO DE ENTRADAS Y SALIDAS	27
2.4.1	CONFIGURACIÓN DE ARCHIVOS	28
2.4.2	CREAR TAGS	28
2.4.3	EDITAR TAGS	29
2.4.4	BORRAR TAGS	30
2.4.5	CONFIGURAR LOS ATRIBUTOS DE LAS ETIQUETAS	30
2.4.6	TIPOS DE DATOS DE TAGS	31
2.4.7	GRUPOS DE TAGS	32
2.4.8	CONFIGURACIÓN DE ALARMAS EN TAGS	32
2.4.9	EL MOTOR DE TAGS	35
2.5	VISUALIZACIÓN Y CONTROL CON LABVIEW DSC	37
2.5.1	LOS PANELES FRONTALES DE UN HMI	37
2.6	DESPLIEGUE DE TENDENCIAS	39
2.6.1	TENDENCIAS DE TIEMPO REAL	39
2.6.2	TENDENCIAS HISTÓRICAS	39
2.7	ALARMAS Y EVENTOS	39
2.7.1	VISUALIZACIÓN DE ALARMAS Y EVENTOS	40
2.8	MANEJO DE DATOS HISTÓRICOS	40
2.8.1	LA BASE DE DATOS CITADEL	41
2.8.2	ARCHIVAR DATOS HISTÓRICOS	41
2.8.3	EXTRAER DATOS HISTÓRICOS	43
2.9	SEGURIDAD	44
2.9.1	CUENTAS DE USUARIO	44
2.10	DESARROLLO DE APLICACIONES EN RED	45

2.10.1	FIJAR LAS APLICACIONES DE RED	46
2.10.2	SERVICIOS DE MONITOREO DE WINDOWS	46
2.10.3	EL SISTEMA RUN TIME	47
2.11	USO DE SQL PARA ACCEDER A DATOS HISTÓRICOS DE CITADEL	47

### **CAPÍTULO III : COMUNICACIÓN DE LA APLICACIÓN SCADA CON EL ENTORNO**

3.1	TARJETAS DE ADQUISICIÓN DE DATOS	49
3.1.1	SAMPLING (MUESTREO).	49
3.1.2	CONVERTIDORES ANALÓGICO/DIGITALES (ADC)	50
3.1.3	TRANSFERENCIA DE DATOS AL ORDENADOR	51
3.1.4	CONVERTIDORES DIGITAL/ANALÓGICOS (DAC)	52
3.1.5	TARJETA NI PCI 6014	52
3.2	FIELD POINT FP 1000	69
3.2.1	CARACTERÍSTICAS DE MODULARIDAD DEL SISTEMA FP - 1000	70
3.2.2	MÓDULOS DE RED	71
3.2.3	MÓDULOS DE ENTRADA Y SALIDA I/O	74
3.2.4	BENEFICIOS DE TRABAJAR EN RED	81
3.2.5	CONEXIÓN DE SENSORES EN LOS DISPOSITIVOS MODULARES EN FORMA DIRECTA	82
3.2.6	SOFTWARE DE FÁCIL CONFIGURACIÓN	83
3.2.7	INTERFASES OPC	84
3.3	PLC SIMATIC S7 200	85
3.3.1	CARACTERÍSTICAS TÉCNICAS GENERALES	85
3.3.2	GENERALIDADES DEL S7-200	88
3.3.3	ENTRADAS Y SALIDAS ANALÓGICAS	90
3.3.4	SELECCIONAR EL MODO DE OPERACIÓN EL S7-200	91
3.3.5	SISTEMAS DE CONTROL DE LAZO PID	92
3.3.6	PRINCIPIOS BÁSICOS DE LA COMUNICACIÓN EN REDES S7-200	96

## **CAPÍTULO IV: REDES TCP/IP**

4.1	INTRODUCCIÓN	102
4.2	MODELO GENERAL DE COMUNICACIONES	102
4.2.1	USO DE LAS CAPAS PARA ANALIZAR PROBLEMAS EN UN FLUJO DE MATERIALES	102
4.2.2	ORIGEN, DESTINO Y PAQUETES DE DATOS	103
4.2.3	MEDIOS	103
4.2.4	PROTOCOLO	104
4.2.5	LAS SIETE CAPAS DEL MODELO DE REFERENCIA OSI	106
4.2.6	ENCAPSULAMIENTO	107
4.3	EL MODELO DE REFERENCIA TCP/IP	108
4.3.1	LAS CAPAS DEL MODELO DE REFERENCIA TCP/IP	108
4.4	DISPOSITIVOS DE LAN BÁSICOS	112
4.4.1	TOPOLOGÍAS BÁSICAS	112
4.4.2	CONCEPTOS BÁSICOS DEL FLUJO DE DATOS A TRAVÉS DE LAS LAN	114
4.5	REDES DE CAMPO	116
4.5.1	TIPOS DE BUS I/O EN REDES	118
4.6	PROTOCOLOS DE LOS BUSES Y REDES DE CAMPO INDUSTRIALES	120
4.7	REDES CON BUSES DE DISPOSITIVOS BYTE-WIDE	122
4.7.1	INTERBUS-S	122
4.8	REDES CON BUSES DE DISPOSITIVOS BIT-WIDE	123
4.8.1	RED ASI	123
4.8.2	INTERBUS LOOP	123
4.8.3	SERIPLEX	123
4.9	REDES CON BUSES DE PROCESO	125
4.9.1	MODBUS	127
4.9.2	MODBUS PLUS	128
4.9.3	FIELDBUS	129
4.9.4	FOUNDATION FIELDBUS	130

4.9.5	PROFIBUS	131
4.9.6	PROFIBUS-DP	134
4.10	OPC	134
4.11	TENDENCIA DE LAS ARQUITECTURA DE APLICACIONES	136
4.12	MICROSOFT SQL SERVER Y VISUALBATCH	137
4.13	BASE DE DATOS	138

## **CAPÍTULO V : DISEÑO E IMPLEMENTACIÓN DEL SISTEMA SCADA PARA EL CONTROL DISTRIBUIDO DE LAS ESTACIONES DE CONTROL DE PROCESOS**

5.1	DISEÑO E IMPLEMENTACIÓN DE UN CONTROL DE PROCESO DE NIVEL BASADO EN FIELD POINT 1000	140
5.1.1	REQUERIMIENTOS DEL PROCESO	140
5.1.2	EQUIPO DE CONTROL A UTILIZAR	141
5.2	DISEÑO E IMPLEMENTACION DE UN CONTROL DE PROCESO DE FLUJO BASADO EN TARJETA DE ADQUISICION DE DATOS USB	147
5.2.1	REQUERIMIENTOS DEL PROCESO	148
5.2.2	EQUIPO DE CONTROL A UTILIZAR	
5.3	DISEÑO DEL CONTROL DEL PROCESO DE PRESIÓN.	
5.3.1	REQUISITOS FUNCIONALES	
5.3.2	IMPLEMENTACIÓN DEL HARDWARE	155
5.3.3	CONFIGURACIÓN DEL MÓDULO POR SOFTWARE	156
5.3.4	IMPLEMENTACIÓN DEL SOFTWARE DE CONTROL	157
5.4	DISEÑO DEL CONTROL DEL PROCESO DE TEMPERATURA	161
5.4.1	REQUISITOS FUNCIONALES	161
5.4.2	IMPLEMENTACIÓN DEL HARDWARE	163
5.4.3	IMPLEMENTACIÓN DEL SOFTWARE DE CONTROL	164
5.4.4	IMPLEMENTACIÓN DE LA INTERFASE DE COMUNICACIÓN POR SERVIDOR DE OPC	166
5.4.5	IMPLEMENTACIÓN DE LA APLICACIÓN DE CLIENTE EN	168

LABVIEW		
5.5	DISEÑO DEL SISTEMA DE SUPERVISIÓN Y REGISTRO	171
5.5.1	REQUISITOS FUNCIONALES	171
5.5.2	CONFIGURACIÓN DE TAGS	176
5.6	DISEÑO DE LA BASE DE DATOS ADMINISTRATIVA (BDD)	182
5.6.1	CONFIGURACIÓN DE LA COMUNICACIÓN DEL ODBC Y CITADEL	183
5.6.2	IMPLEMENTACIÓN DE LA BASE DE DATOS (BACK-END)	185
5.6.3	IMPLEMENTACIÓN DE LA APLICACIÓN (FRONT-END)	185

## **CAPÍTULO VI: CONCLUSIONES Y RECOMENDACIONES**

CONCLUSIONES	189
RECOMENDACIONES	192

## **BIBLIOGRAFÍA**

## INTRODUCCIÓN

En el laboratorio de Control de Procesos e Instrumentación existen cuatro estaciones de control de procesos industriales, cada una de ellas monitorea y controla en forma autónoma o asistida por un PC un proceso industrial: Temperatura, Presión, Caudal y Nivel. El control autónomo se lo realiza con un controlador digital FOXBORO Familia 760, serie 761-C; y el control asistido por PC, a través de una tarjeta de comunicaciones RS485, OPTO 24 y el software AIMAX PLUS versión 2.0.

Por tal motivo se ha planteado como objetivo fundamental de este proyecto diseñar e Implementar un sistema SCADA utilizando para ello LabVIEW DSC (BridgeVIEW) con el que se consiga monitorear, supervisar y controlar en forma distribuida los procesos industriales existentes en el laboratorio de control de procesos e Instrumentación de la ESPE sede Latacunga.

En el presente proyecto, vamos a utilizar las estaciones con un entorno particular; no se va a utilizar el controlador FOXBORO 761C, la tarjeta de comunicaciones OPTO 24, y el software de aplicación general AIMAX PLUS 2.0; es decir de las estaciones se van a utilizar solamente los medios eléctricos y electrónicos auxiliares para realizar el control de cada proceso, esto es: transmisores de temperatura, transmisores de presión absoluta, transmisores de presión diferencial, conversores I/P, válvulas neumáticas, válvulas con mando neumático, válvulas electroneumáticas, actuadores de relé, módulos de control de Triac,

Variadores de frecuencia, etc. Cada estación simula un proceso de una planta industrial y está conectada a un computador, que será quien, con ayuda de un dispositivo de control, comunicaciones y un software especializado realice el control automático del proceso en forma independiente de las demás.

Todo este sistema se colgará a una red Ethernet, a través de un circuito de red conmutado con un switch, que tendrá un computador controlador de la red distribuida y se enlazará con otro computador que simulará la operación de la red administrativa, enlazando así de esta forma todo el sistema administrativo y la red de campo.

Se ha elaborado cuatro capítulos teóricos que permiten comprender de mejor manera el funcionamiento del proyecto.

En el capítulo I se tratan los temas relacionados a los sistemas SCADA tales como definiciones, conceptos asociados, elementos constitutivos; para el entendimiento e introducción a los Sistemas SCADA.

El capítulo II trata temas relacionados al software LabVIEW DSC tales como: Filosofía de LabVIEW, elementos constitutivos del software, módulo DATA LOGGING AND SUPERVISORY CONTROL (DSC), configuraciones de alarmas y eventos, manejo de reportes históricos y de tiempo real, y manejo de servidores de niveles de seguridad para aplicaciones.

En el capítulo III se describen los dispositivos de adquisición de datos tales como el Field point FP 1000, PLC Siemens S7200, con sus respectivas características propias de configuración y manejo de datos estandarizados de corriente y voltaje; además se pone énfasis en la interconectividad que presentan los mismos con el entorno de LabVIEW.

El capítulo IV nos permite entender los temas relacionados a redes computacionales y comunicaciones industriales, los mismo que permitirán el manejo de la red industrial y administrativa de nuestro Sistema SCADA, se detallan temas como: el modelo OSI, conceptos de redes, topologías, protocolos industriales, OPC server, sistemas de bases de datos relacionales y el motor de SQL para el manejo de la base de datos administrativa.

Finalmente en el capítulo V se enfoca en las consideraciones de diseño, tanto del software como el hardware y se procede a realizar la recopilación de datos y análisis de resultados

En el transcurso del desarrollo de este proyecto, se ha ido recopilando información valiosa, las mismas que exponemos en las conclusiones y recomendaciones.

Con este proyecto estamos enlazando los tres aspectos básicos y fundamentales de la carrera de ingeniería electrónica que son: Control electrónico, Redes Informáticas y computacionales, programación y comunicaciones.

# **CAPÍTULO I**

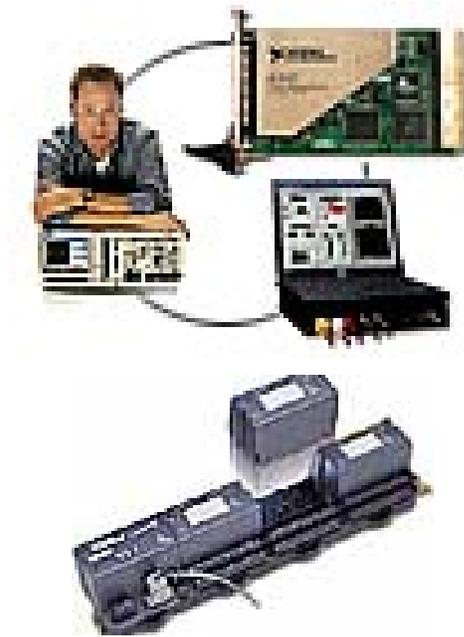
## **INTRODUCCIÓN**

### **1.1. Sistemas SCADA**

SCADA viene de las siglas de "Supervisory Control And Data Acquisition", es decir: adquisición de datos y control de supervisión. Se trata de una aplicación software especialmente diseñada para funcionar sobre ordenadores en el control de producción, proporcionando comunicación con los dispositivos de campo (controladores autónomos, autómatas programables, etc.) y controlando el proceso de forma automática desde la pantalla del ordenador. Además, provee de toda la información que se genera en el proceso productivo a diversos usuarios, tanto del mismo nivel como de otros supervisores dentro de la empresa: control de calidad, supervisión, mantenimiento, etc.

En este tipo de sistemas usualmente existe un ordenador, que efectúa tareas de supervisión y gestión de alarmas, así como tratamiento de datos y control de procesos. La comunicación se realiza mediante buses especiales o redes LAN. Todo esto se ejecuta normalmente en tiempo real, y están diseñados para dar al operador de planta la posibilidad de supervisar y controlar dichos procesos.

Los programas necesarios, y en su caso el hardware adicional que se necesite, se denomina en general sistema SCADA.



**Fig. 1 Interconexión de un sistema SCADA**

### **1.1.1 Prestaciones.**

Un paquete SCADA debe estar en disposición de ofrecer las siguientes prestaciones:

- Posibilidad de crear paneles de alarma, que exigen la presencia del operador para reconocer una parada o situación de alarma, con registro de incidencias.
- Generación de históricos de señal de planta, que pueden ser volcados para su proceso sobre una hoja de cálculo.
- Ejecución de programas, que modifican la ley de control, o incluso anular o modificar las tareas asociadas al autómeta, bajo ciertas condiciones.
- Posibilidad de programación numérica, que permite realizar cálculos aritméticos de elevada resolución sobre la CPU del ordenador.

Con ellas, se pueden desarrollar aplicaciones para ordenadores (tipo PC, por ejemplo), con captura de datos, análisis de señales, presentaciones en pantalla, envío de resultados a disco e impresora, etc.

Además, todas estas acciones se llevan a cabo mediante un paquete de funciones que incluye zonas de programación en un lenguaje de uso general (como C, Pascal, o Basic), lo cual confiere una potencia muy elevada y una gran versatilidad. Algunos SCADA ofrecen librerías de funciones para lenguajes de uso general que permiten personalizar de manera muy amplia la aplicación que desee realizarse con dicho SCADA.

### **1.1.2 Requisitos.**

Un SCADA debe cumplir varios objetivos para que su instalación sea perfectamente aprovechada:

- Deben ser sistemas de arquitectura abierta, capaces de crecer o adaptarse según las necesidades cambiantes de la empresa.
- Deben comunicarse con total facilidad y de forma transparente al usuario con el equipo de planta y con el resto de la empresa (redes locales y de gestión).
- Deben ser programas sencillos de instalar, sin excesivas exigencias de hardware, y fáciles de utilizar, con interfaces amigables con el usuario.

### **1.1.3 Módulos de un SCADA.**

Los módulos o bloques software que permiten las actividades de adquisición, supervisión y control son los siguientes:

- Configuración: permite al usuario definir el entorno de trabajo de su SCADA, adaptándolo a la aplicación particular que se desea desarrollar.

- Interfaz gráfico del operador: proporciona al operador las funciones de control y supervisión de la planta. El proceso se representa mediante sinópticos gráficos almacenados en el ordenador de proceso y generados desde el editor incorporado en el SCADA o importados desde otra aplicación durante la configuración del paquete.
- Módulo de proceso: ejecuta las acciones de mando preprogramadas a partir de los valores actuales de variables leídas.
- Gestión y archivo de datos: se encarga del almacenamiento y procesado ordenado de los datos, de forma que otra aplicación o dispositivo pueda tener acceso a ellos.
- Comunicaciones: se encarga de la transferencia de información entre la planta y la arquitectura hardware que soporta el SCADA, y entre ésta y el resto de elementos informáticos de gestión.

## **1.2. Conceptos asociados a sistemas SCADA**

En casa y en la oficina, el ordenador personal continúa con su progreso. El PC se ha establecido en un gran número de campos. Los componentes hardware y software están siendo cada vez más potentes y más rentables. Es lógico, por tanto, que la industria quiera tomar provecho de este hecho, para reducir costes y/o incrementar la productividad.

Ciertas tareas industriales están actualmente en manos de los ordenadores desde hace tiempo: desde emplear la tecnología Windows cuando se manejan pedidos y/o se ajustan parámetros de maquinaria hasta preparar o visualizar datos prácticamente de cualquier tipo.

No hay que sorprenderse entonces, que los especialistas en automatización y los usuarios estén pensando ahora en qué forma se pueden transferir al PC otras tareas, para poder llegar a un mayor ahorro. Más recientemente un gran número de simuladores de PLC (controladores lógicos programables) por software han

aparecido en el mercado, que están ayudando a transferir el control de tareas al disco duro y presentan una automatización más efectiva en costes en una simple pieza de hardware (el PC).

### **1.2.1 Tiempo real.**

La capacidad en tiempo real se refiere a la capacidad del ordenador en programas de procesamiento de datos para que siempre esté listo para procesar y proporcionar los resultados dentro de un tiempo especificado. En este contexto "estrictamente en tiempo real" significa que un sistema reacciona a los eventos externos dentro de un tiempo especificado en un 100% de los casos. Además si se habla de "tiempo real" el sistema debe responder en tiempos concretos también en un 100% de los casos. Si, de otra forma, los tiempos concretos de reacción pueden superarse en ciertos casos, como en sistemas no críticos, hablamos de "tiempo real suave".

### **1.2.2 Hardware en sistemas de supervisión: PLC y PC.**

El hecho es que las tareas automatizadas de control, visualización y computación pueden ser efectuadas por PLCs (conectados en red mediante los módulos adecuados) mejor que con sistemas exclusivos de control basados en PC. Lo que finalmente es práctico, no obstante, depende de un gran número de factores y la mayoría deben ser considerados individualmente para cada proyecto de automatización.

Así, por ejemplo, los actuales conocimientos y preferencias del usuario pueden jugar un mayor papel que la pura potencia del ordenador. Los factores cruciales, no obstante, son los atributos de capacidad en tiempo real y las propiedades de seguridad que hasta ahora han sido fuertemente asociadas con el PLC, aunque el PC también puede disponer de la característica de capacidad en tiempo real. Un sistema de control es inconcebible sin capacidad en tiempo real. Es común en

sistemas de control por ordenador tener que elegir, según las características del sistema a supervisar, entre el PLC o el PC. Se debe elegir aquel hardware que mejor se adapte a las necesidades del sistema a supervisar.

Los controladores lógicos programables, en la mayoría de los casos, están diseñados específicamente para ser empleados en ambientes industriales exigentes y han sido continuamente desarrollados de forma que sus sistemas operativos en tiempo real representan su mayor virtud. Ellos son y seguirán siendo, no obstante, la primera elección para todo control de tareas críticas o extremas por su rendimiento y simpleza, en los que un PC podría estar simplemente "sobrecargado" debido al trabajo que le pueden suponer otras tareas de ámbito común, como la gestión y visualización de datos, accesos a periféricos, bases de datos, etc.

Si, además del control de tareas, se necesita un procesamiento de datos, trabajo en red o visualización (una aplicación SCADA), un sistema basado en PC debe ser tomado en consideración. En cuanto a sistemas operativos, Windows NT, por ejemplo, no es estrictamente un sistema operativo en tiempo real como el de un PLC, pero puede actuar de forma suficientemente rápida para aplicaciones "suaves" en tiempo real, gracias a su arquitectura de micro-kernel.

### **1.2.3 Tarjetas de expansión.**

Como el sistema operativo sólo puede proporcionar respuestas suaves en tiempo real lo más simple es emplear extensiones hardware para las tareas críticas (placas de expansión PC) y soluciones software para el resto de tareas. Esto nos lleva a una compatibilidad con futuros sistemas operativos y una solución totalmente factible actualmente. Estas tarjetas de expansión asumen las tareas críticas en tiempo real que el ordenador (PC) no puede atender, se está hablando de tarjetas que incorporan DSPs (Procesadores de Señales Digitales) o microcontroladores y que aportan una ayuda a la anterior "sobrecarga" mencionada para los ordenadores (PC).

#### **1.2.4 La arquitectura abierta.**

Aún no se ha establecido un estándar para poseer extensiones compatibles en tiempo real de sistemas operativos. De una forma estrictamente determinante, los sistemas estándar actuales deben ser modificados de forma general, así que la principal ventaja de un sistema basado en PC - su estructura abierta – puede llegar a ser un inconveniente. No obstante, la estructura abierta, permite a la empresa o al desarrollador más libertad en la elección de la herramienta adecuada para el análisis, diseño y programación del sistema SCADA. La solución comienza a ser propietaria nuevamente (cada empresa ofrece su solución) y la conversión a futuras generaciones de sistemas operativos lo hace más difícil.

## **CAPÍTULO II**

### **EL SOFTWARE LabVIEW DSC.**

#### **2.1 INTRODUCCIÓN.**

LabVIEW es una herramienta diseñada especialmente para monitorear, controlar, automatizar y realizar cálculos complejos de señales analógicas y digitales capturadas a través de tarjetas de adquisición de datos, puertos serie y GPIBs (Buses de Intercambio de Propósito General).

Es un lenguaje de programación de propósito general, como es el Lenguaje C o Basic, pero con la característica que es totalmente gráfico, facilitando de esta manera el entendimiento y manejo de dicho lenguaje para el diseñador y programador de aplicaciones tipo SCADA.

Incluye librerías para la adquisición, análisis, presentación y almacenamiento de datos, GPIB y puertos serie. Además de otras prestaciones, como la conectividad con otros programas, por ejemplo de cálculo, en especial Matlab.

Está basado en la programación modular, lo que permite crear tareas muy complicadas a partir de módulos o sub-módulos mucho más sencillos. Además estos módulos pueden ser usados en otras tareas, con lo cual permite una programación más rápida y provechosa.

También ofrece la ventaja de “debugging” (puesta a punto) en cualquier punto de la aplicación. Permite la posibilidad de poner “break points” (ventanas), ejecución paso a paso, ejecución hasta un punto determinado y se puede observar como los datos van tomando valores a medida que se va ejecutando la aplicación.

### **2.1.1 Filosofía de LabVIEW.**

LabVIEW es un lenguaje completamente gráfico, y el resultado de ello es que es totalmente parecido a un instrumento, por ello a todos los módulos creados con LabVIEW se les llama VI (Instrumento Virtual).

Existen dos conceptos básicos en LabVIEW: el Front Panel (Panel Frontal) y el Block diagram (Diagrama de Bloque). El Panel Frontal es el interfaz que el usuario está viendo y puede ser totalmente parecido al instrumento del cual se están recogiendo los datos, de esta manera el usuario sabe de manera precisa cual es el estado actual de dicho instrumento y los valores de las señales que se están midiendo, El diagrama de bloques muestra la conexión de todos los controles y variables, que tendría cierto parecido al diagrama del esquema eléctrico del instrumento.

LabVIEW tiene la característica de descomposición modular ya que cualquier VI que se ha diseñado puede convertirse fácilmente en un módulo que puede ser usado como una sub-unidad dentro de otro VI. Esta peculiaridad podría compararse a la característica de procedimiento en los lenguajes de programación estructurada.

Es un sistema abierto, en cuanto a que cualquier fabricante de tarjetas de adquisición de datos o instrumentos en general puede proporcionar el driver de su producto en forma de VI dentro del entorno de LabVIEW. También es posible programar módulos para LabVIEW en lenguajes como C y C++, estos módulos son conocidos como Sub-VIs y no se diferencian a los VI creados con LabVIEW salvo por el interfaz del lenguaje en el que han sido programados. Además estos Sub-VIs son muy útiles por ejemplo en el campo de cálculos numéricos complejos que no se encuentran incluidos en las librerías de LabVIEW.

### **2.1.2 El panel frontal y el diagrama de bloques.**

Se podría decir que en cualquier VI existen dos caras bien diferenciadas: El Panel Frontal y el Diagrama de Bloques.

El Panel Frontal es la cara que el usuario del sistema está viendo cuando se está monitoreando o controlando el sistema, o sea, el interfaz del usuario. Este contiene controles e indicadores y existe una gran variedad de ellos, pero además incluso se pueden diseñar controles e indicadores personalizados, lo cual permite tener una amplia gama de dichos controles e indicadores.

Un control puede tomar muchas formas, y muchas de estas formas el dibujo real usado en instrumentos reales. Otros son estrictamente conceptos digitales o analógicos. Pero todos los controles tienen una forma visual que indican al usuario cual es el estado de dicho control en el instrumento real.

Es muy importante en un sistema SCADA que el usuario no tenga que interpretar nada, sino que todo le sea claro y conciso, las interpretaciones pueden dar lugar a falsas actuaciones y, por consiguiente, podrían existir lamentables errores. Además, dos usuarios podrían interpretar de manera diferente cualquier evento.

El Diagrama de Bloques del VI sería la cara oculta del Panel Frontal, una cara que el usuario del sistema no puede ver. En ella están todos los controles e indicadores interconectados, pareciéndose mucho a un diagrama de esquema eléctrico. Esta cara es mucho menos conceptual que el Panel Frontal y para el usuario sería muy difícil entenderla.

Todos los módulos están interconectados, mediante líneas de conexión, por donde circulan los diferentes datos o valores del VI., de esta manera se logra que el VI funcione como un conjunto de elementos, módulos y sub-módulos.

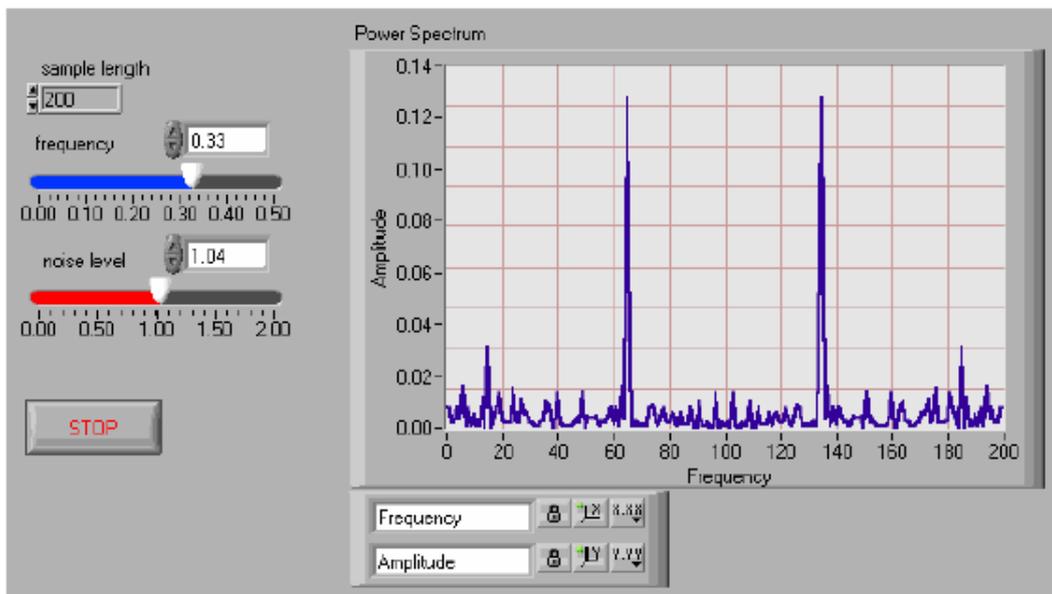
### **2.1.3 Lab VIEW como entorno básico de Instrumentación Virtual.**

LabVIEW es una parte integral de la instrumentación virtual dado que proporciona un medio ambiente de desarrollo de aplicaciones que es fácil de utilizar y está diseñado específicamente teniendo en mente las necesidades de ingenieros y científicos. LabVIEW ofrece poderosas características que facilitan la conexión a una gran variedad de hardware y otros softwares.

### a) Programación gráfica.

Una de las características más poderosas que LabVIEW ofrece a los ingenieros y científicos es un medio ambiente de programación que es gráfico. Con LabVIEW se puede diseñar instrumentos virtuales a medida creando interfases gráficas de usuario en la pantalla de la computadora con la cual puede:

- Operar el programa de instrumentación.
- Controlar el hardware seleccionado.
- Analizar datos adquiridos.
- Visualizar los resultados.



**Figura 2.1 Panel Frontal de un Instrumento Virtual hecho en LabVIEW**

Se pueden personalizar paneles frontales con perillas, botones, diales y gráficos a fin de emular paneles de control de instrumentación tradicionales, crear paneles de ensayo personalizados o representar visualmente el control y operación de procesos, como se muestra en la figura 2.1. La similitud existente entre los diagramas de flujo y los programas gráficos acorta la curva de aprendizaje asociada con lenguajes tradicionales basados en texto.

Se puede también determinar el comportamiento de los instrumentos virtuales conectando iconos entre si para crear diagramas de bloques, que son notaciones de diseño naturales para ingenieros y científicos. Con un lenguaje gráfico se puede desarrollar sistemas más rápidamente que con lenguajes de programación convencionales mientras que conserva la potencia y flexibilidad necesarias para crear una variedad de aplicaciones. La figura 2.2 muestra un diagrama de bloques construido en LabVIEW.

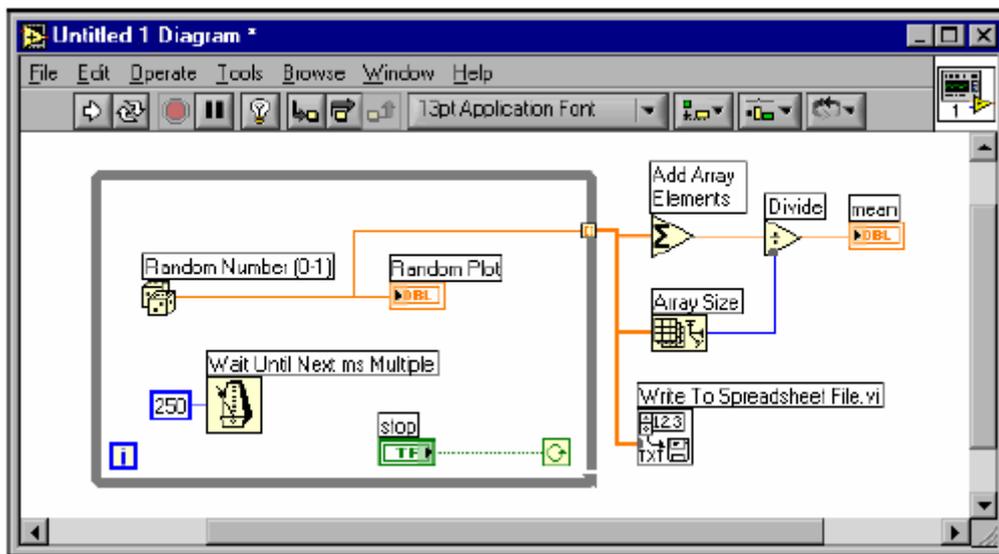


Figura 2.2 Diagrama de bloques de un Instrumento virtual hecho en LabVIEW

## b) Conectividad y control de instrumentos.

La productividad del software de instrumentación virtual es tal que ya incluye el conocimiento de la integración del hardware. Diseñado para crear ensayos, mediciones y control de sistemas, el software de instrumentación virtual incluye una extensa funcionalidad para entradas y salidas prácticamente de cualquier tipo.

LabVIEW posee bibliotecas listas para ser utilizadas con el objeto de integrar instrumentos autónomos, equipos de adquisición de datos, productos para el control de movimientos y de imágenes, instrumentos GPIB/IEEE 488, seriales RS-232 y PLC's, entre otros, lo cual permite construir una solución completa de medición y automatización. LabVIEW también tiene incorporadas las más

importantes normas de instrumentación, tal como VISA, una norma que permite la operación entre instrumentos GPIB, seriales y VXI.

### **c) Medio ambiente abierto.**

Aunque LabVIEW proporciona las herramientas requeridas para la mayoría de las aplicaciones, también constituye un medio ambiente de desarrollo abierto. El cumplimiento de normas por parte del software se basa principalmente en la habilidad del paquete que se eligió para trabajar bien con otros sistemas de software y hardware de medición y control, y normas abiertas, que definen la capacidad de interactuar entre múltiples fabricantes. Seleccionando el software que cumple con estos criterios se asegura que la empresa y las aplicaciones aprovechen los productos ofrecidos por numerosos proveedores. Además, ajustándose a normas comerciales abiertas, se reduce el costo total del sistema.

Un gran número de fabricantes de hardware y software desarrollan mantienen centenares de bibliotecas de LabVIEW y manejadores de instrumentos que le ayudan a utilizar fácilmente sus productos con LabVIEW. Sin embargo, esa no es la única forma de proporcionar conectividad a las aplicaciones basadas en LabVIEW. Este software ofrece maneras simples de incorporar programas en ActiveX, bibliotecas dinámicas (DLL's) y bibliotecas compartidas de otras herramientas. Además se puede compartir código hecho en LabVIEW como un DLL, construir un programa ejecutable o utilizar ActiveX.

LabVIEW también ofrece un completo rango de opciones de comunicaciones y estándares de datos como TCP/IP, OPC, SQL y formato de datos en XML.

### **d) Reducción de costos y preservación de la inversión.**

LabVIEW es un producto versátil dado que se puede utilizar una sola computadora equipada con LabVIEW para innumerables aplicaciones y propósitos. No sólo es versátil sino también extremadamente efectivo desde el punto de vista del costo. La instrumentación virtual con LabVIEW demuestra ser económica, no sólo por los reducidos costos de desarrollo sino también porque

preserva la inversión del capital a lo largo de un extenso período. A medida que cambian las necesidades, se pueden fácilmente modificar los sistemas sin necesidad de adquirir nuevo equipamiento y crear bibliotecas enteras de instrumentación a costo menor que el correspondiente a un solo instrumento comercial tradicional.

#### **e) Plataformas Múltiples.**

La mayoría de los sistemas computacionales utilizan alguna variante del sistema operativo Microsoft Windows<sup>®</sup>; no obstante ello, existen otras opciones que ofrecen claras ventajas para cierto tipo de aplicaciones. El desarrollo de sistemas operativos de tiempo real y embebido continúa creciendo rápidamente en la mayoría de las industrias a medida que la capacidad de cálculo es incorporada en paquetes más especializados y pequeños. Es importante minimizar las pérdidas resultantes del cambio hacia nuevas plataformas y la elección del software correcto para dicho objetivo es un factor clave.

LabVIEW minimiza esta preocupación ya que corre en Windows 2000, NT, XP, Me, 98, 95 y NT embebido, así como también sobre Mac OS, Sun Solaris y Linux. Con LabVIEW también se puede compilar código que corra en el sistema operativo de tiempo real VenturCom ETS a través del módulo LabVIEW Real-Time. Dada la importancia de los sistemas legacy, National Instruments continúa poniendo a disposición versiones más antiguas de LabVIEW para los sistemas operativos Windows, Mac OS y Sun. LabVIEW es independiente de la plataforma seleccionada; los instrumentos virtuales que se creen en una plataforma pueden ser transportados de manera transparente a cualquier otra plataforma LabVIEW simplemente abriendo el instrumento virtual.

Puesto que las aplicaciones de LabVIEW pueden transportarse entre plataformas, se puede asegurar que el trabajo de hoy será utilizable en el futuro. A medida que emerjan nuevas tecnologías computacionales, se puede migrar fácilmente las aplicaciones a nuevas plataformas y sistemas operativos. Además debido a que se pueden crear instrumentos virtuales que son independientes de la plataforma y

transportarlos entre distintas plataformas, puede ahorrarse tiempo de desarrollo y otros inconvenientes relacionados con la portabilidad entre plataformas.

#### **f) Desarrollo distribuido.**

Con LabVIEW se puede desarrollar fácilmente aplicaciones distribuidas, aún entre diferentes plataformas. Con herramientas de servidores fáciles de usar, se pueden descargar rutinas que hacen uso intensivo del procesador a otras máquinas para lograr una ejecución más rápida, o crear aplicaciones de monitoreo y control remoto. Una tecnología de servidores poderosa puede simplificar la tarea de desarrollar aplicaciones grandes y que requieran de múltiples computadoras. Además de ello, LabVIEW incluye tecnologías normalizadas de redes, tales como TCP/IP e incorpora protocolos robustos de publicación y suscripción.

#### **g) Capacidades de Análisis.**

El software de instrumentación virtual requiere complejas herramientas de análisis y procesamiento de señales ya que la aplicación no se detiene justo cuando el dato es recogido. Usualmente, las aplicaciones de medición de alta velocidad en los sistemas de monitoreo y control de maquinaria requieren análisis de orden para lograr obtener datos específicos. Los sistemas de control embebidos de lazo cerrado podrían requerir premedicación punto a punto para lograr que los algoritmos de control mantengan la estabilidad. Además de las bibliotecas de análisis avanzado ya incluidas en LabVIEW, National Instruments proporciona software adicional, tal como el LabVIEW Signal Processing Toolset (Paquete de herramientas para el procesamiento de señales en LabVIEW) a fin de complementar las ofertas de análisis.

#### **h) Capacidades de visualización.**

LabVIEW incluye también un amplio conjunto de herramientas de visualización para presentar datos en la interfase del usuario de la instrumentación virtual, tanto para gráficos continuos como también para visualización de gráficos en 2D y 3D.

Se puede reconfigurar de manera instantánea los atributos de la presentación de datos, tales como: colores, tamaño de fuentes, tipos de gráfico y más, así como también efectuar rotación, enfoque (zoom) y desplazamiento dinámico en los gráficos con el ratón. En lugar de programas gráficos y todos los atributos corrientes desde cero, simplemente se arrastra y coloca los objetos dentro de los paneles frontales de los instrumentos.

### **i) Flexibilidad y escalabilidad.**

Los ingenieros y científicos tienen necesidades y requerimientos que cambian rápidamente. Ellos también necesitan tener soluciones extensas que puedan ser mantenidas y utilizadas por un largo período. Creando instrumentos virtuales basados en un software de desarrollo poderoso, como es LabVIEW, se puede diseñar un ambiente de trabajo abierto que se integre de modo ininterrumpido con el software y el hardware. Esto asegura que sus aplicaciones no sólo funcionarán bien hoy, sino también que se podrán utilizar nuevas tecnologías en el futuro a medida que ellas se hallen disponibles, o extender sus soluciones más allá de su alcance original a medida que se identifiquen nuevos requerimientos. Más aún cada aplicación posee sus propios y únicos requerimientos que precisan una amplia gamma de soluciones.

## **2.2 MÓDULO LabVIEW DATALOGGIN AND SUPERVISORY CONTROL (DSC).**

### **2.2.1 Introducción.**

El Módulo de Control Supervisión y Registro de Datos LabVIEW (DSC) añade características y capacidades a LabVIEW para ayudarle a crear aplicaciones automáticas o aplicaciones que podrían monitorear un largo número de datos. El Módulo de LabVIEW DSC proporciona soluciones para control de supervisión de una amplia variedad de sistemas distribuidos, usando la flexibilidad de programación gráfica de LabVIEW. Antes de usar el Módulo LabVIEW DSC, se debe estar familiarizado con la programación en LabVIEW.

Se utilizará el Módulo de LabVIEW DSC para aplicaciones típicas como: cambiar setpoints o enviar instrucciones de control a dispositivos individuales mientras

esté supervisando el sistema entero. El Módulo LabVIEW DSC proporciona las siguientes características y capacidades:

- Configuración de utilidades y asistentes.
- Base de datos en tiempo real.
- Colección histórica de datos y direcciones.
- Reportes y registros de alarma y evento.
- Seguridad.
- Conexión para PLC y Redes de Dispositivos Industriales.
- Conexión OPC en ambiente cliente servidor.
- Conexión para una amplia selección de dispositivos servidores.

El Modulo LabVIEW DSC instala las siguientes subpaletas en la paleta de controles (panel frontal), como se muestra en la figura 1-1:

- **Recipientes (Vessels).**- Varios tanques, un depósito, y una caja. Todos estos objetos en la subpaleta son numéricos.
- **Tuberías, Bombas y Válvulas (Pipes, Pumps, & Valves).**- Varios objetos como tuberías, bombas, y válvulas booleanas.
- **Alarmas y Eventos.**- Un juego de displays, Herramientas de formato, y un conjunto de herramientas para manejar alarmas y eventos.
- **Direcciones (Trends).**- Tendencias histórica de un gráfico XY y tendencias en tiempo real de un gráfico waveform chart.
- **Tipos de Datos para el Servidor DSC.**- Varias definiciones de tipos de datos que se puede usar para instrumentos virtuales basados en arquitectura cliente servidor.

La figura 2.3 muestra los elementos que se adicionan a la paleta de controles en el panel frontal al instalar el módulo DSC.

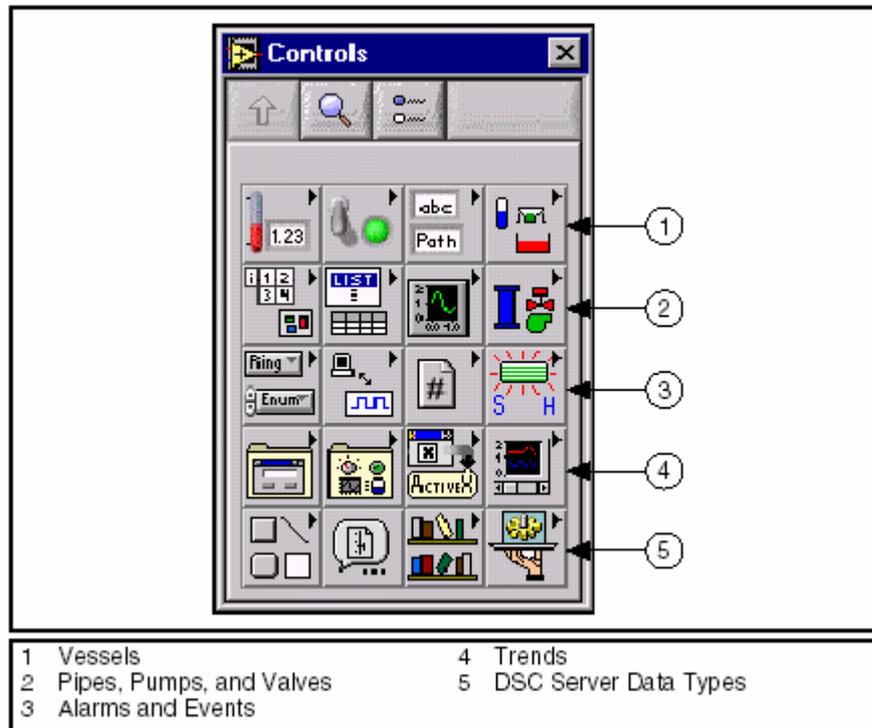


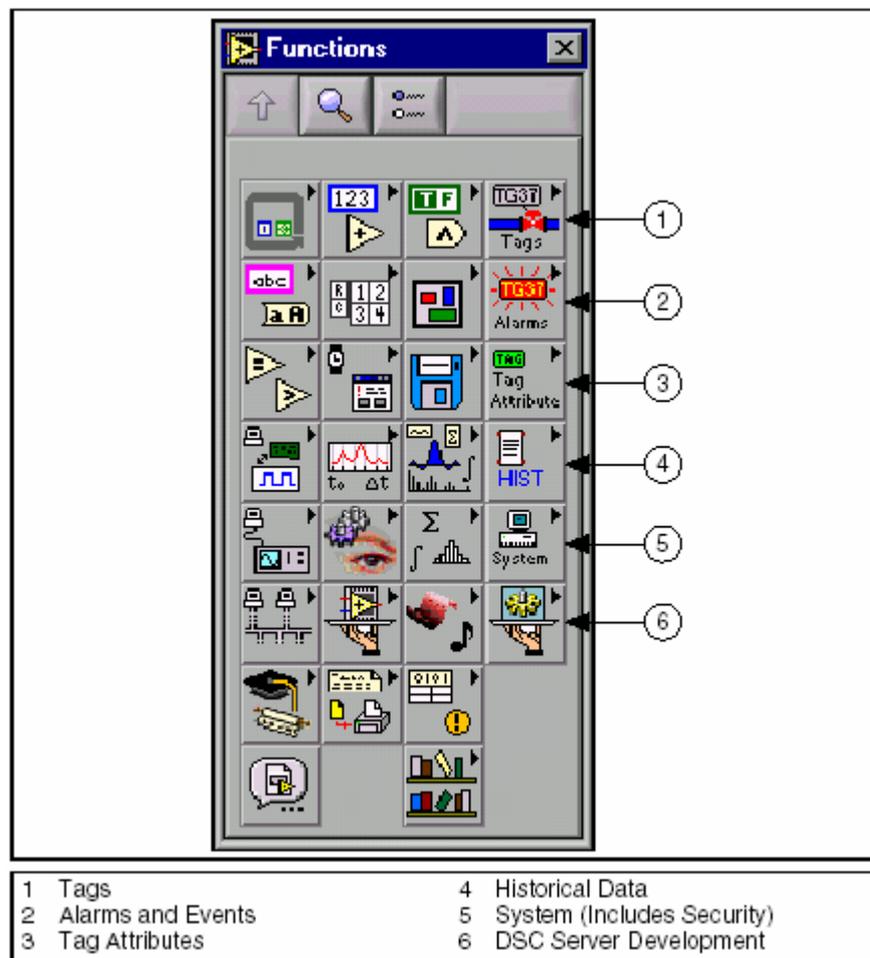
Figura 2.3 Paleta de controles modificada por LabVIEW DSC.

El Módulo LabVIEW DSC también instala una función llamada Tag Control en la paleta Controls»I/O. Use el tag control para seleccionar un tag desde una configuración activa (.scf).

El módulo de LabVIEW DSC también instala las siguientes subpaletas en la paleta de funciones , como se muestra en la figura 2.4.

- **Tags.**- Use estos VIs y funciones para leer los más recientes valores de un tag, escribir un nuevo valor para un tag, u obtener datos para una dirección en tiempo real.
- **Alarmas & Eventos.**- Use estos VIs para reconocer alarmas, desplegar el resumen de la alarma o la información histórica del evento, u obtener el estado de la alarma.
- **Características de un Tag.**- Use estos VIs para leer y escribir en los parámetros de configuración de un tag programáticamente.

- **Datos Históricos.**- Use estos VIs para leer y escribir datos históricos acerca de un tag, para direcciones de datos y muestras, calcular datos estadísticos para una dirección histórica, o convertir datos de una dirección histórica a una hoja de cálculos.
- **Sistema.**- Use estos VIs para activar o parar el motor de Tag, para habilitar o deshabilitar registro de eventos, para imprimir o registrar datos históricos, o para manipular el ambiente de módulo LabVIEW DSC.
- **Sistema»Seguridad.**- Use estos VIs para acceder a la información de seguridad acerca de su aplicación.
- **Desarrollo del Servidor DSC.**- Use estos VIs para desarrollar instrumentos virtuales basados en dispositivos servidores.



**Figura 2.4 Paleta de funciones modificada por LabVIEW DSC**

La figura 2.4 muestra los VI's introducidos en la paleta de funciones con la instalación del módulo LabVIEW DSC.

### 2.2.2 Desarrollando una aplicación.

Para desarrollar una aplicación en el módulo LabVIEW DSC, se debe estar familiarizado con la programación gráfica en LabVIEW. El Módulo de aplicación LabVIEW DSC contiene tres partes que trabajan juntas: la interfase gráfica o aplicación HMI, el motor de Tag, y varios dispositivos servidores, como se muestra en la figura 2.5. El motor de Tag , con cualquier dispositivo servidor, corre como un proceso independiente separado de la aplicación HMI.

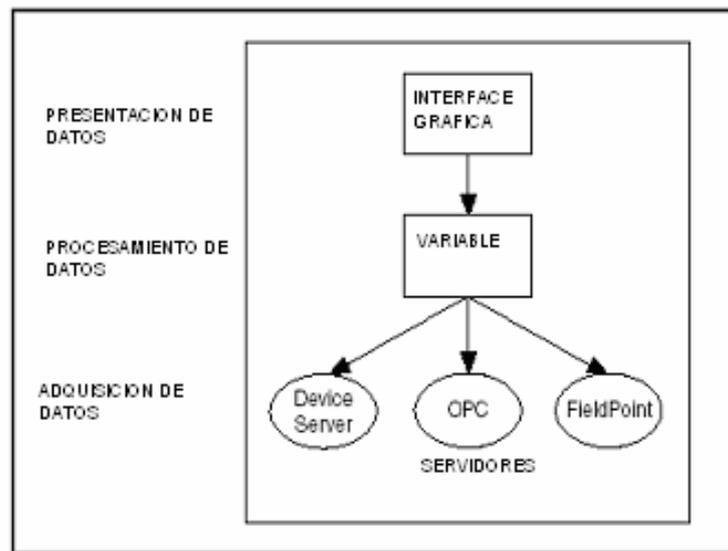


Figura 2.5 Diagrama de partes del módulo DSC.

Los procedimientos básicos que se necesitan para desarrollar una aplicación con el módulo LabVIEW DSC son los siguientes:

- Requiere de un plan específico para la aplicación.
- Determinar el hardware que se va a utilizar.
- Establecer una conexión física a su hardware. Conectar, configurar, probar, y corregir.
- Instalar y configurar servidores para conectar los puntos de I/O. Esto crea el mecanismo para manejar la comunicación con el hardware.
- Cuando sea posible, antes que se acceda a los servidores en LabVIEW, hay que probarlos según las instrucciones en el documento del servidor.

- Usar el editor de configuración del Tag y la configuración Wizard del Tag para crear y configurar tags y guardar la configuración en archivos (.scf).
- Usar el Monitor Tag para probar los servidores y tags.
- Usar LabVIEW y el módulo LabVIEW DSC, VIs y funciones para crear la interfase de usuario o aplicación HMI y cualquier otra funcionalidad. Crear la aplicación HMI de la misma manera que puede crear cualquier aplicación normal en LabVIEW.
- Desplegar, probar, y documentar la aplicación.

### 2.3 SERVIDORES.

En el módulo LabVIEW DSC, un *dispositivo servidor* es una aplicación que comunica y maneja instrumentos de I/O tales como PLCs, dispositivos de I/O remotos, motor de Tag remota, y DAQ. Estos servidores leen elementos de entrada seleccionados y escriben en ellos. El módulo LabVIEW DSC puede conectarse a cualquier servidor OPC y a muchos dispositivos servidores terciarios. También se puede conectar a servidores de la National Instruments, incluyendo DAQ de National Instruments y servidores FieldPoint. Un elemento servidor es un canal, punto de I/O, o una variable en el hardware de un dispositivo. Se asocian a estos elementos con tags. Los Instrumentos servidores monitorean los valores adquiridos por el hardware y el motor de Tag actualiza los tags cuando el servidor envía nuevos datos al motor de Tag. Los Servidores también actualizan cada salida cuando la aplicación HMI escribe un valor del tag, y ellos manejan y reportan las comunicaciones y errores del dispositivo. Un buen dispositivo servidor cubre todo el entorno y detalla las especificaciones del hardware, así establece un dispositivo independiente de I/O para el módulo LabVIEW DSC.

Cuando se ejecuta una aplicación en el módulo LabVIEW DSC éste determina desde el archivo (.scf) la configuración que los servidores necesiten y cuales elementos son necesitados desde estos servidores. El módulo LabVIEW DSC da a cada servidor lo que necesita y monitorea los elementos específicos de interés a través del motor de Tag. El módulo LabVIEW DSC también puede funcionar como un servidor OPC y como una fuente de datos para un protocolo de red.

Un servidor no es lo mismo que un manejador de dispositivo o que un manejador de instrumentos. En general, un manejador de instrumentos es un software que está designado para controlar un instrumento programable tal como un multímetro. Un manejador de dispositivo es típicamente un componente de software que una computadora necesita para trabajar con una interfase. Un manejador también puede funcionar como un servidor si cumple ciertas características, tales como especificaciones de OPC.

### 2.3.1 Tipos de servidores a seleccionar.

El módulo LabVIEW DSC contiene soporte para varios tipos de servidores incluyendo los siguientes:

- **Servidores OPC.-** Conforme con la fundación OPC, los datos OPC acceden a las especificaciones 2.0 y anteriores.
- **Servidores DDE.-** Cualquier servidor que soporte la interfase de intercambio dinámico de datos (DDE).
- **Dispositivo servidor IA.-** Es un tipo de servidor desarrollado por National Instruments. Los dispositivos servidores IA tienen dos aplicaciones: basados en VI Y basados en DLL.
  - **Servidores basados en VI.-** Usa VIs para proporcionar datos al motor de Tag.
  - **Servidores basados en DLL.-** También conocidos como Servidores de Automatización Industrial (IAS).

Se puede también usar servidores proporcionados por un tercer fabricante de hardware, y además se puede construir un servidor basado en VI cuando se tenga una sola configuración o un dispositivo establecido, el cual se desee utilizar con el módulo LabVIEW DSC.

### **2.3.2 Servidores IAK.**

Antes de la existencia de Windows 2000, National Instruments usó servidores IAK y OPC. Sin embargo, con el apareamiento de Windows 2000, usando los servidores IAK basados en DLL hubieron problemas por razones de compatibilidad. Para ser compatible con las versiones futuras de Windows, National Instruments recomienda que use servidores OPC, específicamente los servidores OPC para NI-DAQ (incluyendo SCXI), FieldPoint de National Instruments, y Foundation Fieldbus de National Instruments.

Se pueden adquirir separadamente un CD de servidores de National Instruments, el cual incluye el Manejador del Protocolo Lookout para servidores OPC. Puede usar el Manejador del Protocolo Lookout para servidores OPC para conectarse a una gran variedad de dispositivos a través de protocolos de interfase OPC. Estos dispositivos incluyen Allan-Bradley, Siemens, Modbus, y otros más. El manejador del protocolo Lookout para servidores OPC también incluye un gran tiempo de prueba, y los manejadores son completamente comprobados. Se puede utilizar los manejadores de Lookout para reemplazar a los servidores IAK si es necesario. El CD del Servidores de National Instruments también incluye un parche que permite a los servidores IAK correr en Windows 2000. Este parche es preparado solo para mantener compatibilidad, sin garantizar que los servidores IAK puedan continuar trabajando en versiones futuras de Windows.

### **2.3.3 Proceso de instalación y configuración de los servidores.**

Después que se ha seleccionado el dispositivo servidor que usará con su respectivo hardware, se instala y configura de acuerdo a la información del servidor.

Para muchos servidores, se debe utilizar el utilitario de configuración de servidores para configurar la forma de trabajo de todos los ítems de monitoreo, incluyendo las llamadas a los dispositivos y otros parámetros importantes de configuración.

### **2.3.4 Registro de servidores.**

Es necesario registrar sus dispositivos servidores manualmente para que el módulo LabVIEW DSC pueda acceder a ellos.

#### **a) Servidores OPC.**

Si un servidor cumple con las especificaciones de OPC, este debería registrarse por si mismo de acuerdo a esta especificación. Si un servidor OPC no aparece en cajas de texto (listbox) de los servidores en la configuración Wizard del Tag no podrá ser utilizado.

Si se cambia el registro del servidor mientras el Editor de configuración del Tag está abierto, el cambio no aparece en el listbox de los servidores. Para actualizar el listbox de los servidores mientras el editor de configuración del Tag este abierto, se selecciona: Servidores\Refrescar.

#### **b) Servidores DDE.**

No se necesita registrar servidores OPC.

#### **c) Servidores basados en VI.**

Los Servidores basados en VI incluyen un VI que se usa para registrar el servidor. Antes que pueda usar un servidor basado en VI, puede correr este registro VI. El módulo LabVIEW DSC instalará servidores basados en VI, los cuales son usados en los ejemplos del módulo de LabVIEW DSC. Estos servidores incluyen servidores tanques, el servidor SIM, y el servidor cookie. Ellos residen en el archivo: labview\examples\lvdsc\servers.

### **2.3.5 Carga del utilitario de configuración de servidores.**

Cuando se registra un dispositivo servidor basado en VI o un dispositivo servidor IAK en su sistema, el Modulo LabVIEW DSC también registra la dirección de esta

configuración. En Windows NT, es necesario que se registre como administrador para acceder a las utilidades de configuración del servidor.

También puede abrir las utilidades de configuración del servidor registrándose desde el editor de configuración del Tag, seleccionando: Servers»*Server Name* Configuration, donde *Server Name* es el nombre de el servidor.

### **2.3.6 Revisión de la configuración del servidor.**

Usar el Servidor Browser para observar la información acerca de los dispositivos servidores configurados en su sistema y en la red. También se puede usar el Servidor Browser para ver ciertas propiedades de servidores OPC y basados en VI. Además también puede usar el manejador del motor para ver información acerca de la ejecución de los servidores.

### **2.3.7 Uso LabVIEW DSC como un servidor OPC.**

LabVIEW sin el módulo DSC puede actuar con un cliente OPC a través de la conexión DataSocket connection. Sin embargo, LabVIEW con el módulo DSC, el motor de tag puede actuar de ambas formas como cliente OPC y servidor OPC.

Ya que es un servidor OPC, un cliente OPC puede arrancar el motor de Tag cuando el cliente OPC arranca el motor de Tag y se carga en un archivo con extensión .scf. El motor de Tag puede entonces cargar cualquier servidor que el tag requiera y se almacena en el archivo .scf. Sin embargo un cliente de OPC no puede especificar que archivo .scf debe usar el motor de tag, este motor puede utilizar solo el archivo actualizado, de tal forma que se debe asegurar que el archivo .scf sea el correcto.

Cuando se ingresa datos desde el motor de Tag que actúa como un servidor OPC, un software cliente, como Lookout o algún otro software cliente de OPC, este ve el motor de tag como un servidor OPC llamado "National

Instruments.OPCLabVIEW. donde los nombres que se fija en el archivo .scf aparece como nombres de items OPC.

Si el archivo .scf contiene tags de memoria que manejan valores calculados por uno de los VI's, aquellos valores no se actualizan a menos que se haga correr LabVIEW manualmente.

### **2.3.8 Ingreso al servidor OPC usando LabVIEW DSC como cliente OPC.**

El módulo LabviewDSC puede funcionar como un cliente OPC y comunicarse con cualquier servidor que implementa la interfase del servidor OPC de la fundación OPC, que es una plataforma basada en el estándar COM de Microsoft. El módulo LabVIEW DSC encuentra todos los servidores OPC instalados en su computadora y lee cualquier información sobre capacidades del servidor y los items manejados directamente por el servidor.

#### **a) Configuración de LabVIEW DSC como cliente OPC.**

Es necesario configurar el módulo LabVIEW DSC como cliente OPC por las siguientes razones:

- Si el servidor no soporta la comunicación asincrónica, se puede forzar al cliente para usar comunicación sincrónica
- Si el servidor tiene muchos items OPC se puede aumentar el número máximo de items que se despliega mientras que se lee el servidor.

Si se ejecuta muchas escrituras en el servidor y recibe un sobreflujo de salida, se puede incrementar la longitud de la fila de escrituras asincrónicas completas al servidor OPC.

### **2.3.9 Uso de LabVIEW DSC con servidores DDE.**

El módulo LabVIEW DSC puede comunicarse con cualquier servidor DDE como interfase. Un servidor DDE es un servidor en el cual se escribe un dispositivo o ítem para seleccionar un punto de datos específicos a la cual se conecta.

Terceros servidores DDE no se registran con el módulo LabVIEW DSC, Entonces, el módulo no puede arrancar el servidor DDE automáticamente cuando se hace funcionar una aplicación HMI.

Para usar el servidor DDE, se debe ejecutar el servidor DDE antes que arranque el motor de tag.

El módulo LabVIEW DSC emitirá mensajes de error de sistema si no puede conectarse al servidor DDE cuando arranca el motor de Tag. Después de esto, intenta reconectarse al servidor periódicamente.

## **2.4 USO DE TAGS PARA MANEJO DE ENTRADAS Y SALIDAS.**

En el Módulo LabVIEW DSC se puede usar un *tag* para crear y mantener una conexión con entradas y salidas exteriores. Se puede también usar un *memory tag* para datos que por su aplicación necesita usar o realizar trazos. Un *network tag* es una etiqueta concertada remotamente o cualquier tipo de etiqueta residente otra motor de tag. Las tareas que se hace a través de etiquetas dependen de como se configura los atributos de Tag (etiqueta). Los atributos de tag son las maneras en que los tags se pueden actualizar y escalar, y las formas de cómo, cuándo y dónde los tags son almacenados en una base de datos histórica, los niveles de alarma y las prioridades de los datos en los tags.

Además, se puede organizar etiquetas en grupos lógicos para conveniencia y eficacia, configurar el tipo de datos de etiqueta, fijar valores de iniciación, fijar bandas muertas separadas para cargar o actualizar datos, pegar unidades de medidas a los datos, pegar un mensaje de alarma a una etiqueta cuyos valores

ingresan rangos de alarmas y colocar bandas de tolerancia de alarma desde la carga y actualizar bandas muertas.

#### **2.4.1 Configuración de Archivos.**

Después de crear etiquetas y configurar sus atributo, se graba la información en un archivo de configuración (.scf) Cualquier utilitario del módulo LabVIEW DSC que necesita información acerca de una etiqueta utiliza un archivo .scf. Estos utilitarios incluyen motores de etiqueta, monitores de etiqueta y HMI Wizard, los cuales generalmente acceden al archivo .scf para encontrar una lista de etiquetas activas y otra información de configuración.

El archivo .scf no contiene ninguna información sobre el HMI. En efecto, no necesita especificar ninguna aplicación independiente. Múltiples aplicaciones pueden correr concurrentemente usando el mismo .scf.

#### **2.4.2 Crear Tags.**

Se pueden crear etiquetas usando los siguientes procedimientos:

- Generar Tags automáticamente en el utilitario Tag Configuration Wizard.
- Crear Tags manualmente en el utilitario Tag Configuration Monitor.
- Crear Tags en el HMI Wizard, pero antes se debe enlazar un objeto del panel frontal de la aplicación con un nombre o una etiqueta.

##### **a) Crear etiquetas automáticamente.**

Se usa el Tag Configuration Wizard para generar etiquetas desde el servidor, si se quiere que el motor de Tag monitoree un gran numero de puntos de I/O en su sistema. Cuando se ejecuten los utilitarios de configuración del servidor para todos los servidores del sistema, se puede definir dispositivos e items para puntos de I/O que el servidor monitorea y controla.

El asistente (Wizard) usa el nombre de la etiqueta, tipo de dato, grupo I/O, conexión I/O y atributos de escala, para cada ítem del servidor para crear las etiquetas. Para servidores IAK y servidores basados en VI's, el asistente lee la información del servidor desde la base de datos de configuración común (CCDB).

**b) Crear etiquetas manualmente.**

- Cuando se genera etiquetas, se las puede añadir una configuración existente o se puede crear un nuevo archivo de configuración (.scf), y se puede cambiar luego manualmente la configuración de cualquier etiqueta.

**c) Crear etiquetas con el asistente HMI Wizard.**

Se puede crear una etiqueta haciendo click con el botón derecho de un control o indicador del panel frontal y luego ejecutar el HMI Wizard.

### **2.4.3 Editar Tags.**

Utilizando el módulo LabVIEW DSC se pueden editar tags de la siguiente manera:

**a) Editar etiquetas manualmente.**

**b) Editar etiquetas programáticamente.**

**c) Editando etiquetas en una hoja de cálculo.**

Con el utilitario Tag Configuration Editors se puede exportar información de configuración una etiqueta a archivos tipo hoja de cálculo o importar datos desde un archivo de este tipo como configuración de una etiqueta. Los archivos pueden tener extensión .txt con delimitadores como tabuladores y fin de líneas.

Si se quiere usar archivos de hoja de cálculo con el editor de configuración de etiqueta es importante que se considere lo siguiente:

- Si no se seleccionan todos los campos cuando se exporta datos desde la hoja de cálculo, se pierde información de configuración del tag al importarlos de regreso al Tag Configuration Editor.

- Se puede exportar una parte de la información y luego llenar los parámetros que faltan en etiqueta cuando se importan de regreso al Tag Configuration Editor. Sin embargo cada fila en el archivo de la hoja de cálculo debe contener el nombre de la etiqueta y los campos de tipo de datos.
- Algunos parámetros de configuración, tales como aquellos que residen en las cajas de diálogo Historical Loggin Configuration y en Event Configuration son inherentes y están directamente relacionados al archivo .scf actualmente abierto.
- Al importar se pueden adicionar tags al archivo de configuración .scf actual y que se encuentra cargado.
- Si se crea un archivo de hoja de cálculo para importar la configuración de una etiqueta, use el mismo formato del archivo creado al exportar una configurando de etiqueta existente.
- De la misma forma puede importar la configuración de un tag desde un archivo formateado como hoja de cálculo.

#### **2.4.4 Borrar Tags.**

- Se pueden borrar los tags configurados en una aplicación realizada con el módulo LabVIEW DSC.

#### **2.4.5 Configurar los atributos de las etiquetas.**

Los atributos de etiqueta fijan la forma de cómo el motor de tags maneja una etiqueta. Hay cinco categorías de atributos de etiqueta: General, Connection, Operations, Scaling y Alarmas. Cuando se crea una etiqueta usando el asistente de tags (Tag Wizard), se asignan los valores por defecto para cada atributo de etiqueta.

Además se visualizan las opciones de los tag's:

- **General.**- Atributos tales como nombre de la etiqueta, grupo y descripción.
- **Connection.**- Atributos que describen donde el motor manejador de tags envía o recibe valores de la etiqueta y como se acceden a los datos.  
Estas etiquetas tienen acceso directo a entrada, salida o entrada/salida. Las etiquetas de memoria no son conectadas a un punto de I/O; se deben fijar a sus accesos en forma específica.
- **Operations.**- Atributos que describen funcionalidad adicional que el motor de tags desarrolla en una etiqueta o sus valores.
- **Scaling:** Atributos que describen que funciones de escalamiento es aplicada a una etiqueta.
- **Alarms:** Atributos que describen condiciones anormal de proceso para una etiqueta.

#### 2.4.6 Tipos de datos de Tags.

La forma en que se configura una etiqueta varía levemente dependiendo del tipo de datos. Los tipos de datos de las etiquetas pueden ser: análogos, discreto, arreglos de bits y etiquetas cadenas de caracteres.

##### a) Tags Análogos.

Un *analog tag* es una representación de valor continuo de una conexión a un punto de I/O o variable de memoria. Este tipo de etiqueta puede variar continuamente sobre un rango de valores dentro de un rango determinado de una señal.

##### b) Tags Discretos.

Un *discrete tag* (Control o Indicador booleano en LabVIEW) es una representación de dos estados (prendido/apagado) de un punto de I/O o variable de memoria.

### **c) Tags de arreglos de bits.**

Un *bit array tag* es una representación de valor de multi bits de una conexión a un punto de I/O o variable de memoria. Este tipo de etiqueta puede comprender hasta 32 valores discretos.

### **d) Tags de cadena de caracteres.**

Un *string tag* es un ASCII o la representación del carácter binario de una conexión a un punto de I/O o variable de memoria.

## **2.4.7 Grupos de Tags.**

Se usa grupos de etiquetas para definir un subconjunto de etiquetas en el sistema. Se puede usar grupos de tags para examinar estados de alarma para un conjunto de tags, es decir un conjunto de variables.

Los grupos I/O son usados para configurar velocidades de trabajo y bandas muertas para varios ítems de un servidor y para seleccionar un dispositivo específico, si es que el servidor usa dispositivos. Para servidores que usan configuración de fuente, también se puede usar grupos I/O para configurar dispositivos y fuentes de comunicación. Para servidores OPC, un grupo I/O conforma el concepto de un grupo OPC, el cual es definido por el usuario y su tiempo de control. Cada grupo I/O crea mapas a un grupo OPC, en el servidor OPC con los mismos atributos. Un grupo I/O está asociado con un solo servidor y si ese servidor usa dispositivos con un solo dispositivo. Un servidor puede tener grupos múltiples I/O asociados con él.

## **2.4.8 Configuración de alarmas en tags.**

Las alarmas son útiles para notificar a los usuarios las condiciones anormales dando una etiqueta, es decir en una variable.

Estas características incluyen la habilitación de las alarmas, bajo que circunstancias la etiqueta está en alarma, el nivel de prioridad de la alarma, y

como reconocer estas alarmas. Cada alarma tiene un límite de prioridad, reconocidos entre 1 y 15. En el módulo de LabView DSC, 15 es la prioridad más alta y 1 es la prioridad más baja.

Las alarmas tienen dos tipos principales:

- Alarmas basado en estados.
- Alarmas basados en valores de etiquetas.

La configuración basada en valores de etiquetas está especificada por el tipo de dato. Por consiguiente, muchos atributos de las alarmas aplican un solo subconjunto de los tipos de datos etiquetados.

#### **a) Ajuste de alarmas para etiquetas análogas.**

Las etiquetas tienen cuatro niveles de alarmas: Alto\_Alto; Alto, Bajo y Bajo\_Bajo. Para proporcionar distintos niveles de alarma, se puede además proporcionar más información acerca de donde vino la condición de la alarma. Las alarmas son calculadas después de desarrollado su escalamiento. Se expresa los niveles de alarma en unidades de ingeniería.

#### **b) Ajuste de tolerancia para alarmas de etiquetas análogas.**

La zona muerta de una alarma define cuanto puede variar una alarma desde sus niveles máximo o mínimo configurado como normal. Por ejemplo, si una etiqueta que representa valores de temperatura flotan cerca del límite de la alarma de 40 grados Celsius, la etiqueta de la alarma podría entrar y salir varias veces en periodos relativamente cortos. La tabla 2.1 muestra ejemplos de eventos con zona muerta de la alarma fijada para 0.0%.

<b>Tiempo</b>	<b>Valores</b>	<b>Eventos</b>	<b>Tipo de alarma</b>
9 : 15 : 05	40.1	Si	HI
9 : 15 : 10	39.9	Si	Normal

9 : 15 : 15	40.1	Si	HI
9 : 15 : 20	38.5	Si	Normal

**Tabla 2.1 Eventos con zona muerta de la alarma al 0.0%**

Este tipo de situaciones detienen los eventos de archivos con información redundante y puede causar algunas inconvenientes en los operadores que tienen que reconocer las alarmas continuamente, cuando la etiqueta no ha cambiado significativamente. Se puede usar la zona muerta de una alarma para evitar este problema.

Para ir a la etiqueta con alarma, se debe actuar sobre el valor exacto de la alarma (en el ejemplo anterior 40). Sin embargo para ser considerado normal otra vez, debe dejar el valor de la alarma en una cantidad mayor a la zona muerta de la alarma. Por ejemplo, si el rango es de 0 a 100 grados Celsius, una zona muerta de la alarma de 1.0 % ( un grado Celsius) elimina los eventos innecesarios. La tabla 2.2 muestra ejemplos de eventos con alarma inactiva al 1.0 %.

<b>Time</b>	<b>Valor</b>	<b>Evento</b>	<b>Tipo de Alarma</b>
9 : 15 : 05	40.1	Si	HI
9 : 15 : 10	39.9	No	HI
9 : 15 : 15	40.1	No	HI
9 : 15 : 20	38.5	No	Normal

**Tabla 2.2 Eventos con zona muerta de la alarma al 1.0%**

### **c) Ajuste de alarmas para etiquetas discretas.**

Las etiquetas discretas tienen dos estados de alarma, ambos indican que la etiqueta está en alarma o no. Se puede determinar si una etiqueta discreta está en alarma cuando está en estado alto o cuando está en estado bajo, esta situación está en dependencia directa de quien interprete su proceso.

#### **d) Ajuste de alarmas para etiquetas de arreglo de bits.**

Se puede habilitar uno o dos tipos de alarmas para una etiqueta de arreglo de bits. Alarm On Any indica que una tag de este tipo está en alarma si es que cualquiera los bits del arreglo está en estado de alarma.

Alarm On All indica que toda la etiqueta está en alarma, si es que todos los bits del arreglo están en estado de alarma. Se puede usar Invert Mask para determinar que bits deberían estar en alarma baja (OFF) y cuales están en estado de alarma alta (ON). Se puede usar Select Mask (operación lógica AND) para determinar los bits que deberían ser considerados como alarma. Si se tiene bits en Select Mask que son cero, estos bits no son usados para el cálculo de la alarma del tag.

#### **e) Ajuste de alarmas para etiquetas de cadenas de caracteres.**

Las etiquetas de cadena de caracteres no tienen alarmas de estado basadas en el valor de la etiqueta. Solamente utilizan estados de alarma para estados anormales.

### **2.4.9 EL MOTOR DE TAGS.**

El motor de etiquetas, es prácticamente el administrador y manejador de las variables asignadas a cada una de las etiquetas, para observar su funcionamiento se debe cargar el motor de tags (Tag Engine) y abrir el administrador de dicho motor (Engine Manager).

Se puede dejar minimizado el Engine Manager, a no ser que se desee realizar aplicaciones con él, como arrancar o detener el motor de tags, arrancar o detener los registros históricos de las variables, observar los eventos ocurridos con las variables, etc.

El motor de tags trabaja directamente con la configuración de las etiquetas almacenada en el archivo .scf y creados en el Tag Configuration Editor, el motor de tags únicamente muestra el estado actual de los tags configurados previamente.

La Figura 2.6 describe los campos que administra la ventana de Administrador del motor de tags. Adicionalmente la Tabla 2.3 describe los campos sobre los cuales trabaja el Engine Manager.



**Figura 2.6 Ventana de Engine Manager**

<b>CAMPO</b>	<b>DESCRIPCION</b>
Estado del motor de tag. <b>Engine Status</b>	Muestra estado actual del motor de tags (Tag Engine) si está cargado, corriendo o detenido.
Anotando datos <b>Carga de datos (Log Data)</b>	Activa o desactiva la carga de los registros históricos de las variables en un archivo.
Anotando eventos <b>Carga de eventos (Log Events)</b>	Activa o desactiva la carga automática de estados de alarmas y eventos para una o varias etiquetas hacia un archivo.
Imprime los eventos <b>Impresión de eventos (Print Events)</b>	Activa o desactiva la impresión de alarmas o eventos hacia una línea de impresión.
Diálogo de errores (Error Dialog)	Habilita o deshabilita la caja de dialogo de Error. Si ésta contiene un indicador se despliega un mensaje de error del sistema con la finalidad de que se acepte el evento producido cuando un error ha ocurrido.

**Tabla 2.3 Descripción de los campos del Engine Manager.**

## **2.5 VISUALIZACIÓN Y CONTROL CON LabVIEW DSC.**

Una HMI es la interfase a través de la cual un operador de proceso interactúa con el sistema LabVIEW y con el ambiente externo que LabVIEW monitorea y controla. En LabVIEW, se usa Instrumentos virtuales con el HMI. El módulo LabVIEW DSC instala controles personalizados, indicadores, VI's y funciones que hacen fácil la creación de una HMI.

El módulo de LabVIEW DSC incluye un conjunto de VI's con los cuales se puede controlar la HMI, proporciona también acceso a la base de datos en tiempo real y a la base de datos histórica Citadel, puede realizar cálculos matemáticos y lógicos y cambios entre diferentes pantallas de una o varias aplicaciones. El módulo de LabVIEW DSC tiene librerías de VI's que contemplan VI's de alarmas y eventos, VI's de históricos de datos, VI's del Sistema, VI's de Tags, y VI's de atribuciones.

Hay allí generalmente muchos principios de programación que LabVIEW ha incluido con los cuales ya se ha trabajado y es un entorno sobre el que se va a desenvolver la HMI. Para programas que construyan y manejen HMI más avanzadas se pueden utilizar Nodos de propiedad (Property Nodes) y los elementos del VI Server.

### **2.5.1 Los paneles frontales de un HMI.**

El panel frontal es la interfase gráfica de usuario de un VI. Se puede dividir un HMI en muchos paneles frontales, de esta forma el operador pueda navegar a través de ellos usando varios botones en las pantallas. El panel Wizard puede ayudar automáticamente a generar este código y vincular a los botones del panel frontal.

Los gráficos y decoraciones, de los controladores e indicadores, pueden hacer en el panel frontal más fácil la operación y proporcionar mayor información.

Al construir un panel frontal con controles e indicadores, estos representan los valores de los tags, tomando en consideración que los controles entregan datos y los indicadores despliegan datos, es decir en una aplicación HMI los controles e indicadores del panel frontal de LabVIEW se convierten en las variables a monitorear ya sean de entrada o salida al sistema.

El módulo de LabVIEW DSC también incluye una extensa librería de control y supervisión de otras imágenes que tienen formas geométricas básicas o que representan equipos específicos.

Si la dimensión u otras consideraciones de la aplicación se hacen muy grandes, se debe considerar dividir el HMI dentro de varios paneles frontales así el operador puede navegar a través de ellos usando botones para desplazarse por varias ventanas.

Se utiliza el asistente HMI Wizard para generar secciones comunes de código para diagramas de bloque.

Muchos módulos de LabVIEW DSC y funciones requieren un nombre de etiqueta como entrada. Además del estándar del tipo de datos de LabVIEW, el módulo de LabVIEW DSC incluye los tipos de dato de etiqueta (Tag data Types). Todas las funciones de los módulos de LabVIEW DSC operan las etiquetas o grupos de etiquetas usando Tag data Types. La etiqueta Tag data Types está marcada con un icono de válvula. Las constantes y alambres en el diagrama de bloques que contengan esta información se despliegan en púrpura. Muchos VI's del módulo LabVIEW DSC operan con arreglos de este tipo de tags.

La función Not a Tag es una constante que no informa a una etiqueta. Esta constante se usa más a menudo en estructuras de tipo case cuando una condición contiene vinculada un Tag name y otra no.

Los Tag data type usan las etiquetas y los nombres de grupo del archivo de configuración (.scf). Cuando se arranca LabVIEW, los datos de etiqueta disponible de una lista nombres o de nombres de grupo de etiqueta, actualizan

automáticamente el archivo .scf por defecto. El valor predeterminado .scf es el último archivo que se revisó en Tag Configuration Editor. Si la lista de nombres está vacía, se necesita abrir un archivo .scf en el en Tag Configuration Editor.

## **2.6 DESPLIEGUE DE TENDENCIAS.**

Una tendencia es un indicador de los valores de una etiqueta en función del tiempo. LabVIEW DSC posee dos alternativas para indicar gráficos de tendencias: Las tendencias de tiempo real y las tendencias históricas.

### **2.6.1 Tendencias de tiempo real.**

Una tendencia de tiempo real es un despliegue de un conjunto de valores de una etiqueta en tiempo real es decir al instante, y en un período relativamente corto de tiempo.

### **2.6.2 Tendencias históricas.**

Una tendencia histórica es un despliegue de valores de la etiqueta que se han guardado al disco, normalmente por un período relativamente largo de tiempo.

## **2.7 ALARMAS Y EVENTOS.**

Una alarma es una condición de proceso anormal que pertenece a una etiqueta. En el módulo de LabVIEW DSC, se generan las alarmas basado en los cambios de un valor etiqueta o estado.

Un evento es algo que pasa dentro del módulo de sistema de LabVIEW DSC. Los eventos pueden ser divididos en dos grupos: Tag events que pertenecen a las etiquetas individuales, y eventos del sistema que pertenecen al módulo del sistema de LabVIEW DSC. Un ejemplo de Tag event es un cambio de un estado de la alarma por un Tag. Los ejemplos de eventos del sistema incluyen a un usuario que accione, el Tag engine, poniéndolo en marcha.

Debido a que las alarmas son generadas por los valores de la etiqueta, se puede configurar más atributos de alarma como una parte de configuración de Tags.

### **2.7.1 Visualización de alarmas y eventos.**

Se puede usar varios acercamientos diferentes para desplegar y manejar las alarmas y los eventos generados en el modulo de aplicaciones de LabVIEW DSC. Algunos acercamientos operan a través de VI's tradicionales; otros usan capacidades construidas en la red de computadoras de National Instruments. La multiplicidad de acercamientos no se proporciona sólo para la flexibilidad, sino también para la compatibilidad con la aplicación BridgeVIEW.

Alarms & Events entran en la RTDB (base de datos de tiempo real) y se guardan entonces en la base de datos histórica Citadel. Por las razones de compatibilidad con las versiones más actuales de BridgeVIEW, también se anotan los eventos a un archivo de ASCII nombrado en el formato YYYYMMDDHHMM.evt que indica el tiempo del primer evento para ser anotado. YYYY es el año, el MM es el mes, DD es el día, HH es la hora, el MM, es el minuto, y el .evt es la extensión para todos los archivos de evento.

Una Alarm Summary es una colección de todas las alarmas que actualmente existen en el sistema. Adicionalmente, si un tag previamente en alarma retorna a su estado normal pero es desconocida, una notificación es anulada en el resumen de alarma.

Las alarmas desarrolladas en el sumario de alarmas puede ser filtrada usando el HMI Wizard por grupo o nombres, prioridad y conocimiento de estatus.

Se puede reportar el estado de alarmas actuales en el sistema.

## **2.8 MANEJO DE DATOS HISTÓRICOS.**

La base de datos de tiempo real (RTDB Real Time Data Base) contiene en memoria toda la información monitoreada, aquí residen los archivos creados para

mantener los datos importantes de un proceso. Cuando el motor de los tags está detenido, la RTDB retiene la última información grabada de los datos, pero no actualiza ni cambia los valores hasta que el motor de tags arranque nuevamente. Esto sucede porque la RTDB no puede registrar los datos en Citadel, ya que los datos históricos registrados en Citadel provienen de la RTDB y ningún dato puede registrarse allí cuando el motor está detenido.

### **2.8.1 La Base de datos CITADEL.**

El módulo LabVIEW DCS utiliza una base de datos histórica propia de National Instruments llamada CITADEL. LabVIEW DSC incluye también el manejador Citadel ODBC que es un software especial que se utiliza para la transformación de los datos, para recuperación, manipulación y análisis histórico automático desde una aplicación LabVIEW.

En el sistema operativo Windows NT/2000 Citadel corre en la computadora como un servicio accesible a través de un asistente administrador de servicios.

La información que se configura puede ser almacenada en Citadel en un grupo de archivos en un directorio fuente para poder ser registrados. Esta información puede incluir valores de la aplicación como de las alarmas y eventos. Se puede adicionalmente controlar qué información está generada, en qué localización, de qué configuración y la configuración de alarmas y eventos. Se puede registrar datos en una computadora local o en una computadora remota de una red, pero el directorio con el cual se desea registrar los datos debe estar en la computadora donde reside el motor de tags.

### **2.8.2 Archivar datos históricos.**

Cuando se registran los datos históricos para cualquier aplicación, hay un acuerdo entre la configuración del archivo .scf y la base de datos Citadel. Cuando se decide archivar datos de tipo histórico se copia el archivo .scf con los datos históricos hacia una nueva localización. Aunque se pueden recuperar los datos históricos sin el archivo .scf no se va a poseer la información de la configuración

de la etiqueta, como rango, unidades, etc., a menos que se archiven con el archivo .scf.

Preferiblemente se deberá mantener un camino relativo entre el archivo .scf y los archivos históricos en esta nueva localización por ejemplo si se guardó el archivo .scf en C:\Archivo, mantenga los archivos históricos en C:\Archivo\Datos. Si es que se graba un nuevo archivo .scf y no se tiene especificado un directorio de datos históricos se está obligado a crear un nuevo directorio.

Los datos históricos son grabados en un número de archivos creados por el módulo LabVIEW DSC y la base de datos Citadel, consecuentemente se debe siempre registrar los datos en un directorio exclusivo reservado para estos archivos.

Cuando se crea un nuevo archivo .scf el sitio por defecto en donde se graban los datos creados para los tags previamente configurados es un directorio llamado *data* localizado en el directorio en el cual se grabó el archivo .scf. Si se resetea la localización de destino de los archivos y ellos se pierden o se confunden con otros archivos, se necesita localizar los archivos con las siguientes extensiones:

- .evt
- .ale
- .adx
- .dat
- .mdx
- .bak
- .tbd
- .tdx
- .thd

Estos archivos no tienen acceso independiente.

### **2.8.3 Extraer datos históricos.**

En el módulo de LabVIEW DSC hay tres métodos para visualizar y extraer los datos históricos que están registrados en el disco, estos son:

- a) Usando los VI's de datos históricos (Historical Data VI's).
- b) Usando el desplegador de datos históricos (HTV).
- c) Usando un programa compilador ODBC que se encadene a la base de datos Citadel.

#### **a) Acceso a los datos históricos usando VI's.**

Hay muchos VI's que se pueden utilizar en una HMI para manipular los datos residentes en Citadel. Estos VI's acceden al disco sin necesidad de que el motor de tags este corriendo. Se pueden utilizar estos VI's para revisar el contenido de los archivos, extraer la información en un formato que pueda desplegarse en un indicador de tendencias históricas (Gráfico XY), o exportar los datos a un archivo formateado como hoja de cálculo.

Los VI's principales para manipular los datos de un archivo histórico son los siguientes:

- Call HTV
- Decimate Historical Trend
- Decimate Historical Trends
- Get Historical Tag List
- Get Historical Trend Info
- Historical Trend Statistics
- Historical Trends to Spreadsheet
- Historical Trends to Spreadsheet File
- Read Historical Trend
- Read Historical Trends

- Read Historical String Trend
- Read Historical String Trends

#### **b) Acceso a los datos históricos usando el Visualizador de tendencias históricas (Historical Trend Viewer).**

El Visualizador de tendencias históricas (HTV Historical Trend Viewer) es un utilitario que ayuda a visualizar los datos históricos de cualquier aplicación en un sistema. El HTV limita a no visualizar más de 8 tags a la vez, si se desea monitorear más tags en un gráfico de tendencias históricas, se debe construir un utilitario propio utilizando los VI's de manejo de datos históricos.

#### **c) Imprimir datos históricos.**

Se pueden imprimir los datos históricos grabados en un archivo.

### **2.9 SEGURIDAD.**

Para implementar seguridad en una aplicación hay que configurar cuentas de usuario y grupos de usuarios, para poder asignar privilegios o restringir accesos a la aplicación o simplemente al HMI.

Un sistema con seguridad basado en permisión, es un sistema en el cual los usuarios pueden tener varios niveles de acceso, los que les otorgan ciertos privilegios dentro del sistema y también pueden restringir su acceso a diferentes niveles. Estos accesos o restricciones dependen del nivel con el que se configuró su cuenta el momento de crearla.

#### **2.9.1 Cuentas de usuario.**

Se puede crear y editar las propiedades de las cuentas de usuarios o grupos de usuarios, asignar usuarios a uno o más grupos, y por otra parte maneja la seguridad para las cuentas de usuarios registrados en un sistema o aplicación

escrito en LabVIEW o Lookout. Solamente el administrador de cuentas o alguna otra persona encargada de acreditar las cuentas de usuario puede crear, revisar o borrar las cuentas de usuario.

El utilitario National Instruments User Manager tiene muchas cuentas de usuario y grupos pre – construidos que incluyen cuentas como: Administrador, Todos, Invitado y Nadie; y grupos pre – construidos como: Administrador, Invitados, Operadores y Operadores del Sistema. No es posible borrar ninguna de esas cuentas, sin embargo se pueden revisar todas las propiedades que poseen algunos de ellos.

La cuenta del Administrador está sobre todas las demás cuentas y posee acceso a todas las configuraciones del sistema, es más es quien puede asignar nuevas cuentas de usuario con sus códigos de acceso, los niveles de seguridad y privilegios. Esta característica se extiende a todos los participantes del grupo de cuentas de Administrador.

No se puede borrar la cuenta del Administrador o cambiar sus niveles de seguridad, lo que se puede realizar con esta cuenta es colocarle un nombre y una descripción del usuario y definir la clave de entrada a la cuenta y además se puede adicionar o quitar cuentas del grupo de administración.

## **2.10 DESARROLLO DE APLICACIONES EN RED.**

National Instruments utiliza una tecnología especial de red conocida como Logos Networking Protocol. Es un servicio que se instala con el módulo LabVIEW DSC. El protocolo de red Logos funciona con una red, sin necesidad de desarrollar configuraciones especiales de trabajo. Solamente se necesita buscar los puntos de datos a los cuales se quiere acceder con el software de cliente que debe tener capacidades de interpretación de Logos (como el Tag Configuration Editor o el Tag Configuration Wizard) para ver los datos que Logos administra en la red.

LabVIEW sin el módulo DSC puede adquirir datos tipo Logos a través de DataSocket pero no puede actuar como un servidor Logos. En consecuencia se deberá utilizar el módulo DSC para conectar los datos y los tags directamente.

Antes de que se pueda acceder a los datos de tipo Logos es necesario registrar la computadora y todos los dispositivos que corren en la red. Para registrar datos históricos en forma exacta, se deben además sincronizar los relojes internos de los computadores y dispositivos de la red.

El módulo LabVIEW DSC también incorpora un módulo muy funcional de comunicaciones tipo OPC, permitiendo que se puedan realizar aplicaciones en el modelo Cliente – Servidor. En este caso tampoco se necesita realizar ninguna configuración especial. Cualquier aplicación de LabVIEW es transparente para cualquier aplicación cliente de OPC. El asistente Tag Configuration Editor o el Tag Configuration Wizard pueden ingresar como una aplicación OPC en una red y revisar el contenido de los tags que el sistema esté monitoreando.

#### **2.10.1 Fijar las aplicaciones de red.**

Para realizar aplicaciones utilizando el módulo de LabVIEW DSC en una red, se necesita registrar la computadora o los dispositivos que están utilizándose en la red de modo que sean usuarios del Logos Networking Protocol, se debe estar completamente seguro de que los relojes de todas las computadoras y dispositivos conectados a la red estén perfectamente sincronizados; hay que asegurarse también de que los archivos de seguridad que se vayan a utilizar sean compatibles con la red que se está manejando, Finalmente se debe estar seguro de que todos los servicios de red necesarios están corriendo.

Para realizar aplicaciones en LabVIEW o Lookout o utilizar hardware de FieldPoint con el Logos Networking Protocol, se deberán registrar dichas aplicaciones mientras corren o el dispositivo FieldPoint mientras está encendido.

#### **2.10.2 Servicios de monitoreo de Windows.**

El Logos Networking Protocol requiere que tres servicios de background corran en Windows fuera de cualquier aplicación de National Instruments. Estos servicios son conocidos como Citadel Sever, Classified Ads y Time Synchronization. En el

administrador de tareas de Windows 2000/NT estos servicios aparecen como: Classifieds, TimeService y CitadelService, bajo Windows 2000/NT estos servicios corren automáticamente.

En Windows 9x el módulo de LabVIEW DSC instala un administrador de servicios identificado por el icono de un faro localizado en la barra de tareas. Cuando se hace un click derecho en es icono un menú pop up se despliega con las características de los servicios, allí se puede arrancar o detener los mencionados servicios.

### **2.10.3 El sistema Run Time.**

Se pueden desarrollar aplicaciones de LabVIEW DSC, utilizando el sistema Run Time que el módulo proporciona. El cual tiene soporte para construir capacidades especiales de LabVIEW. Se debe comprar el sistema de Run Time de LabVIEW DSC en forma separada, pues no se pueden correr las aplicaciones del módulo LabVIEW DSC sin el sistema Run Time.

Cuando se desarrollen aplicaciones se debe asegurar que todo el hardware y los drivers del hardware han sido instalados correctamente y sean los apropiados y que trabajen en arquitectura cliente servidor.

## **2.11 USO DE SQL PARA ACCEDER A DATOS HISTÓRICOS DE CITADEL.**

La base de datos histórica Citadel incluye un driver de ODBC (Open Data Base Connectivity) el que habilita otras aplicaciones para recuperación de datos desde la base de datos Citadel usando SQL (Structured Query Language).

ODBC es un estándar desarrollado por Microsoft que define los mecanismos para acceder a datos que residen en un sistema de administración de base de datos (DBMS). Casi todas las aplicaciones Windows pueden recuperar datos desde de un sistema de banco de datos utilizando algún tipo de soporte ODBC.

SQL es un lenguaje industrial estandarizado usado para recuperar actualizar y manejar datos. En LabVIEW con el toolkit Enterprise Connectivity se puede usar SQL para construir procedimientos que extraigan datos de Citadel. El driver ODBC de Citadel también incluye muchos constructores de datos simplificados para análisis estadístico simple y recuperación de datos.

Se deberán utilizar sistemas que manejen drivers de ODBC para acceder a los datos históricos de Citadel.

## **CAPÍTULO III**

### **COMUNICACIÓN DE LA APLICACIÓN SCADA CON EL ENTORNO**

#### **3.1 TARJETAS DE ADQUISICIÓN DE DATOS.**

Otra forma de medir las señales y transferir los datos al ordenador es usando Tarjetas de Adquisición de Datos, llamadas comercialmente tarjetas DAQ. Estas tarjetas poseen Convertidores Analógico/Digitales (ADC) y Convertidores Digital/Analógicos (DAC) que permiten la entrada/salida de señales analógicas y digitales.

##### **3.1.1 Sampling (Muestreo).**

Los datos son capturados mediante un ADC usando un proceso de muestreo. Muestrear una señal analógica requiere la toma de muestras de dicha señal en tiempos determinados, tiempos discretos. La frecuencia en que la señal es muestreada es conocida como Frecuencia de Muestreo. El proceso de muestreo genera valores de la señal a intervalos de tiempo conocidos.

La frecuencia de Muestreo determina la calidad de la señal analógica que se convierte. Cuanto más alta sea la frecuencia de muestreo mejor será la resolución de la señal analógica, tendrá mayor parecido con la señal real. La frecuencia mínima de muestreo requerida para representar la señal adecuadamente debe ser como mínimo dos veces la frecuencia máxima de la señal analógica que se desea convertir (Principio de Nyquist). Si la señal es muestreada a una Frecuencia de Muestreo menor que dos veces la frecuencia máxima de la señal, se produce el efecto de Aliasing.

### **3.1.2 Convertidores Analógico/Digitales (ADC).**

Una vez la señal ha sido muestreada, sus muestras necesitan ser convertidas a código digital. Este proceso se llama conversión Analógico/Digital. La mayoría de tarjetas también poseen un multiplexor que actúa como un switch para los diferentes canales del ADC. Esto hace posible capturar diferentes señales analógicas en paralelo, el inconveniente es que la Frecuencia de Muestreo debe ser dividida por el número de canales en paralelo que posee la tarjeta.

#### **3.1.2.1 Resolución del ADC.**

La precisión de la señal analógica de entrada convertida en formato digital depende del número de bits que el ADC usa. La resolución de la señal convertida está en función del número de bits que el ADC usa para representar el dato digital. El rango de voltaje entre voltaje máximo y voltaje mínimo con el que la señal real puede ser representada analógicamente se subdivide en función del número de bits de resolución digital. Por ejemplo un ADC de 8 bits puede dar hasta 256 niveles de representación digital ( $2^8 = 256$ ). Esto quiere decir que el rango de voltaje de la señal real se dividirá en 256 niveles, en el que cada uno de ellos tomará un valor digital determinado.

Dependiendo de la precisión del ADC, puede ocurrir que los cambios más pequeños de voltaje de la señal analógica no puedan ser representados digitalmente. Por ello la resolución es una característica muy importante en las tarjetas ADQ.

#### **3.1.2.2 ADC No lineales**

Idealmente si el voltaje aplicado a la entrada de un ADC se incrementa linealmente, se espera que el resultado al convertir la señal a información digital también se incremente linealmente

No siempre ocurre de esta manera. Sólo las tarjetas DAQ perfectas son así, estas tarjetas, aunque existen, tienen un elevado costo, y normalmente se encuentran en el mercado tarjetas que tienen una pequeña variación lineal.

### **3.1.2.3 ADC Setting Time (Tiempo de fijación de la señal).**

En una tarjeta típica, la señal analógica primero es seleccionada por un multiplexor, y luego amplificada antes de ser convertida por el ADC. El amplificador usado entre el multiplexor y el ADC debe ser capaz también de rastrear, retener y fijar la señal que se desea convertir, si no el ADC convertirá una señal que aún está en transición. Un tiempo de fijación erróneo puede llegar a ser un gran problema. Para determinar un Tiempo de Fijación correcto debe tenerse en cuenta la frecuencia de muestreo y la ganancia de la tarjeta DAQ.

### **3.1.3 Transferencia de Datos al Ordenador**

Normalmente, las tarjetas DAQ se instalan en los buses de alta velocidad del PC como los buses PCI. En función de la velocidad de la placa base del PC, la velocidad de transferencia de datos máxima entre componentes de dicha placa base suele estar entre el microprocesador y la memoria con valores que van desde los 20Mhz hasta los 40Mhz. Para mejorar la transferencia de datos, se implementa el Bus Mastering, que permite a las tarjetas DAQ transferir datos directamente a la memoria, y con ello se logra acelerar el proceso de adquisición de datos.

El microprocesador está participando en la transferencia de datos, empleando en ello un tiempo en que podría estar haciendo otras tareas. Por otro lado se hace la transferencia de datos directa a la memoria, con lo cual el microprocesador está libre para ser utilizado para otras tareas, esto se logra mediante la tecnología Bus Mastering.

### **3.1.4 Convertidores Digital/Analógicos (DAC)**

Las tarjetas multifunción también tienen integrado, normalmente, un convertidor Digital/Analógico (DAC). Un DAC puede generar una señal analógica en función de un dato digital.

Esto permite a las tarjetas de este tipo generar una señal analógica de salida, con voltajes en corriente continua (DC) o corriente alterna (AC). Al igual que el ADC, los DAC se encuentran limitados por el número de muestras que pueden procesar y el número de bits que usan para convertir el dato digital en señal analógica.

Es también importante en un DAC un tiempo de fijado (Settling Time) pequeño, ya que así podrá generar señales de frecuencia alta, debido a que el tiempo usado en fijar la señal de salida para un nuevo nivel de voltaje será bajo.

### **3.1.5 Tarjeta NI PCI 6014**

La tarjeta de adquisición de datos NI PCI 6014 posee 16 canales de entrada analógica de 16 bits (ocho canales en forma diferencial), dos canales de salidas análogas de 16 bits, un conector de 68 pines y ocho líneas de entrada/salida digitales.

Utiliza además un sistema de adquisición de datos de National Instruments con control temporizado para funciones relacionadas con el tiempo (DAQ-STC). El DAQ-STC se compone de tres grupos temporizados para controlar entradas analógicas, salidas analógicas, y funciones de conteo y temporización de propósito general.

Los mencionados grupos consisten en siete contadores de 24 bits y tres de 16 bits, con una resolución máxima de tiempo de 50nS. El DAQ-STC hace posible también aplicaciones como generación de pulsos buferizadas, muestreo en tiempo real y cambio rápido de velocidades de muestreo.

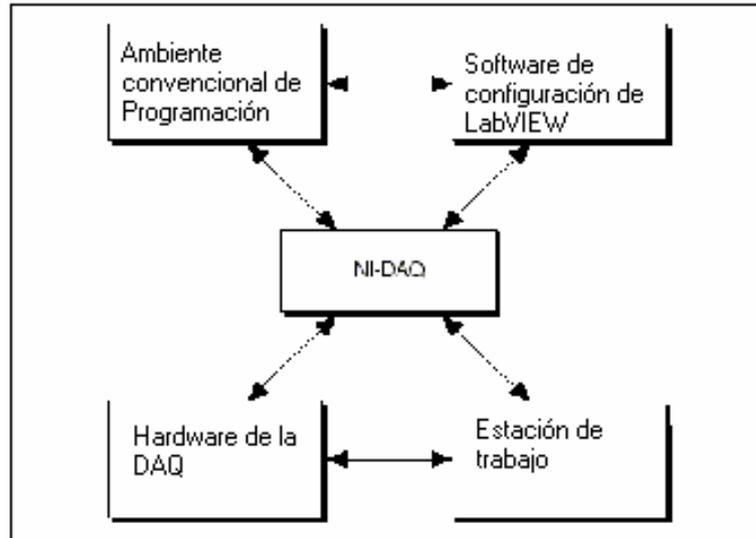
Para que una tarjeta NI PCI 6014 funcione adecuadamente como un sistema de adquisición de datos necesita los siguientes requerimientos:

- El Manual de usuario NI PCI 6013/6014.
- Software NI DAQ compatible con la tarjeta.
- Un paquete de software de instrumentación virtual con su documentación (puede ser LabVIEW, Measurement Studio).
- Un computador con bus PCI.

Cuando se programe un hardware de adquisición de datos de National Instruments, puede utilizar un ambiente de desarrollo de aplicación de National Instruments (ADE) u otro ambiente de control, en ambos casos deberá utilizar el utilitario NI-DAQ.

El NI-DAQ proporcionado conjuntamente con los drivers de la tarjeta NI PCI 6014, tiene una extensa biblioteca de funciones que se ejecutan desde el ADE, estas funciones son las que le permiten utilizar todas las características funcionales de la tarjeta 6014.

NI-DAQ lleva a cabo muchas de las complejas interacciones, como por ejemplo las interrupciones entre la computadora y dispositivo de adquisición de datos, necesarias para el eficiente uso del procesador; le proporciona también la posibilidad de migración, es decir mantiene una interfase de software que puede ejecutarse en diferentes versiones, tal como se muestra en la figura 3.1



**Figura 3.1 Relaciones entre el entorno de programación NI-DAQ y el hardware**

El ambiente de desarrollo de aplicación puede ser LabVIEW que proporciona gráficos interactivos una interfase muy innovadora y un lenguaje de programación gráfico muy poderoso, posee una librería de instrumentos virtuales muy versátil y completa que interactúan directamente con su hardware de aplicación.

### **3.1.5.1 Características Técnicas.**

En este apartado se describen las características del hardware incorporado en la tarjeta DAQ NI PCI 6014. En la Figura 3.2 se muestra el diagrama de bloques constituyentes de la tarjeta de adquisición de datos NI PCI 6014, los cuales se detallan a continuación en orden de importancia:

#### **a) Entradas análogas.**

Poseen dos modos de conexión de las entradas: NRSE (de terminación simple sin referenciación) y DIFF (modo Diferencial), NRSE proporciona 16 canales de entrada, y DIFF solamente presenta 8 canales de entrada, cada canal de entrada deberá ser configurado con un modo particular, y se deberá realizar programáticamente en algún utilitario de configuración de las entradas, por

ejemplo se podrían configurar 12 canales de entrada, cuatro de ellos en modo diferencial (utilizan 8 entradas) y 8 canales de entrada en modo NRSE.

La Tabla 3.1 muestra los dos modos de entrada posibles en la tarjeta NI PCI 6014.

MODO	DESCRIPCION
DIFERENCIAL	Un canal configurado en modo diferencial usa dos líneas de entrada análoga, una línea conecta la entrada positiva del amplificador de instrumentación de ganancia programable del dispositivo, y la otra entrada se conecta a la entrada negativa
NRSE	Un canal configurado en modo NRSE usa una sola línea de entrada análoga para cada señal, la cual se conecta a la entrada positiva del amplificador operacional de instrumentación del dispositivo, la entrada negativa del amplificador se conecta a la señal Aisense.

Tabla 3.1 Modos de entrada de la tarjeta NI-PCI-6014

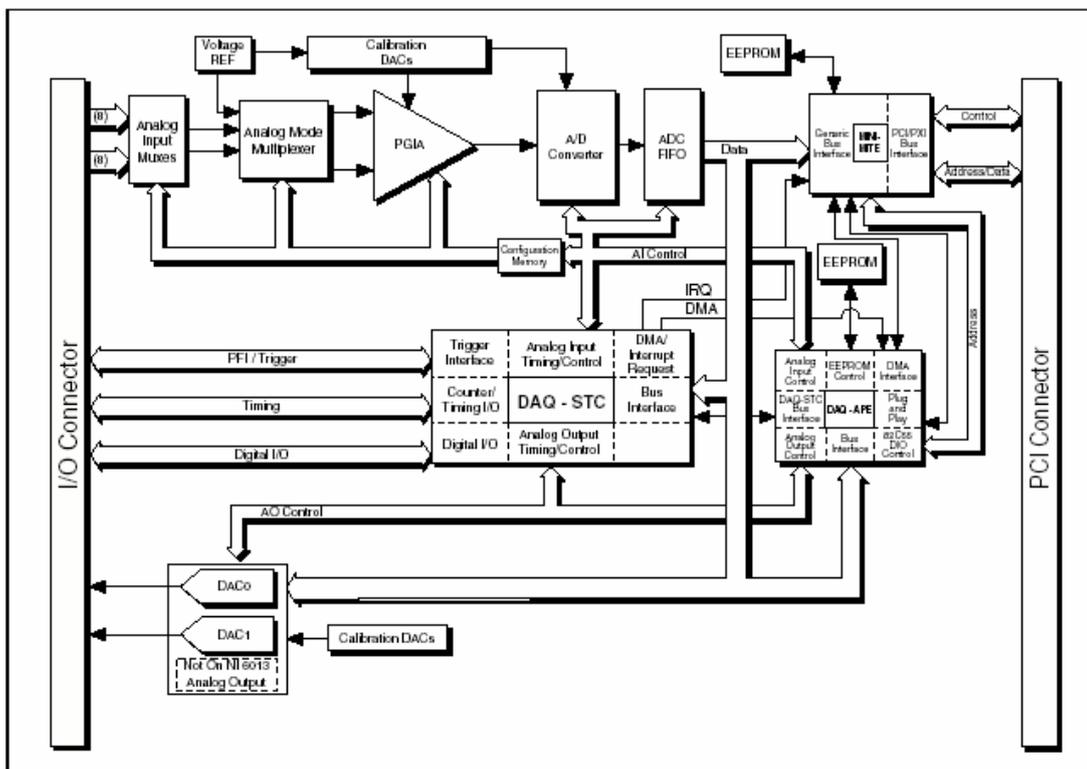


Figura 3.2 diagrama de bloques de la tarjeta NI PCI 6014

## b) Rango de entrada.

La tarjeta NI 6014 tiene un rango de entrada bipolar que puede cambiar de acuerdo a la ganancia que se programa por el utilitario de software, cada canal puede ser configurado con ganancias de: 0.5, 1, 10 y 100 para la máxima resolución del conversor AD. Con la adecuada ganancia configurada, usted puede utilizar toda la resolución del conversor para la medición de la señal de entrada. La tabla 3.2 muestra el rango de entrada y precisión de acuerdo a la ganancia seleccionada.

Gain	Input Range	Precision
0.5	-10 to +10 V	305.2 $\mu$ V
1.0	-5 to +5 V	152.6 $\mu$ V
10.0	-500 to +500 mV	15.3 $\mu$ V
100.0	-50 to +50 mV	1.53 $\mu$ V

Tabla 3.2 Rangos de entrada y precisión

## c) Monitoreo de múltiples canales.

El dispositivo puede monitorear múltiples canales en forma simultánea como si fuera un solo canal a una velocidad determinada, sin embargo hay que prestar mucha atención con los límites predeterminados por cada canal. No es necesario configuraciones de tiempo adicionales cuando la ganancia es la misma o las ganancias de cada canal son altas y las impedancias de las fuentes de señal son bajas.

Cuando se monitorean canales con ganancias diferentes, la configuración de los tiempos debe modificarse, cuando el utilitario cambia a una ganancia más alta, la señal del canal anterior puede estar fuera de rango, por ejemplo suponga que una

señal de 4V está conectada al canal 0 y una señal de 1mV conectada al canal 1, y suponga que el utilitario está programado para colocar una ganancia de uno en el canal cero y de 100 en el canal 1, cuando el multiplexor cambia al canal 1 y el utilitario cambia la ganancia a 100, el nuevo rango máximo es de  $\pm 50\text{mV}$ . De los 4V que se monitoreaban, pasa a monitorearse aproximadamente 1mV esto es un 4000% de disminución, esto hace que la circuitería interna se demore unos 100 $\mu\text{S}$  para establecer el cambio de 1 bit menos significativo, en general este tiempo extra de establecimiento de funcionamiento correcto no es necesario cuando el utilitario cambia a una ganancia más baja.

#### **d) Salidas Análogas.**

La tarjeta NI 6014 proporciona dos salidas de voltaje análogas de 16 bits de resolución cada uno, cada salida tiene un rango bipolar fijo de salida,  $\pm 10\text{V}$ . Los datos escritos en el convertor digital análogo son interpretados en el formato de complemento a dos

#### **e) Entradas y salidas digitales.**

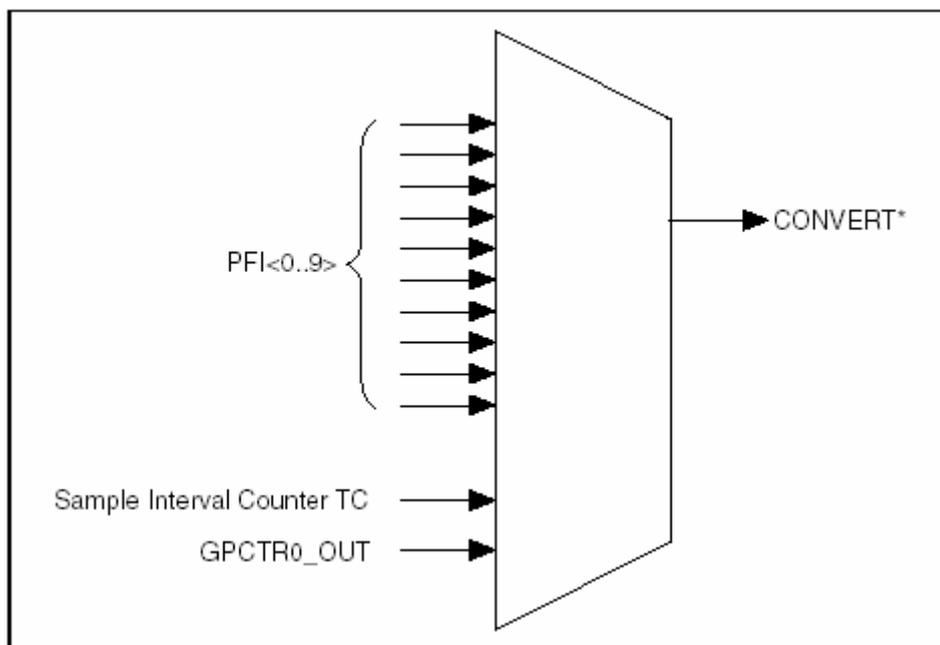
La tarjeta NI 6014 contiene ocho líneas que corresponden a entradas y salidas discretas (DIO<0...7>) para propósito general. Se puede configurar por software en forma independiente cada línea ya sea como entrada o salida cuando el sistema arranca o se resetea los puertos digitales se colocan en el estado de alta impedancia. A través de estas líneas se pueden controlar los disparos de los contadores y temporizadores del dispositivo.

#### **f) Señales temporizadas.**

Proporciona una interfase flexible para conectar señales sincronizadas para otros dispositivos o circuitos externos. La tarjeta NI 6014 usa una función de entrada

programable (PFI) colocada en uno de los pines del bloque conector para conectarla con el dispositivo o circuito externo, estas conexiones están diseñadas para habilitar a la tarjeta para controlar y ser controlada por otros dispositivos o circuitos externos.

La tarjeta DAQ-STC tiene 13 señales internas temporizadas que pueden ser controladas por una fuente externa, estas señales temporizadas pueden también ser controladas por señales generadas internamente, y las mencionadas selecciones son completamente programables.



**Figura 3.3 Asignación de ruta para las señales del convertidor**

La figura 3.3 muestra que la señal convert puede generarse de fuentes diferentes, incluyendo de señales externas y señales internas

### g) Entradas de función programable.

10 pines del dispositivo están conectados al multiplexor que determina la ruta de las señales, para cada señal temporizada que puede ser seleccionada por software. Es importante que se note que se puede usar las entradas temporizadas una a una, o todas las señales temporizadas en forma simultánea con el mismo control del utilitario.

### 3.1.5.2 Especificaciones del producto.

En esta sección se determinarán algunas especificaciones de conexión que recomienda National Instruments para el empleo de la Tarjeta DAQ NI PCI 6014.

La figura 3.4 muestra la asignación de los pines, para un conector de entrada y salida de 68 pines.

ACH8	34	68	ACH0
ACH1	33	67	AIGND
AIGND	32	66	ACH9
ACH10	31	65	ACH2
ACH3	30	64	AIGND
AIGND	29	63	ACH11
ACH4	28	62	AISENSE
AIGND	27	61	ACH12
ACH13	26	60	ACH5
ACH6	25	59	AIGND
AIGND	24	58	ACH14
ACH15	23	57	ACH7
DAC0OUT1	22	56	AIGND
DAC1OUT1	21	55	AOGND
RESERVED	20	54	AOGND
DIO4	19	53	DGND
DGND	18	52	DIO0
DIO1	17	51	DIO5
DIO6	16	50	DGND
DGND	15	49	DIO2
+5V	14	48	DIO7
DGND	13	47	DIO3
DGND	12	46	SCANCLK
PFI0/TRIG1	11	45	EXTSTROBE*
PFI1/TRIG2	10	44	DGND
DGND	9	43	PFI2/CONVERT*
+5V	8	42	PFI3/GPCTR1_SOURCE
DGND	7	41	PFI4/GPCTR1_GATE
PFI5/UPDATE*	6	40	GPCTR1_OUT
PFI6/WFTRIG	5	39	DGND
DGND	4	38	PFI7/STARTSCAN
PFI9/GPCTR0_GATE	3	37	PFI8/GPCTR0_SOURCE
GPCTR0_OUT	2	36	DGND
FREQ_OUT	1	35	DGND

\* Not available on the NI 6013

Figura 3.4 Bloque conector NI 6014

La descripción de cada una de las entradas y salidas a este conector, se encuentran consignadas en la tabla 3.3 a continuación.

SEÑAL	REFERENCIA	DIRECCIÓN	DESCRIPCIÓN
AIGND			Tierra de referencia de las señales de entrada
ACH<0..15>	AIGND	Entrada	Canales análogos de entrada de 0 a 15 cada par de canales de entrada se puede configurar en modo diferencial o NRSE.
AISENSE	AIGND	Entrada	Sensor de entrada análogo, sirve como referencia para cualquiera de los canales análogos de entrada de 0 a 15 en configuración NRSE. AISENSE puede ser conectado a AIGND directamente o con una tierra externa para referencias RSE
DAC0OUT	AOGND	Salida	Canal 0 de salida análoga
DAC1OUT	AOGND	Salida	Canal 1 de salida análoga
AOGND			Tierra de referencia para la salida de señales análogas, los voltajes análogos de salida son referenciados a este nodo.
DGND			Tierra digital, este pin proporciona la referencia de las señales de entrada y salida digitales
DIO <0..7>	DGND	Entrada Salida	Señales de entrada – salida digital, las señales DIO6 y DIO7 controlan la forma de conteo UP/DOWN de los contadores de la tarjeta.
+5V	DGND	Salida	Fuente de +5Vdc
SCANCLK	DGND	Salida	Reloj de escaneo, emite un pulso por cada conversión del conversor análogo – digital cuando el modo de escaneo está habilitado
EXTSTROBE	DGND	Salida	Strobe externo, esta salida puede ser conmutada por software para retardar señales o disparar eventos de dispositivos externos.
PFI0/TRIG1	DGND	Entrada Salida	Como entrada es una señal PFI programable, como salida es una señal de disparo 1.
PFI1/TRIG2	DGND	Entrada Salida	Como entrada es un PFI, y como salida es el disparador 2.

PFI2/CONVERT	DGND	Entrada Salida	Como entrada es un PFI y como salida indica que se ha realizado una conversión A-D.
PFI3/GPCTR1_SOURCE	DGND	Entrada Salida	Como entrada es un PFI y como salida indica la fuente a la que está conectada el contador 1.
PFI4/GPCTR1_GATE	DGND	Entrada Salida	Como entrada es un PFI y como salida indica el puente conectado al contador 1.
GPCTR1_OUT	DGND	Entrada Salida	Salida del contador1.
PFI5/UPDATE	DGND	Entrada Salida	Como entrada es un PFI y como salida indica que un grupo de salidas análogas han sido actualizadas.
PFI6/WFTRIG	DGND	Entrada Salida	Como entrada es un PFI y como salida indica la iniciación de una generación de forma de ondas.
PFI7/STARTSCAN	DGND	Entrada Salida	Como entrada es un PFI y como salida indica que ha empezado un barrido de las señales.
PFI8/GPCTR0_SOURCE	DGND	Entrada Salida	Como entrada es un PFI y como salida indica la actual fuente conectada al contador 0.
PFI9/GPCTR0_GATE	DGND	Entrada Salida	Como entrada es un PFI y como salida indica el puente actual conectado al contador 0.
GPCTR0_OUT	DGND	Salida	Salida del contador 0.
FREQ_OUT	DGND	Salida	Salida de frecuencia del generador interno.

**Tabla 3.3 Descripción de las entradas y salidas**

### **3.1.5.3 Modos de conexión de entradas análogas.**

La tarjeta NI PCI 6014 que contiene internamente amplificadores operacionales de instrumentación (PGIA), puede usarse de distintas formas dependiendo de cómo se configure el modo de operación que puede ser DIFF y NRSE.

La señal AIGND es una señal común para todas las entradas análogas y está directamente conectada al punto de tierra del dispositivo, se puede utilizar esta señal para obtener un punto de referencia común de las señales análogas de entrada.

La figura 3.5 muestra los diversos tipos de conexión de las señales dentro de una tarjeta DAQ.

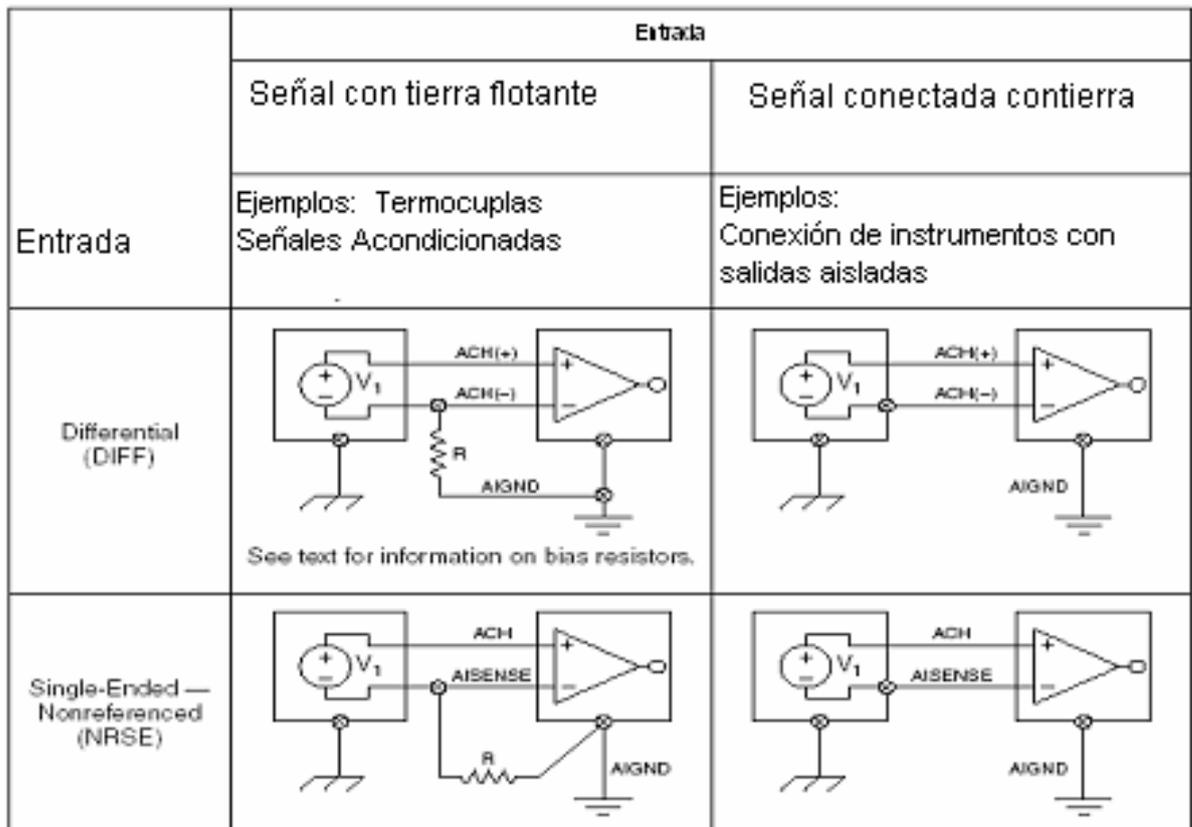


Figura 3.5 Modos de conexión de las entradas análogas

### a) Consideraciones de conexión diferencial.

Como se dijo anteriormente, una conexión diferencial es aquella en la cual la señal análoga de entrada tiene su propia referencia. Esta conexión está disponible cuando se programa la entrada como modo de canal diferencial (DIFF). En este modo los canales de entrada análoga se consideran como pares, donde ACH<math>i</math> es la señal de entrada y ACH<math>8+i</math> es sus señal de referencia. Por

ejemplo el ACH<0> es el par de entrada de ACH<8>, y donde el canal de la señal de entrada está conectado al terminal positivo del PGIA y la referencia al terminal negativo.

Cuando se configura el canal de entrada en modo diferencial cada señal utiliza dos entradas del multiplexor, una para la señal y una para la referencia, por lo que con este tipo de modo la tarjeta solo posee ocho canales de entrada. Se recomienda utilizar el modo diferencial cuando una señal de entrada tenga alguna de las siguientes condiciones:

- La señal de entrada es muy pequeña (menor que 1V).
- La fuente de señal esta conectada al dispositivo a una distancia mayor de 3m.
- La señal de entrada requiere un punto de referencia independiente.
- La señal de entrada está expuesta a un ambiente lleno de ruidos e interferencias.

Las conexiones de modo diferencial reducen el ruido introducido dentro de la señal, e incrementa la relación de rechazo de modo común.

La figura 3.6 muestra la forma de conexión para señales que poseen una fuente de señal con referencia a tierra con un canal en un dispositivo configurado en modo diferencial. Como se puede observar, con este tipo de conexión el amplificador elimina el ruido de modo común de la señal y la diferencia de potencial de tierra entre la tierra de la fuente de señal y la tierra del dispositivo, representados como  $V_{cm}$  en la figura 3.6.

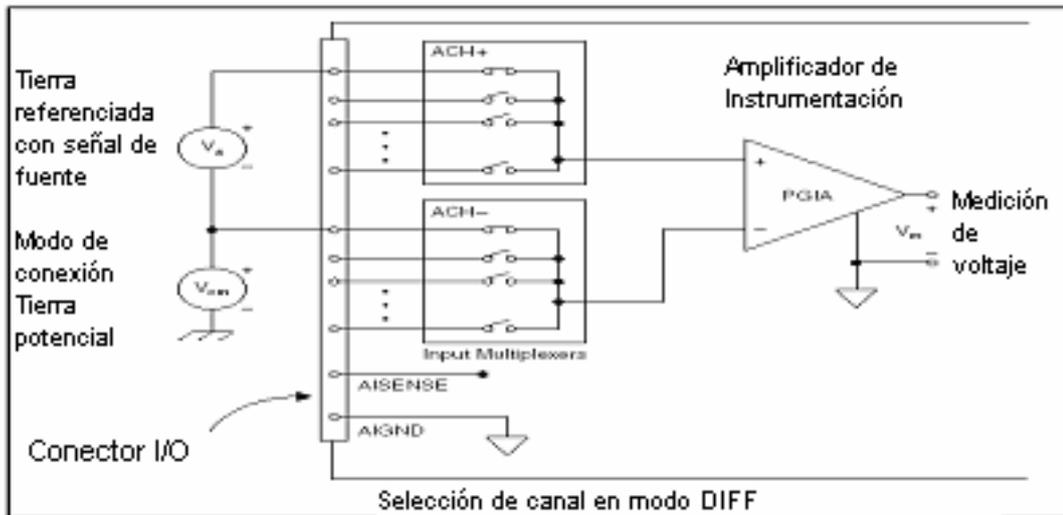


Figura 3.6 Conexión diferencial para señales referenciadas

La figura 3.7 muestra la forma de conexión de señales flotantes a un canal de un dispositivo configurado en modo diferencial, aquí se puede observar dos resistencias tolerantes conectadas en paralelo con la señal de entrada flotante, si no se usan estas resistencias es poco probable que la señal este dentro del rango del voltaje de modo común en el amplificador, y este puede saturarse y producir lecturas erróneas de la señal.

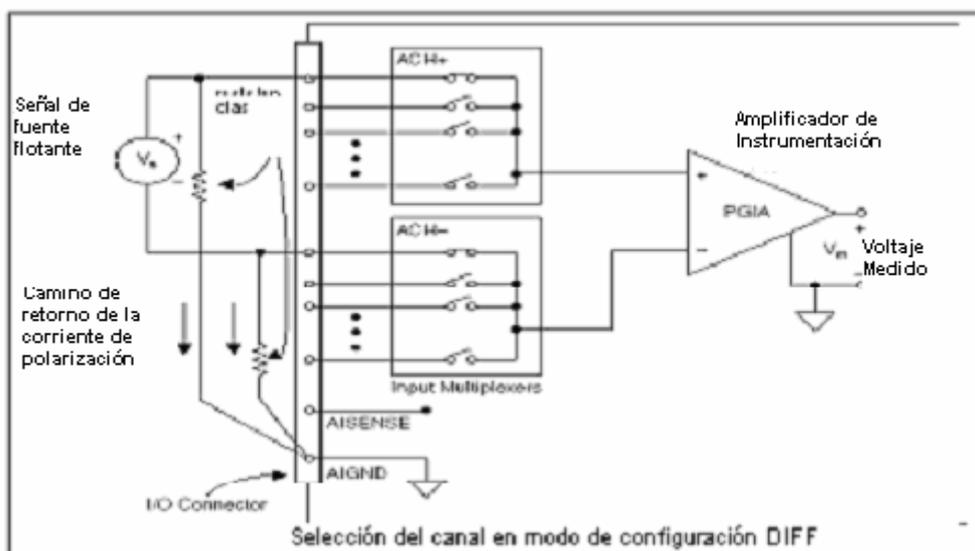


Figura 3.7 Conexión diferencial para señales flotantes

Como se observa en la Figura 3.7, se debe conectar la fuente de señal al terminal AIGND, es decir conectar el terminal positivo de la fuente de señal al terminal positivo del amplificador, y el terminal negativo de la fuente de señal al terminal AIGND, así como al terminal negativo del amplificador. Esta conexión es recomendada para fuentes acopladas de corriente continua con baja impedancia de entrada (menos de  $100\Omega$ ).

#### **b) Consideraciones de conexión RSE.**

Una conexión de terminación simple referenciada (RSE Referenced Single-ended) es aquella en la que la señal de entrada análoga está referenciada a un punto de tierra común que puede estar compartido con otras señales de entrada. La señal de entrada está conectada al terminal positivo del amplificador operacional y la tierra común está conectada al terminal negativo de entrada del amplificador usando la entrada AISENSE.

Cuando un canal está configurado en RSE están disponibles 16 canales de entrada en la tarjeta NI 6014. Se podrá utilizar la configuración de RSE para señales de entrada que posean las siguientes características:

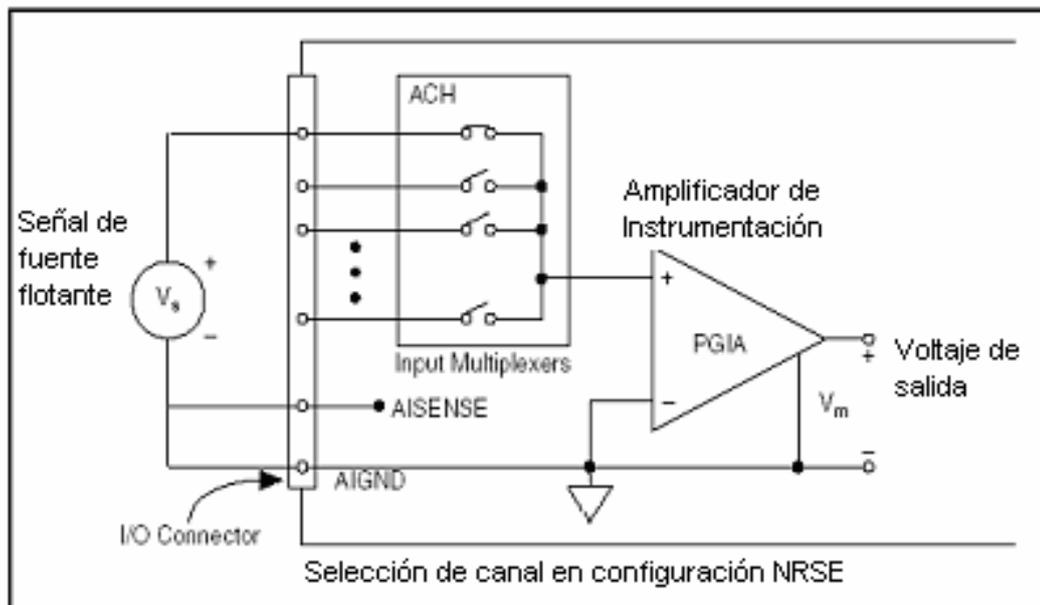
- Voltajes de entrada grandes (mayores a 1V)
- La fuente de señal está conectada al dispositivo a una distancia menor de 3m.
- La fuente de señal comparte la tierra común con otras señales.

El modo de conexión NRSE es solamente una configuración RSE con modificaciones de la referencia. La conexión de la entrada AISENSE se diferencia para conectar fuentes de señal flotante y fuentes de señal referenciadas. Para fuentes de señal de entrada flotante AISENSE está conectada directamente a AIGND, y la tarjeta proporciona la referencia de tierra para señales externas, en cambio para fuentes de señal referenciada AISENSE está conectada a la

referencia externa de tierra, previniendo de ésta forma lazos de realimentación de corriente por tierra y errores en la medición.

En el modo de conexión RSE se introduce mayor cantidad de ruido e interferencias electromagnéticas que en la configuración de modo diferencial, la interferencia por inducción magnética es proporcional al área de los conductores con los que está conectada la señal, y la interferencia eléctrica es resultado de la diferencia de campo que existe entre los conductores.

La figura 3.8 muestra la forma de cómo conectar fuentes de señal flotantes al canal de un dispositivo configurado en modo NRSE.



**Figura 3.8 Conexión de señales flotantes en modo NRSE.**

Para medir una fuente de señal de entrada referenciada con un canal de un dispositivo configurado en NRSE deberá realizarse la configuración respectiva del canal (Utilitario de configuración MAX).

La señal está conectada entonces al terminal positivo del amplificador y el punto de referencia local está conectado al terminal negativo del amplificador. El punto

de referencia de tierra de la señal debe, por consiguiente estar conectado a la entrada AISENSE.

Cualquier diferencia de potencial entre la tierra del dispositivo y la tierra de la señal aparece como una señal de modo común entre la entrada positiva y negativa del amplificador operacional, y ésta diferencia es rechazada por el amplificador. Si AISENSE está conectada a AIGND la diferencia entre los potenciales de tierra aparece como un error en el voltaje medido.

La figura 3.9 muestra la forma de conexión de una señal referenciada en modo NRSE.

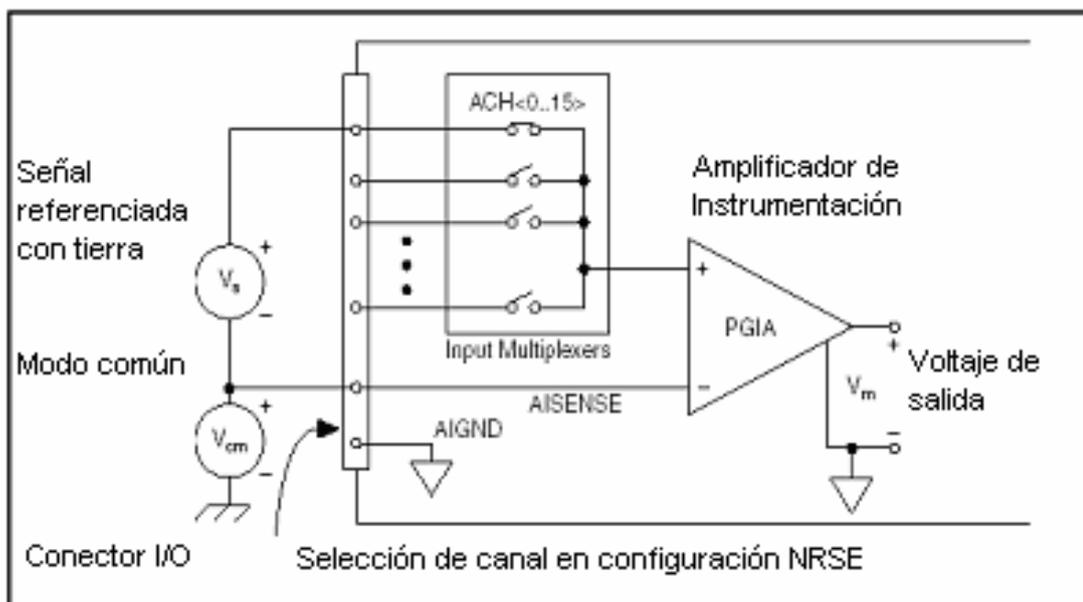


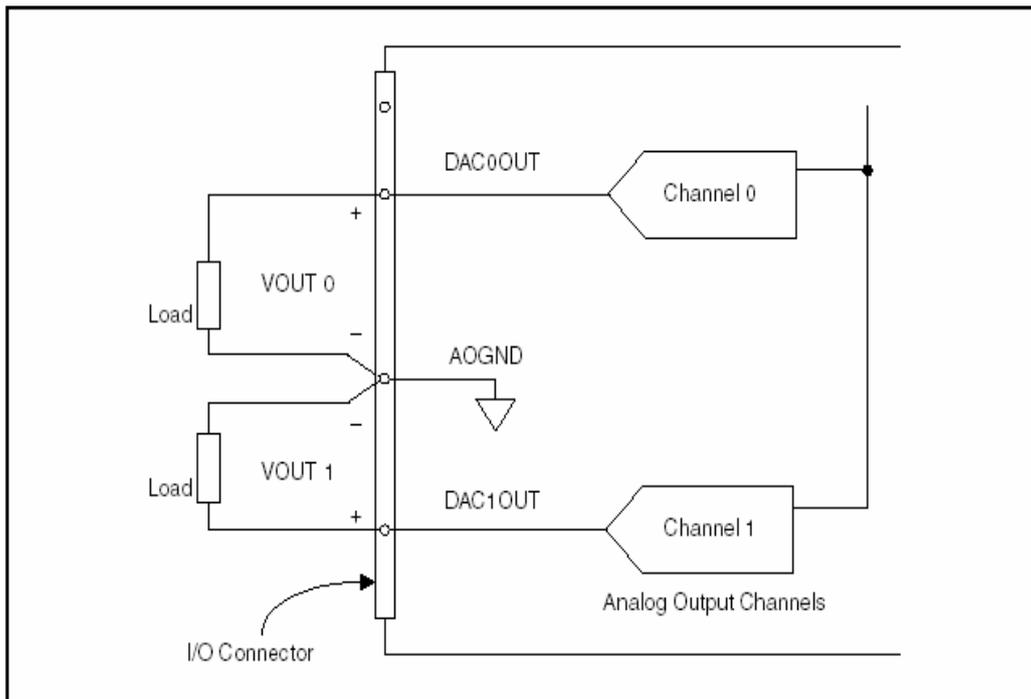
Figura 3.9 Conexión de señales referenciadas en modo NRSE

### 3.1.5.4 Conexión de salidas análogas.

Disponibles solamente en la tarjeta NI PCI 6014, las señales a considerar para las salidas análogas son: DAC0OUT, DAC1OUT y AOGND. DAC0OUT es la señal de salida de voltaje del canal 0, DAC1OUT es la señal de salida de voltaje para el

canal 1 y AOGND es la tierra de referencia para ambos canales, y la referencia externa del sistema DAQ.

La figura 3.10 muestra la forma de conexión de las salidas análogas para la tarjeta NI 6014.



**Figura 3.10 Conexión de las Salidas análogas**

### **3.1.5.5 Conexión de Entradas – Salidas Digitales.**

Las entradas – salidas digitales en la tarjeta NI 6014 son: DIO<0..7> y DGND, las primeras son los puertos de entrada salida digital y DGND es la referencia común de todas estas señales, estas señales pueden ser programadas individualmente como entrada o salida según sea la necesidad. Hay que tener precaución de exceder los límites aceptados de entrada a estos puertos, pues si así sucede, puede ocasionar daños irreversibles a la tarjeta y al computador conectado a ella.

La figura 3.11 muestra la forma de conexión de las señales de entrada – salida digital para tres aplicaciones típicas.

En ella se puede observar aplicaciones de entrada y salida como son: manejo de un led indicador (salida), entrada de un tren de pulso, y una entrada de un switch, que podría ser un final de carrera, pulsador de paro general, etc. Aquí se puede observar la configuración de los canales DIO<0..3> como entrada digital y los canales DIO<4..7> como salida digital.

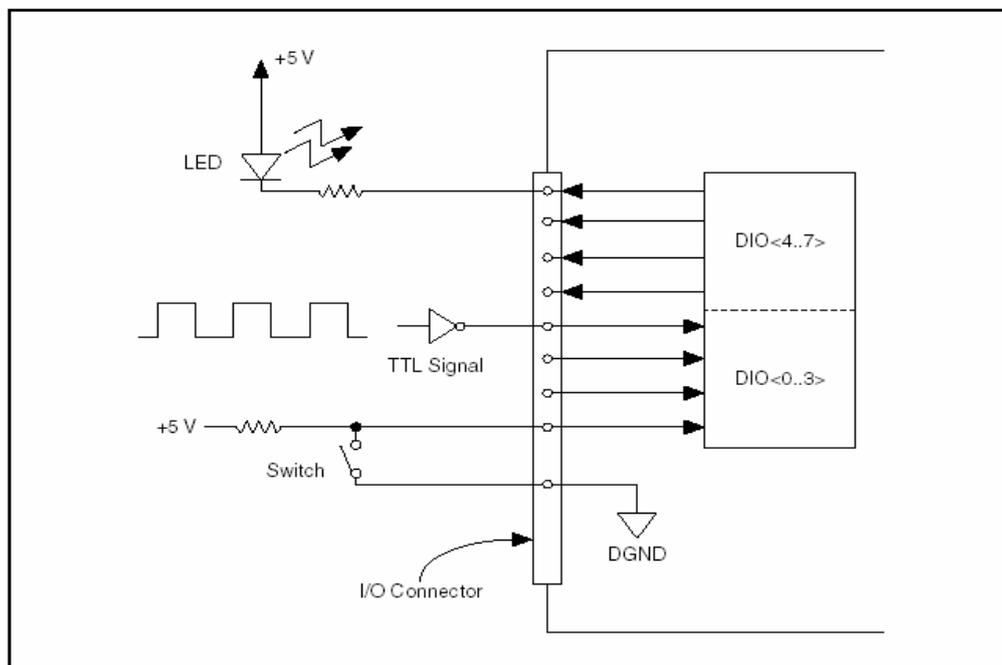


Figura 3.11 Conexión de los canales de entrada – salida digital

### 3.2 FIELD POINT FP 1000

El FieldPoint de National Instruments es un sistema distribuido de I/O modulares para utilización a nivel industrial y con un excelente software integrado. Con las I/O FieldPoint, usted puede fácilmente, configurar, construir y mantener soluciones confiables para I/O distribuidas.

### **3.2.1 Características de modularidad del sistema FP - 1000**

FieldPoint ofrece una arquitectura innovadora que modulariza las comunicaciones, funciones I/O y terminación de señales. Por lo tanto, se puede independientemente escoger las I/O, o la red industrial, y el estilo de terminación de señales que mejor se ajusta a la aplicación particular. FieldPoint incluye tres clases de componentes que lo hacen flexible: Módulos I/O, bases de conexiones y módulos de red de trabajo.

El sistema FieldPoint incluye una variedad de módulos I/O analógicos y digitales aislados eléctricamente, una base de conexiones, e interfaces de redes de trabajo de fácil conexión para tecnologías estándar y abiertas. Este sistema Fieldpoint también incluye algunas opciones de software, tales como LabVIEW, Real Time. Además de las fáciles herramientas de configuración que posee, el software de FieldPoint incluye un servidor OPC para conexiones estandarizadas para cualquier paquete de software compatible con OPC.

#### **3.2.1.2 Módulo Field Point FP-1000**

El FieldPoint es un sistema distribuido de I/O modulares para utilización a nivel industrial y con un software integrado.

Se utilizan para acondicionamiento de las señales estándar entregadas por sensores como RTD's y termocuplas, además poseen entradas analógicas para señales entregadas por otro tipo de sensores; a diferencias de los PLC's, el FielPoint adecua las señales analógicas de los sensores para ser procesadas en el computador, es decir que las funciones de control y programación se las accede desde un PC, lo que no sucede con un PLC que su programación y ejecución de órdenes se da desde el propio dispositivo.

### **3.2.2 Módulos de red**

Los módulos de red de trabajo proveen una conexión abierta para redes industriales. Estos módulos se comunican con los módulos locales de I/O mediante el bus local de alta velocidad formado mediante las conexiones de las bases de terminales. Las opciones para el trabajo en red son RS-232, RS-485, Ethernet (de 10 y 100 Mb/s), y FOUNDATION Fieldbus.

El sistema FieldPoint está diseñado para operar en los duros ambientes de las aplicaciones industriales. La mayoría de los componentes de FieldPoint pueden operar dentro de un amplio rango de temperatura que va de -40 a 70°C. Los bancos de aislamiento están estandarizados y dentro de un rango de 2500 Vrms, con doble aislamiento para una trabajo seguro con un voltaje de 250Vrms.

Mediante el FP-1000 y el FP-1001, se tienen módulos de interfase serial para RS-232 y RS-485 respectivamente, usted puede fácilmente implementar I/O y aplicaciones de adquisición de datos basados en PC. La interfase RS-232 permite una conexión directa al puerto COM de la PC, mientras que el RS-485 provee a un bajo costo, una conexión de larga distancia (1,2Km), una red de trabajo multipunto para I/O FieldPoint.

#### **3.2.2.1 INTERFASE FP-1000 RS 232/485**

El FP-1000 conecta una red de uno a nueve módulos entrada /salida FieldPoint bajo una red de datos con norma serie RS-232 normal y RS-485; NI FP-1000 entrega una conexión para FieldPoint que es fácil unir a un PC y fácil usar. El FP-1000 maneja las comunicaciones entre el anfitrión (host) PC y los módulos de I/O sobre un bus local de alta velocidad formado por las bases terminales de FieldPoint.

El módulo de interfaz de red también proporciona el diagnóstico y auto configuración para simplificar la instalación, uso y mantenimiento. Las aplicaciones pueden comunicarse fácilmente con el FP-1000 para intercambio de datos ; la interfaz de datos serial puede comunicar con un FP-20xx como controlador o con una computadora de Windows corriendo LabVIEW , LabWindows/CVI , Measurement Studio o aplicaciones de software de aplicación de OPC-cliente ; usando Optomux se puede comunicar con el FP-1000 con plataformas que no sean Windows como Mac OS y Linux. Usando el FP-100x, se puede construir modulares distribuidos flexibles rápidamente para medida y sistemas de automatización; la figura 3.12 indica el módulo de aplicaciones.

## RS-232 and RS-485 Serial Network Interfaces

**NI FP-1000, NI FP-1001**

- PC-based distributed I/O serial network interfaces
- Standard serial networking
  - RS-232 serial to a PC port
  - RS-485 for industrial multidrop applications
- Connects up to 25 FieldPoint banks to a serial bus
- Industrial-grade reliability
  - Automatic self-diagnostics
  - Isolated communication bus to I/O modules
  - Network watchdog timer
  - Configurable I/O power up states

**Operating Systems**

- Windows 2000/NT/XP

**Recommended Software**

- LabVIEW
- LabVIEW Datalogging and Supervisory Control Module

**Other Compatible Software**

- LabWindows/CVI
- Measurement Studio
- Lookout
- VI Logger

**Driver Software (included)**

- Measurement & Automation Explorer
- OPC server (2.0 compliant)

FieldPoint Serial Network Interfaces



Figura 3.12 Interfase serial RS 232 /485

El FP-1000 incluye un pórtico de 9 pines RS-232 normal y un pórtico aislado Full duplex RS-485 se puede comunicar hasta con 24 módulos con el FP1001 en nuestro proyecto utilizaremos el pórtico RS-232 según se indica la figura 3.13a.

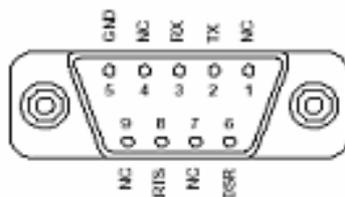


Figura 3.13a conexión RS232



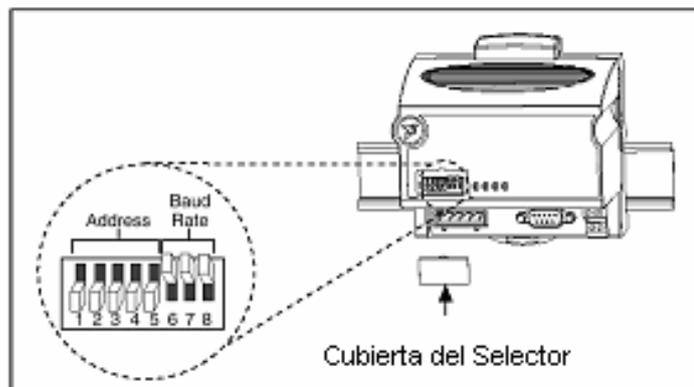
Figura 3.13b Conexión 485

Tiene la opción de cronómetros guardián (watchdog timer) para poder detectar y responder a fallas de la red, en ese momento se pone inactivo y las salidas son fijadas a un estado predefinido según sea configurado. El tipo de alimentación oscila en un rango de 11 a 30 VDC redistribuyendo la energía a todos módulos de I/O.

Para configurar el FP-1000 existen 8 interruptores. La figura 3.14 indica la forma que se configuran:

- La dirección del equipo switch 1 – 5 hasta 25 direcciones 0, 10,20....240
- La velocidad de transferencia (baud rate) switch 6-8 desde 300 bps. hasta 115.2 Kbps

Para un solo FP 1000 el equipo está preconfigurado con dirección cero y un baud rate 115.2 Kbps. permitiendo un rápido desempeño y no será necesario cambiarlo al menos que se experimente problemas de comunicación, los módulos adyacentes I/O asumirán las direcciones siguientes es decir la dirección 1, 2, etc.; el único cuidado es que si se va instalar más de un módulo en configuración 485 se debe verificar que dicha velocidad de transferencia sea la misma.



**Figura 3.14 Configuración del FP 1000**

Al aplicar la energía al módulo FP-1000 según Fig. 3.15 se tiene 4 leds de indicación de estatus:

- POWER en verde cuando la energía aplicada es correcta y no existe sobrecarga.
- NETWORK en amarillo cuando se encuentra en comunicación con el computador anfitrión (host) y el módulo.
- ACCESS en amarillo cuando el FP 1000 o cualquier I/O responde al anfitrión indicando que el módulo fue diseccionado correctamente.
- STATUS en rojo al detectar falla por los siguientes motivos: cuando la inicialización ha fallado, al detectar error en la conexión y las bases o conectores y cuando un nuevo módulo I/O no fue aceptado (máximo 9 por dirección) .

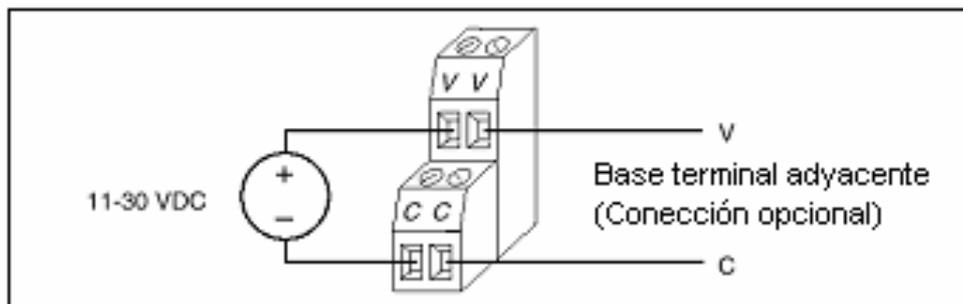


Figura 3.15 alimentación de módulo FP-1000

### 3.2.3 Módulos de entrada y salida I/O

El sistema FieldPoint incluye dos tipos de módulos I/O estándar, uno de 8 y 16 canales y otro de doble canal para máxima combinación de flexibilidad. Los módulos I/O poseen salidas y entradas analógicas aisladas eléctricamente para una amplia variedad de señales y tipos de sensores, y los cuales son intercambiables aún en funcionamiento y autoconfigurables para una fácil instalación y mantenimiento.

Los módulos I/O FieldPoint entregan un nivel de calidad apropiado para la industria, robustez y flexibilidad. Primeramente hay una amplia variedad de módulos I/O que entregan una alta precisión en las entradas I/O analógicas, conexión directa de sensores, I/O discretas para el campo industrial, alta velocidad de conteo, y generación de pulsos. Además las I/O FieldPoint ayuda a minimizar el tiempo muerto con un alto voltaje de aislamiento, instalación Plug and Play aún en estado de funcionamiento, diagnóstico de cableado, estados de encendido programables y estados de modos de falla, y tarjetas de etiquetación con documentación de cableado para una referencia rápida.

FieldPoint también entrega un sin igual nivel de flexibilidad, ya sea mediante los módulos I/O de 4 ó 16 canales, o mediante los módulos de doble canal que tiene una mayor modularidad. Con los módulos de doble canal, se puede evitar el tener canales sin uso, lo cual permite una implementación que se ajuste a las necesidades sin gastos innecesarios y sistemas más compactos. Además los módulos de doble canal proveen aislamiento individual al igual que el módulo a módulo para módulos analógicos.

### **3.2.3.1. Módulo Salida Analógica FP-AO-200**

Los FP-AO-200 son versátiles módulos de salida analógica que pueden ser usados para control de válvulas, salida como una medida acondicionada en corriente, y otros actuadores industriales. Todos los módulos incluyen protección de sobre rango y diagnósticos de operación (sobrecargas o lazo abierto) para asegurar la instalación y mantenimiento sin preocupaciones. Los módulos aceptan unidades predefinidas por el usuario en este caso salida en amperios o miliamperios.

El FP-AO-200 es un módulo FieldPoint de salida analógico con ocho lazos de salida de corriente en un rango de 0-20/4-20 mA. Se puede usar el FP-AO-200 para manejar dispositivos que usan lazos de corriente de control estándar en los

rangos ya indicados Las características principales del módulo según la figura 3.16 son:

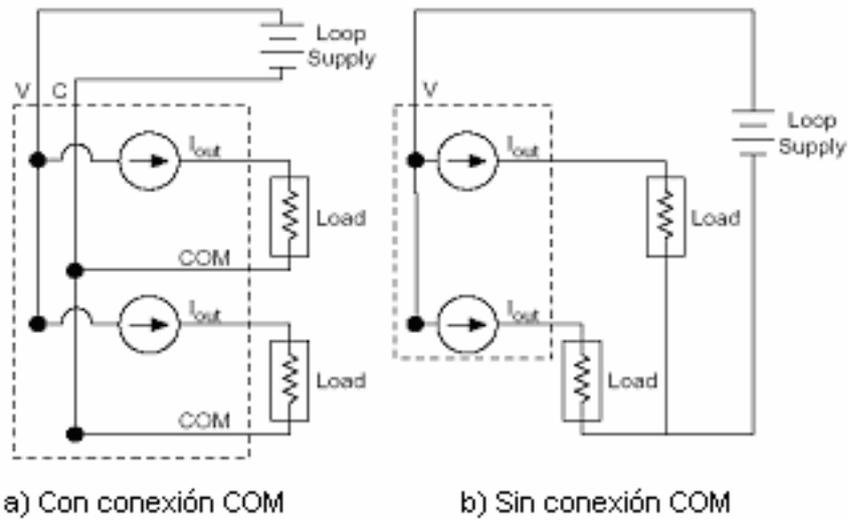
- Ocho salidas 0-20/4-20 mA.
- 0.5 mA de disponibilidad de sobre rango para evitar sobrecargas.
- 12 bits de resolución de DAC, 1 en 4096 6uA/bit
- Indicador de lazo en circuito abierto.
- Impedancia de carga sobre 1 KOhm (alimentación de 24 voltios).
- Protección contra corto circuitos.
- Temperatura de operación entre -40° a +70° C.
- Rango de máxima conversión 200actualizaciones /s. ; 200Hz.
- Velocidad de cambio .4 mA/us.



**Figura 3.16 Módulo AO 200 salida en mA.**

FP-AO-200 suministra una corriente a la carga o dispositivo del campo desde una fuente externa de alimentación. Las figuras 3.17a y 3.17b muestran las conexiones básicas según sea el requerimiento; el terminal positivo se representa con V y el negativo con C; cada canal tiene una salida Iout; un terminal común, COM (internamente se conecta al terminal C); y un terminal del suministro, Vsup

(internamente conectado a V) guía la energía a los aparatos que requieran energía suplementaria .



Figuras 3.17 a/b tipos de conexión de AO 200

La tabla 3.3 asigna los terminales asociados con cada canal:

Channel	Terminal Numbers		
	$I_{out}$	COM	$V_{sup}$
0	1	2, 18	17
1	3	4, 20	19
2	5	6, 22	21
3	7	8, 24	23
4	9	10, 26	25
5	11	12, 28	27
6	13	14, 30	29
7	15	16, 32	31

Tabla 3.3 FP AO-200

EL FPAO-200 puede operar con una fuente externa en rango de 5 a 24 Vdc; voltaje que dictamina la máxima impedancia de carga que el módulo puede manejar, es decir con 24 Voltios cada salida podrá manejar hasta 1 Kohm, con 5 voltios la carga soportada será de 100 Ohms; el módulo detecta y reporta condición de error en excesivas cargas o insuficiente voltaje de suministro; El esquema de lazo de salida se representa en la figura 3.18, cada canal tiene un circuito de monitoreo el cual compara la salida actual con el valor deseado de corriente; si no alcanza al valor deseado de corriente el estatus cambia a rojo por cada canal afectado .

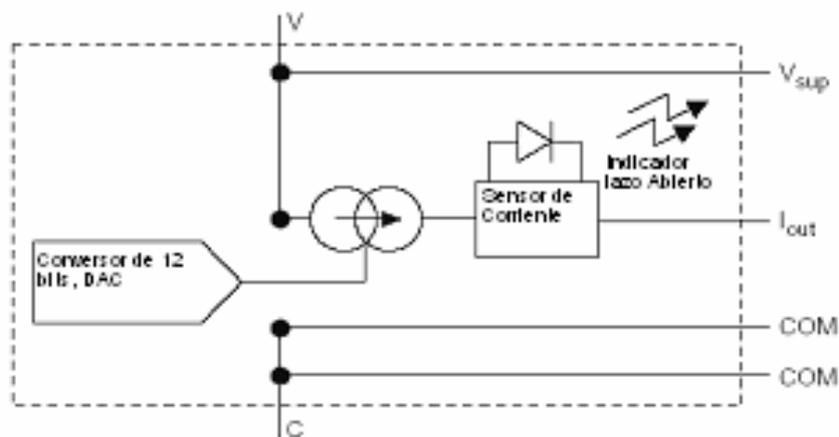


Figura 3.18 lazo de salida corriente

El módulo actualiza continuamente la salida enviado por la red, el tiempo que toma en responder a un cambio en un solo canal está entre los 3 y 6 ms; mientras que el tiempo para cambiar en todos los 8 canales está entre 24 a 27 ms.

### 3.2.3.2 Módulo de Entrada Analógica FP-AI-110

El FP-AI-110 es un módulo de entrada analógico con 8 canales como se muestra en la figura 3.19 que puede ser configurado en voltaje o corriente indistintamente; siendo ideal para señales de baja frecuencia. Se incluyen 3 filtros para rechazar el ruido; entre las características principales se tiene:

- Ocho entradas analógicas de voltaje o corriente
- 11 rangos en las entradas  $\pm 10$  V,  $\pm 5$  V,  $\pm 1$  V,  $\pm 300$  mV,  $\pm 60$  mV, 0-10 V, 0-5 V, 0-1 V,  $\pm 20$  mA, 0-20 mA, 4-20 mA.
- Selección de 3 filtros frecuencias 50 Hz, 60 Hz y 500 Hz
- 16 bit resolución
- Voltaje de aislamiento 3000 V con doble aislamiento de 250 V



Figura 3.19 FP-AI-110

El sistema de conexionado del módulo es específico para cada aplicación según la figura 3.20 la tabla de conexionado del lazo se presenta en la tabla 3.4.

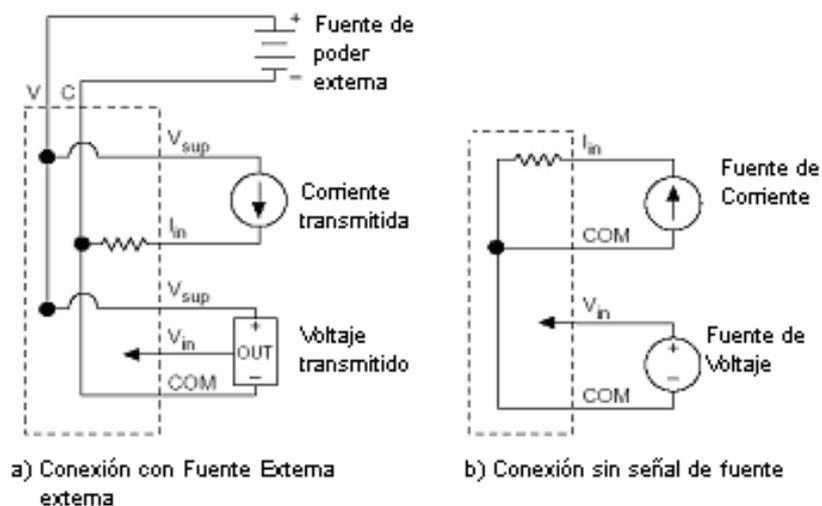


Figura 3.20 Conexionado en FP AI-110

Cada canal tiene una entrada separada para el terminal de voltaje ( $V_{in}$ ) o corriente ( $I_{in}$ ); ambos voltaje y corriente son referenciados en un terminal común.

La sección de entrada del FP AI-110 es eléctricamente aislada, el punto de referencia es flotante, es decir internamente no está referenciado a tierra o a cualquier otra señal referenciada de otro módulo, este tipo de entrada es generalmente llamada seudo diferencial; la figura 3.21 indica un circuito de un solo canal aplicado

Channel	Terminal Numbers			
	$V_{in}$	$I_{in}$	$V_{sup}$	COM
0	1	2	17	18
1	3	4	19	20
2	5	6	21	22
3	7	8	23	24
4	9	10	25	26
5	11	12	27	28
6	13	14	29	30
7	15	16	31	32

Tabla 3.4 Terminales de conexión FP AI-110

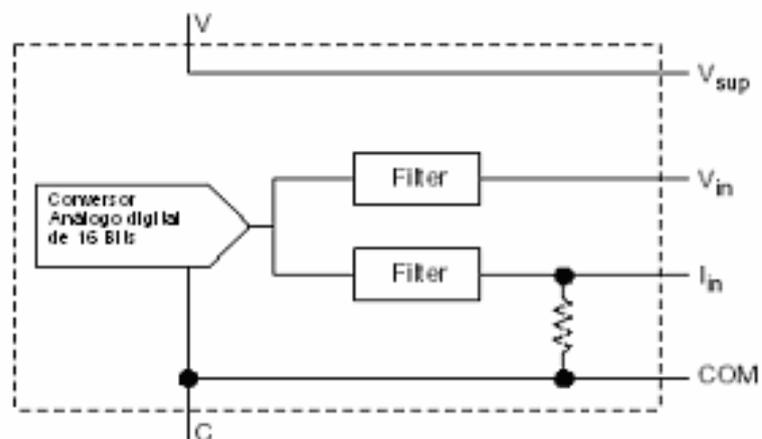


Figura 3.21 Canal entrada FP AI-110

Cuando los canales son fijados con filtros a 50/60Hz cada canal muestreará cada 1.47 hasta 1.23 segundos, mientras que cuando se filtra a 500 Hz cada canal será muestreado cada .173 segundos.

### **3.2.4 Beneficios de trabajar en red**

Un módulo de red FieldPoint conecta un banco de módulos de I/O analógicos y digitales a una red industrial, como Ethernet, la FOUNDATION Fieldbus, RS-232, RS-485, e inalámbrico.

El sistema FieldPoint incluye una variedad de módulos I/O analógicos y digitales aislados eléctricamente, una base de conexiones, e interfaces de redes de trabajo de fácil conexión para tecnologías estándar y abiertas. Este sistema Fieldpoint también incluye algunas opciones de software, tales como LabVIEW, Real Time. Además de las fáciles herramientas de configuración que posee, el software de FieldPoint incluye un servidor OPC para conexiones estandarizadas para cualquier paquete de software compatible con OPC.

Los módulos de red de trabajo proveen una conexión abierta para redes industriales. Estos módulos se comunican con los módulos locales de I/O mediante el bus local de alta velocidad formado mediante las conexiones de las bases de terminales. Las opciones para el trabajo en red son RS-232, RS-485, Ethernet (de 10 y 100 Mb/s), y FOUNDATION Fieldbus.

Mediante el FP-1000 y el FP-1001, se tienen módulos de interfase serial para RS-232 y RS-485 respectivamente, usted puede fácilmente implementar I/O y aplicaciones de adquisición de datos basados en PC. La interfase RS-232 permite una conexión directa al puerto COM de la PC, mientras que el RS-485 provee a un bajo costo, una conexión de larga distancia (1,2Km), una red de trabajo multipunto para I/O FieldPoint

El módulo de red FP-1000 comunica un máximo porcentaje de 115.2 Kb/s. El porcentaje de baudios y dirección de red son fijados con interruptores localizados sobre el módulo de red. La dirección del módulo de red fija la dirección de base para que los módulos I/O sean conectados al módulo de red. Por ejemplo, si se fija el módulo de red en 20, la dirección de los módulos I/O conectados directamente al módulo de red son automáticamente fijados a 21, 22 y así sucesivamente.

El FP-1000 incluye un puerto estándar RS-232 de 9 pines y un puerto aislado full-duplex RS-485. El puerto RS-485 incluye un repetidor de red que puede manejar una red de 1200m (4000ft.).

### **3.2.5 Conexión de sensores en los dispositivos modulares en forma directa**

Los módulos I/O FieldPoint entregan un nivel de calidad apropiado para la industria, robustez y flexibilidad. Primeramente hay una amplia variedad de módulos I/O que entregan una alta precisión en las entradas I/O analógicas, conexión directa de sensores, I/O discretas para el campo industrial, alta velocidad de conteo, y generación de pulsos. Además las I/O FieldPoint ayuda a minimizar el tiempo muerto con un alto voltaje de aislamiento, instalación Plug and Play aún en estado de funcionamiento, diagnóstico de cableado, estados de encendido programables y estados de modos de falla, y tarjetas de etiquetación con documentación de cableado para una referencia rápida.

También los módulos I/O FieldPoint que se pueden conectar directamente son los de los sensores tales como:

- FP-RTD-122:
  - Módulo que tiene 8 canales
  - 3 líneas RTD
  - Un módulo de entrada de resistencias
  - Referirse a la figura 3.22

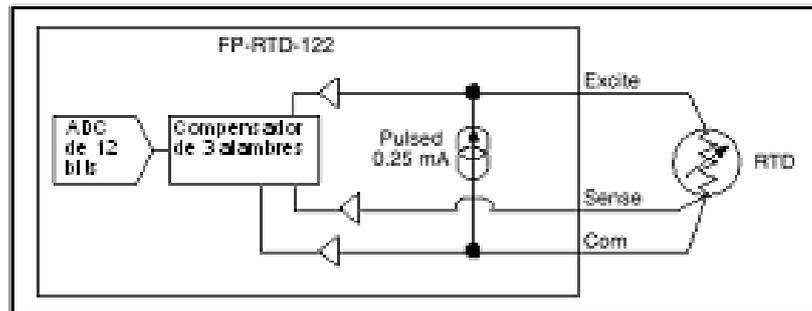


Figura 3.22 FP-RTD-122 Circuito de entrada

- FP-TC-120:
  - Módulo de 8 canales de entrada de Termocupla
  - Referirse a la figura 3.23

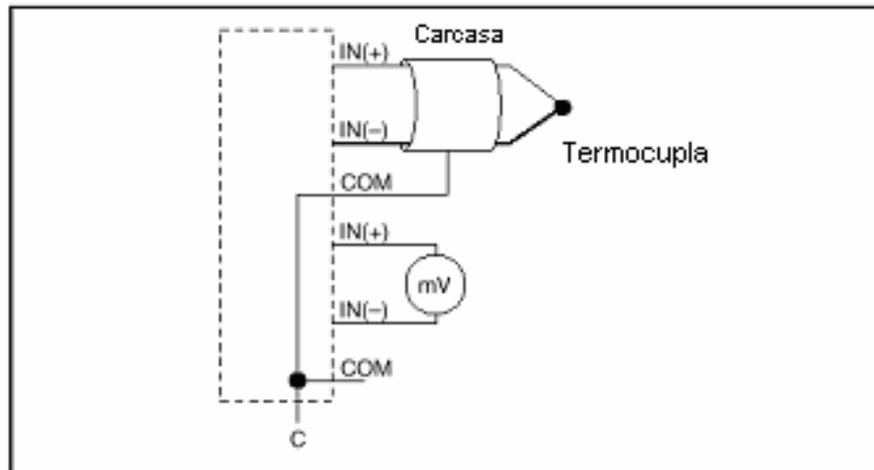


Figura 3.23 Presentación de una conexión básica con uso de dos Canales

### 3.2.6 Software de fácil Configuración

Cuando se instala el software de FieldPoint, una librería de FieldPoint es creada si LabVIEW está instalado en el computador. Los VI'S que se indican a continuación acceden directamente a las I/O que se configura con el programa FieldPoint Explorer.

### **3.2.6.1 FIELD POINT EXPLORER**

El FieldPoint Explorer es el utilitario de hardware y software por medio del cual LabVIEW se comunicará con los módulos de FieldPoint; las funciones principales son:

- Configurar las características de hardware a ser conectados en una red de datos
- Configurar el nombre del módulo que reconocerá FieldPoint (llamado tag name)
- Escribir y leer valores desde un módulo I/O de FieldPoint ya reconocido

El paso previo para correr el software de FieldPoint es correr primero el FieldPoint Explorer en el que se configura los ajustes programables de los módulos que se integrarán para realizar el controlador, el rango de entrada del módulo de entrada y el rango de salida del módulo de salida del que se aplicará la señal de control al proceso como también el pórtico de salida.

### **3.2.7 Interfases OPC**

El sistema FieldPoint también incluye un servidor OPC gratis para conexiones estándar en la industria. Por lo tanto, se puede fácilmente integrar sistemas FieldPoint con cualquier software que tenga la capacidad de cliente OPC. Porque OPC usa tecnología DCOM, las aplicaciones de cliente pueden acceder a datos de un servidor OPC remoto que se encuentra dentro de la red de trabajo.

El sistema FieldPoint también incluye VIs para fácil acceso desde LabVIEW o LabVIEW Real-Time, integrando controladores de Lookout, y un controlador de instrumentos para Measurement Studio.

### **3.3 PLC SIMATIC S7 200.**

La gama S7-200 comprende diversos sistemas de automatización pequeños (Micro-PLCs) que se pueden utilizar para numerosas tareas.

Gracias a su diseño compacto, su bajo costo y su amplio juego de operaciones, los sistemas de automatización S7-200 son idóneos para controlar tareas sencillas. La gran variedad de modelos S7-200 y el software de programación basado en Windows ofrecen la flexibilidad necesaria para solucionar las tareas de automatización.

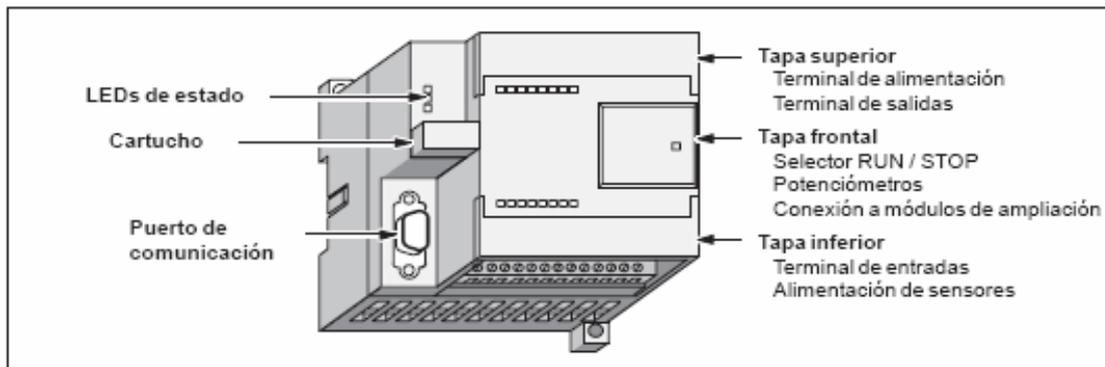
La gama S7-200 SIMATIC cumple las siguientes normas:

- ❖ Directiva de Baja Tensión de la Comunidad Europea 73/23/CEE (EN 61131-2): Autómatas programables - requisitos del equipo.
- ❖ Directiva EMC de la Comunidad Europea (CE) 89/336/CEE
  - Norma de emisión electromagnética
  - EN 50081-1: entornos residenciales, comerciales y semi-industriales
  - EN 50081-2: entornos industriales
  - Norma de inmunidad electromagnética
  - EN 61000-6-2: entornos industriales
- ❖ Underwriters Laboratories, Inc.  
UL 508 registrado (Industrial Control Equipment) N° de registro: E75310
- ❖ Canadian Standards Association: CSA C22.2 n° 142, certificado (Process Control Equipment)
- ❖ Factory Mutual Research: FM clase I, categoría 2, grupos A, B, C y D “Hazardous Locations” y clase I, sección 2, IIC, T4.

#### **3.3.1 Características técnicas generales.**

La CPU S7-200 incorpora en una carcasa compacta un microprocesador, una fuente de alimentación integrada, así como circuitos de entrada y de salida que conforman un potente Micro-PLC, tal como se muestra en la figura 3.24. Tras

haber cargado el programa en el S7-200, éste contendrá la lógica necesaria para observar y controlar los aparatos de entrada y salida de la aplicación.



**Figura 3.24 PLC Simatic S7 200**

Siemens ofrece diferentes modelos de CPU's S7-200 que incorporan una gran variedad de funciones y prestaciones para crear soluciones efectivas de automatización destinadas a numerosas aplicaciones. En la tabla 3.5 se comparan de forma resumida algunas de las funciones de la CPU.

<b>Función</b>	<b>CPU 221</b>	<b>CPU 222</b>	<b>CPU 224</b>	<b>CPU 226</b>	<b>CPU 226XM</b>
Dimensiones físicas (mm)	90 x 80 x 62	90 x 80 x 62	120,5 x 80 x 62	190 x 80 x 62	190 x 80 x 62
Memoria del programa	2048 palabras	2048 palabras	4096 palabras	4096 palabras	8192 palabras
Memoria de datos	1024 palabras	1024 palabras	2560 palabras	2560 palabras	5120 palabras
Memoria de backup	50 horas (típ.)	50 horas (típ.)	190 horas (típ.)	190 horas (típ.)	190 horas (típ.)
E/S integradas	6 E / 4 S	8 E / 4 S	14 E / 10 S	24 E / 16 S	24 E / 16 S
Módulos de ampliación	0	2	7	7	7
Contadores rápidos					
Fase simple	4 a 30 kHz	4 a 30 kHz	6 a 30 kHz	6 a 30 kHz	6 a 30 kHz
2 fases	2 a 20 kHz	2 a 20 kHz	4 a 20 kHz	4 a 20 kHz	4 a 20 kHz
Salidas de impulsos (DC)	2 a 20 kHz	2 a 20 kHz	2 a 20 kHz	2 a 20 kHz	2 a 20 kHz
Potenciómetros analógicos	1	1	2	2	2
Reloj de tiempo real	Cartucho	Cartucho	Incorporado	Incorporado	Incorporado
Puertos de comunicación	1 RS-485	1 RS-485	1 RS-485	2 RS-485	2 RS-485
Aritmética en coma flotante	Sí				
Tamaño de la imagen de E/S digitales	256 (128 E / 128 S)				
Velocidad de ejecución booleana	0,37 microsegundos/operación				

**Tabla 3.5 Características técnicas de la CPU S7 200.**

### **3.3.1.1 Módulos de ampliación S7 200.**

La gama S7-200 incluye una gran variedad de módulos de ampliación para poder satisfacer aún mejor los requisitos de la aplicación. Estos módulos se pueden

utilizar para agregar funciones a la CPU S7-200. En la tabla 3.6 se muestra una lista de los módulos de ampliación disponibles en la actualidad.

Módulos de ampliación		Tipos		
Módulos digitales	Entradas	8 entradas DC	8 entradas AC	
	Salidas	8 salidas DC	8 salidas AC	8 salidas de relé
	Combinación	4 entradas DC / 4 salidas DC	8 entradas DC / 8 salidas DC	
		4 entradas DC / 4 salidas de relé	16 entradas DC / 16 salidas DC	8 entradas DC / 8 salidas de relé
			16 entradas DC / 16 salidas de relé	
Módulos analógicos	Entradas	4 entradas analógicas	4 entradas termopar	2 entradas RTD
	Salidas	2 salidas analógicas		
	Combinación	4 entradas analógicas / 1 salida analógica		
Módulos inteligentes		Posición	Módem	PROFIBUS-DP
Otros módulos		AS-Interface		

**Tabla 3.6 módulos de ampliación a la CPU S7 200**

### 3.3.1.2 Paquete de programación.

El paquete de programación STEP 7-Micro/WIN constituye un entorno de fácil manejo para desarrollar, editar y observar el programa necesario con objeto de controlar la aplicación. STEP 7-Micro/WIN comprende tres editores que permiten desarrollar de forma cómoda y eficiente el programa de control. Para encontrar fácilmente las informaciones necesarias, STEP 7-Micro/WIN incorpora una completa Ayuda en pantalla y un CD de documentación.

Para ejecutar el paquete de aplicación de software descrito los requisitos del sistema deberán ser los siguientes:

- ❖ Sistema operativo:  
Windows 95, Windows 98, Windows 2000, Windows ME (Millennium Edition), Windows XP o Windows NT 4.0(o una versión posterior).
- ❖ 50 MB libres en el disco duro (como mínimo).
- ❖ Ratón (recomendado).

### **3.3.1.3 Opciones de comunicación.**

Siemens ofrece dos opciones de programación para conectar el PC al S7-200, a saber: una conexión directa vía un cable PC/PPI, o bien un procesador de comunicaciones (CP) con un cable MPI para redes MPI y PROFIBUS-DP.

El cable de programación PC/PPI es el método más usual y más económico de conectar el PC al S7-200. Este cable une el puerto de comunicación del S7-200 con el puerto serie del PC. El cable de programación PC/PPI también se puede utilizar para conectar otros dispositivos de comunicación al S7-200.

Para poder utilizar el cable MPI es preciso instalar también un procesador de comunicaciones (CP) en el PC. El CP incorpora el hardware adicional necesario para establecer enlaces a velocidades de transferencia más elevadas, así como para procesar la comunicación rápida en la red.

### **3.3.2 Generalidades del S7-200.**

La función principal del S7-200 consiste en vigilar las entradas de campo y, conforme a la lógica de control, activar o desactivar los aparatos de salida de campo.

El S7-200 ejecuta cíclicamente la lógica de control del programa, leyendo y escribiendo datos. El funcionamiento básico del S7-200 es muy sencillo:

- ❖ El S7-200 lee el estado de las entradas.
- ❖ El programa almacenado en el S7-200 utiliza las entradas para evaluar la lógica. Durante la ejecución del programa, el S7-200 actualiza los datos.
- ❖ El S7-200 escribe los datos en las salidas.

La figura 3.25 muestra cómo se procesa un esquema de circuitos simple en el S7-200. En este ejemplo, el estado del interruptor para arrancar el motor se combina con los estados de otras entradas. El resultado obtenido establece entonces el estado de la salida que corresponde al actuador que arranca el motor.

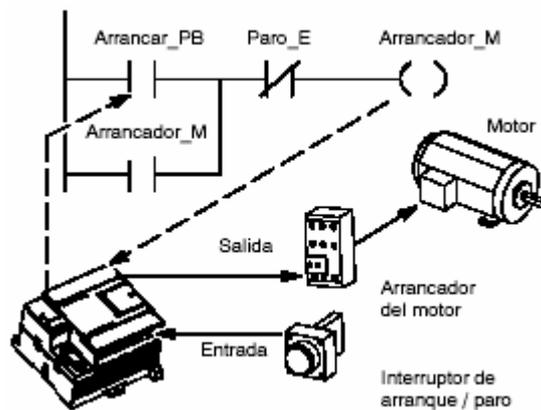


Figura 3.25 Control de entradas y salidas

El S7-200 ejecuta una serie de tareas de forma repetitiva. Esta ejecución se denomina ciclo. Como muestra en la figura 3.26, el S7-200 ejecuta la mayoría de las tareas siguientes (o todas ellas) durante un ciclo:

- ❖ Leer las entradas: el S7-200 copia el estado de las entradas físicas en la imagen del proceso de las entradas.
- ❖ Ejecutar la lógica de control en el programa: el S7-200 ejecuta las operaciones del programa y guarda los valores en las diversas áreas de memoria.
- ❖ Procesar las peticiones de comunicación. El S7-200 ejecuta las tareas necesarias para la comunicación.
- ❖ Efectuar el autodiagnóstico de la CPU: el S7-200 verifica si el firmware, la memoria del programa y los módulos están trabajando correctamente.
- ❖ Escribir en las salidas: los valores almacenados en la imagen del proceso de las salidas se escriben en las salidas físicas.

La ejecución del ciclo depende de si el S7-200 está en modo STOP o RUN. El programa se ejecutará si el S7-200 está en modo RUN. En cambio, no se ejecutará en modo STOP.

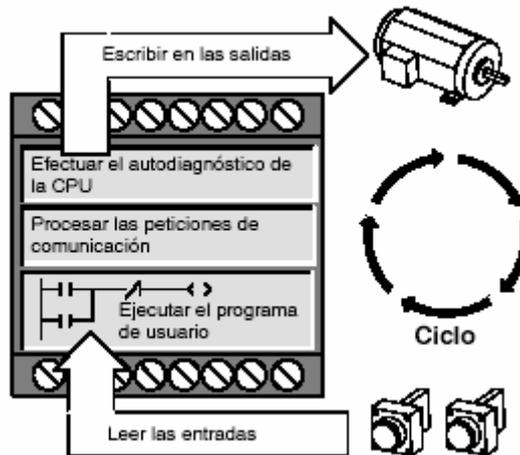


Figura 3.26 Ciclo del S7-200

### 3.3.2.1 Acceso a los datos del S7-200.

El S7-200 almacena información en diferentes áreas de la memoria que tienen direcciones unívocas. Es posible indicar explícitamente la dirección a la que se desea acceder. El programa puede acceder entonces directamente a la información. La tabla 3.7 muestra el rango de números enteros representables en diversos tamaños de datos.

Representación	Byte (B)	Palabra (W)	Palabra doble (D)
Entero sin signo	0 a 255 0 a FF	0 a 65.535 0 a FFFF	0 a 4.294.967.295 0 a FFFF FFFF
Entero con signo	-128 a +127 80 a 7F	-32.768 a +32.767 8000 a 7FFF	-2.147.483.648 a +2.147.483.647 8000 0000 a 7FFF FFFF
Real IEEE en coma flotante de 32 bits	<i>No aplicable</i>	<i>No aplicable</i>	+1,175495E-38 a +3,402823E+38 (positivo) -1,175495E-38 a -3,402823E+38 (negativo)

Tabla 3.7 Rangos de los datos

### 3.3.3 Entradas y salidas analógicas.

El S7-200 convierte valores reales analógicos (por ejemplo, temperatura, tensión, etc.) en valores digitales en formato de palabra (de 16 bits). A estos valores se accede con un identificador de área (AI), seguido del tamaño de los datos (W) y de la dirección del byte inicial.

El S7-200 convierte valores digitales en formato de palabra (de 16 bits) en valores reales analógicos (por ejemplo, intensidad o tensión). Estos valores analógicos son proporcionales a los digitales.

A los valores analógicos se accede con un identificador de área (AQ), seguido del tamaño de los datos (W) y de la dirección del byte inicial.

Las entradas y salidas integradas en la unidad central de procesamiento (CPU) tienen direcciones fijas. Para agregar a la CPU entradas y salidas adicionales, se pueden conectar módulos de ampliación a la derecha de la CPU S7-200, formando así una cadena de entradas y salidas (E/S).

Las direcciones de las E/S de cada módulo vienen determinadas por el tipo de E/S y por la posición relativa del módulo en la cadena (con respecto al anterior módulo de E/S del mismo tipo).

Por ejemplo, un módulo de salidas no afecta a las direcciones de un módulo de entradas y viceversa. Igualmente, los módulos analógicos no afectan al direccionamiento de los módulos digitales y viceversa.

#### **3.3.4 Seleccionar el modo de operación el S7-200.**

El S7-200 tiene dos modos de operación, a saber: STOP y RUN. El diodo luminoso (LED) ubicado en la parte frontal de la CPU indica el modo de operación actual. En modo STOP, el S7-200 no ejecuta el programa.

Entonces es posible cargar un programa o configurar la CPU. En modo RUN, el S7-200 ejecuta el programa.

- ❖ Para cambiar el S7-200 a modo STOP es posible introducir la correspondiente operación (STOP) en el programa. Ello permite detener la ejecución del programa en función de la lógica.

### **3.3.5. Sistemas de control de lazo PID.**

La operación Regulación PID (PID) ejecuta el cálculo de un lazo de regulación PID en el LOOP referenciado, conforme a las informaciones de entrada y configuración definidas en TABLE (TBL).

La tabla del lazo almacena nueve parámetros que sirven para controlar y supervisar la operación del mismo. Incluye el valor actual y previo de la variable del proceso (valor real), la consigna, la salida o magnitud manipulada, la ganancia, el tiempo de muestreo, el tiempo de acción integral, el tiempo de acción derivada y la suma integral (bias).

Para poder realizar el cálculo PID con el intervalo de muestreo deseado, la operación PID deberá ejecutarse bien dentro de una rutina de interrupción temporizada, o bien desde el programa principal, a intervalos controlados por un temporizador. El tiempo de muestreo debe definirse en calidad de entrada para la operación PID a través de la tabla del lazo.

#### **3.3.5.1 Algoritmo PID.**

En modo estacionario, un regulador PID varía el valor de su salida para llevar a cero el error de regulación ( $e$ ). El error es la diferencia entre el valor de consigna (SP) (el punto de trabajo deseado) y la variable del proceso (PV) (el punto de trabajo real). El principio de una regulación PID se basa en la ecuación que se indica a continuación y que expresa la salida  $M(t)$  como una función de un término proporcional, uno integral y uno diferencial:

Salida = término proporcional + término integral + término diferencial

$$M(t) = K_C * e + K_C \int_0^t e dt + M_{\text{inicial}} + K_C * de/dt$$

donde:  $M(t)$  es la salida del lazo en función del tiempo  
 $K_C$  es la ganancia del lazo  
 $e$  es el error de regulación (diferencia entre la consigna y la variable de proceso)  
 $M_{\text{inicial}}$  es el valor inicial de la salida del lazo

Para poder implementar la función de regulación en un sistema digital, la función continua deberá cuantificarse mediante muestreos periódicos del valor del error, calculándose seguidamente el valor de la salida. La ecuación que constituye la base de la solución en un sistema digital es:

$$M_n = K_C * e_n + K_I * \sum_1^n e + M_{\text{inicial}} + K_D * (e_n - e_{n-1})$$

Salida = término proporcional + término integral + término diferencial

donde:  $M_n$  es el valor de salida del lazo calculado en el muestreo n-ésimo  
 $K_C$  es la ganancia del lazo  
 $e_n$  es el valor del error del lazo en el muestreo n-ésimo  
 $e_{n-1}$  es el valor previo del error de regulación (en el muestreo (n-1)-ésimo)  
 $K_I$  es la constante proporcional del término integral  
 $M_{\text{inicial}}$  es el valor inicial de la salida del lazo  
 $K_D$  es la constante proporcional del término diferencial

Para esta ecuación, el término integral se muestra en función de todos los términos del error, desde el primer muestreo hasta el muestreo actual. El término diferencial es una función del muestreo actual y del muestreo previo; mientras que el término proporcional sólo es función del muestreo actual. En un sistema digital no es práctico almacenar todos los muestreos del término del error, además de no ser necesario.

Puesto que un sistema digital debe calcular el valor de salida cada vez que se muestre el error, comenzando en el primer muestreo, basta con almacenar el valor previo del error y el valor previo del término integral. Debido a la naturaleza repetitiva de la solución basada en un sistema digital, es posible simplificar la ecuación a resolver en cada muestreo

Para calcular el valor de salida del lazo, el S7-200 utiliza una forma modificada de la ecuación simplificada anterior. Esta ecuación modificada equivale a:

$M_n$	=	$MP_n$	+	$MI_n$	+	$MD_n$
Salida	=	término proporcional	+	término integral	+	término diferencial
donde:	$M_n$	es el valor de salida del lazo calculado en el muestreo n-ésimo				
	$MP_n$	es el valor del término proporcional de salida del lazo en el muestreo n-ésimo				
	$MI_n$	es el valor del término integral de salida del lazo en el muestreo n-ésimo				
	$MD_n$	es el valor del término diferencial de salida del lazo en el muestreo n-ésimo				

La operación PID es simple, pero ofrece grandes prestaciones para ejecutar cálculos PID. Si se precisan funciones de postprocesamiento (tales como funciones de alarma o cálculos especiales en base a las variables de lazo), ello deberá implementarse utilizando las operaciones básicas soportadas por el S7-200 en cuestión.

La operación PID no comprueba si todos los valores de entrada en la tabla del lazo respetan los límites de rango. Es decir, el usuario deberá vigilar que la variable del proceso y la consigna (al igual que la suma integral y la variable del proceso previa, si se utilizan como entradas) sean números reales comprendidos entre 0.0 y 1.0.

### 3.3.6 Principios básicos de la comunicación en redes S7-200.

El S7-200 soporta redes compuestas por maestros y esclavos, pudiendo actuar tanto de maestro como de esclavo en la red. En cambio, STEP 7-Micro/WIN actúa siempre de maestro.

Los maestros pueden enviar una petición a otros aparatos de la red. Un maestro también puede responder a las peticiones de otros maestros incorporados en la red. Algunos maestros típicos son STEP 7-Micro/WIN, interfaces hombre-máquina (HMI), tales como el TD 200, así como sistemas de automatización S7-300 o S7-

400. El S7-200 actúa de maestro cuando le solicita información a otro S7-200 (comunicación punto a punto).

Un aparato que se haya configurado como esclavo sólo puede responder a las peticiones de un maestro. Un esclavo no puede iniciar una petición. El S7-200 actúa de esclavo en la mayoría de las redes. En su calidad de esclavo, el S7-200 responde a las peticiones de un maestro de la red, por ejemplo, de un panel de operador o de STEP 7-Micro/WIN.

La velocidad de transferencia de los datos en la red se indica, por lo general, en kilobits por segundo (kbit/s), o bien, en megabits por segundo (Mbit/s). La velocidad de transferencia mide cuántos datos se pueden transmitir en un determinado período.

Todos los aparatos que conforman la red se deben configurar de manera que transfieran datos a una misma velocidad de transferencia. Por tanto, el aparato más lento de la red determina la velocidad de transferencia máxima.

En la tabla 3.8 figuran las velocidades de transferencia que soporta el S7-200. La dirección de estación es un número unívoco que se asigna a cada aparato de la red. La dirección de estación unívoca garantiza que los datos sean enviados al o recibidos del aparato correcto. El S7-200 soporta direcciones de estación comprendidas entre 0 y 126. Si el S7-200 dispone de dos puertos, cada uno de ellos puede tener su propia dirección de estación. En la tabla 3.9 figuran los ajustes estándar (de fábrica) de los equipos S7-200.

<b>Red</b>	<b>Velocidad de transferencia</b>
Red estándar	9,6 kbit/s a 187,5 kbit/s
Utilizando un EM 277	9,6 kbit/s a 12 Mbit/s
Modo Freeport	1200 bit/s a 115,2 kbit/s

**Tabla 3.8 Velocidades de transferencia.**

Equipo S7-200	Dirección estándar
STEP 7-Micro/WIN	0
HMI (TD 200, TP u OP)	1
CPU S7-200	2

**Tabla 3.9 Direcciones estándar de los equipos.**

Las CPUs S7-200 soportan uno o varios de los protocolos de comunicación siguientes. Estos protocolos permiten configurar la red conforme al rendimiento y a la funcionalidad que exige la aplicación:

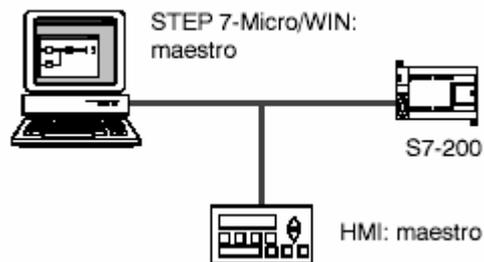
- ❖ Interfase punto a punto (PPI)
- ❖ Interfase multipunto (MPI)
- ❖ PROFIBUS

Basándose en la intercomunicación de sistemas abiertos (OSI) de la arquitectura de siete capas, estos protocolos se implementan en una red “token ring” (red de anillo con testigo) conforme al estándar PROFIBUS, definido en la Norma Europea EN 50170. Se trata de protocolos asíncronos de caracteres que utilizan un bit de inicio, ocho bits de datos, un bit de paridad par y un bit de parada. Los bloques de comunicación dependen de los caracteres especiales de inicio y de parada, de las direcciones de estación de fuente y de destino, de la longitud de los bloques y de la suma de verificación para garantizar la integridad de los datos. Los protocolos se pueden utilizar simultáneamente en una red sin que interfieran entre sí, con la condición de que usen una misma velocidad de transferencia.

PPI es un protocolo maestro-esclavo. Los maestros envían peticiones a los esclavos y éstos responden, como se muestra en la figura 3.27. Los esclavos no inician mensajes, sino que esperan a que un maestro les envíe una petición o solicite una respuesta.

Los maestros se comunican con los esclavos vía un enlace compartido que es gestionado por el protocolo PPI. El protocolo PPI no limita el número de maestros

que se pueden comunicar con un mismo esclavo. Sin embargo, la red no puede comprender más de 32 maestros.



**Figura 3.27 Red PPI**

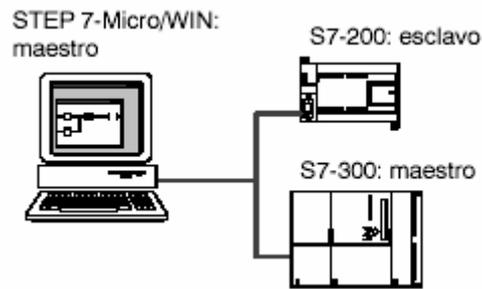
Si se selecciona el protocolo PPI Avanzado es posible establecer un enlace lógico entre los aparatos. En este caso, cada aparato soporta un número de enlaces limitado. En la tabla 3.10 figura el número de enlaces que soporta el S7-200.

Módulo		Velocidad de transferencia	Enlaces
CPU S7-200	Puerto 0	9,6 kbit/s, 19,2 kbit/s ó 187,5 kbit/s	4
	Puerto 1	9,6 kbit/s, 19,2 kbit/s ó 187,5 kbit/s	4
Módulo EM 277		9,6 kbit/s a 12 Mbit/s	6 por módulo

**Tabla 3.10 Enlaces soportados por S7-200**

El protocolo MPI soporta la comunicación maestro-maestro y maestro-esclavo como se muestra en la figura 3.28. Para comunicarse con una CPU S7-200, STEP 7-Micro/WIN establece un enlace maestro-esclavo. El protocolo MPI no sirve para comunicarse con una CPU S7-200 que actúe de maestra.

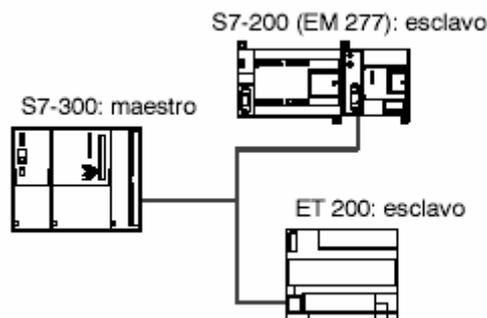
Los aparatos de la red se comunican a través de enlaces separados (gestionados por el protocolo MPI) entre dos aparatos cualquiera. La comunicación entre los aparatos se limita la cantidad de enlaces que soportan la CPU S7-200 o el módulo EM 277.



**Figura 3.28 Red MPI.**

El protocolo PROFIBUS se ha diseñado para la comunicación rápida con unidades periféricas descentralizadas (E/S remotas). Hay numerosos aparatos PROFIBUS ofrecidos por diversos fabricantes. Estos aparatos abarcan desde módulos sencillos de entradas o salidas hasta controladores de motores y sistemas de automatización (autómatas programables).

Por lo general, las redes PROFIBUS incorporan un maestro y varios esclavos, tal como se ve en la figura 3.29. La configuración del maestro permite detectar los tipos de esclavos conectados, así como sus respectivas direcciones. El maestro inicializa la red y verifica si los esclavos coinciden con la configuración. Continuamente, el maestro escribe los datos de salida en los esclavos y lee de allí los datos de entrada.



**Figura 3.29 Red Profibus.**

Una vez que un maestro DP haya configurado correctamente a un esclavo, éste último le pertenecerá. Si hay otro maestro en la red, tendrá apenas un acceso muy limitado a los esclavos del primer maestro.

### **3.3.6.1 Configurar la Red.**

Se debe instalar los dispositivos de supresión de sobretensiones apropiados en el cableado susceptible de recibir sobretensiones causadas por rayos.

También se debe colocar los conductores de señalización y los cables de comunicación en una misma canalización junto con los cables de corriente AC y los cables DC de alta tensión y de conmutación rápida. El cableado deberá efectuarse por pares; con el cable de neutro o común combinado con el cable de fase o de señal.

El puerto de comunicación del S7-200 no está aislado. Es recomendable que prevea un repetidor RS-485 o un módulo EM 277 para garantizar el aislamiento de la red.

Si se interconectan equipos con potenciales de referencia diferentes, podrían circular corrientes indeseadas por el cable de conexión.

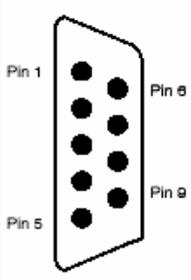
Estas corrientes pueden causar errores de comunicación o deteriorar los equipos. Para evitar corrientes indeseadas, vigile que todos los equipos conectados con un cable de comunicación compartan un circuito de referencia, o bien que estén aislados entre sí.

Los aspectos a considerar en la configuración de la red son:

- a) Determinar distancias, la velocidad de transferencia y el cable de red.**
- b) Usar repetidores de red.**
- c) Seleccionar el cable de red.**
- d) Asignación de pines.**

Los puertos de comunicación de las CPUs S7-200 son compatibles con el estándar RS-485 vía un conector D subminiatura de 9 pines, conforme al estándar PROFIBUS definido en la norma europea EN 50170. La tabla 3.11 muestra el

conector que ofrece el enlace físico para el puerto de comunicación, indicándose también las asignaciones de pines de los puertos de comunicación.

Enchufe	Nº de pin	Señal PROFIBUS	Puerto 0/Puerto 1
	1	Blindaje	Tierra
	2	Hilo de retorno 24 V	Hilo lógico
	3	Señal B RS-485	Señal B RS-485
	4	Petición de transmitir	RTS (TTL)
	5	Hilo de retorno 5 V	Hilo lógico
	6	+5 V	+5 V, 100 $\Omega$ resistor en serie
	7	+24 V	+24 V
	8	Señal A RS-485	Señal A RS-485
	9	No aplicable	Selección protocolo de 10 bits (entrada)
	Carcasa del enchufe	Blindaje	Tierra

**Tabla 3.11 Puerto de comunicaciones S7-200.**

### e) Polarizar y cerrar el cable de red.

Siemens ofrece dos tipos de conectores de bus que permiten conectar fácilmente varios aparatos a una red, a saber: un conector de bus estándar (en la tabla 7-6 figura la asignación de pines) y un conector que incorpora un puerto de programación, permitiendo conectar un PC/una PG o un aparato HMI (interface hombre-máquina) a la red, sin perturbar ningún enlace existente. El conector con puerto de programación transmite todas las señales del S7-200 (incluyendo los pines de potencia) a través del puerto de programación, siendo especialmente apropiado para conectar aparatos alimentados por el S7-200 (por ejemplo, un TD 200).

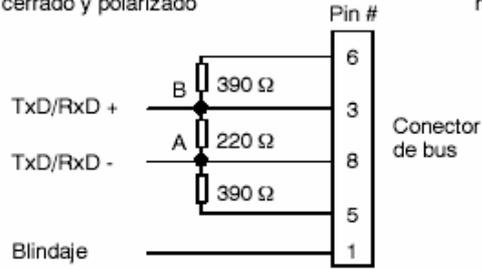
Ambos conectores poseen dos juegos de tornillos para fijar los cables de entrada y salida. Asimismo, disponen de interruptores para polarizar y cerrar la red de forma selectiva. La figura 3.30 muestra cómo polarizar y cerrar el cable de red.

Ambos extremos del cable se deben cerrar y polarizar.



Blindaje del cable: aprox. 12 mm deben hacer contacto con la guía de metal en todos los puntos.

Selector en posición ON:  
cerrado y polarizado



Selector en posición OFF:  
no cerrado ni polarizado

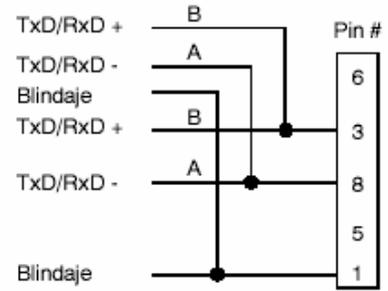


Figura 3.30 Polarizar y cerrar el cable de red.

## **CAPÍTULO IV**

### **REDES TCP/IP**

#### **4.1 INTRODUCCIÓN.**

Durante las últimas dos décadas ha habido un enorme crecimiento en la cantidad y tamaño de las redes. Muchas de ellas sin embargo, se desarrollaron utilizando implementaciones de hardware y software diferentes. Como resultado, muchas de las redes eran incompatibles y se volvió muy difícil para las redes que utilizaban especificaciones distintas poder comunicarse entre sí. Para solucionar este problema, la Organización Internacional para la Normalización (ISO) realizó varias investigaciones acerca de los esquemas de red. La ISO reconoció que era necesario crear un modelo de red que pudiera ayudar a los diseñadores de red a implementar redes que pudieran comunicarse y trabajar en conjunto (interoperabilidad) y por lo tanto, elaboraron el modelo de referencia OSI en 1984.

#### **4.2 MODELO GENERAL DE COMUNICACIONES.**

##### **4.2.1 Uso de las capas para analizar problemas en un flujo de materiales.**

El concepto de *capas* ayudará a comprender la acción que se produce durante el proceso de comunicación de un computador a otro. En la figura 4.1 se plantean preguntas que involucran el movimiento de objetos físicos como por ejemplo, el tráfico de autopistas o los datos electrónicos. Este desplazamiento de objetos, sea este físico o lógico, se conoce como flujo. Existen muchas capas que ayudan a describir los detalles del proceso de flujo. Otros ejemplos de sistemas de flujo son el sistema de suministro de agua, el sistema de autopistas, el sistema postal y el sistema telefónico.

#### 4.2.2 Origen, destino y paquetes de datos.

Como lo ilustra la figura 4.1, la información que viaja a través de una red se conoce como *paquete*, *datos* o *paquete de datos*. Un paquete de datos es una unidad de información, lógicamente agrupada, que se desplaza entre los sistemas de computación. Incluye la información origen junto con otros elementos necesarios para hacer que la comunicación sea factible y confiable en relación con los dispositivos destino. La dirección origen de un paquete especifica la identidad del computador que envía el paquete. La dirección destino especifica la identidad del computador que finalmente recibe el paquete.

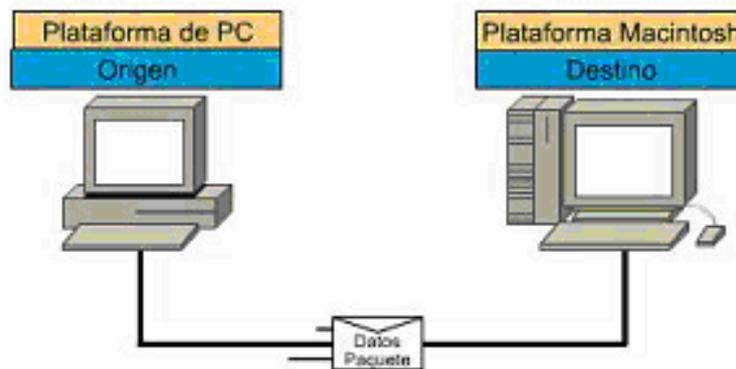


Figura 4.1 Origen y destino de los datos

#### 4.2.3 Medios.

En networking, un medio es el material a través del cual viajan los datos. Puede ser cualquiera de los siguientes materiales: Cable coaxial, par trenzado y fibra óptica.

Existen otros dos tipos de medios que son menos evidentes, pero que no obstante se deben tener en cuenta en la comunicación por redes. En primer lugar, está la atmósfera (en su mayor parte formada por oxígeno, nitrógeno y agua) que transporta ondas de radio, microondas y luz.

La comunicación sin ningún tipo de alambres o cables se denomina inalámbrica o comunicación de espacio abierto. Esto es posible utilizando ondas electromagnéticas (EM). Entre las ondas EM, que en el vacío viajan a velocidad de la luz, se incluyen las ondas de energía, ondas de radio, microondas, luz infrarroja, luz visible, luz ultravioleta, rayos x y rayos gama. Las ondas EM viajan a través de la atmósfera (principalmente compuesta de oxígeno, nitrógeno y agua), pero también viajan a través del vacío del espacio exterior (donde no existe prácticamente materia, ni moléculas ni átomos).

#### 4.2.4 Protocolo.

Para que los paquetes de datos puedan viajar desde el origen hasta su destino a través de una red, es importante que todos los dispositivos de la red hablen el mismo lenguaje o *protocolo*. Un *protocolo* es un conjunto de reglas que hacen que la comunicación en una red sea más eficiente.

Una definición técnica de un protocolo de comunicaciones de datos es: un conjunto de normas, o un acuerdo, que determina el formato y la transmisión de datos. La capa  $n$  de un computador se comunica con la capa  $n$  de otro computador. Las normas y convenciones que se utilizan en esta comunicación se denominan colectivamente *protocolo de la capa  $n$* . Así lo muestra la figura 4.2.

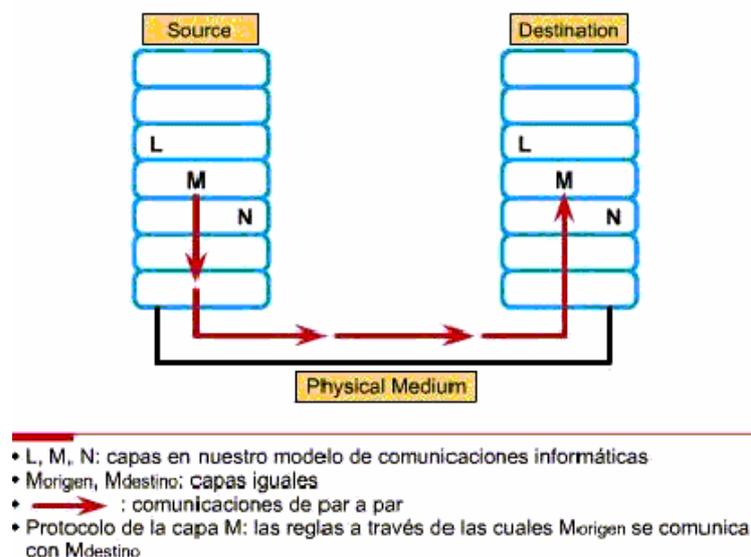


Figura 4.2 Protocolos computacionales

El modelo de referencia OSI es el modelo principal para las comunicaciones por red. Aunque existen otros modelos, en la actualidad la mayoría de los fabricantes de redes relacionan sus productos con el modelo de referencia OSI, especialmente cuando desean enseñar a los usuarios cómo utilizar sus productos. Los fabricantes consideran que es la mejor herramienta disponible para enseñar a enviar y recibir datos a través de una red.

El modelo de referencia OSI permite que los usuarios vean las funciones de red que se producen en cada capa. Más importante aún, el modelo de referencia OSI es un marco que se puede utilizar para comprender cómo viaja la información a través de una red. Además, puede usar el modelo de referencia OSI para visualizar cómo la información o los paquetes de datos viajan desde los programas de aplicación (por ej., hojas de cálculo, documentos, etc.), a través de un entorno de red (por ej., cables, etc.), hasta otro programa de aplicación ubicado en otro computador de la red, aún cuando el remitente y el receptor tengan distintos tipos de red.

En el modelo de referencia OSI, hay siete capas numeradas, cada una de las cuales ilustra una función de red particular. Esta división de las funciones de networking se denomina *división en capas*. La división de la red en siete capas presenta las siguientes ventajas:

- Divide la comunicación de red en partes más pequeñas y sencillas.
- Normaliza los componentes de red para permitir el desarrollo y el soporte de los productos de diferentes fabricantes.
- Permite a los distintos tipos de hardware y software de red comunicarse entre sí.
- Impide que los cambios en una capa puedan afectar las demás capas, de manera que se puedan desarrollar con más rapidez.

Divide la comunicación de red en partes más pequeñas para simplificar el aprendizaje tal como se muestra en la figura 4.3.



Figura 4.3 Modelo de referencia OSI

#### 4.2.5 Las siete capas del modelo de referencia OSI.

El problema de trasladar información entre computadores se divide en siete problemas más pequeños y de tratamiento más simple en el modelo de referencia OSI. Cada uno de los siete problemas más pequeños está representado por su propia capa en el modelo. Las siete capas del modelo de referencia OSI son:

Capa 7: La capa de aplicación

Capa 6: La capa de presentación.

Capa 5: La capa de sesión.

Capa 4: La capa de transporte.

Capa 3: La capa de red.

Capa 2: La capa de enlace de datos.

Capa 1: La capa física

Al estudiar una por una las capas del modelo de referencia OSI, se comprenderá de qué manera los paquetes de datos viajan a través de una red y qué dispositivos operan en cada capa a medida que los paquetes de datos las

atraviesan. Como resultado, comprenderá cómo diagnosticar las fallas cuando se presenten problemas de red especialmente durante el flujo de paquetes de datos.

Cada capa individual del modelo OSI tiene un conjunto de funciones que debe realizar para que los paquetes de datos puedan viajar en la red desde el origen hasta el destino.

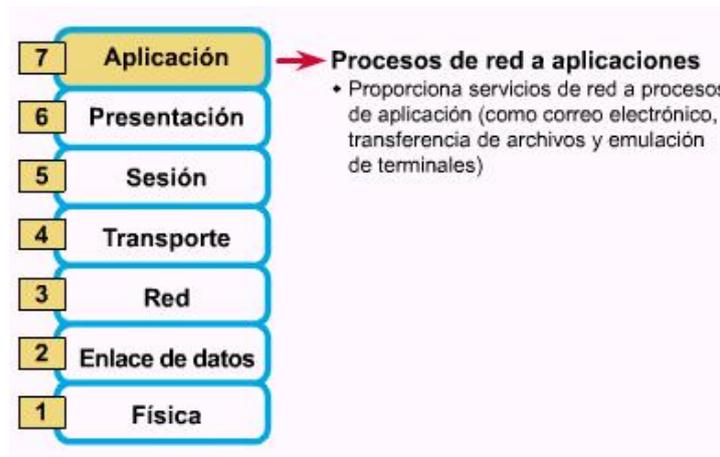


Figura 4.4 Capa de Aplicación

#### 4.2.6 Encapsulamiento.

Las comunicaciones de una red parten de un origen y se envían a un destino, y la información que se envía a través de una red se denomina datos o paquete de datos.

Si un computador (host A) desea enviar datos a otro (host B), en primer término los datos deben empaquetarse a través de un proceso denominado encapsulamiento.

El encapsulamiento rodea los datos con la información de protocolo necesaria antes de que se una al tránsito de la red. Por lo tanto, a medida que los datos se desplazan a través de las capas del modelo OSI, reciben encabezados, información final y otros tipos de información.

### **4.3 EL MODELO DE REFERENCIA TCP/IP.**

Aunque el modelo de referencia OSI sea universalmente reconocido, el estándar abierto de Internet desde el punto de vista histórico y técnico es el *Protocolo de control de transmisión/Protocolo Internet (TCP/IP)*. El modelo de referencia *TCP/IP* y la *pila de protocolo TCP/IP* hacen que sea posible la comunicación entre dos computadores, desde cualquier parte del mundo, a casi la velocidad de la luz. El modelo TCP/IP tiene importancia histórica, al igual que las normas que permitieron el desarrollo de la industria telefónica, de energía eléctrica, el ferrocarril, la televisión y las industrias de vídeos.

#### **4.3.1.Las capas del modelo de referencia TCP/IP.**

El modelo TCP/IP tiene cuatro capas: la capa de aplicación, la capa de transporte, la *capa de Internet* y la capa de red. Es importante observar que algunas de las capas del modelo TCP/IP poseen el mismo nombre que las capas del modelo OSI. No se debe confundir las capas de los dos modelos, porque la capa de aplicación tiene diferentes funciones en cada modelo.

##### **4.3.1.1 Capa de aplicación.**

Los diseñadores de TCP/IP sintieron que los protocolos de nivel superior deberían incluir los detalles de las capas de sesión y presentación. Simplemente crearon una capa de aplicación que maneja protocolos de alto nivel, aspectos de representación, codificación y control de diálogo. El modelo TCP/IP combina todos los aspectos relacionados con las aplicaciones en una sola capa y da por sentado que estos datos están correctamente empaquetados para la siguiente capa.

##### **4.3.1.2 Capa de transporte.**

La capa de transporte se refiere a los aspectos de calidad del servicio con respecto a la confiabilidad, el control de flujo y la corrección de errores. Uno de sus protocolos, el protocolo para el control de la transmisión (TCP), ofrece

maneras flexibles y de alta calidad para crear comunicaciones de red confiables, sin problemas de flujo y con un nivel de error bajo. TCP es un protocolo orientado a la conexión. Mantiene un diálogo entre el origen y el destino mientras empaqueta la información de la capa de aplicación en unidades denominadas segmentos. Orientado a la conexión no significa que el circuito exista entre los computadores que se están comunicando (esto sería una conmutación de circuito). Significa que los segmentos de la Capa 4 viajan de un lado a otro entre dos hosts para comprobar que la conexión exista lógicamente para un determinado período. Esto se conoce como conmutación de paquetes.

#### **4.3.1.3 Capa de Internet.**

El propósito de la *capa de Internet* es enviar paquetes origen desde cualquier red en Internetwork de redes y que estos paquetes lleguen a su destino independientemente de la ruta y de las redes que se utilizaron para llegar hasta allí. El protocolo específico que rige esta capa se denomina Protocolo Internet (IP). En esta capa se produce la determinación de la mejor ruta y la conmutación de paquetes. Esto se puede comparar con el sistema postal. Cuando envía una carta por correo, usted no sabe cómo llega a destino (existen varias rutas posibles); lo que le interesa es que la carta llegue.

#### **4.3.1.4 Capa de red.**

El nombre de esta capa es muy amplio y se presta a confusión. También se denomina capa de host a red. Es la capa que se ocupa de todos los aspectos que requiere un paquete IP para realizar realmente un enlace físico y luego realizar otro enlace físico. Esta capa incluye los detalles de tecnología de LAN y WAN y todos los detalles de las capas: física y de enlace de datos del modelo OSI.



Figura 4.5 Protocolo TCP/IP.

El diagrama que aparece en la figura 4.5 se denomina *gráfico de protocolo*. Este gráfico ilustra algunos de los protocolos comunes especificados por el modelo de referencia TCP/IP. En la capa de aplicación, aparecen distintas tareas de red que probablemente usted no reconozca, pero como usuario de Internet, probablemente use todos los días. Estas aplicaciones incluyen las siguientes:

- *FTP*: File Transfer Protocol (Protocolo de transporte de archivos)
- *HTTP*: Hypertext Transfer protocol (Protocolo de transferencia de hipertexto)
- *SMTP*: Simple Mail Transport Protocol (Protocolo de transporte de correo simple)
- *DNS*: Domain Name Service (Servicio de nombre de dominio)
- *TFTP*: Trivial File transport protocol (Protocolo de transporte de archivo trivial)

El modelo TCP/IP enfatiza la máxima flexibilidad, en la capa de aplicación, para los diseñadores de software. La capa de transporte involucra dos protocolos: el protocolo de control de transmisión (TCP) y el *protocolo de datagrama (UDP)*. La capa inferior, la capa de red, se relaciona con la tecnología LAN o WAN que se utiliza en particular.

En el modelo TCP/IP existe solamente un protocolo de red: el protocolo Internet, o IP, independientemente de la aplicación que solicita servicios de red o del protocolo de transporte que se utiliza. Esta es una decisión de diseño deliberada. *IP* sirve como protocolo universal que permite que cualquier computador en

cualquier parte del mundo pueda comunicarse en cualquier momento. La figura 4.6 muestra los servicios que puede presentar TCP/IP.

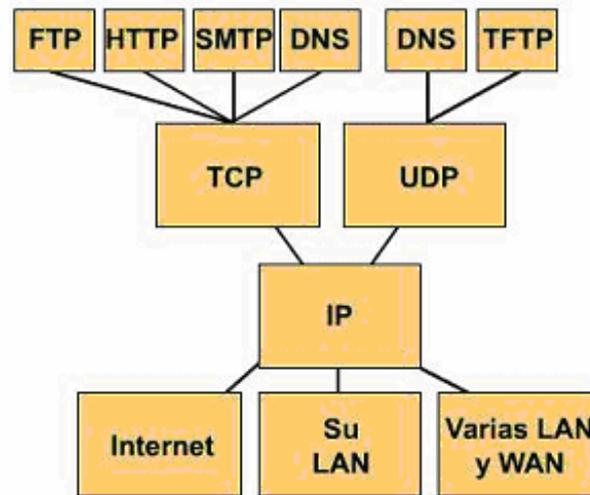


Figura 4.6 Servicios de TCP/IP.

Si compara el modelo OSI y el modelo TCP/IP (figura 4.7), observará que ambos presentan semejanzas y diferencias.

Los protocolos TCP/IP son los estándares en torno a los cuales se desarrolló Internet, de modo que la credibilidad del modelo TCP/IP se debe en gran parte a sus protocolos. En comparación, no se crean redes a partir de protocolos específicos relacionados con OSI, aunque todo el mundo utiliza el modelo OSI como guía.

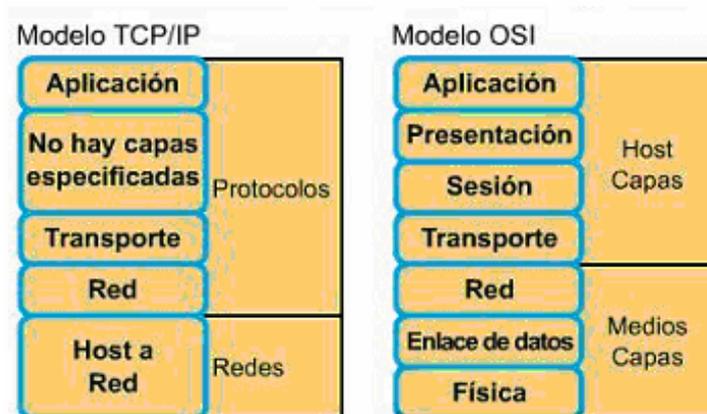


Figura 4.7 Cuadro comparativo entre los protocolos

## 4.4 DISPOSITIVOS DE LAN BÁSICOS.

### 4.4.1 Topologías Básicas.

La *topología* define la estructura de una red. La definición de topología está compuesta por dos partes, la topología física, que es la disposición real de los cables (los medios) y la topología lógica, que define la forma en que los hosts acceden a los medios.

Las topologías físicas que se utilizan comúnmente son de bus, de anillo, en estrella, en estrella extendida, jerárquica y en malla. Estas topologías se indican en la figura 4.8

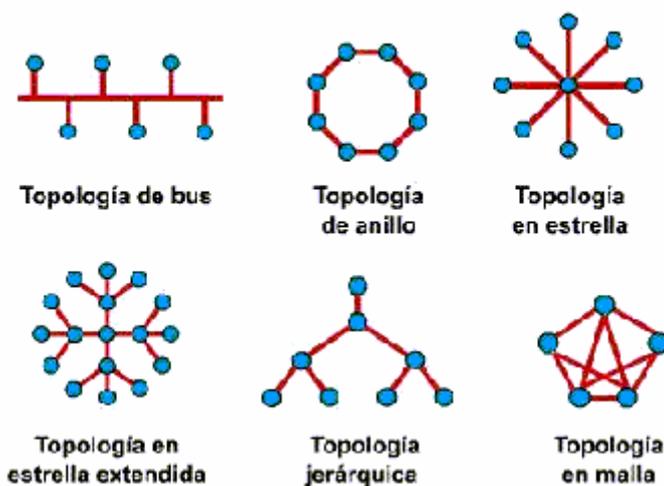


Figura 4.8 Topologías básicas de red.

- La topología de bus utiliza un único segmento backbone (longitud del cable) al que todos los hosts se conectan de forma directa.
- La topología de anillo conecta un host con el siguiente y al último host con el primero. Esto crea un anillo físico de cable.
- La topología en estrella conecta todos los cables con un punto central de concentración. Por lo general, este punto es un hub o un switch, que se describirán más adelante en este capítulo.

- La topología en estrella extendida se desarrolla a partir de la topología en estrella. Esta topología enlaza estrellas individuales enlazando los hubs/switches. Esto, como se describe más adelante en este capítulo, permite extender la longitud y el tamaño de la red.
- La topología jerárquica se desarrolla de forma similar a la topología en estrella extendida pero, en lugar de enlazar los hubs/switches, el sistema se enlaza con un computador que controla el tráfico de la topología.
- La topología en malla se utiliza cuando no puede existir absolutamente ninguna interrupción en las comunicaciones, por ejemplo, en los sistemas de control de una central nuclear. De modo que, como puede observar en el gráfico, cada host tiene sus propias conexiones con los demás hosts. Esto también se refleja en el diseño de la Internet, que tiene múltiples rutas hacia cualquier ubicación.

La topología lógica de una red es la forma en que los hosts se comunican a través del medio. Los dos tipos más comunes de topologías lógicas son broadcast y transmisión de tokens.

La topología de broadcast simplemente significa que cada host envía sus datos hacia todos los demás hosts del medio de red. Las estaciones no siguen ningún orden para utilizar la red, el orden es el primero que entra, el primero que se sirve. El segundo tipo es transmisión de tokens. La transmisión de tokens controla el acceso a la red al transmitir un token electrónico de forma secuencial a cada host. Cuando un host recibe el token, eso significa que el host puede enviar datos a través de la red. Si el host no tiene ningún dato para enviar, transmite el token hacia el siguiente host y el proceso se vuelve a repetir.

En el diagrama de la figura 4.9 se pueden observar varias topologías. Este diagrama muestra una LAN de Networking moderada que es típica de una escuela o de una pequeña empresa. Tiene muchos símbolos, y describe varios conceptos de Networking que lleva cierto tiempo aprender. Esta LAN es típica de un campus pequeño, y de la mayoría de los dispositivos que estudiará para el CCNA

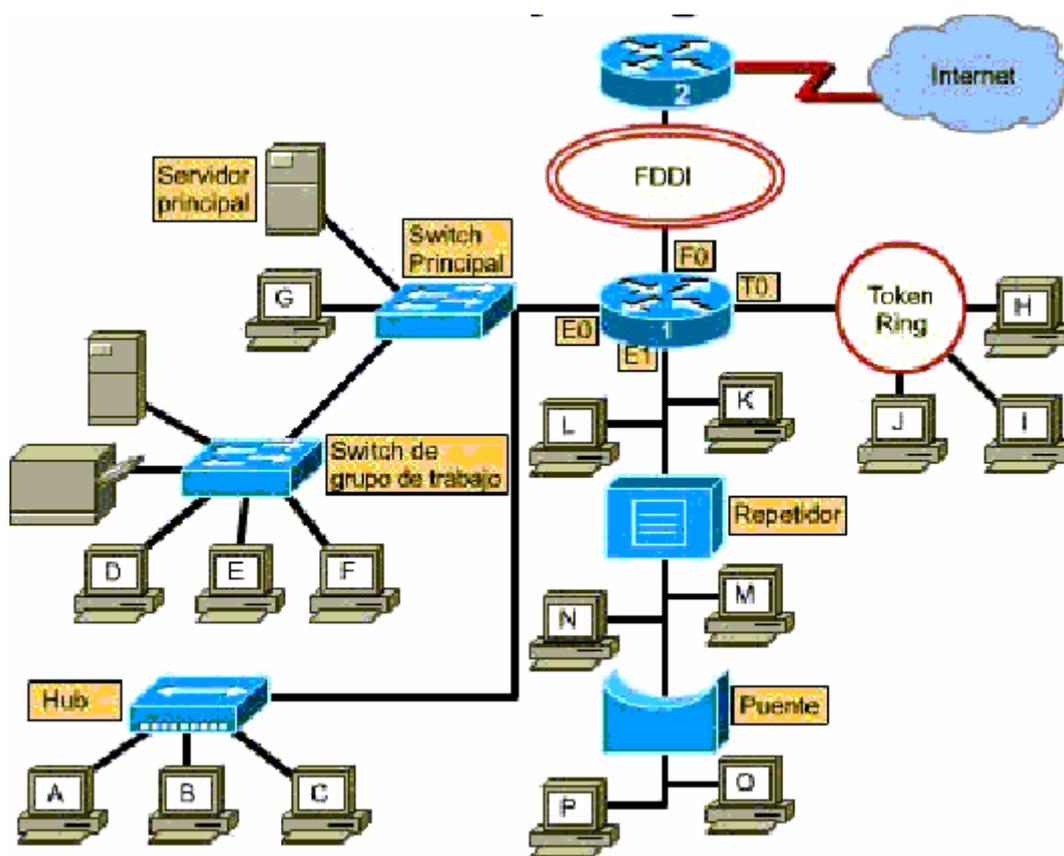


Figura 4.9 Topologías de red

## 4.4.2 Conceptos básicos del flujo de datos a través de las LAN.

### 4.4.2.1 Encapsulamiento de los paquetes.

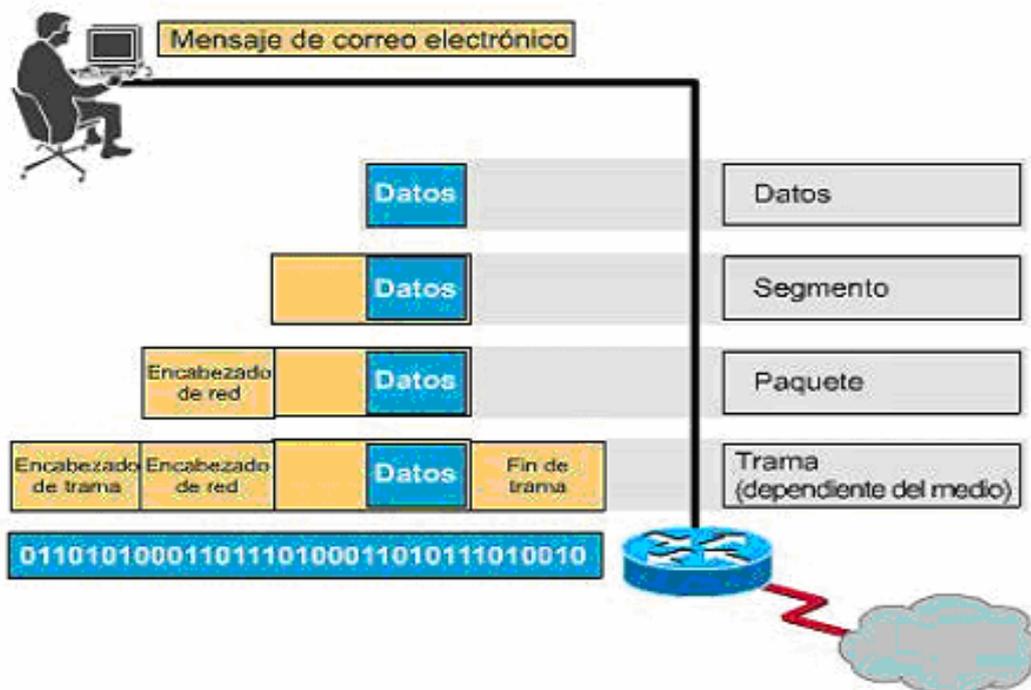
Para que se puedan producir comunicaciones confiables a través de una red, los datos que se deben enviar se deben colocar en paquetes que se puedan administrar y rastrear. Esto se realiza a través del proceso de encapsulamiento. Un breve repaso del proceso indica que las tres capas superiores (aplicación, presentación y sesión) preparan los datos para su transmisión creando un formato común para la transmisión (figura 4.10).

La capa de transporte divide los datos en unidades de un tamaño que se pueda administrar, denominadas segmentos. También asigna números de secuencia a los segmentos para asegurarse de que los hosts receptores vuelvan a unir los datos en el orden correcto. Luego la capa de red encapsula el segmento creando

un paquete. Le agrega al paquete una dirección de red destino y origen, por lo general IP.

En la capa de enlace de datos continúa el encapsulamiento del paquete, con la creación de una trama. Le agrega a la trama la dirección local (MAC) origen y destino. Luego, la capa de enlace de datos transmite los bits binarios de la trama a través de los medios de la capa física.

Cuando los datos se transmiten simplemente en una red de área local, se habla de las unidades de datos en términos de tramas, debido a que la dirección MAC es todo lo que se necesita para llegar desde el host origen hasta el host destino. Pero si se deben enviar los datos a otro host a través de una red interna o Internet, los paquetes se transforman en la unidad de datos a la que se hace referencia. Esto se debe a que la dirección de red del paquete contiene la dirección destino final del host al que se envían los datos (el paquete) .



**Figura 4.10 Encapsulamiento de datos**

Las tres capas inferiores (red, enlace de datos, física) del modelo OSI son las capas principales de transporte de los datos a través de una red interna o de

Internet. La excepción principal a esto es un dispositivo denominado gateway. Este es un dispositivo que ha sido diseñado para convertir los datos desde un formato, creado por las capas de aplicación, presentación y sesión, en otro formato. De modo que el gateway utiliza las siete capas del modelo OSI para hacer esto.

#### 4.5 REDES DE CAMPO

En esta sección se introduce el concepto de las redes de campo industriales o, como también se las denomina, el Bus I/O (Entrada / Salida). Existen dos tipos de buses I/O; el bus de dispositivos y el bus de procesos. Las redes industriales o buses I/O permite a un PLC comunicarse con dispositivos de entrada y salida de manera similar a como en una red de área local, PCs o PLCs supervisores se comunican con PLCs individuales como lo indica la Figura 4.11.

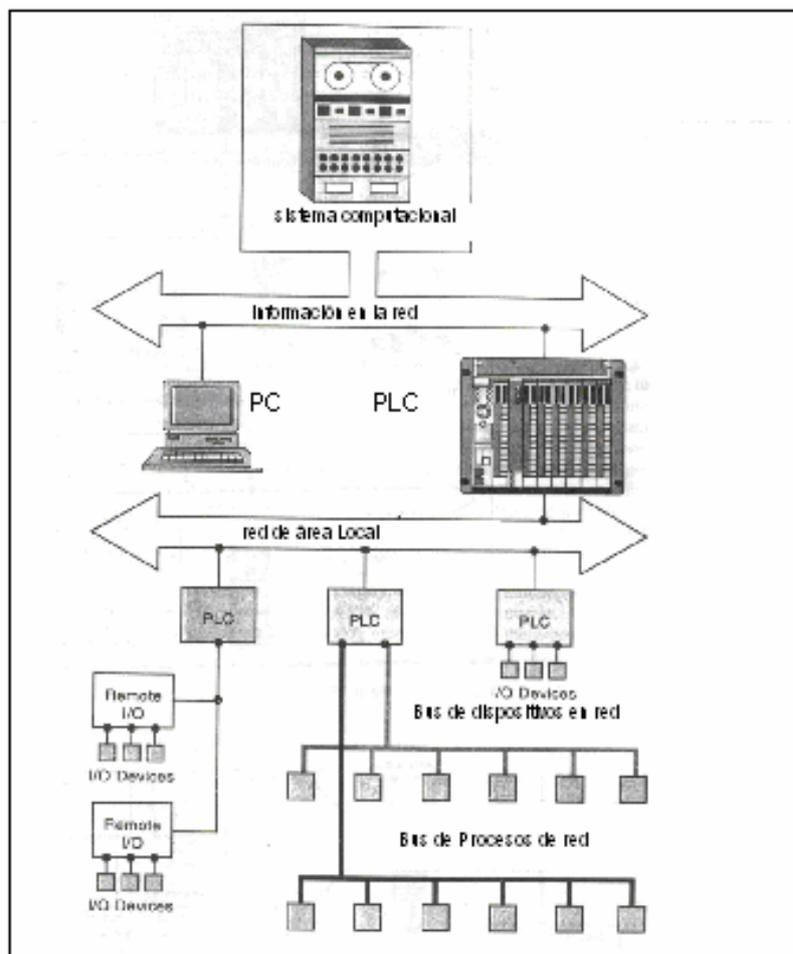
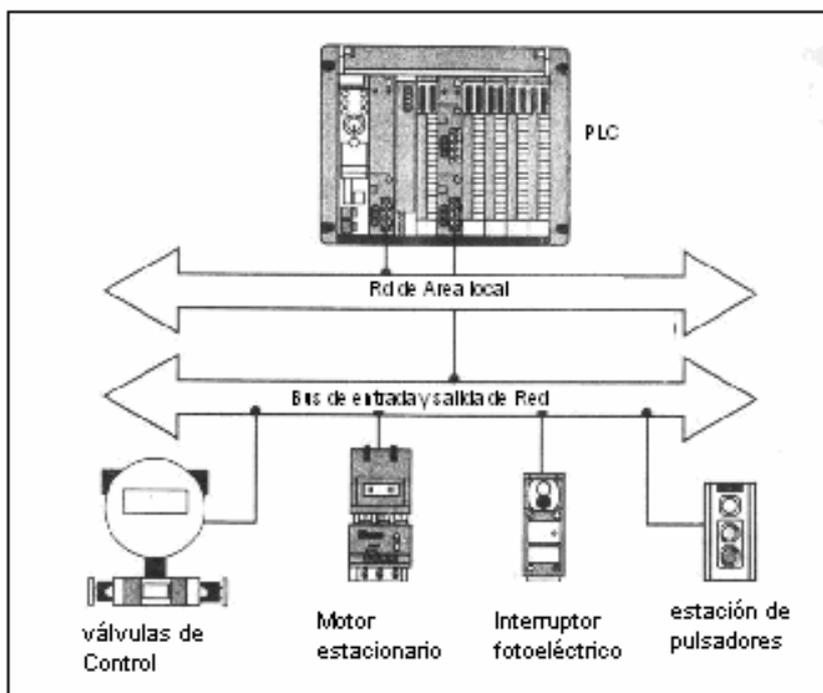


Figura 4.11 Bus I/O o Red Industrial de campo

Esta configuración descentraliza el control en un sistema para que los PLCs se encarguen de controlar partes de un todo, dando como resultado un sistema de control más grande y rápido. Rápido pues la información tenderá a viajar solamente hasta / desde un PLC. Poco tráfico será el que tenga que recorrer muchos caminos en la red con lo que se optimiza el flujo de datos en la misma. La topología o arquitectura física de un bus I/O sigue la configuración de un bus o bus extendido, esto permite que dispositivos inteligente (interruptores de fin de carrera, fotoeléctricos y de proximidad) se conecten directamente sea a otro PLC o a una LAN. En estos buses se debe hacer notar una diferencia: el bus, a más de tener líneas para transmitir datos, puede también tener líneas de alimentación. La Figura 4.12 ilustra una conexión típica entre un PLC, una red de área local y un bus I/O.



**Figura 4.12 Conexiones típicas**

A un bus I/O se debe conectar dispositivos inteligentes donde lo de inteligente se puede traducir como un dispositivo que tiene los circuitos y la lógica para compartir el medio físico con otros dispositivos.

En un bus I/O, el PLC maneja los dispositivos inteligentes directamente sin el uso de módulos de entrada y salida; lo que quiere decir que el PLC se conecta y se comunica con los dispositivos inteligentes de entrada y salida de acuerdo a un

protocolo de bus. Para esto el PLC usa un I/O bus Network scanner. Un bus I/O scanner lee y escribe a la dirección de cada uno de los dispositivos inteligentes, al mismo tiempo que lee la información contenida en los paquetes de la red. Una red grande con bus de topología de árbol puede tener hasta 2048 dispositivos de campo discretos o más conectados.

#### **4.5.1 TIPOS DE BUS I/O EN REDES.**

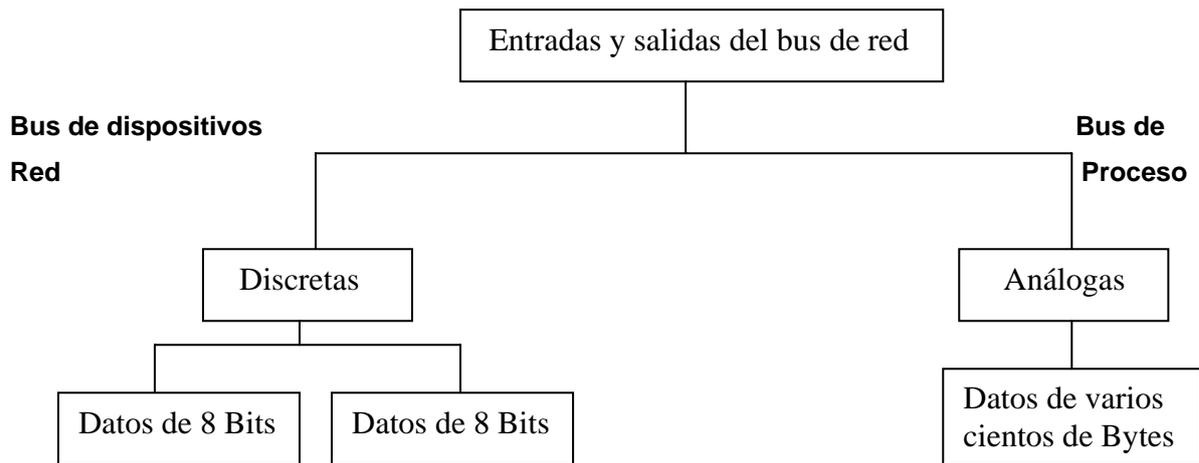
Los buses I/O de redes pueden dividirse en dos diferentes categorías: una que tiene que ver con dispositivos de bajo nivel que son típicos de operación de manufactura discretas y los otros son dispositivos de alto nivel utilizados en procesos industriales.

Las categorías de los buses de red son:

- Buses de dispositivos.
- Buses de proceso.

**4.5.1.1 Bus de Dispositivos.-** Es la interfaz con los dispositivos de campo de bajo nivel (pulsadores, interruptores de fin de carrera, etc), cuyo fin es proporcionar información respecto al estado de los dispositivos (ON/OFF) o al estado de operación (operación correcta / incorrecta). Estas redes generalmente transmiten solo desde unos pocos bits hasta varios bytes de datos en un determinado tiempo.

**4.5.1.2 Bus de Proceso.-** Por otro lado, estos buses se conectan a dispositivos de campo capaces de generar un alto nivel de información (válvulas de proceso inteligentes, medidores de nivel inteligentes, etc), que típicamente se emplean en aplicaciones de control de procesos en donde se requiere un control más “fino” de sus variables. El bus de procesos maneja grandes lotes de datos (varias centenas de bytes), dando información acerca del proceso, así como de los mismos dispositivos de campo (marca del equipo, fecha de último mantenimiento, etc). La figura 4.13 ilustra la clasificación de los dos tipos de buses I/O.



**Figura 4.13 Clasificación de los buses tipo I/O**

La mayoría de dispositivos conectados a un bus de procesos son analógicos, cuya información por supuesto debe ser convertida a digital para su transmisión; los dispositivos conectados a un bus de dispositivos son discretos. Sin embargo, el bus de dispositivos algunas veces incluye dispositivos analógicos como termocuplas y controladores de velocidad, que transmiten unos pocos bytes de información.

El bus de dispositivos que incluye dispositivos discretos, tanto como pequeños dispositivos analógicos, son llamados “byte-wide bus networks” que podría traducirse como redes con bus para transmitir bytes. Estas redes pueden transferir entre 1 a 50 o más bytes de datos en un determinado momento. El bus de dispositivos que se comunica solo con dispositivos discretos es llamado bit-wide bus networks, este transfiere menos de 8 bits de datos desde dispositivos discretos simples a distancias relativamente cortas.

La razón principal para que el bus de dispositivos trabaje con dispositivos discretos y el bus de procesos con analógicos es el requerimiento diferente de transmisión de datos. El tamaño de los paquetes de información tiene un efecto inverso en la velocidad con que estos datos viajan en la red. Puesto que los buses de dispositivos transmiten poca información pueden trabajar a gran velocidad y por lo mismo deberían emplearse en aplicaciones donde se requiere de rapidez de accionamiento de ciertos dispositivos, caso de una electro válvula por ejemplo. El bus de procesos trabaja lentamente por el gran tamaño de los paquetes de datos

y, por lo mismo, se los aplica más para el control de dispositivos de entrada/salida analógicas que no requieren una respuesta rápida de tiempo, por ejemplo una servo válvula.

Las velocidades de transmisión para ambos tipos de buses I/O pueden ser de hasta 1 y 2.5 Mbytes por segundo. Sin embargo un bus de dispositivos puede entregar más paquetes de información que el bus de procesos.

Puesto que por un bus de proceso se puede transmitir varios cientos de bytes de datos, son apropiados para aplicaciones que requieren una transmisión de datos complejos. Por ejemplo, un bus de procesos compatible con un transmisor de presión puede proveer de control con mucha mayor información que solo presión, está puede también transmitir información acerca del flujo de temperatura flotante y así como de su operación interna. Este tipo de información requiere un paquete largo de datos a transmitir y para esto es apropiado el bus de proceso pues en el de dispositivos simplemente no cabría.

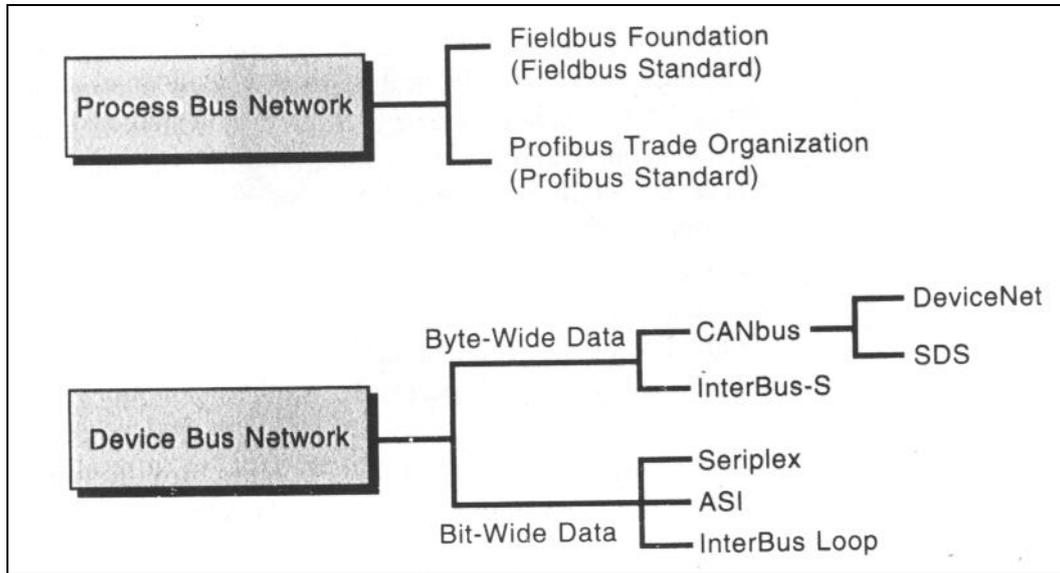
#### **4.6 PROTOCOLOS DE LOS BUSES Y REDES DE CAMPO INDUSTRIALES**

Ninguno de los dos buses I/O tienen un protocolo estándar; sin embargo algunas organizaciones están trabajando para desarrollar tanto especificaciones para los buses de dispositivos como de proceso.

En el área del bus de proceso dos organizaciones principales, la Fundación Fieldbus (que es el resultado de la unión de la Fundación ISP y FIP) y la Organización de Comercio Profibus (bus de proceso inteligente) están trabajando para producir estándares para protocolos y redes industriales.

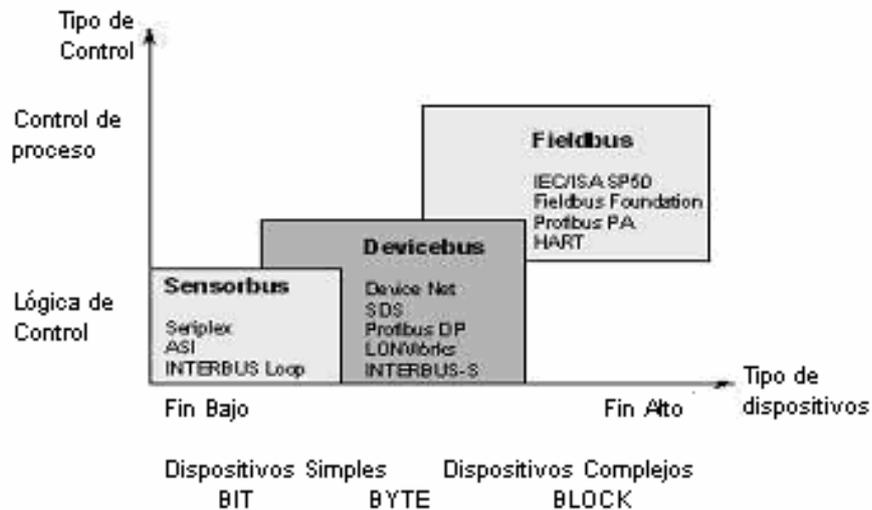
Otras organizaciones como la ISA y el IEC están también participando en el desarrollo de estos estándares. Esta es la razón por la que algunos fabricantes especifican que sus productos analógicos son compatibles con Profibus, Fieldbus u otro tipo de esquema de protocolo de comunicación.

La Figura 4.14 ilustra un diagrama de bloques de los protocolos disponibles.



**Figura 4.14 Estándares de protocolos**

Otra manera de intentar una clasificación de los buses de campo se muestra en la Figura 4.15 a continuación.



**Figura 4.15 Otra clasificación de las redes o buses de campo industriales.**

Tanto los buses de dispositivos como los de proceso transmiten su información de la misma manera; esto es, digitalmente. De hecho, la necesidad de la

comunicación digital fue uno de las mayores razones para el establecimiento de las redes industriales. La comunicación digital es la que permite que más de un dispositivo pueda conectarse a un mismo medio y compartirlo debido a la capacidad de direccionamiento que ofrecen y a que los nodos involucrados sean capaces de reconocer los datos. Por otro lado, los datos digitales son menos propensos a sufrir de distorsión producto de la EMI o RFI. También, los PLCs pueden transmitir datos digitales sin tener que convertirlos a análogos y viceversa. De esta manera hay menos errores asociados a la conversión A/D y D/A. Por último otra de las grandes ventajas tiene que ver con la enorme reducción de cableado al no tener que conectar los dispositivos de campo con cables de conexión individuales.

#### **4.7 REDES CON BUSES DE DISPOSITIVOS BYTE-WIDE**

Los buses de dispositivos byte-wide más comunes están basados en las redes InterBus S y CANbus.

##### **4.7.1 InterBus-S**

Es una red de sensores y actuadores que conecta estos dispositivos de campo a un PLC o a una computadora (Soft PLC) en una configuración tipo anillo. Tiene incorporadas interfaces I/O en sus 256 nodos posibles, que también incluyen bloques terminales de conexión que posibilitan y facilitan la conexión a más dispositivos I/O. Gracias a estos bloques terminales es posible conectar hasta 4096 dispositivos de campo I/O (dependiendo de la configuración) a una velocidad de 500 Kbps con detección de error tipo CRC.

Las direcciones de los dispositivos en la red InterBus-S son automáticamente determinados por su ubicación física y así se elimina la necesidad de asignar manualmente las direcciones. El dispositivo controlador continuamente barre los dispositivos de I/O leyendo todas las entradas en un barrido y escribiendo datos en las salidas. InterBus-S trabaja en las capas Física, Enlace de Datos y Aplicación del modelo OSI.

## **4.8 REDES CON BUSES DE DISPOSITIVOS BIT-WIDE**

Se usan con simples dispositivos de campo discretos; es decir, sensores y actuadores tipo ON-OFF. Pueden transmitir tan solo 4 bits (un nibble) que es suficiente para transmitir los datos desde estos dispositivos. Los dispositivos más pequeños requieren de solo un bit para operar. Minimizando la cantidad de datos transmitidos, estos buses pueden proveer un rendimiento óptimo a bajos costos. Los buses más comunes son: ASI, InterBus Loop y Seriplex.

### **4.8.1 Red ASI**

Se usan en redes que no requieren más de 124 dispositivos de Entrada / Salida (I/O). Estos 124 dispositivos de entrada y salida pueden conectarse hasta 31 nodos sea en una topología de anillo, árbol o estrella. Los dispositivos I/O se conectan al PLC o a una PC por medio de una interfaz controlada (una tarjeta que se conecta al bus). con simples dispositivos

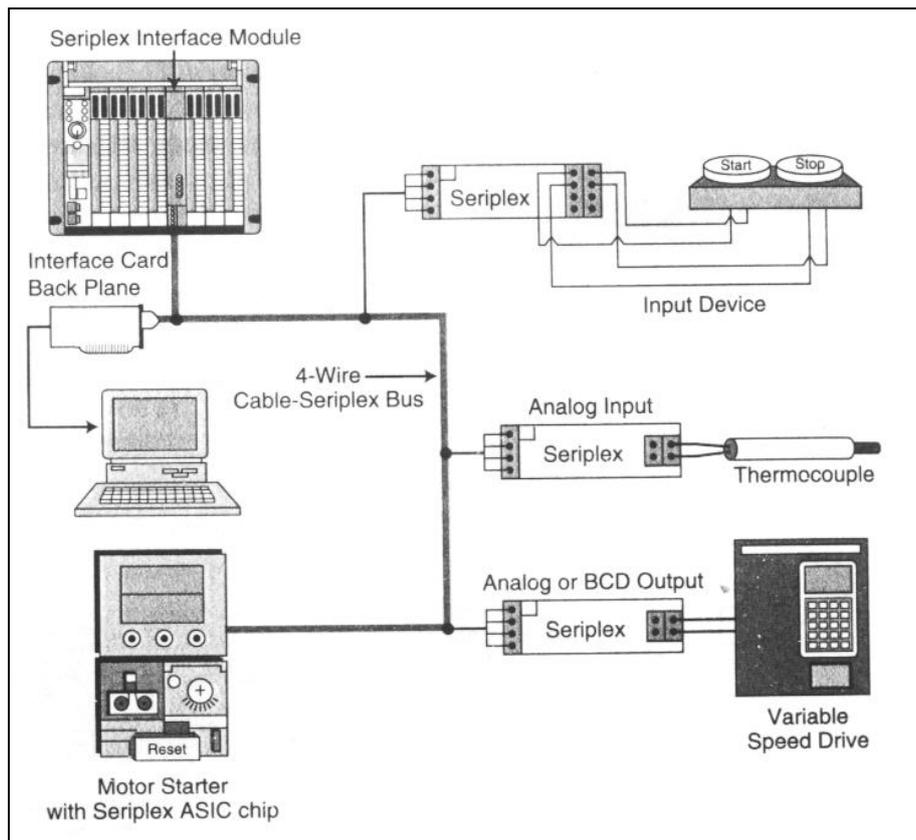
### **4.8.2 InterBus Loop**

La red bit-wide InterBus loop fue desarrollada por la compañía Phoenix Contact Inc. y es empleada para conectar a un PLC dispositivos sensores y actuadores simples. InterBus Loop emplea una tecnología de alimentación y comunicación denominada PowerCom para enviar la señal con el protocolo InterBus por las líneas de alimentación; es decir, el protocolo es modulado en las líneas de poder. Esto permite reducir el número de cables requerido por la red a solamente dos conductores que llevan tanto la alimentación como las señales de comunicación a los dispositivos de campo.

### **4.8.3 Seriplex**

La red Seriplex permite la conexión de hasta 510 dispositivos de campo a un PLC en una configuración Maestro / Esclavo o peer to peer. La red Seriplex se basa en el chip ASIC (Application Specific Integrated Circuit) el cual debe estar presente en todos los dispositivos que se conectan en esta red. Los dispositivos que no

tiene este chip puede conectarse a la red por medio de un módulo Seriplex I/O que contiene un chip ASIC esclavo. La interfaz ASCII I/O contiene 32 funciones lógicas booleanas que sirven para proveer la lógica necesaria para comunicar, direccionar e inteligencia necesaria para controlar los dispositivos de campo conectados a la red (Figura 4.16).



**Figura 4.16 Red Seriplex con módulo controlador**

La red Seriplex a nivel de capa física puede extenderse hasta 2000m con una topología estrella, bus o árbol. Esta red también puede operar sin un controlador. A diferencia de la red ASI, la red Seriplex se puede conectar con dispositivos análogos de entrada y salida; sin embargo, la señal analógica digitalizada se debe leer o escribir bit por bit en cada ciclo de barrido. La figura 4.17 muestra una red Seriplex típica sin controlador.

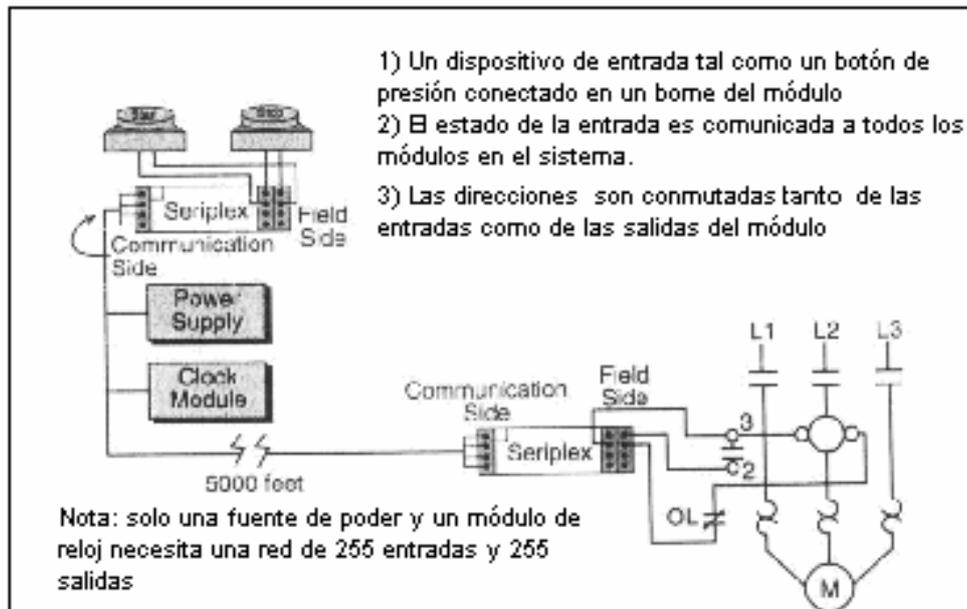
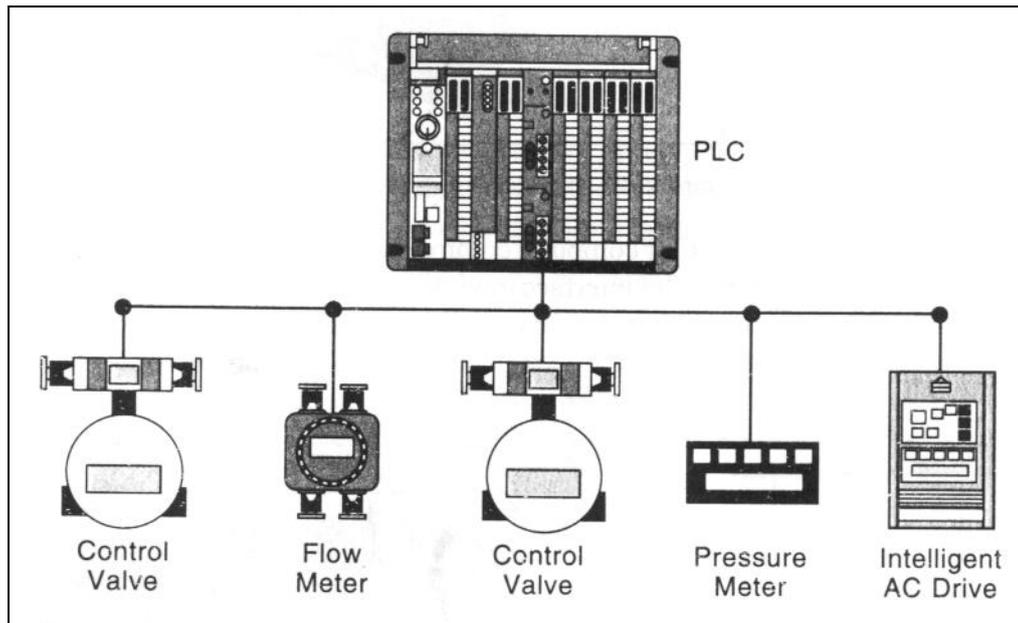


Figura 4.17 Interfaz Seriplex I/O sin un controlador

#### 4.9 REDES CON BUSES DE PROCESO

Una red con buses de proceso es una red digital de comunicaciones, abierta y de alto nivel que se emplea para conectar dispositivos de campo análogos a un sistema de control. Como se mencionó antes, una red de proceso se emplea en aplicaciones donde los sensores o actuadores análogos de entrada / salida responden más lentamente que aquellos en aplicaciones con dispositivos discretos (redes de buses de dispositivos). El tamaño de los paquetes de información de / desde estos dispositivos de campo análogos es grande, debido a la naturaleza de la información que se recoge.

Los protocolos que se emplean en estas redes de proceso transmiten a velocidades de 1 a 2 Mbps, aún así se dice que su velocidad va de lenta a mediana debido a la gran cantidad de datos que deben transferir. De todas maneras, esta velocidad es adecuada debido a que los dispositivos análogos no responden instantáneamente como los discretos. En la figura 4.18 se muestra una red de bus de proceso típica.



**Figura 4.18 Configuración típica de una red de bus de proceso**

Las redes de proceso pueden transmitir una gran cantidad de información a un PLC de ahí que pueden mejorar notablemente la operación de una planta o proceso. Por ejemplo, un arranque de motor inteligente, compatible con una red de proceso, puede proveer información acerca de la cantidad de corriente consumida por el mismo, tal que, si la cantidad de corriente sube, o hay algún síntoma de problema, el sistema puede alertar al operador y evitar una falla potencial del motor en una línea de producción crítica.

Se cree que las redes de proceso eventualmente reemplazarán a todas las redes análogas 4-20 mA. Estas proveen mayor exactitud y repetibilidad, así como añaden comunicación bidireccional entre los dispositivos de campo y los controladores (como los PLCs).

Un PLC o computadora se comunica con una red de proceso por medio de módulo o tarjeta interfaz controladora que pueden emplear sea el formato del protocolo MODBUS, Fieldbus o Profibus. El procesador del bus de proceso generalmente se inserta dentro del gabinete del PLC.

### 4.9.1 MODBUS

Los controladores programables de Modicon pueden comunicarse entre sí y con otros dispositivos sobre una variedad de redes, dentro de estas se incluyen las redes industriales Modicon Modbus y Modbus Plus, y las redes estándar MAP y Ethernet. Las redes son accedidas por medio de puertos incorporados en los controladores o por adaptadores de red, módulos opcionales, o gateways disponibles desde Modicon.

El idioma común empleado por todos los controladores de Modicon es el protocolo Modbus. Este protocolo define una estructura de mensaje que los controladores reconocen y usan, sin importar el tipo de redes sobre el que se comunican. Describe el proceso que un controlador empleará para solicitar acceso a otro dispositivo, cómo responderá a las demandas de los otros dispositivos, y cómo se detectarán y reportarán los errores. Establece un formato común para el diseño y contenidos de los campos de un mensaje.

El protocolo Modbus proporciona el estándar interno que los controladores Modicon usan para el análisis de los mensajes. Durante la comunicación sobre una red Modbus, el protocolo determina cómo cada controlador conocerá la dirección de un dispositivo, como reconocerá que un mensaje es dirigido a él, como determinará el tipo de acción a ser ejecutada, y como extraerá la información o cualquier dato contenido en el mensaje. Si se requiere una respuesta, el controlador construirá el mensaje de respuesta y lo enviará usando el protocolo Modbus.

Sobre otras redes, los mensajes del protocolo Modbus se integran en la trama o estructura del paquete utilizada sobre esa red. Por ejemplo, para los controladores de red Modicon para Modbus Plus o MAP. Con software de aplicación asociado – drivers y librerías - se proporciona la conversión entre el mensaje de protocolo Modbus y las tramas específicas de los protocolos que esas redes utilizan para comunicar entre sus dispositivos nodo.

## **4.9.2 MODBUS PLUS**

### **4.9.2.1 Aplicaciones de Modbus Plus**

Modbus Plus es un sistema de red de área local diseñado para aplicaciones de mando industriales. Cada red soporta 64 dispositivos de nodo direccionables, a una tasa de transferencia de datos de 1,000,000 bps. Las aplicaciones incluyen supervisión de mando de un proceso y los mensajes de supervisión.

Modbus Plus mantiene el protocolo de comunicación par-a-par en los diferentes niveles de la red. Modicon proporciona una variedad de controladores programables y adaptadores de red. La red también permite módulos de comunicación I/O distribuidos (DIO) en los que los controladores Modicon se comunican directamente con los subsistemas I/O.

Cada controlador Modicon acepta que una red de computadoras se conecte directamente a su puerto Modbus Plus. Agregando módulos opcionales de red, pueden configurarse redes adicionales para extender las comunicaciones de I/O en la aplicación del usuario.

El bus de la red usa un solo-cable o un cable dual. El diseño de cable dual aumenta la protección contra las fallas del cable o el ruido que puede aparecer en cualquier corrida del cable, permitiendo que ante la presencia de un problema la comunicación continúe por el camino libre de errores.

A 32 dispositivos nodo puede conectarse directamente el bus de red la cual puede cubrir una longitud de 1500 pies (450 metros). Es posible emplear repetidores para extender la distancia del cable a su máximo de 6000 pies (1800 metros), y el número de nodos a su máximo de 64.

### **4.9.2.2 Comunicación entre Redes**

Es posible emplear un puente (bridge) para unir redes que estén jerárquicamente en el mismo o diferente nivel. El puente permite el enrutamiento de los mensajes originados en el nodo de una red a través de uno o mas puentes a un nodo

destino en otra red. Ésto optimiza el tráfico en las redes pues solamente el tráfico necesario circulará por cada red lo cual es útil para aplicaciones críticas en cuanto al tiempo en que cada red emplea sólo los dispositivos requerido por cronometrar del proceso local. Solamente los mensajes que se requieren pasarán por los puentes para llegar a otras redes.

Modbus y Modbus Plus pueden emplear Puentes Multiplexores. El Multiplexor del Puente proporciona cuatro puertos seriales que se pueden configurar separadamente para protocolo Modbus o como dispositivos RS232 / RS485.

### 4.9.3 FIELDBUS

Al igual que Modbus y Profibus, Fieldbus busca conectar las redes de campo y las administrativo-financieras de una forma jerárquica, tal como se ilustra en la figura 4.19.

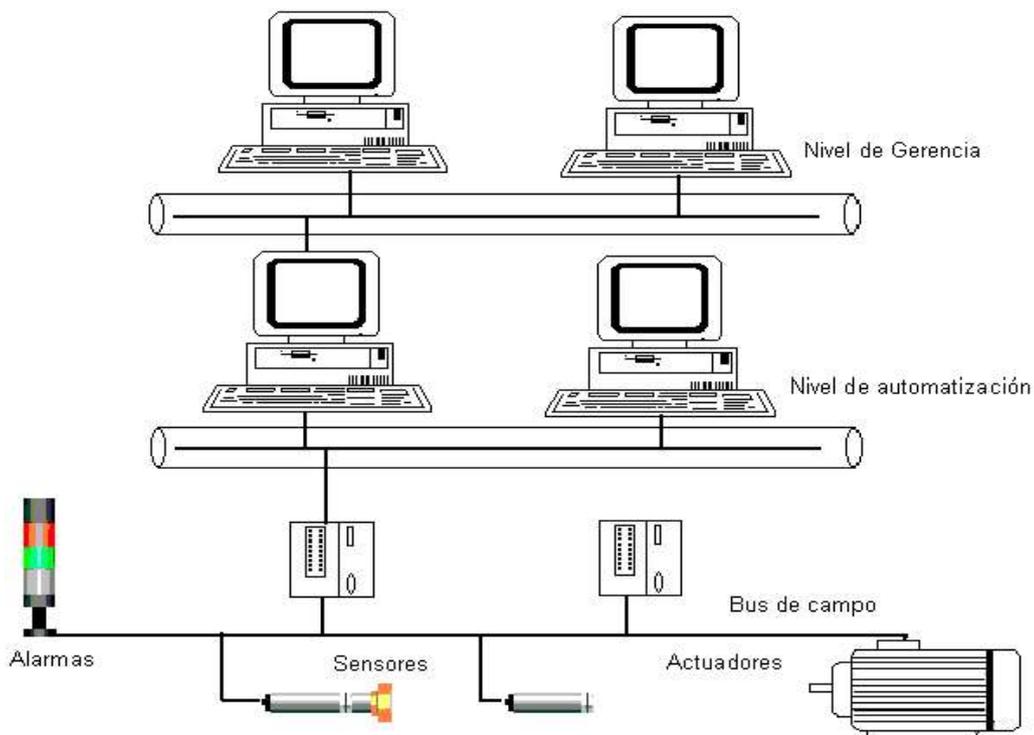


Figura 4.19. Red administrativa y redes de campo.

Se ha dicho que cada protocolo tiene sus características de funcionamiento propias. Al unir redes con diferentes protocolos, como se muestra en la figura 4.40 surgen incompatibilidades. En muchos casos no se trata de protocolos que compitan entre sí, sino que funcionan bajo reglas diferentes y hay que procurar que se complementen, particularmente en una arquitectura de comunicación de varios niveles tal como se muestra arriba.

#### **4.9.4 FOUNDATION FIELDBUS**

Este es el nombre que se da a la organización que da soporte a este protocolo para lograr su difusión a nivel mundial. La red de bus de proceso Fieldbus es un sistema de comunicación de dos vías, serial, digital y multipuerto que conecta equipo de campo, como sensores y actuadores inteligentes, con controladores, como PLCs.

Con Fieldbus (figura 4.20) no se trata simplemente de reemplazar las redes análogas de corriente de 4-20 mA por una red digital. Por medio de la comunicación digital se permite enviar no sólo una señal de medición o control, sino también anexar importante información de diagnóstico, que facilita el trabajo de mantenimiento y da mayor seguridad al sistema, así como alarmas, parámetros de configuración, etc. Además, siendo la comunicación bi-direccional se facilitan las labores de configuración y calibración.

El protocolo está desarrollado a partir del modelo de comunicaciones OSI.

Es un protocolo para redes industriales pensado específicamente para aplicaciones de control distribuido. Puede comunicar grandes volúmenes de información, ideal para aplicaciones con varios lazos complejos para control de procesos y automatización de una fábrica. Provee bloques de función: IA, ID, OA, OD, PID, que pueden intercambiarse entre la estación maestra y los dispositivos de campo. La longitud máxima por mensaje es de 256 bytes, lo que permite transferir funciones de control con el concepto de objetos.

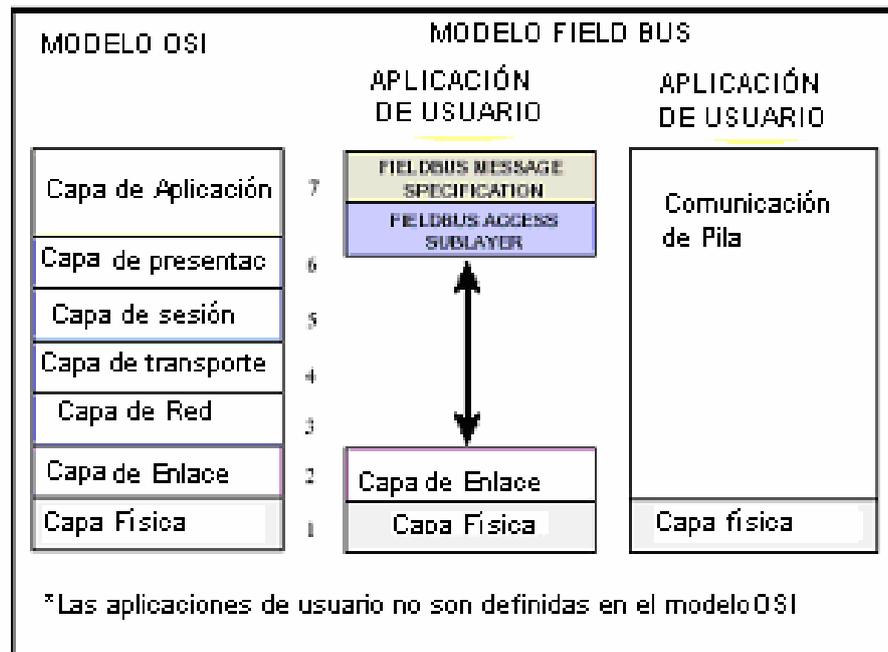


Figura 4.20. Protocolo Fieldbus y su comparación con el modelo OSI

Fundamentalmente consta de tres niveles:

- a) Capa física,
- b) Capa o pila (Stack) de comunicaciones, y
- c) Capa de usuario.

En este protocolo no se implementan los niveles 3, 4, 5 y 6 del modelo OSI, pues estos no se requieren en aplicaciones de control de procesos, pero si se tiene en cuenta dentro del stack de comunicaciones a la capa de aplicación que tiene que comunicarse con el Nivel de Usuario. La capa de usuario no pertenece al modelo OSI y en el protocolo Fieldbus provee varias funciones clave como son: bloques de funciones, servicios de descripción de dispositivos y administración del sistema.

#### 4.9.5 PROFIBUS

Existen tres variantes principales de PROFIBUS de acuerdo a las características de la aplicación:

Automatización de fábricas	de	Automatización para propósitos generales	para	Automatización de procesos	de
<b>PROFIBUS DP</b>		<b>PROFIBUS FMS</b>		<b>PROFIBUS PA</b>	
(DIN 19245 T1 + T3)		(DIN 19245 T1 + T2)		(DIN 19245 T4)	
pr EN 50170		pr EN 50170		en preparación	
Alta velocidad de transferencia de data para periféricos descentralizados		<b>Perfiles de aplicación específicos:</b> Máquinas textiles Automatización de edificios Drivers, sensores y actuadores, PLCs, Switch gear de bajo voltaje		Técnicas de transmisión con seguridad intrínseca de acuerdo a IEC 1158-2	

TABLA 4.1. La familia PROFIBUS

En la Figura 4.21 se muestra como se conectan y relacionan estas redes de la familia Profibus.

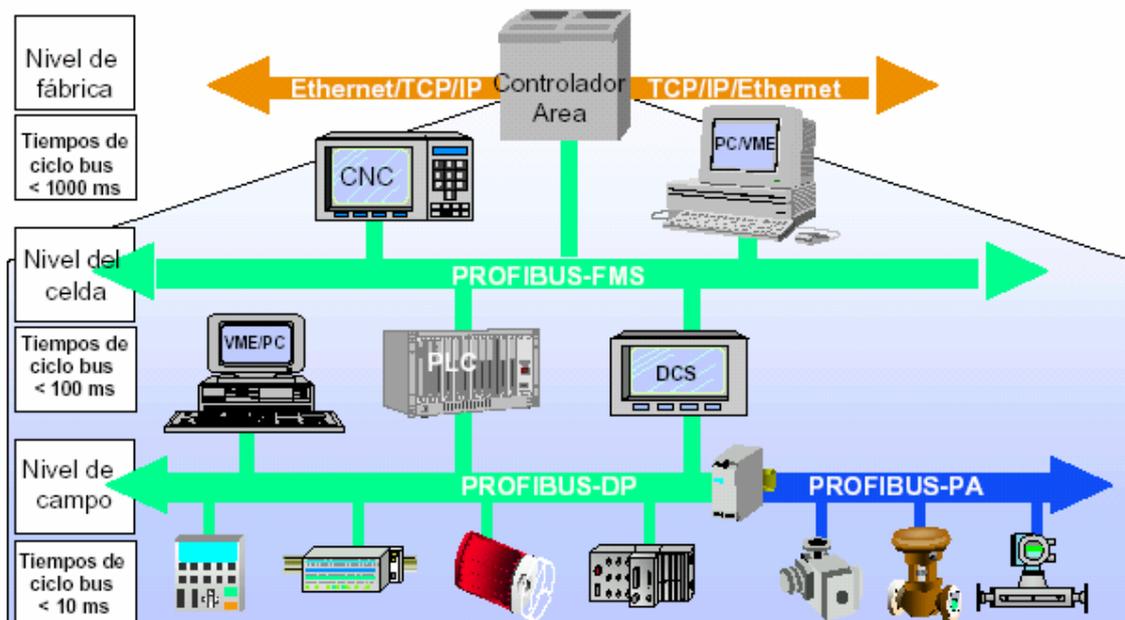


Figura 4.21 Jerarquía de la Red Profibus

#### **4.9.5.1 PROFIBUS-FMS**

Profibus FMS es la solución universal para la comunicación entre el nivel superior (nivel de celda) y el nivel de campo de acuerdo a la jerarquía de comunicación industrial de Profibus (Figura 4.45). Para llevar a cabo tareas de comunicación extensivas con transferencia de datos en forma cíclica o acíclica a una velocidad de transmisión mediana, el servicio FMS "*Especificación de Mensaje de Bus de campo (Fieldbus Message Specification, por sus siglas en inglés)*" ofrece una amplia gama de funcionalidad y flexibilidad.

El servicio FMS está definido como una subparte de las funciones MMS (Manufacturing Message Specification "Especificaciones de Mensajes del Fabricante", ISO 9506) del protocolo MAP, protocolo de automatización de fabricación (Manufacturing Automation Protocol).

El control a nivel de celda se da en áreas (celdas) individuales que proveen de control durante la producción. Los controladores a este nivel deben comunicarse con otros sistemas supervisores.

#### **4.9.5.2 Configuración PROFIBUS-FMS**

Un sistema típico de PROFIBUS-FMS está compuesto por varios equipos de automatización inteligentes (Figura 4.22):

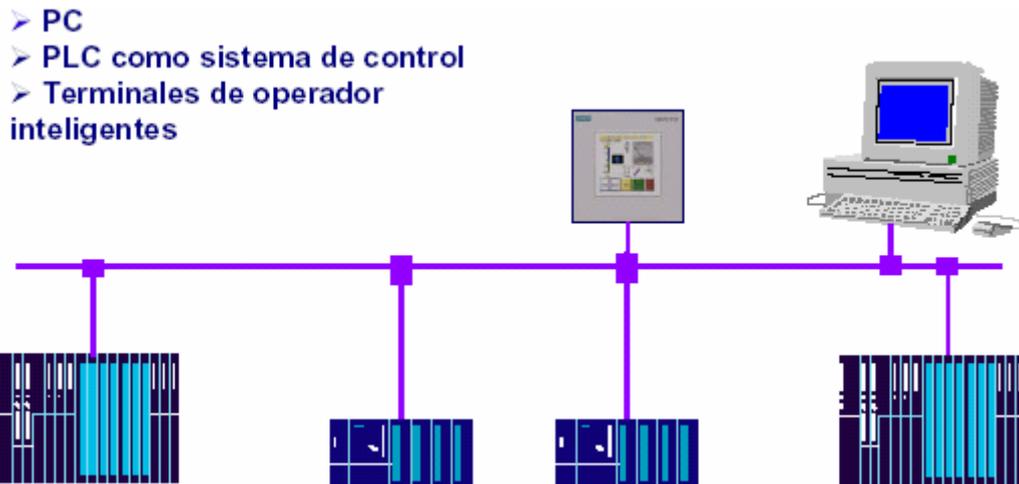


Figura 4.22 Configuración PROFIBUS-FMS

#### 4.9.6 PROFIBUS-DP

Ésta es la versión de desempeño optimizado de la red PROFIBUS, dedicado específicamente a comunicaciones de tiempo crítico entre sistemas de automatización y periféricos distribuidos.

#### 4.10 OPC

Es un estándar para la comunicación de numerosas fuentes de datos, bien sean dispositivos de campo en una industria o bien una base de datos en una habitación de control.

La arquitectura de información para el proceso industrial que se muestra en la figura 4.23 incorpora los siguientes niveles.

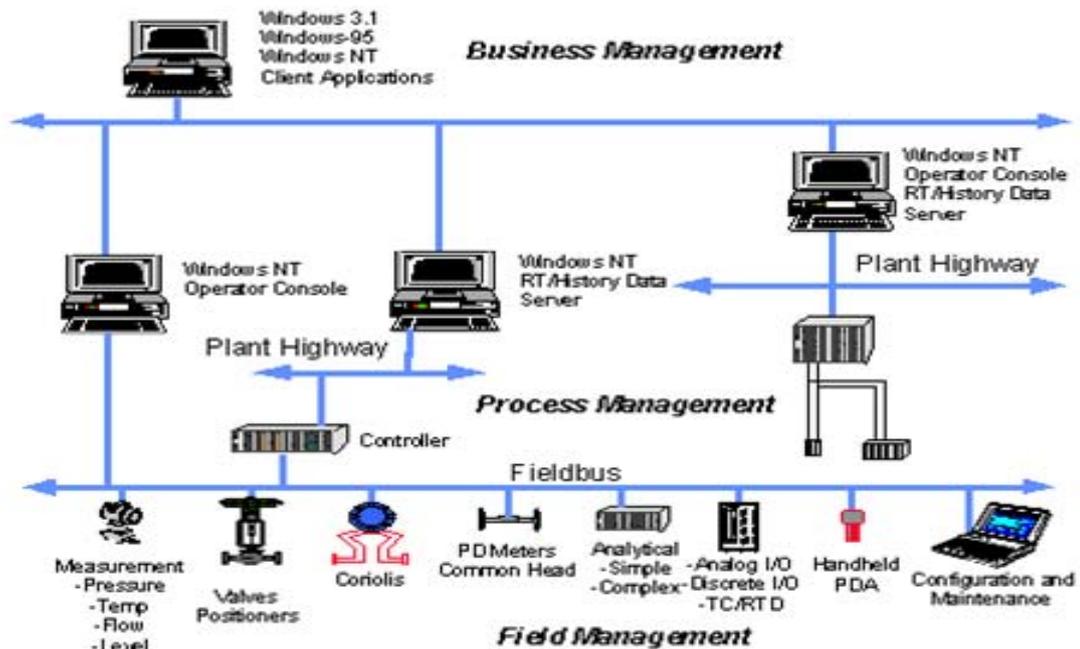


Figura 4.23 Configuración normal de red

Lo que se necesita es una forma común con la que las aplicaciones accedan a los datos de cualquier fuente de datos, ya sea un dispositivo o una base de datos.

Hay muchas aplicaciones clientes que requieren datos de una fuente de datos y que para ello requieren del desarrollo de drivers propios.

Esto lleva a los siguientes problemas:

- **Mucho esfuerzo de duplicación.** Cada uno debe escribir un driver para cada vendedor de hardware particular.
- **Inconsistencias entre drivers de los vendedores.** existen características de ciertos dispositivos hardware que no son soportadas por todos los desarrolladores de drivers.

Dos paquetes no pueden acceder generalmente al mismo dispositivo simultáneamente ya que cada uno contiene drivers independientes.

Los fabricantes de hardware intentan resolver estos problemas desarrollando drivers, pero son entorpecidos por las diferencias entre los protocolos de los clientes.

OLE for Process Control (OPC) establece una línea entre los proveedores de hardware y los desarrolladores de software. Esto proporciona un mecanismo para proporcionar datos desde una fuente de datos y comunicar los datos a cualquier aplicación cliente de una forma estándar. Un vendedor puede desarrollar un servidor muy optimizado y reusable para comunicar las fuentes de datos. Proporcionar el servidor con un interfaz OPC permite a cualquier cliente acceder a sus dispositivos.

#### **4.11 TENDENCIA DE LAS ARQUITECTURA DE APLICACIONES**

Es cada vez mayor el número de aplicaciones son desarrolladas en ambientes como Visual Basic, Delphi, Power Builder, etc. OPC ha tenido en cuenta esta tendencia. Microsoft comprende esta tendencia y ha diseñado OLE/COM de forma que permita utilizar componentes. Los fabricantes escribirán componentes software en C y C++ para encapsular las características del acceso a los datos de un determinado dispositivo, de forma que los desarrolladores de aplicaciones puedan escribir código en VB que utilice los datos de la planta industrial

El propósito de todas las especificaciones es facilitar el desarrollo de servidores OPC en C y C++, y facilitar el desarrollo de aplicaciones clientes OPC en el lenguaje escogido. Aunque también se intenta que los servidores OPC sean desarrollados en otros lenguajes.

OLE for Process Control (OPC) se ha diseñado para permitir a las aplicaciones clientes acceder a la planta industrial de una manera consistente. Si se produce una aceptación amplia en la industria, OPC proporcionará muchos beneficios.

Los fabricantes de hardware solo tendrán que fabricar un conjunto de componentes software para inicializar sus aplicaciones.

Los desarrolladores de software no tendrán que reescribir drivers debido al cambio de características o incorporaciones en una nueva versión de hardware

Los compradores tendrán más opciones con las que implementar su sistema de producción integrado en clases.

#### **4.12 MICROSOFT SQL SERVER Y VISUALBATCH**

La tendencia tradicional del desarrollo de software era integrar tanta funcionalidad como fuese posible en una aplicación software. Mientras que esto podría dar lugar a un potente producto software, habían problemas inherentes debido a la dificultad para utilizar dichas aplicaciones. Para el usuario, trabajar y comprender el software se convirtió en algo cada vez más complicado, incluso aunque muchos fabricantes establecían importantes requerimientos de facilidad de uso en sus productos.

Los sistemas modernos permiten a los usuarios componer sus aplicaciones usando la modularidad ofrecida por los diferentes productos hardware y software. Esta aproximación sin un estándar como OPC llevaría a la aparición de problemas eventuales con el interfaz. Sin embargo, con OPC, es posible para el usuario integrar fácilmente una variedad de aplicaciones.

Los programas software que están integrados con OPC pueden ser combinados con los productos de Microsoft Office y Back Office. Esto significa que el intercambio de datos entre los programas serán mucho más fáciles de llevar a cabo en el futuro. Con estos mecanismos, los datos pueden ser intercambiados entre aplicaciones desde el nivel de producción, nivel de control de procesos, sistemas de gestión de información y otros sistemas de administración de una forma eficiente, fácil de usar.

Un excelente ejemplo de lo mencionado es el hecho de que el nuevo producto de Intellution, VisualBatch , estará fuertemente ligado con el software Microsoft's SQL Server. Esta relación entre las aplicaciones no solo muestra el claro movimiento en este ambiente hacia ofrecer una solución *completa* a los compradores sino también muestra la fuerza de Microsoft en este ambiente. Microsoft SQL Server se está convirtiendo rápidamente en la plataforma estándar de los sistemas de bases de datos al mismo tiempo que Windows NT se ha convertido en el sistema operativo en este ambiente.

#### 4.13 BASE DE DATOS

El término de bases de datos fue escuchado por primera vez en 1963, en un simposio celebrado en California –USA.

Una base de datos se puede definir como un conjunto de información relacionada que se encuentra agrupada ó estructurada.

Desde el punto de vista de la informática, la base de datos es un sistema formado por un conjunto de datos almacenados en discos que permiten el acceso directo a ellos y un conjunto de programas que manipulen ese conjunto de datos.

Una base de datos tiene mucha importancia en el ritmo de vida que llevamos en los actuales momentos, ya que, está acelera el ritmo en el momento realizar una búsqueda de información.

El análisis de requerimientos para una base de datos incorpora las mismas tareas que el análisis de requerimientos del software. Es necesario un contacto estrecho con el cliente; es esencial la identificación de las funciones e interfaces; se requiere la especificación del flujo, estructura y asociatividad de la información y debe desarrollarse un documento formal de los requerimientos.

Requerimientos administrativos: se requiere mucho más para el desarrollo de sistemas de bases de datos que únicamente seleccionan un modelo lógico de base de datos. La bases de datos son una disciplina organizacional, un método, más que una herramienta o una tecnología. Requiere de un cambio conceptual y organizacional.

Elementos claves de organización en un ambiente de Bases de Datos son:

- Sistema de administración de base de datos
- Administración de información
- Tecnología de administración de base de datos
- Usuarios
- Planeación de información y tecnología de modelaje

Una base de datos contiene entidades de información que están relacionadas vía organización y asociación. La arquitectura lógica de una base de datos se define mediante un esquema que representa las definiciones de las relaciones entre las entidades de información. La arquitectura física de una base de datos depende de la configuración del hardware residente. Sin embargo, tanto el esquema (descripción lógica) como la organización (descripción física) deben adecuarse para satisfacer los requerimientos funcionales y de comportamiento para el acceso al análisis y creación de informes.

El sistema manejador de base de datos(DBMS) es un conjunto de programas que se encargan de manejar la creación y todos los accesos a las bases de datos. Se compone de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta.

Una de las ventajas del DBMS es que puede ser invocado desde programas de aplicación que pertenecen a Sistemas Transaccionales escritos en algún lenguaje de alto nivel, para la creación o actualización de las bases de datos, o bien para efectos de consulta a través de lenguajes propios que tienen las bases de datos.

## **CAPÍTULO V**

### **DISEÑO E IMPLEMENTACIÓN DEL SISTEMA SCADA PARA EL CONTROL DISTRIBUIDO DE LAS ESTACIONES DE CONTROL DE PROCESOS**

#### **5.1 DISEÑO E IMPLEMENTACIÓN DE UN CONTROL DE PROCESO DE NIVEL BASADO EN FIELD POINT 1000**

En este capítulo se implementará lo revisado en los capítulos anteriores con el fin de lograr un control PID con el uso del Field Point y LabVIEW

##### **5.1.1 Requerimientos del proceso**

Los elementos propios de la estación de control de Nivel son:

- Estación de proceso de nivel
- Transmisor de presión diferencial (P/D)
- Convertidor Corriente a Presión (C/P)
- Controlador Foxforo

El controlador propio de la estación, no va ser el encargado de realizar la acción de control sobre esta estación, ya que para fines demostrativos se ha seleccionado al Field Point 1000 y a LabVIEW como elementos del sistema de control.

Además se ha seleccionado al Field Point porque este dispositivo tiene bancos adicionales de entradas y salidas análogas de corriente, que dan la facilidad de la adquisición de datos para manipular desde LabVIEW, con el fin de realizar el control en la estación

## 5.1.2 Equipo de control a utilizar.

### 5.1.2.1 Configuración De Comunicación Field Point Con LabVIEW

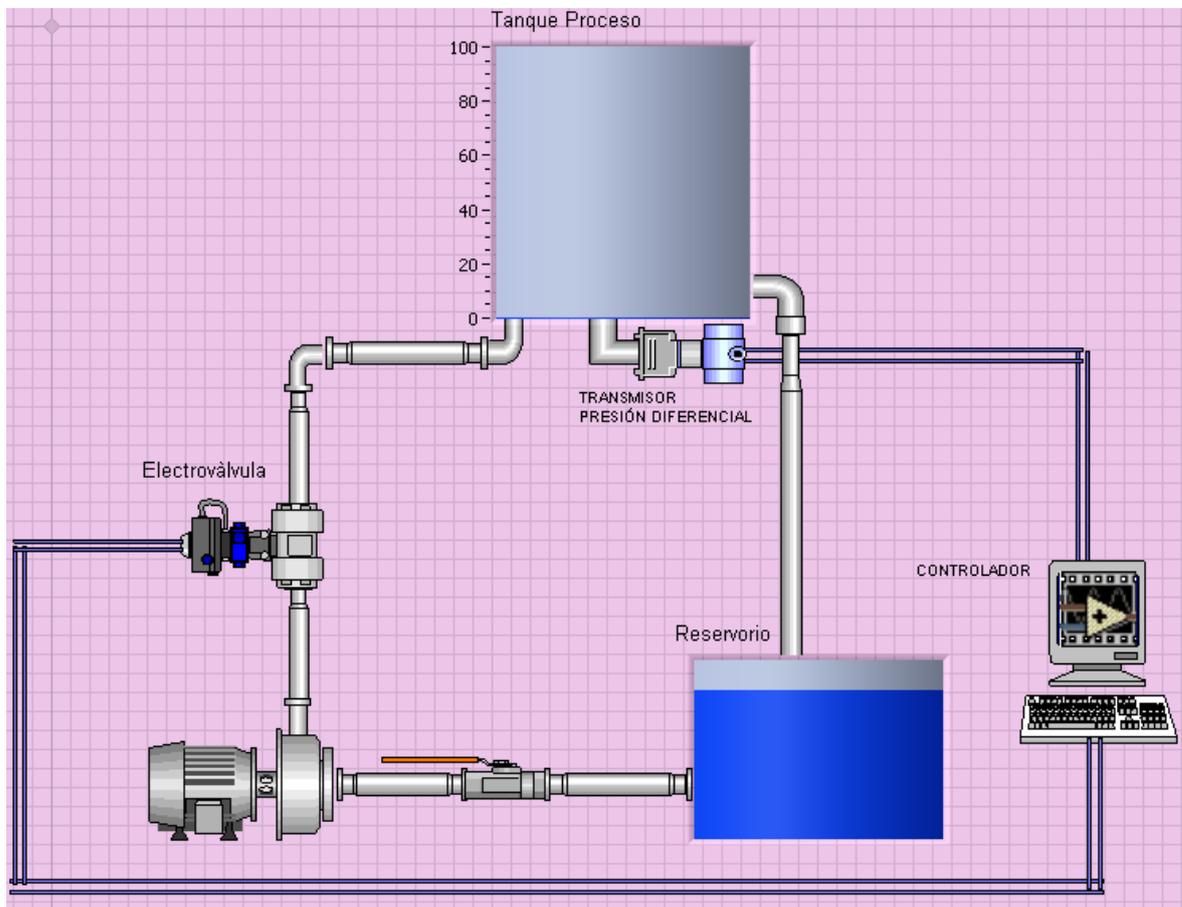


Figura 5.1 Estación de Control Nivel con Field Point

Los valores de proceso a ser considerados son:

- Límites en 0 y 80"
- Acción del actuador 4-20 mA con FP AO 200
- Señal de medida del proceso PV 4-20 mA / 0-80"

### 5.1.2.2 Implementación de Comunicación con Field Point

La figura 5.2 requiere un lazo de control con el Field Point FP-1000 y los módulos FP AO 200, FP AI-110 y LabView 7.0; en los tags correspondientes en el archivo FP.IAK se determinó:

Interfase de comunicación FP 1000	dirección FP-1000@0
Señal de medida FP AI-110	dirección FP-AI-110@1
Señal de actuador FP AO-200	dirección FP-AO-200@2

En primer instante se diseñan las pantallas de presentación con el respectivo diagrama del VI, que se presenta en la Figura 5.2

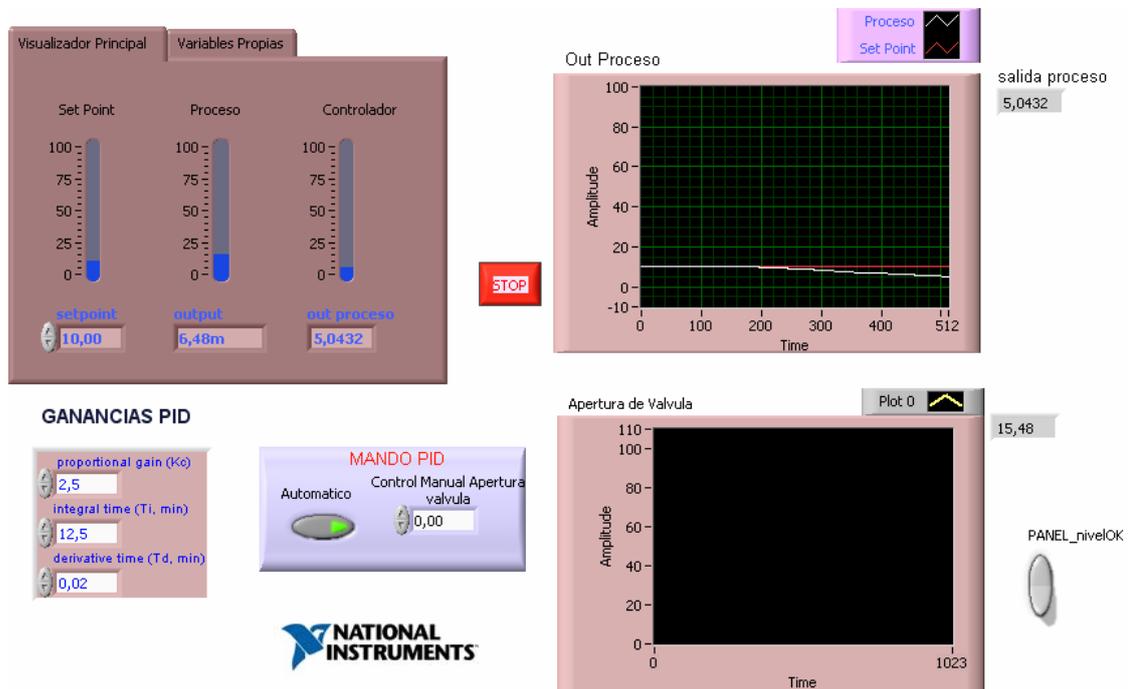
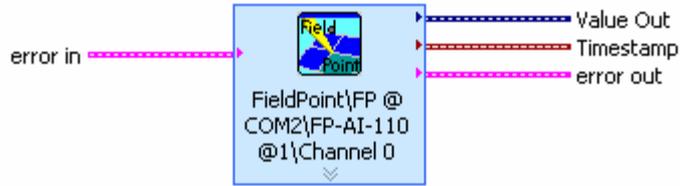
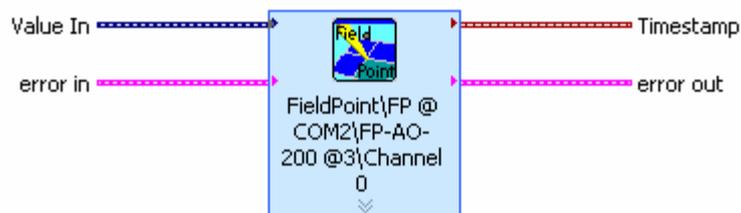


Figura 5.2 Configuración del Panel de presentación

Para la realización del Vi de control se utiliza Field Point Express según se indica en las figuras 5.3(a) y 5.3 (b).



**Figura 5.3 (a) Field Point Express Entrada**



**Figura 5.3 (b) Field Point Express Salida**

El Field Point Express permite leer y escribir en los canales de los módulos del Field Point. Para manejo de un canal simple o multi canal se debe configurar cada módulo en el MAX.

Parámetros	Descripción
<b>error in</b>	Describe la condición de errores ocurridos antes de que el VI o las funciones se hayan ejecutado.
<b>error out</b>	Contiene una información similar al error in con la diferencia que éste indica el estado de error que ha producido el VI o las funciones ejecutadas.
<b>Timestamp</b>	Maneja los tiempos realizados en la operación de lectura del Field Point en cada punto.
<b>Value In.</b>	Indica el valor de escritura que van hacia la salida del Field Point IO point.
<b>Value Out</b>	Es un dato de tipo arreglo que contiene los valores de lectura del Field Point.

Field Point Express Entrada toma los datos de la dirección específica en este caso FP-AI-110@1 para ser evaluados por el programa; mientras que Field Point Express Salida FP escritura describe el valor desde un programa de aplicación hacia la dirección dada por el tag FP-AO-200@2, figura 5.3 (b); los dos están relacionados con las gráficas de tendencias en medida del proceso PV y salida del controlador respectivamente según la figura 5.4.

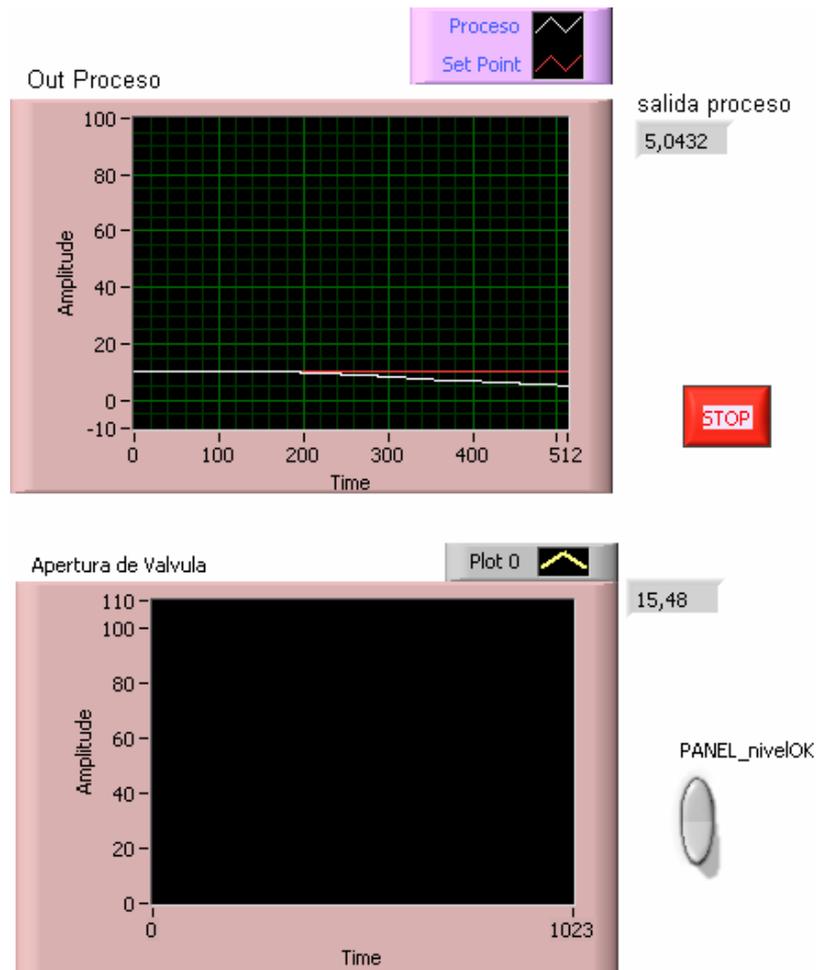
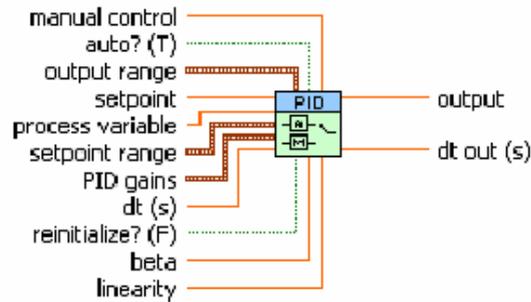


Figura 5.4 tendencia valores control

### 5.1.2.3 Implementación del PID con LabVIEW

Para el diseño de los Vi's que manejan la parte de control mediante el computador, se escogió el Vi PID.advanced de la pantalla de diagrama de bloque, que tiene una distribución de entradas y salidas como se indica en la figura 5.5.



**Figura 5.5 PID avanzado**

El algoritmo PID avanzado tiene características opcionales a más de las propias del PID, tiene un modo de control que se puede realizar manualmente o automáticamente, acción integral no lineal.

En el Vi a operar, se puede usar un solo valor a la entrada o una serie de valores de la entrada.

Los detalles se describen en la tabla 5.1:

Manual control (control manual).	Es el valor de salida usado cuando auto está fijado en Falso
Auto?(T)	Cuando está en Verdadero se selecciona el control automático (por defecto), Falso selecciona el modo de control manual.
Output Rango (rango de salida)	La salida del control es forzada a un rango específico, para salida alta y salida baja. Los rangos por defecto son: -100 a 100.
Setpoint (punto de consigna)	Representa el valor del setpoint de la variable del proceso a ser controlada. Es el valor deseado para la variable del proceso
Process variable (variable del proceso)	Es el valor medido de la variable del proceso que está siendo controlado. Éste es un valor de realimentación de un lazo de control cerrado
Setpoint rango (rango del setpoint)	Son los valores máximos y mínimos para el manejo del setpoint / rango de la variable del proceso. Este rango es usado para el cálculo de la acción integral no lineal. Por defecto es un rango de 0 a 100.
PID gains (ganancias del PID)	Es un Cluster que define los parámetros de la ganancia proporcional, integral y tiempo derivativo (Kp,Ki,Td).

dt(s)	Es un intervalo (en segundos) al cual es llamado este Vi, es usado en los cálculos. Si este valor es menor o igual que cero, un temporizador interno de 1 milisegundo es usado.
Reinitialize?(F) (reset)	Si se pone Verdadero, se reinician los parámetros internos (tales como el error integral), por defecto los valores son 0.0
Beta	Especifica el énfasis relativo de rechazo de perturbación al setpoint. El valor predefinido de 1 es apropiado para la mayoría de las aplicaciones. Un valor más pequeño entre 0 y 1 puede usarse para especificar el énfasis en el rechazo de perturbación (como los cambios de carga de proceso).
Linearity (linealidad)	Fija un valor de finalidad a los errores, su rango es de 0 a 1. 1.0 da una respuesta lineal, mientras que 0.1 da una respuesta cuadrática.
Output (salida)	Es la salida de control del algoritmo PID el que es aplicado al proceso controlado
dt out (s)	El intervalo de tiempo real en segundos.

Tabla 5.1 Descripción de las líneas del VI PID avanzado

El VI matemático desarrollado por LabVIEW se indica en la figura 5.6

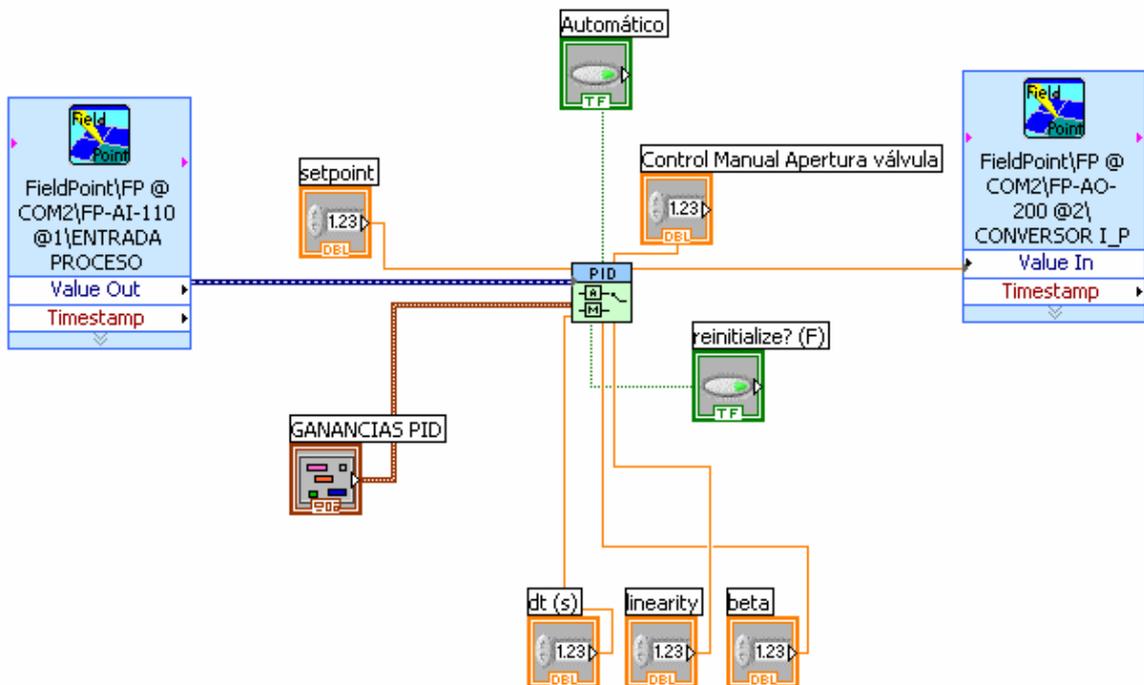


Figura 5.6 Diseño del VI para el control

## **5.2 DISEÑO E IMPLEMENTACION DE UN CONTROL DE PROCESO DE FLUJO BASADO EN TARJETA DE ADQUISICION DE DATOS USB**

En esta sección se implementará lo revisado en los capítulos anteriores con el fin de lograr un control PID con el uso de la tarjeta de adquisición de datos DAQ USB y LabVIEW

### **5.2.1 Requerimientos del proceso**

Los elementos propios de la estación de control de flujo son:

- Estación de proceso de flujo
- Transmisor de presión diferencial (P/D)
- Variador de velocidad

El controlador propio de la estación no va ser el encargado de realizar la acción de control sobre esta estación ya que para fines demostrativos se ha seleccionado a la Tarjeta de Adquisición de Datos USB y a LabVIEW como elementos del sistema de control.

Además se ha seleccionado la Tarjeta de Adquisición de Datos USB porque este dispositivo tiene un puerto de comunicación de tecnología innovadora ya que es uno de los primeros en disponer de su comunicación con la Pc mediante el puerto USB, tiene sus entradas y salidas analógicas que se adaptan para la adquisición y control de la estación de flujo. Cabe indicar que sus entradas y salidas son totalmente configurables para voltajes estandarizados, además este dispositivo no necesita de fuente externa para su alimentación.

En la figura 5.7 se indica el hardware de la DAQ USB 6009 que se usa en el control para la estación de flujo.



**Figura 5.7 Tarjeta de Adquisición de Datos DAQ USB 6009**

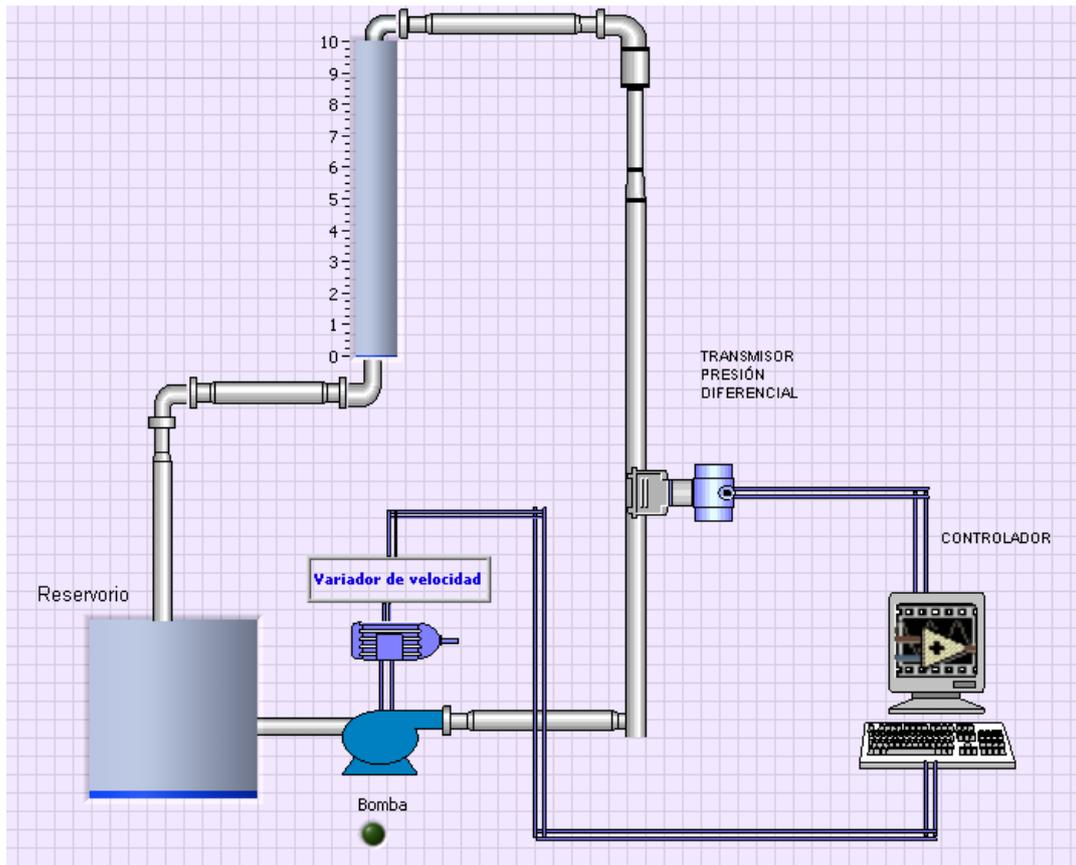
## **5.2.2 Equipo de control a utilizar.**

### **5.2.2.1 Configuración de Comunicación DAQ USB con Labview**

Este dispositivo incorpora un drive para su configuración y reconocimiento en la computadora el mismo que es NI-DAQmx que contiene el Task Configuration Utility, que es una aplicación que permite crear tareas de las entradas y salidas de la tarjeta, pero solo sirve para ver el estado de las variables.

La manera más efectiva de usar los USB-6009 no es configurando la tarea virtual desde el configuration utility sino desde LabVIEW mismo. Debido a que el driver NI-DAQmx base, no permite hacer la búsqueda automática de canales. Por esta razón, en LabVIEW hay que hacer un programa que consta de seis pasos, y es recomendable añadir un séptimo de aviso de errores.

En primer instante se diseñan las pantallas de presentación con el respectivo diagrama de bloques del VI que se presenta en la figura 5.8



**Figura 5.8 Estación de Control Nivel con DAQ USB**

### 5.2.2.2 Implementación de la comunicación DAQ USB con LabVIEW

Los pasos consisten en funciones, las cuales están en la paleta de Funciones >> All functions >> NI Measurements >> DAQmx Base Data Acquisition. También se usa la función DAQmxBASE CREATE TASK que está en la sub-paleta DAQmx Base - Advanced Task Options.

Los pasos son los siguientes:

- a) Crear Tarea.- Permite empezar una tarea de la tarjeta de adquisición de datos DAQ USB 6009.
- b) Configurar Canal.- Se selecciona el canal en el modo de operación que se desee operar la DAQ y estos pueden ser tanto en modo simple y modo

diferencial, adicional en este VI se configura el número de dispositivo y el rango de voltaje del mismo.

- c) Iniciar tarea.- Inicia la tarea que se ha creado para la DAQ USB
- d) Leer canal / Escribir canal.- Dependiendo de la ubicación de este VI sea en la entrada o salida, permitirá adquirir las señales de la variable del proceso para poder ingresar en el control PID. Si se encuentra en modo de escritura éste permite recibir la señal de control a ser enviada al módulo para su control.
- e) Detener tarea.- Una vez realizado el proceso anterior, este Vi permite detener la tarea cada vez que se realice un ciclo completo.
- f) Borrar tarea.- Limpia la tarea para empezar con una nueva en cada ciclo de repetición del lazo de control.
- g) Error.- Vi que facilita la visualización de los errores que se puedan producir en los seis pasos anteriores.

La secuencia de configuración se indica en la figura 5.9 (a), donde se presenta solo la adquisición de la variable del proceso, y la figura 5.9 (b) presenta el programa para sacar el valor de la salida del controlador.

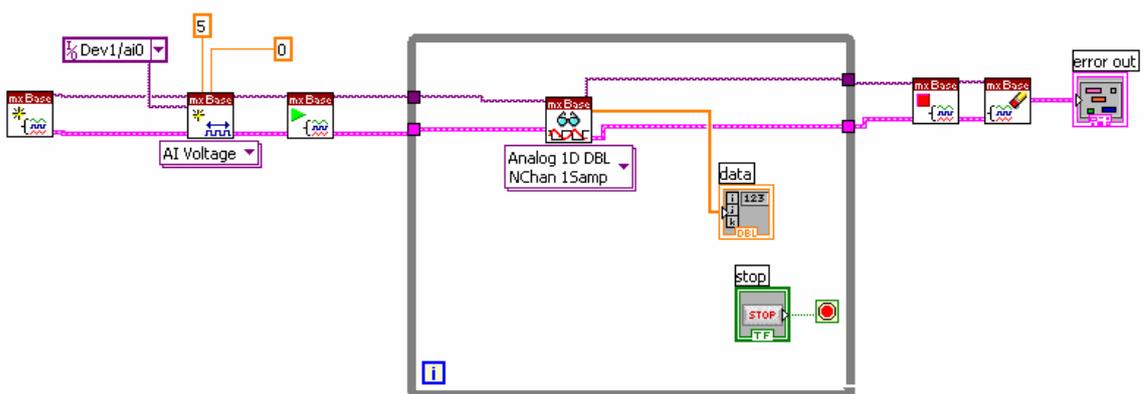
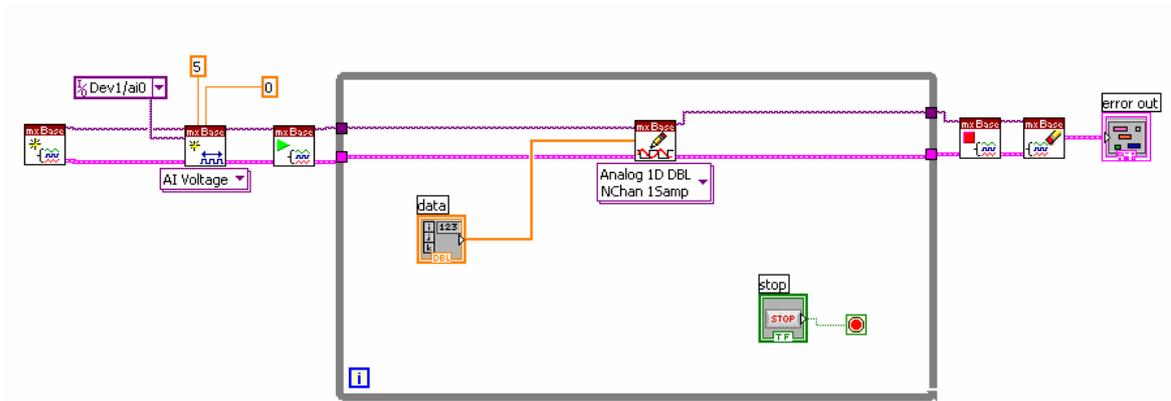


Figura 5.9 (a) Configuración de Tarjeta de Adquisición de Datos para entrada de datos



**Figura 5.9 (b) Configuración de Tarjeta de Adquisición de Datos para salida de datos**

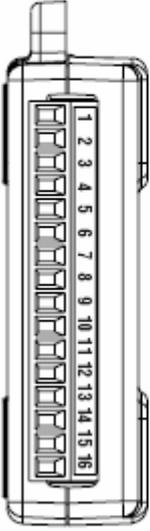
La manera más fácil para configurar cada canal es conocer la sintaxis de los canales virtuales que se desea usar; la misma que se detalla a continuación:

Dev#,ai# donde el primer # es el número del dispositivo USB que se está usando, y el segundo # es el número del canal analógico de donde se quiere adquirir. Si es salida analógica, entonces usar ao#, y así sucesivamente.

### 5.2.2.3 Implementación del PID con LabVIEW

Esta implementación se la realiza utilizando un canal de entrada analógica para adquirir la señal de la variable del proceso y una salida analógica para la salida del controlador.

La configuración de los canales y voltajes tanto de entrada y salida se pueden seleccionar de las tablas 5.2 y 5.3 respectivamente:

MODULO	TERMINAL	SEÑAL MODO SIMPLE	SEÑAL MODO DIFERENCIAL
	1	GND	GND
	2	AI 0	AI 0+
	3	AI 4	AI 0-
	4	GND	GND
	5	AI 1	AI 1+
	6	AI 5	AI 1-
	7	GND	GND
	8	AI 2	AI 2+
	9	AI 6	AI 2-
	10	GND	GND
	11	AI 3	AI 3+
	12	AI 7	AI 3-
	13	GND	GND
	14	AO 0	AO 0
	15	AO 1	AO 1
	16	GND	GND

**Tabla 5.2 Descripción de los canales de la DAQ-USB**

Entradas analógicas	
Modo	Rango
Simple	$\pm 10$ V
Diferencial	$\pm 20$ V, $\pm 10$ V, $\pm 5$ V, $\pm 4$ V, $\pm 2.5$ V, $\pm 2$ V, $\pm 1.25$ V, $\pm 1$ V
Salidas analógicas	
Rango de salida	0 a +5 V

**Tabla 5.3 Descripción de los voltajes de la DAQ-USB**

Para el diseño del VI de control se debe realizar un escalamiento de la señal para poder manejarla con los datos propios de LabVIEW, y su diseño quedaría determinado como indica la figura 5.10.

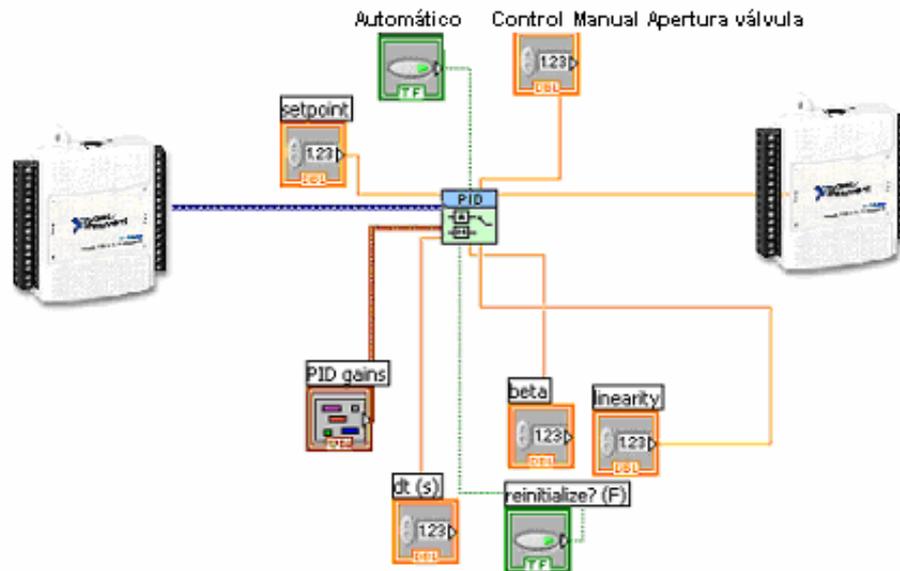


Figura 5.10 Diseño del VI para el control

### 5.3 DISEÑO DEL CONTROL DEL PROCESO DE PRESIÓN.

#### 5.3.1 Requisitos funcionales.

La estación de control de proceso de presión, simula la operación de dos tanques de almacenamiento de un gas cualquiera (en nuestro caso aire a temperatura ambiental), que se pueden conectar a una carga consistente en un filtro de aire, o simplemente se puede dejar abierta la válvula de salida del segundo tanque, para que produzca el decremento de la presión en los cilindros, y de esta forma simular la caída de presión con la que tiene que trabajar el controlador. Para esto se sirve de varios elementos adicionales que se van a describir a continuación: Posee un convertor I/P, que tiene una entrada de alimentación neumática de hasta 20 PSI, que será realmente la presión que se regule al estándar de 3 a 15 para el control de la válvula de posicionamiento. Este convertor toma la corriente proveniente del controlador, que está calibrado en un rango estándar de 4 a 20 mA, y realiza la conversión proporcional a presión de 3 a 15 PSI, que alimentará a la válvula anteriormente descrita.

En este contexto entonces, el detector de error, dará la señal de corriente correspondiente a la acción de control al conversor I/P, éste a su vez convierte el estándar de corriente al estándar de presión; y regula la posición de una válvula neumática que es el posicionador. La salida de ésta válvula se conecta directamente a la entrada de los tanques de almacenamiento del aire para aumentar o disminuir la presión de los mismos. A la salida del tanque simultáneamente con la carga, se conecta la entrada neumática del transmisor de presión absoluta Foxboro 823DP quién mide la presión que está presente en ese momento en los tanques y envía la señal medida al controlador con un rango estándar de 4 a 20 mA.

Por ser la entrada del controlador, tanto como la salida de él, señales en estándar de red de 4 a 20 mA, se necesita elementos de control y adquisición de señales que cumplan con dichas características. Como se puede observar en el capítulo dos, la tarjeta DAQ PCI 6014 que posee el laboratorio, es una tarjeta que maneja entradas y salidas análogas en el orden de hasta  $\pm 20V$ , pero no presenta la posibilidad de realizar mediciones o generar señales de corriente, como es el caso del Field Point FP100, que sí presenta esta posibilidad. Debido a esto, es necesario que se realice un acondicionamiento de señales tanto para la entrada como para la salida de la tarjeta. La entrada no presenta muchos inconvenientes, ya que al hacer circular una corriente de 4 a 20 mA por una resistencia estándar de instrumentación de  $250 \Omega$ , se produce en ella una tensión de 1 a 5 voltios, que se pueden fácilmente medir en la tarjeta NI-PCI-6014.

El inconveniente se presenta en la salida, ya que los niveles de voltaje que genere la salida de la tarjeta tienen que convertirse en una fuente de corriente controlada por voltaje y deben ser exactamente en niveles adecuados.

Afortunadamente no es necesario realizar ninguna de estas acciones ya que la tarjeta se puede conectar a un sistema de acondicionamiento de señales, manejado por un módulo de control SCC. Este módulo posee la facilidad de conectar en un transportador SCC 2345 hasta 16 módulos de acondicionamiento de diferentes características como entradas para termopares, RTD,

amplificadores y atenuadores de señal, entradas y salidas de corriente, conversores frecuencia voltaje, filtros, strain gauge, etc.

Así es como se debe utilizar en conjunto con el transportador SCC 2345, los módulos de entrada y salida de corriente que son: NI-SCC-CI20 módulo de entrada de corriente que posee 2 canales de forma diferencial que acepta entradas de 0 a 20 mA, y produce una salida de voltaje de 0 a 5 V. El módulo NI-SCC-CO20, de salida de corriente que posee un canal de salida no referenciado, en un rango de entrada de 0 a 10 Voltios que genera la tarjeta DAQ y los convierte proporcionalmente en salida de corriente de 0 a 20 mA.

### **5.3.2 Implementación del Hardware.**

El módulo transportador de los dispositivos acondicionadores de señal, se conecta directamente a la tarjeta DAQ a través del cable de conexión SHC68-68EP. Los módulos de acondicionamiento SCC-CI20 y SCC-CO20, se conectarán internamente en los zócalos adecuados para su conexión. Debe tenerse muy en cuenta cuál de los módulos se va a conectar y en qué zócalo. Es necesario referirse al manual de usuario del módulo para verificar las conexiones, pues existen zócalos similares pero algunos de ellos corresponden a entradas y otros a salidas.

En nuestro caso se utilizarán los zócalos J2 para el módulo de entrada de corriente y J18 para el módulo de salida de corriente. Internamente se asignarán a cada uno de ellos uno de los canales analógicos de entrada y salida de la tarjeta NI-PCI-6014, en este caso se configurará el canal de entrada 0 para el módulo SCC-CI20 y el canal 8 para el módulo de salida de corriente, tal como se puede observar en la figura 5.11.

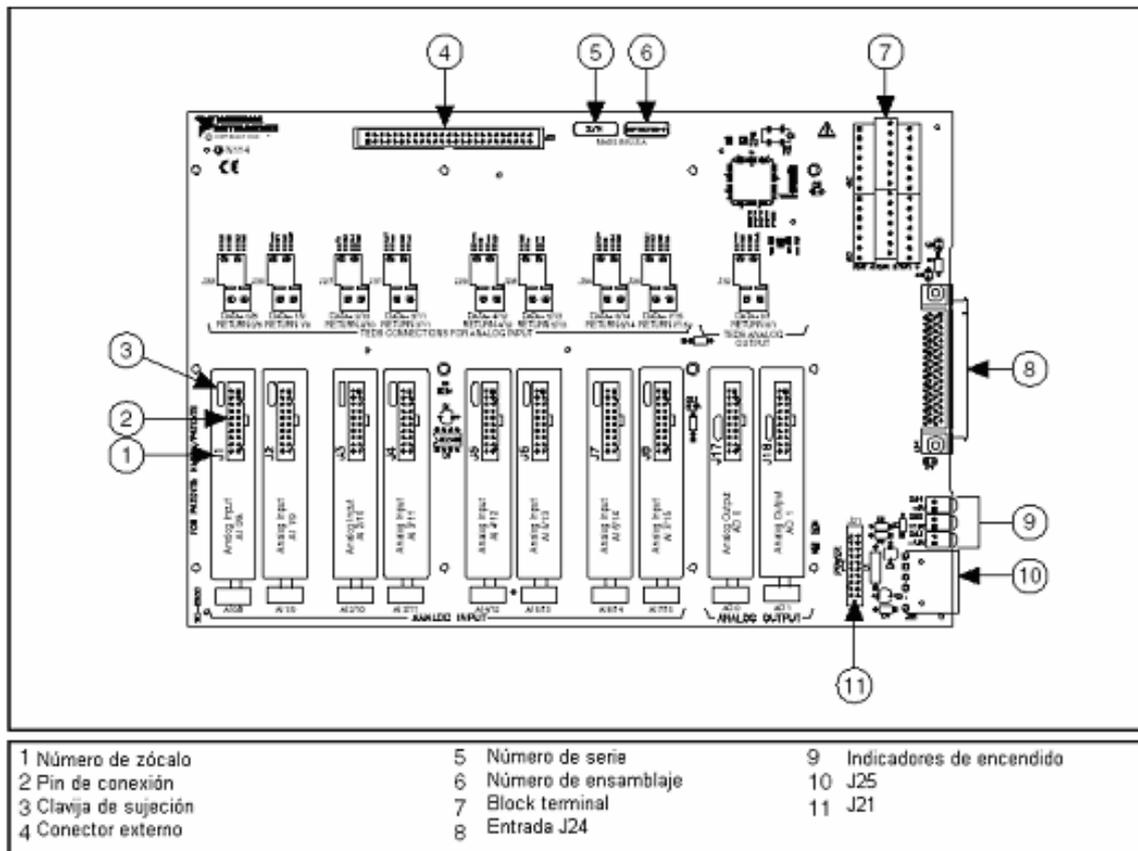
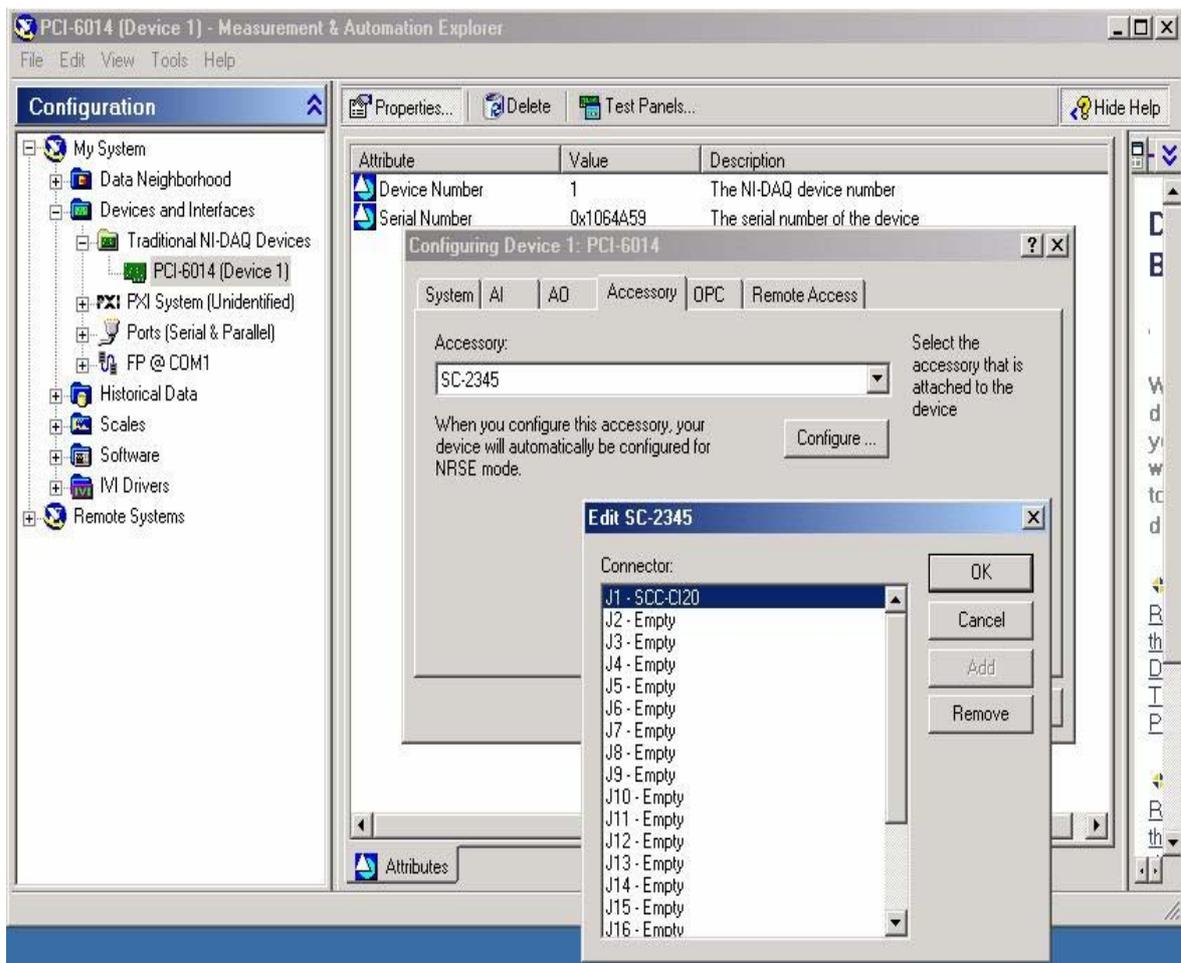


Figura 5.11 Portador de sistemas SCC.

### 5.3.3 Configuración del módulo por software.

El sistema no es reconocido por la tarjeta automáticamente, es por ello necesario que se lo configure en el asistente de configuración MAX. Al arrancar MAX se deberá seleccionar Devices and Interfaces e indicar que la tarjeta que se va a utilizar es la NI-PCI-6014 y configurarla en accesorios, en esta pestaña también existe la posibilidad de configurar cada módulo acondicionador. Para el presente caso, se seleccionará en los canales de entrada el módulo SCC-CI20 en el canal 0, y para los canales de salida el módulo SCC-CO20 en el canal 8. Al cerrar MAX, el sistema ya acepta las entradas y salidas de corriente para el acondicionador como sus proporcionales de voltaje en la tarjeta; el procedimiento se indica en la figura 5.12.



**Figura 5.12. Configuración del módulo SCC por software.**

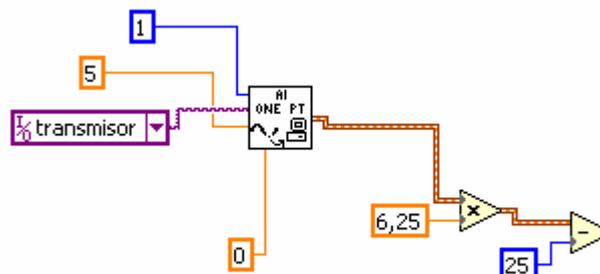
### **5.3.4 Implementación del Software de Control.**

Para la implementación del software de control se utiliza el software LabVIEW versión 7.0, por la compatibilidad que tiene con todos los dispositivos de hardware mencionados en la utilización del módulo (tarjeta DAQ NI-PCI 6014, NI-SCC-CI20, CO20, SCC 2345), al ser del mismo fabricante y obviamente no existir dificultades de comunicación entre los dispositivos.

En la entrada, se debe previamente crear un canal virtual para la entrada del canal 0, que está ya configurada para el SCC-CI20. Para ello se debe utilizar el asistente MAX. En la opción data neighborhood, se selecciona un canal virtual de voltaje, escalado a la señal de entrada de corriente presente a través del SCC. Seguidamente, se debe configurar un canal virtual de salida que utilice la salida

configurada para el SCC-CO20, en este caso el canal 8, se selecciona en el asistente escalando la salida en un rango de corriente de 0 a 20 mA.

En LabVIEW se creará un nuevo instrumento virtual que maneje las opciones del MAX ya configuradas. La entrada solamente necesita una función de muestreo simple, pues al ser una única señal, que solamente tiene variaciones poco rápidas en un solo sentido, la adquisición no necesita ni sincronizaciones, ni buffer de almacenamiento temporal. Solamente se necesita utilizar el VI de adquisición e ingresar las variables ya configuradas como el canal virtual de entrada creado (transmisor) y las constantes de adquisición. El proceso descrito se detalla en la figura 5.13.

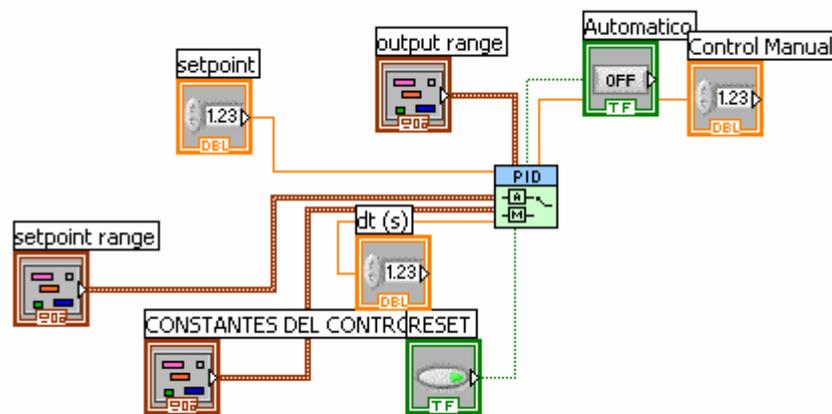


**Figura 5.13 Uso del VI de adquisición.**

En este caso, se debe asegurar que el canal virtual creado sea el que se utilizará, y además los límites del canal físico de entrada (voltaje de 0 a 5 V) y el dispositivo a usar (device 1).

Luego de ello se deberá realizar un convertor de niveles con la señal obtenida de esta función, ya que el módulo lee una entrada de corriente de 0 a 20 mA a pesar de que la fuente (el transmisor foxboro 823 DP) proporcione una señal de 4 a 20 mA. La necesidad de que esto se realice, radica en que la función que maneja el control PID, internamente opera la variable de entrada como un porcentaje de 0 a 100%.

Luego se deberá utilizar esta función de entre unas cuantas de control PID. La razón de utilización de esta función, es que da la opción de ingresar desde el programa los valores de las constantes del proceso ( $K_p$ ,  $T_I$ ,  $T_D$ ), con la alternativa de configurar programáticamente las variables propias del control, como tiempos muertos, tiempos de muestreo, etc. La figura 5.14 muestra el VI configurado para el control PID.

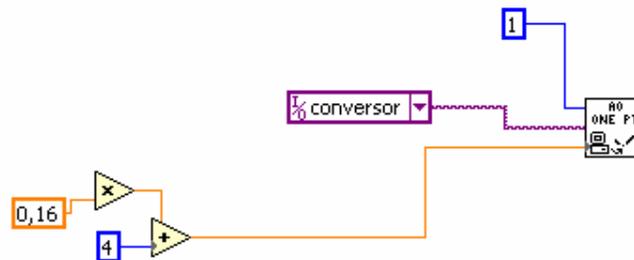


**Figura 5.14 Control PID configurado en el proceso.**

En la figura anterior, es necesario configurar programáticamente los parámetros especificados. La entrada de Set Point es solamente un control que estará programado de 0 a 100, el rango de salida es un cluster en donde se verifica los parámetros requeridos de salida (0 a 100%), el control de automático y manual, no es más que un interruptor que selecciona el modo de operación, en automático se controla el VI por sí sólo, mientras que en manual, las salida del VI depende del valor del control conectado a esta salida. El setpoint range, sirve para definir los límites del set point hacia arriba y abajo, y finalmente las constantes del control poseen un cluster que determina los valores de  $K_p$ ,  $T_I$  y  $T_D$ . Adicionalmente reset, resetea la operación del VI y  $dt(s)$  configura la sincronización del muestreo y la operación del VI.

En la salida, el software utiliza una función de DAQ simple, pues al ser actualizada por el control PID un solo punto a la vez, no es necesaria ni

configuración especial del canal ni uso del buffer de almacenamiento. En este punto se debe tener el cuidado de utilizar el dispositivo disponible (device 1) y el canal de salida configurado (conversor). Finalmente es necesario que se realice otra vez un conversor de niveles a la salida para obtener los niveles de corriente requeridos. La figura 5.15 muestra los procedimientos en éste párrafo mencionado.



**Figura 5.15 Configuración del proceso de salida.**

Con las mencionadas configuraciones, se arma todo el diagrama de bloques considerando, como ya se había dicho, en primer lugar la etapa de entrada, con el VI de adquisición simple y el conversor de niveles adecuado. En segundo lugar el control PID, que necesita una entrada convertida en porcentaje para procesar cualquier parámetro, la configuración programática de las constantes del proceso y rangos de operación, tanto de entrada como de salida. Finalmente, en la tercera parte, se debe utilizar la función de adquisición de datos de salida simple y el escalamiento de la señal proveniente del control PID, ya que al ser ésta una señal obtenida para cualquier proceso, es decir una señal porcentual de 0 a 100%, se la debe acondicionar a una que presente en corriente, no solo para 0 a 20 mA como opera el módulo, sino más bien para una señal de 4 a 20 mA, que es la que necesita el conversor I/P.

Luego de operar los parámetros descritos de las formas indicadas, el panel frontal del VI construido se presenta de la forma que se ilustra en la figura 5.16.

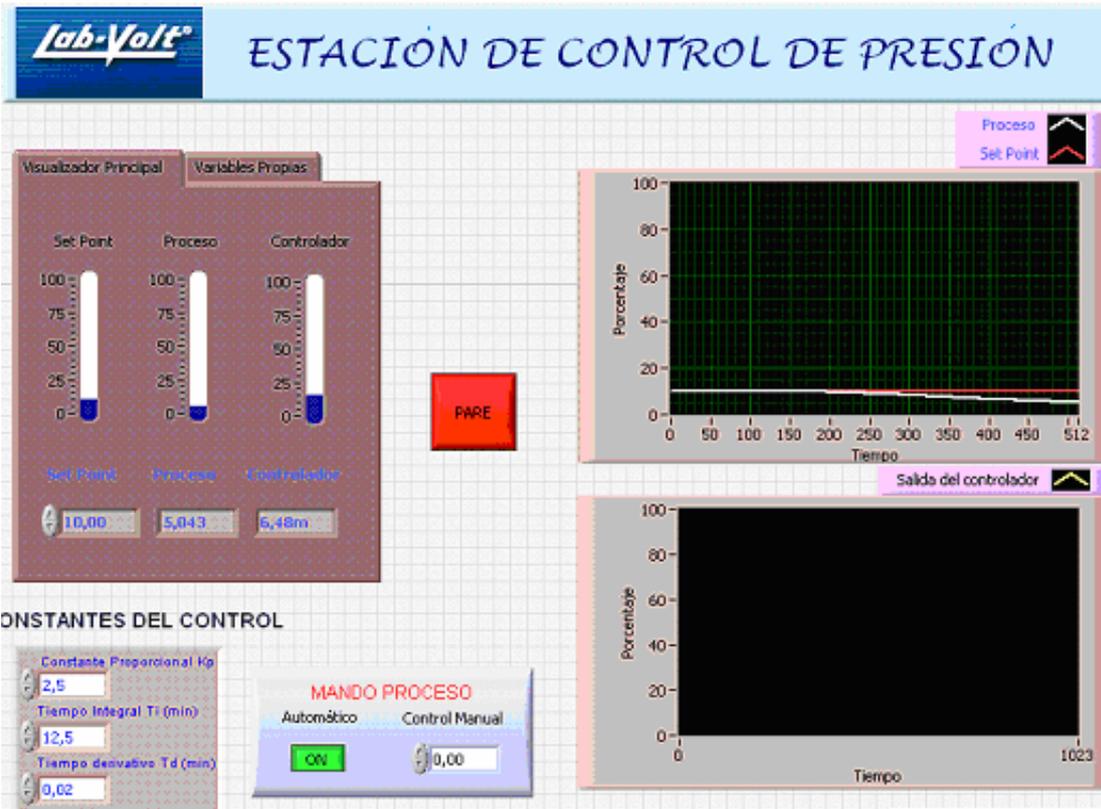


Figura 5.16 Panel Frontal del Control del Proceso de Presión.

## 5.4 DISEÑO DEL CONTROL DEL PROCESO DE TEMPERATURA.

### 5.4.1. Requisitos funcionales.

El sistema de control de temperatura, permite mantener constante la temperatura de un horno de control por niquelina, para ello se sirve de un elemento controlador electrónico, dos transmisores de temperatura foxboro 4726 y omega 2123. El primero utiliza como elemento primario de medición un termopar tipo J (hierro-constantan) e internamente hace el acondicionamiento a una señal estándar de 4 a 20 mA, que se utilizará como entrada al controlador. El segundo utiliza como elemento primario de medición una RTD Pt100, y la linealiza a una red de 4 a 20 mA dentro de él. El actuador en esta estación es solamente un control de accionamiento de varios triacs que actúan solamente en base a un control de fase directo, que controla la potencia media suministrada en forma proporcional a la

niquelina por medio de la señal estándar de 4 a 20 mA entregada por el controlador.

En estas condiciones, nuevamente se observa la necesidad de contar con un dispositivo controlador, al que se le puedan introducir señales de entrada de corriente en un rango de 4 a 20 mA. Y que tenga la posibilidad de proporcionar una salida de las mismas características para alimentar al control de disparo de los triacs en el mismo tipo de red.

Bajo estas consideraciones, se utilizará un PLC Siemens Simatic S7-200, CPU 224, con un módulo de expansión EM 235; el mismo que presenta una CPU que realiza el control de una manera muy eficiente y puede comunicarse con un PC a través de una interfase RS485, suministrando un procedimiento de conversión 232/485, que habla Modbus como protocolo de comunicaciones básico en red y tiene la posibilidad de comunicarse a través de servidores de OPC. Estas características hacen que el dispositivo antes mencionado sea el elemento ideal para realizar el control.

Como se puede haber deducido, el problema fundamental de utilizar el PLC referido es que, al no ser un dispositivo de la marca National Instruments, las comunicaciones y las interfaces de utilización deberán ser desarrolladas por el usuario, cuestión que resulta ser bastante probable, ya que en el software LabVIEW es posible someterse a un driver especial conocido como el Lookout protocol drivers y comunicarse a través de él con el PLC. Otra alternativa es la de manejar OPC control en LabVIEW, Datasocket Manager o por último la comunicación mediante VISA por el puerto serial; ésta tiene la limitación de que se deberá conocer absolutamente la trama de datos enviados por el PLC para poder “entender” el significado de los bits recibidos.

Por otro lado la ventaja del PLC es que es un dispositivo autónomo y no necesita elementos adicionales para realizar el control, únicamente es necesario la utilización del software MicroWin Step7, que lamentablemente en sus versiones más bajas no contiene las librerías de manejo de protocolos Modbus. Es por esta

razón que se utiliza una versión Beta de Internet que posee todos los recursos pero tiene un tiempo de utilización limitado.

#### 5.4.2 Implementación del Hardware.

La implementación del hardware de control a través del PLC no es una cuestión por demás complicada, consiste en únicamente determinar el número de entradas y salidas análogas a utilizar. Para el presente caso se utilizará una sola entrada en modo analógico con un valor de 4 a 20 mA y una sola salida de igual rango.

El módulo de expansión de I/O análogas del PLC EM235, proporciona la facilidad de uso de 04 entradas análogas, configurables como voltaje o corriente según la posición de los interruptores de un dip switch incorporado en el módulo, y la conexión especial en los tres terminales de entrada del módulo para cada señal. La única salida que posee, permite obtener señal en el orden de voltaje o corriente para lo cual (pero no de forma simultánea) posee tres terminales de salida, los cuales determinan si se obtendrá la respuesta en voltaje o corriente. Las conexiones que deberán realizarse se presentan esquemáticamente en la figura 5.17. y en la tabla 5.4, se muestra la posición de los dip switches para configurar las entradas en corriente de 0 a 20 mA.

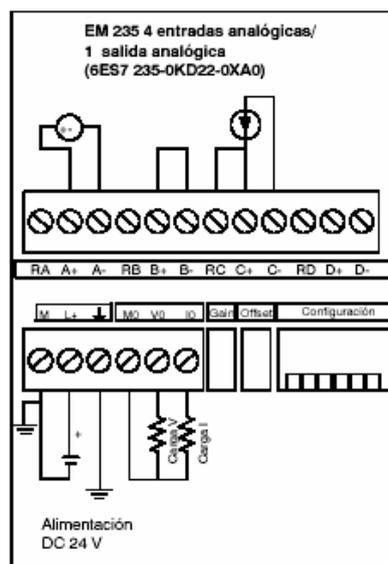


Figura 5.17 Conexión de las entradas y salidas análogas en el módulo EM235

Unipolar						Rango de tensión	Resolución
Int1	Int2	Int3	Int4	Int5	Int6		
ON	OFF	OFF	ON	OFF	ON	0 a 50 mV	12,5 $\mu$ V
OFF	ON	OFF	ON	OFF	ON	0 a 100 mV	25 $\mu$ V
ON	OFF	OFF	OFF	ON	ON	0 a 500 mV	125 $\mu$ V
OFF	ON	OFF	OFF	ON	ON	0 a 1 V	250 $\mu$ V
ON	OFF	OFF	OFF	OFF	ON	0 a 5 V	1,25 mV
ON	OFF	OFF	OFF	OFF	ON	0 a 20 mA	5 $\mu$ A
OFF	ON	OFF	OFF	OFF	ON	0 a 10 V	2,5 mV
Bipolar						Rango de tensión	Resolución
Int1	Int2	Int3	Int4	Int5	Int6		
ON	OFF	OFF	ON	OFF	OFF	$\pm$ 25 mV	12,5 $\mu$ V
OFF	ON	OFF	ON	OFF	OFF	$\pm$ 50 mV	25 $\mu$ V
OFF	OFF	ON	ON	OFF	OFF	$\pm$ 100 mV	50 $\mu$ V
ON	OFF	OFF	OFF	ON	OFF	$\pm$ 250 mV	125 $\mu$ V
OFF	ON	OFF	OFF	ON	OFF	$\pm$ 500 mV	250 $\mu$ V
OFF	OFF	ON	OFF	ON	OFF	$\pm$ 1 V	500 $\mu$ V
ON	OFF	OFF	OFF	OFF	OFF	$\pm$ 2,5 V	1,25 mV
OFF	ON	OFF	OFF	OFF	OFF	$\pm$ 5 V	2,5 mV
OFF	OFF	ON	OFF	OFF	OFF	$\pm$ 10 V	5 mV

**Tabla 5.4 Configuración de los interruptores del EM 235 para selección del rango de entradas**

### 5.4.3 Implementación del software de control.

Como se mencionó anteriormente, al ser el PLC un dispositivo de control autónomo, no es necesario ningún software para mantener un programa interno funcionando conectado a una PC, lo que si se debe hacer es: realizar el programa de control, en diagrama Ladder, AWL o KOP (según el tipo de PLC), para ello en el caso del Siemens Simatic S7-200, se utiliza el software de control Step 7 MicroWin, en el cual se escribirán todas las rutinas de control que se desean ejecutar el PLC, luego se las descargarán utilizando un elemento utilitario del mismo software y finalmente, ya sea desde el software o desde el mismo PLC, se correrá la aplicación, pudiendo en ese momento desconectarse la PC sin perjuicio de ejecución del programa en el autómatas.

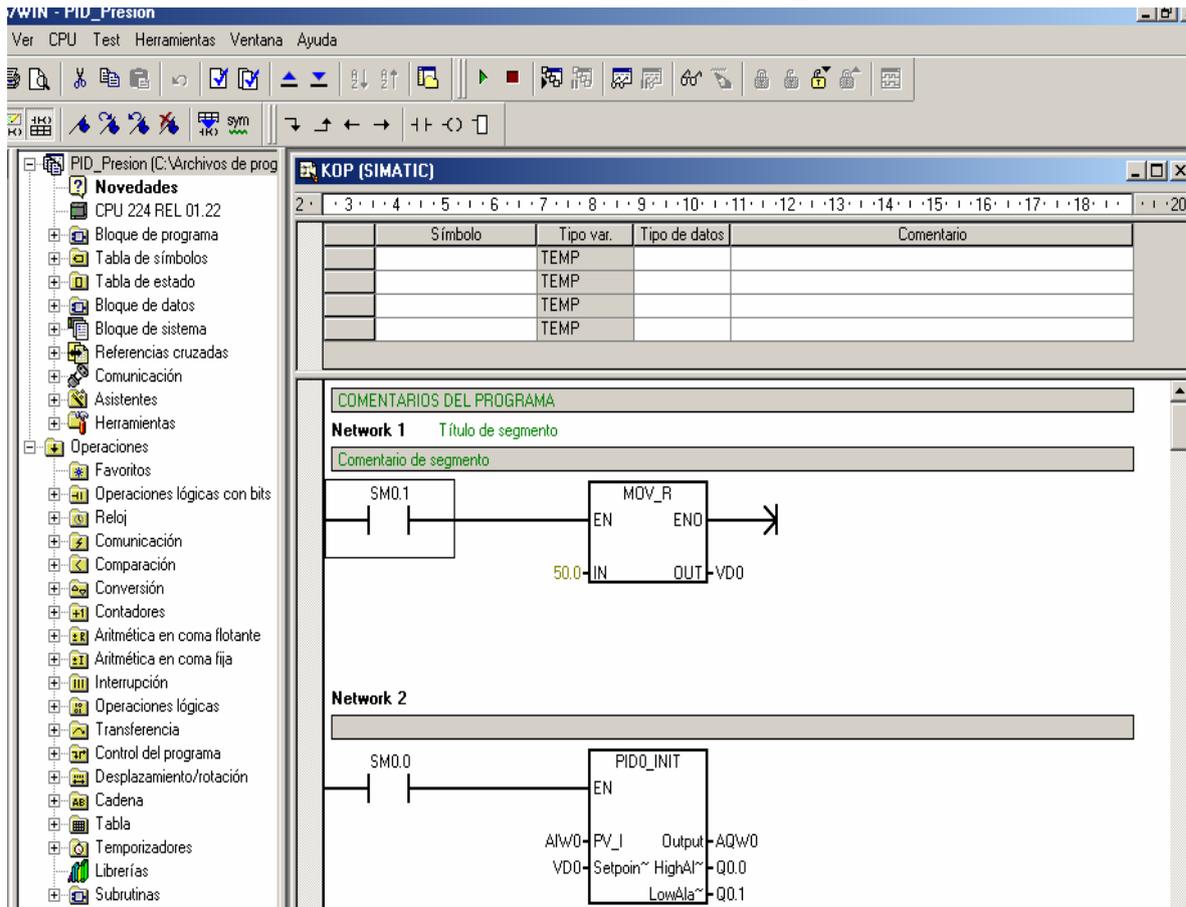
En MicroWin existe un asistente de configuración del control de tipo PID, pues los registros y las operaciones de orden matemático superior, que usa, las realiza el PLC en forma transparente al usuario. Incluso el momento de que el programa

corre dentro del PLC en Step7, se pueden monitorear las entradas, salidas y registros para observar si se encuentran programados de manera correcta.

El programa diseñado para el control de la estación de temperatura utiliza la entrada analógica AIW0, que se accede mediante los terminales RA, A+ y A-. Y la salida analógica AQW0 (única) utiliza los terminales M0, I0. Adicionalmente se ha programado la utilización de dos salidas de relé para control de alarmas en modo de encendido y apagado de altavoces o luces; estas salidas son DO0 y DO1 manejadas por los terminales 1M, 0.0 y 0.1.

En el programa es muy importante definir la tabla de vectorización de los datos de entrada y de salida del PLC, pues en el momento de definir las características del control PID se establecen las nombradas direcciones, esto no sería importante si se monitoreara el PLC con algún software de Siemens (como WinCC por ejemplo), pero en este caso se va a utilizar un Servidor de OPC para comunicarse con una aplicación OPC cliente (en este caso LabVIEW), y este servidor de OPC (TopServer) necesita la tabla de vectorización del PLC para la definición, control y operación de los Tags encadenados a ellos.

A continuación, en la figura 5.18, se puede observar el programa de control descargado en el PLC para su funcionamiento conjunto con el transmisor de termocupla como entrada, y la salida para el módulo de disparo de los triacs.



**Figura 5.18 Programa de control para el PLC Siemens Simatic S7-200**

#### 5.4.4 Implementación de la Interfase de comunicación por servidor de OPC.

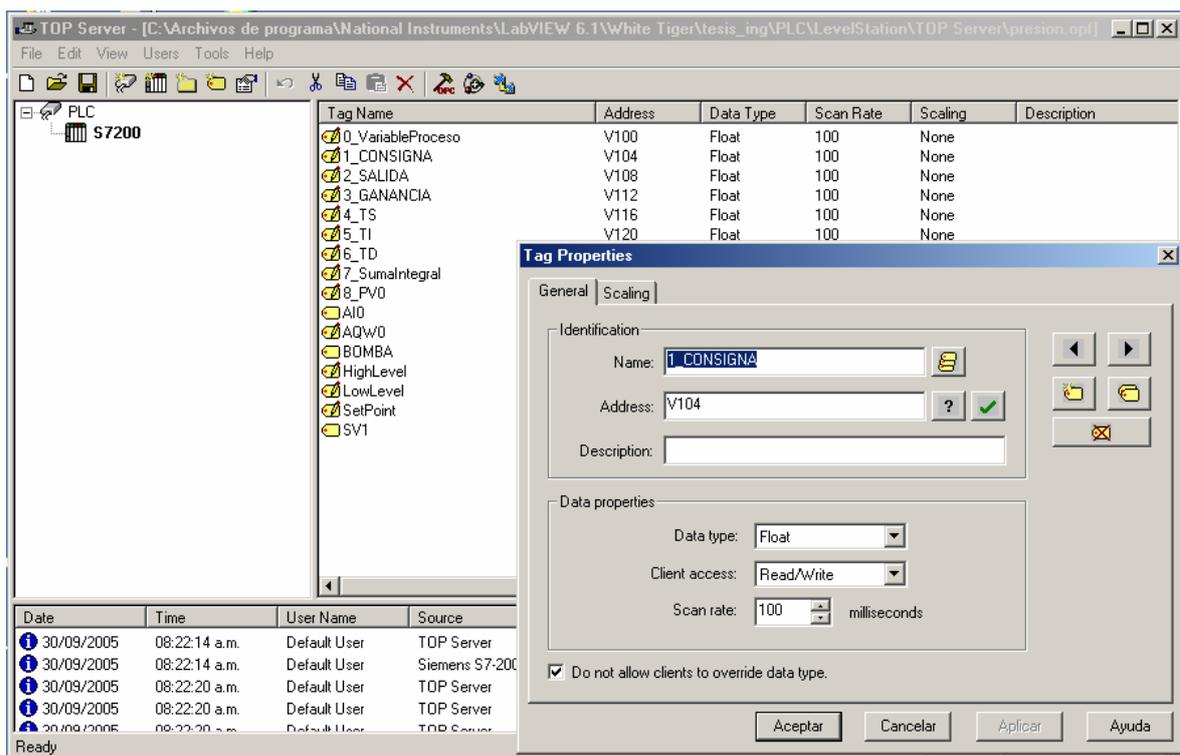
Como se había mencionado anteriormente, el PLC funciona en forma autónoma, y se lo puede monitorear desde el software MicroWin, pero no se pueden cambiar los parámetros de control. El programa de operación del PLC fue completamente descargado y corre sin modificaciones. Para efectos de control y supervisión es necesario que algunos parámetros de dicho control puedan ser cambiados, así por ejemplo: el set point, las constantes de control Kp, TI, TD, etc.

Estos cambios serían fácilmente realizables al poseer los drivers de comunicación entre el PLC y su SCADA, pero como se recordará, la supervisión no se la realiza con WinCC, sino a través de LabVIEW. Es necesario entonces establecer otro tipo de comunicación.

En la actualidad, la tendencia de comunicaciones entre sistemas de diferente protocolo propietario, está siendo llevada a cabo, y muy eficientemente a través

de herramientas DEE, ActiveX, COM y OPC. En esta estación se utiliza la comunicación por OPC. Para el efecto se sirve del software de aplicación TopServer, que configura etiquetas de operación para los parámetros de control que se desea monitorear o cambiar.

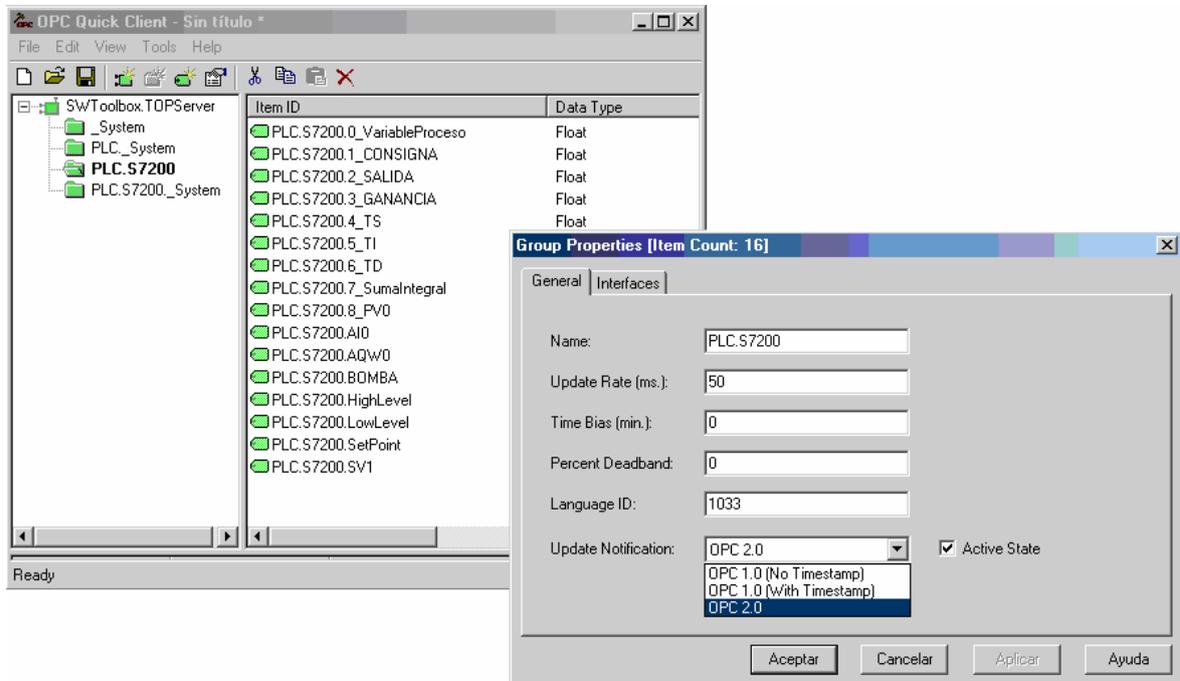
En este caso, se han establecido varias etiquetas para el control como son: el Set Point del Proceso, la salida del controlador (es decir la respuesta PID), el proceso de temperatura (es decir la entrada desde el transmisor) y las constantes como: la ganancia, el tiempo de muestreo, el tiempo integral, el tiempo derivativo, y algunas otras de supervisión. Estos Tags se generan y configuran de tal forma que sean compatibles en: formato, orden y dirección con la tabla de datos del PLC, tal como se muestra en la figura 5.19.



**Figura 5.19 Configuración de los Tags en Top Server**

Luego de haber realizado este procedimiento, deberá establecerse la comunicación entre los dos dispositivos y cargar el servidor de la aplicación para el cliente que deberá tener la opción de manejo de OPC (Intouch, LabVIEW DSC, RSview, CX-Supervisor). Esta situación se ilustra en la figura 5.20. Se debe tener

mucho cuidado con que el servidor de OPC no esté corriendo junto con el monitoreo del PLC por MicroWin, pues existen conflictos en la comunicación si es simultánea.



**Figura 5.20 Configuración de comunicaciones OPC en TOP Server.**

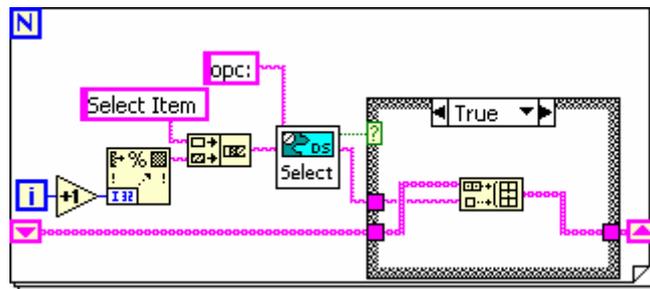
Se debe desarrollar la aplicación cliente y conectarla. En LabVIEW hay varias opciones para ello.

#### **5.4.5 Implementación de la aplicación de cliente en LabVIEW.**

Como se mencionó anteriormente, existen varias alternativas para comunicarse desde LabVIEW con el PLC, una de ellas es de utilizar VISA para monitorear la entrada y la salida del puerto serie, que es por donde se comunica el PLC con su software de monitoreo; al hacerlo de ésta forma, se deberá conocer absolutamente toda la trama de datos que envía el PLC, estos datos no están disponibles en ningún manual de usuario. Otra opción es comunicarse mediante OPC con los drivers de Lookout Protocol Drivers, pero en este caso se va a producir un conflicto entre la comunicación del control del DCS y el SCADA realizado con el módulo DSC ya que el OPC del PLC no existe en Lookout

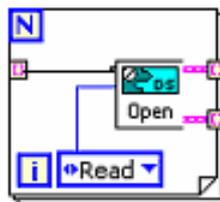
protocol drives. Es decir, la alternativa más viable es crear una conexión de tipo datasocket con el servidor de OPC, que en este momento ya consiguió la comunicación con el PLC.

Al abrir la conexión por datasocket, es necesario determinar la imposibilidad de abrir una comunicación con varios tags de un solo espacio de comunicación, es por esto que se lo debe hacer en un lazo seleccionando los tags que se ha de utilizar. Esta situación se muestra en la figura 5.21.



**Figura 5.21 Selección de la conexión Datasocket.**

Luego de seleccionar los tags de comunicación se abre para cada uno de ellos el recurso de comunicaciones y se monitorea o cambian los valores entre el cliente y el servidor, haciendo que ésta realmente sea entre el PLC y LabVIEW, como se muestra en la figura 5.22.



**Figura 5.22 Apertura del recurso de comunicaciones entre Cliente y Servidor OPC**

Para finalizar, se despliegan los resultados que están encapsulados, es por esta razón que es necesario manipularlos y desencapsularlos en arreglos de diferentes longitudes para luego visualizarlos. Este procedimiento se muestra en la figura 5.23.

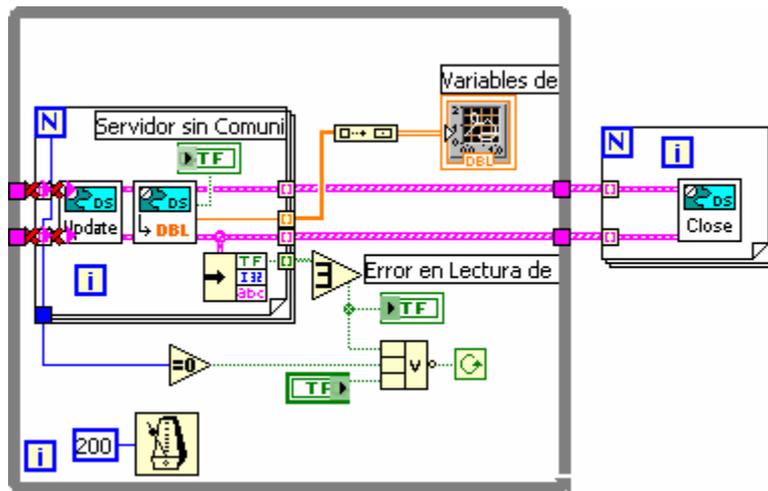


Figura 5.23 Desencapsulado y presentación de los datos.

Finalmente, se construye el panel frontal para que la interfase gráfica de usuario sea la que establezca la correcta intercomunicación entre el usuario y su HMI. Esta se desarrolló tal como indica la figura 5.24



Figura 5.24 Panel frontal del control del Proceso de Temperatura

## **5.5 DISEÑO DEL SISTEMA DE SUPERVISIÓN Y REGISTRO.**

De la misma forma que se establecieron los requisitos funcionales en el diseño de los sistemas de control de cada una de las estaciones, para la integración total también se necesitan establecer los requisitos que se indican a continuación.

### **5.5.1 REQUISITOS FUNCIONALES.**

El sistema deberá configurar y manejar por lo menos tres tags por módulo correspondientes a la variable del proceso (medida por los transmisores), la salida del controlador (respuesta del control matemático de LabVIEW y el Set Point).

Para ello, en cada uno de los procesos se configuraron canales virtuales para cada uno de los dispositivos empleados, situación que se describe en los párrafos siguientes.

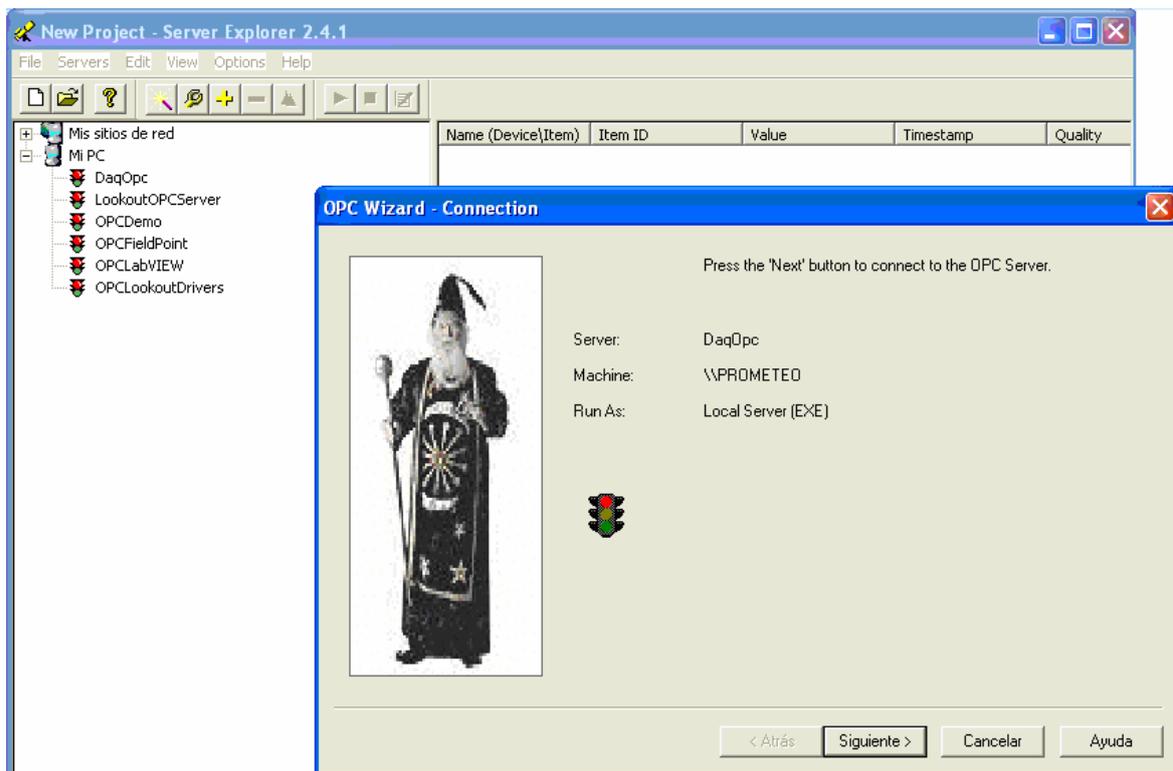
#### **5.5.1.1 Estación de Presión.**

En primer lugar, para la estación de presión que utiliza una tarjeta de adquisición de datos, se configuran los canales virtuales de entrada y salida del proceso: canal de entrada análoga 08 y canal de salida análoga 01. Tal como se había ilustrado en la sección 5.3, la configuración de un canal virtual de DAQ se lo realiza por MAX.

Para comprobar que los canales creados se están comunicando adecuadamente se utilizará el módulo de DSC. Se levantará el asistente de comunicaciones OPC desde Inicio >> todos los programas >> National Instruments >> Server Explorer. Cuando se abra este asistente, se deberá seleccionar DaqOPC e inmediatamente presionar el botón Wizard, de esta forma se levanta el asistente de comunicaciones OPC.

Se puede observar también que se despliega el nombre del computador en el cual se trabajará estableciendo los Tags y determinando la comunicación. Se deberán

dejar los nombres predeterminados en Group Name y Update Rate y se continuará con el procedimiento; automáticamente se añaden todos los canales virtuales configurados en MAX. Los datos de estos canales virtuales están disponibles en el servidor DaqOpc para ser manejador por cualquier servidor OPC (no necesariamente puede ser LabVIEW DSC, podría ser Intouch, RSview, CX-Supervisor, etc). En la figura 5.25 se puede visualizar el mencionado procedimiento.



**Figura 5.25 Asistente de Comunicaciones con DaqOpc.**

En el mismo asistente se puede comprobar el funcionamiento de los diferentes ítems OPC configurados y que están sirviendo para la comunicación. De esta manera se confirma que la tarjeta está usando el servidor OPC para leer y escribir los datos en los canales.

### **5.5.1.2 Estación de nivel.**

Para la estación de nivel, que es controlada a través de los módulos AI210 y AO200 del Field Point FP1000, el procedimiento es sumamente parecido al de la configuración anterior, con la diferencia que el asistente de comunicaciones que se deberá levantar es OPCFieldPoint.

Los pasos a realizar en la mencionada configuración son: Configurar en MAX el canal virtual de Field Point. Si los dispositivos que están conectados al FP1000 no aparecen en la pantalla del asistente, se debe presionar el botón Find Devices y se visualizarán todos los módulos conectados.

Seleccionar el módulo FP-AI\_210@1 para configurar sus entradas en el sistema estándar de 4 a 20 mA. De igual forma se selecciona el módulo FP-AO-200@1 para configurar las salidas en el estándar de 4 a 20 mA. Este procedimiento se describió ya en el control de nivel en la sección 5.1.

De igual forma que en el caso anterior, se debe levantar el Server Explorer y seleccionar ahora OPCFieldPoint, y luego el asistente Wizard. Se puede dar un nombre cualquiera al grupo de elementos que se han configurado y un Update rate (se recomienda no cambiar los valores predeterminados).

Finalmente desde el Server Explorer y de manera similar a la anterior, se puede comprobar el funcionamiento de los módulos solamente cambiando en el dispositivo a las entradas y visualizándolas en el asistente. Verificar que en la parte derecha del asistente se lea Value good, para confirmar la comunicación OPC.

### **5.5.1.3 Estación de flujo.**

Para la estación de flujo, la situación es sumamente parecida a la estación de presión, ya que ella está controlada con una tarjeta de adquisición de datos por puerto USB. La diferencia fundamental radica en que al momento de configurar los canales virtuales dentro de MAX, para la tarjeta USB-6009 no se utiliza el

Traditional NI-DAQ Virtual Channel, sino NI-DAQmx Task, por todo lo demás la configuración es exactamente igual. Es decir, configurado el canal virtual, que en este caso es un canal de entrada análoga de voltaje y para la salida otro canal análogo pero de salida en voltaje.

La comunicación por OPC se realiza de la misma forma: levantando el Server Explorer; seleccionando DaqOpc, y presionando Wizard; luego visualizando y probando los canales virtuales de DAQmx configurados.

Es recomendable no cambiar el Group name y el Update rate que se visualizan por defecto, pues todo el grupo de comunicaciones se está concatenando para una sola aplicación.

#### **5.5.1.4 Estación de temperatura.**

Para la estación de temperatura hay que recordar que el control no reside realmente en LabVIEW, ni en un dispositivo hardware especial NI. El control lo realiza independientemente de la plataforma de National Instruments, un PLC Siemens Simatic S7-200.

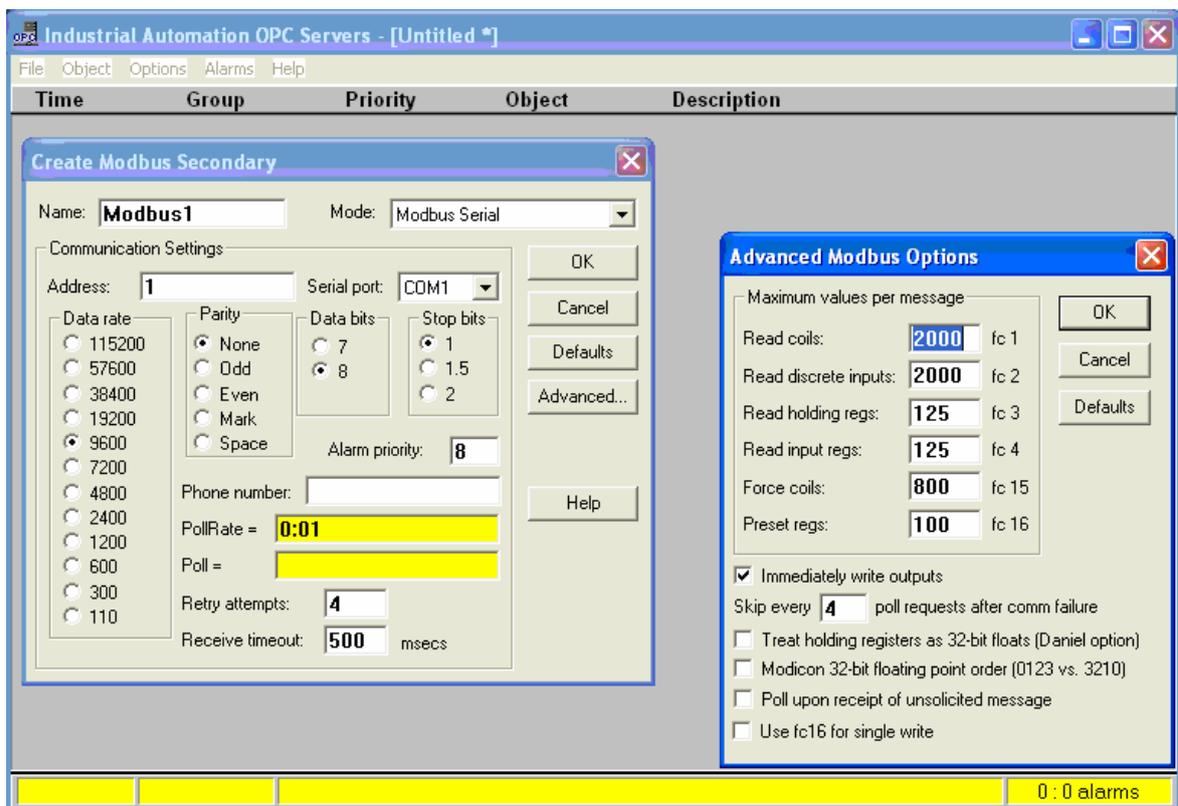
En este contexto, la configuración de los canales virtuales por MAX ya no es necesaria, mas bien se utilizarán los LookoutOPCServer. El procedimiento se describe a continuación.

El control descrito en la sección 5.4. monitoreaba al PLC a través de una conexión Datasocket, pero se había mencionado que dicho monitoreo se podía realizar por muchos otros medios, uno de esos medios es utilizar el módulo DSC y los Lookout OPC Servers; que son utilitarios generalmente concebidos para Lookout pero que han sido adaptados para BridgeVIEW y LabVIEW DSC.

El utilitario LPD (Lookout Potocol Drivers) tiene la ventaja de que está diseñado para proporcionar drivers y comunicarse con muchos PLC, entre ellos los drivers de comunicación con PLC de Siemens Simatic existen, pero lamentablemente no para la versión S7-200, ni para la CPU que se ha utilizado (CPU 224).

Lo que sí es relevante saber es que el PLC Siemens Simatic S7-200 habla protocolo Modbus, entonces simplemente es necesario saber como se comunica Modbus y adaptarlo a nuestra necesidad. Los pasos para configurar la aplicación son los siguientes.

Para realizar las siguientes configuraciones, se debe poseer las licencias de los drivers en Industrial Automation Server, sin ellos no puede ser posible realizar la comunicación. En la figura 5.26 se presenta la pantalla para la selección IAS.



**Figura 5.26 Selección del protocolo de I.A.S**

Finalmente, para comprobar que la comunicación OPC está perfectamente establecida, se debe levantar nuevamente el Server Explorer y seleccionar ésta vez OPCLookoutDrivers. Se añadirán automáticamente las localidades de memoria del PLC que se seleccionaron y se puede comprobar la comunicación OPC (Value Good) cambiando los valores de las entradas análogas y la salida análoga del PLC, que según la configuración son las localidades LV100 y LV 150.

### 5.5.2 CONFIGURACIÓN DE TAGS.

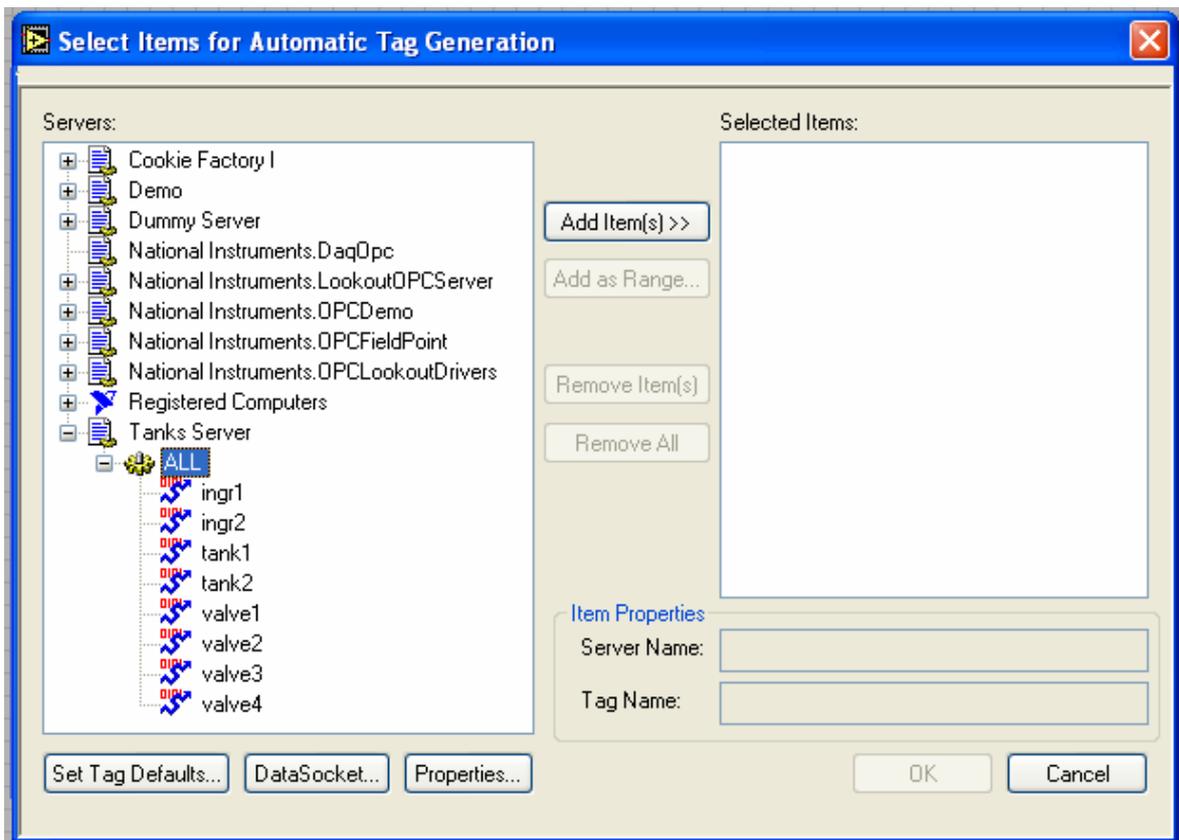
La configuración de todos los Tags utilizados es la misma, pues al estar los ítems de cada dispositivo conectados, solamente se les debe asignar las etiquetas sobre las cuales actúan y monitorearlos. Desde LabVIEW hay que cargar el TAE (Tag Configuration Editor).

El procedimiento para hacerlo puede ser, desde el Wizard o utilizando los íconos Edit/Create/Copy Tag. En este sistema se define la información sobre los atributos, el escalamiento, las alarmas y los niveles de acceso que tiene cada etiqueta. La configuración realizada se almacena en un nuevo archivo de extensión .scf, el cual sirve para utilización de cualquier software de SCADA. Solamente es posible cargar un archivo de configuración a la vez, de tal modo que se debe tener en cuenta esta situación y almacenar todos los tags necesarios en un solo archivo.

En este caso se han configurado dos tags, uno de entrada y otro de salida para cada estación. Los tags de entrada representan a la medida de la variable que está obtenida por diferentes tipos de dispositivos: en la estación de nivel se obtienen a través de una de las entradas del módulo FP-AI-210 que corresponde a una señal de corriente de 4 a 20 mA; en la estación de flujo, se obtiene a través de una entrada análoga de voltaje de 1 a 5 V, desde el canal 0 de la entrada análoga del NI-USB 6009.

Para la estación de presión, se la obtiene desde el canal 8, el cual está conectado al módulo SCC-AI20, con portador NI-SCC-2345, zócalo J2, entrada de corriente de 4 a 20 mA; y finalmente en la estación de temperatura solamente se monitorea la localidad de memoria VD100 del PLC Siemens Simatic S7-200, a través de la comunicación Modbus disponible en el PLC; pero físicamente la señal de entrada de corriente de 4 – 20 mA, está conectada a la entrada AWI0.0 del PLC.

Lo dicho anteriormente se muestra en la figura 5.27.

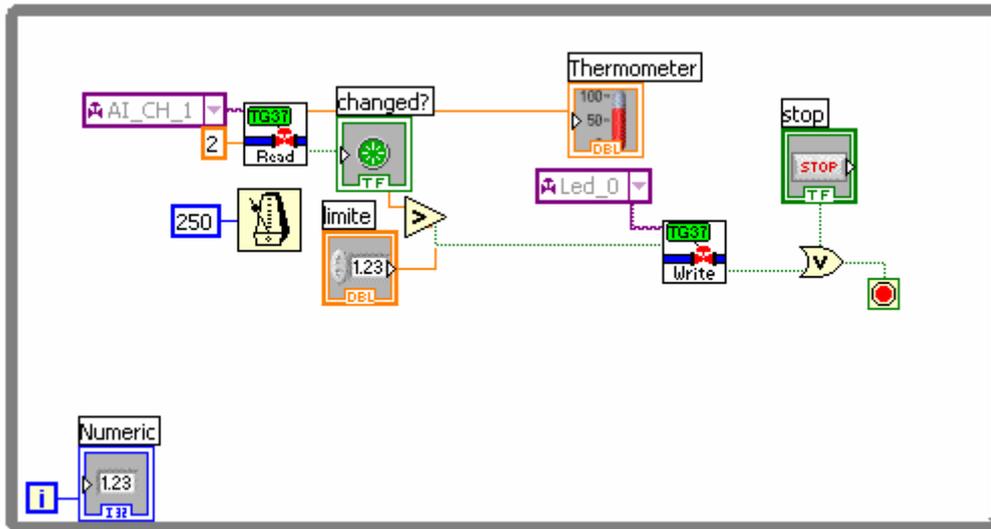


**Figura 5.27 Creación de etiquetas utilizando Wizard**

En la figura anterior se escoge el canal configurado y se le adiciona en el TCE para grabarlos en un solo archivo .scf (SCADA Configurations File). Finalmente se debe monitorear el Tag creado arrancando el Tag Monitor. A partir de ese momento Tag engine ya está registrando los datos de la etiqueta en la base de datos CITADEL.

Luego de ello y para cada estación, se deben generar las lecturas y escrituras, según sea el caso, de los tags configurados. En este caso los principales son la lectura de la variable del proceso y la escritura de la señal del actuador en cada estación.

La figura 5.28 muestra el código generado para la lectura y escritura de tags.

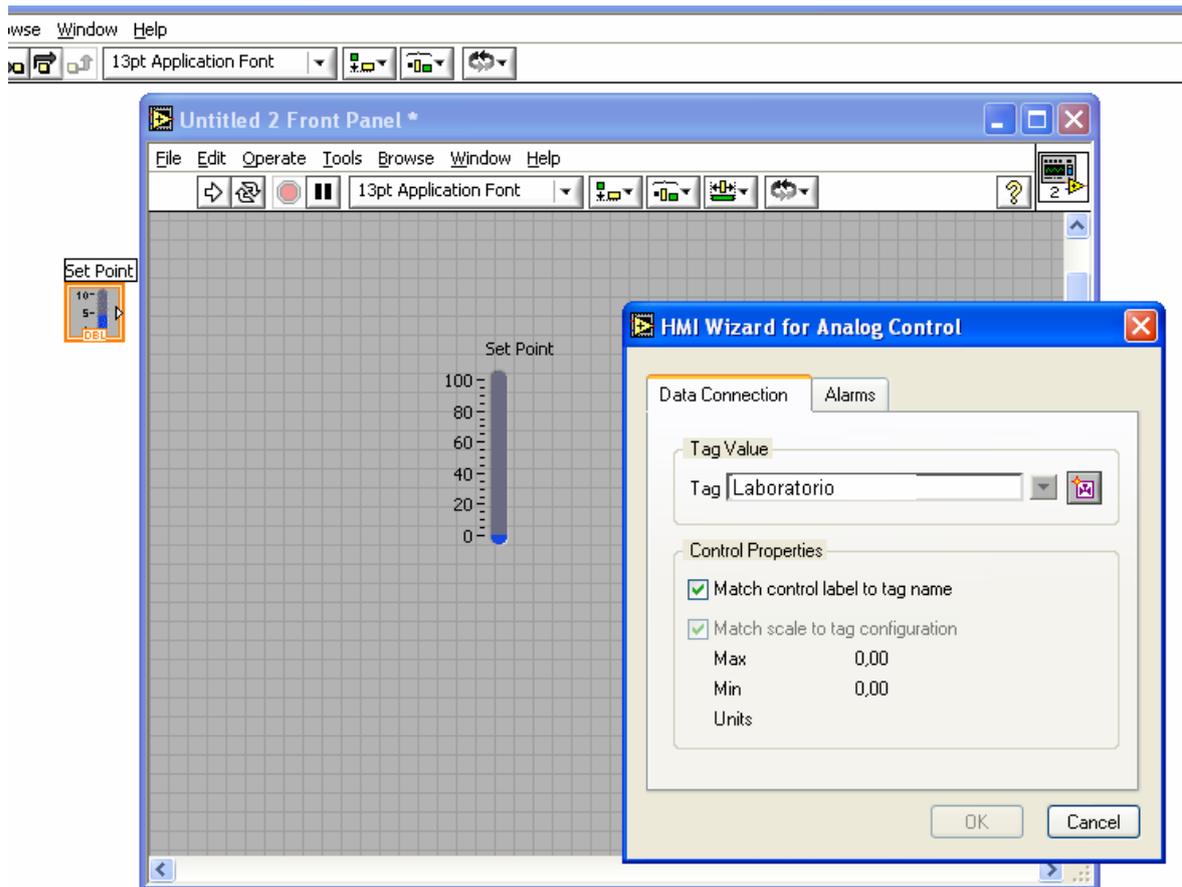


**Figura 5.28 Lectura y escritura de los valores de las tags configuradas en un subVI.**

A continuación será necesario crear el dispositivo servidor para que cualquier cliente (SubVI) pueda utilizar tanto la configuración del archivo .scf como los tags creados a partir de ello, este procedimiento se lo realiza colocando controles e indicadores y asignándolos a las etiquetas creadas.

En la figura 5.29 se muestra la asignación de tags a los elementos del panel frontal.

Los tags ya configurados servirán entonces para configurarlos para el registro de datos, manejo de alarmas, eventos y niveles de seguridad. Esto se realiza a través del TCE. Cabe recordar que en la página principal tiene todos los tags configurados.

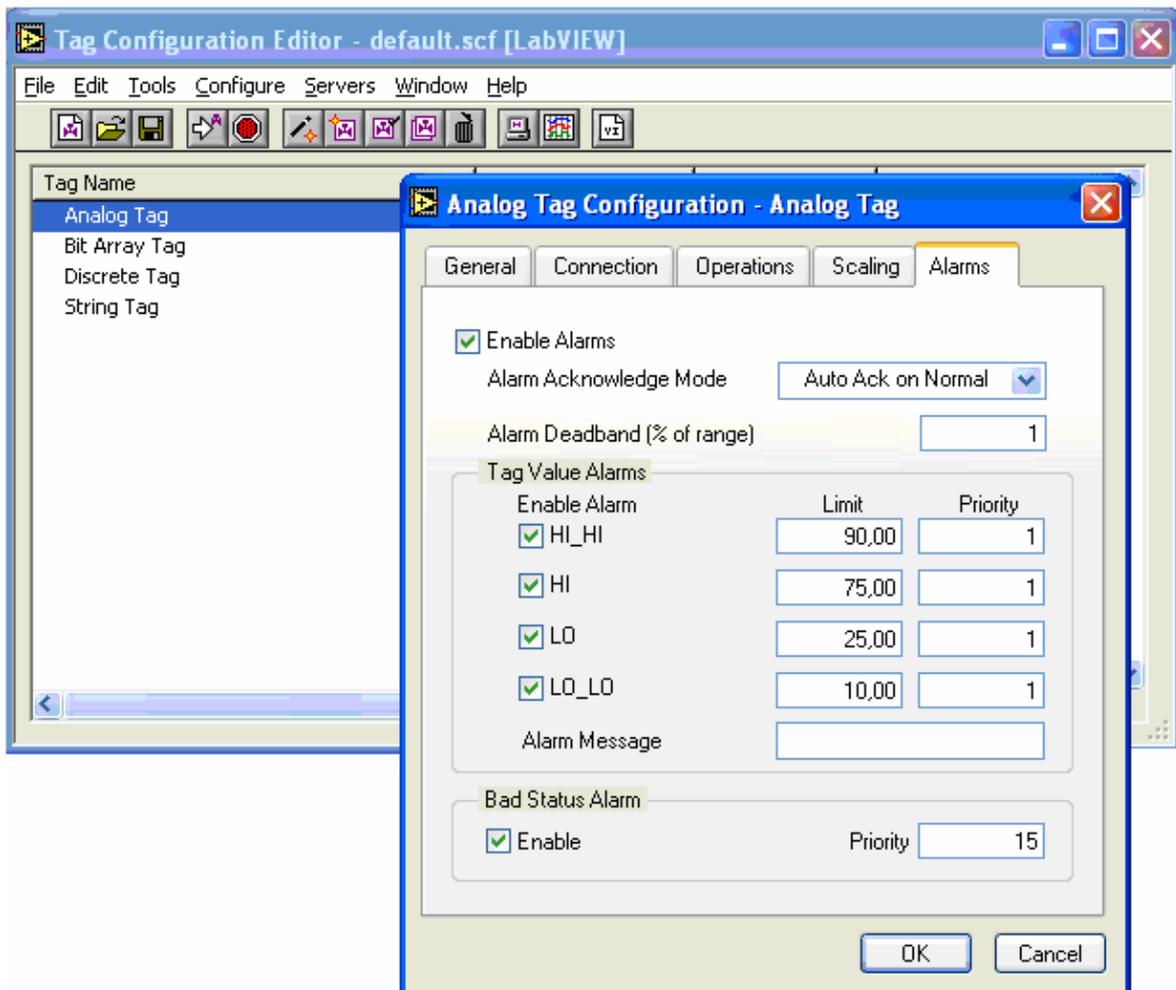


**Figura 5.29 Asignación de los tags a elementos del panel frontal.**

En esta opción del TCE, existe la posibilidad también de configurar los niveles de alarma que se verificarán para dicho tag, ésto se lo realiza en la pestaña de alarmas (Alarms).

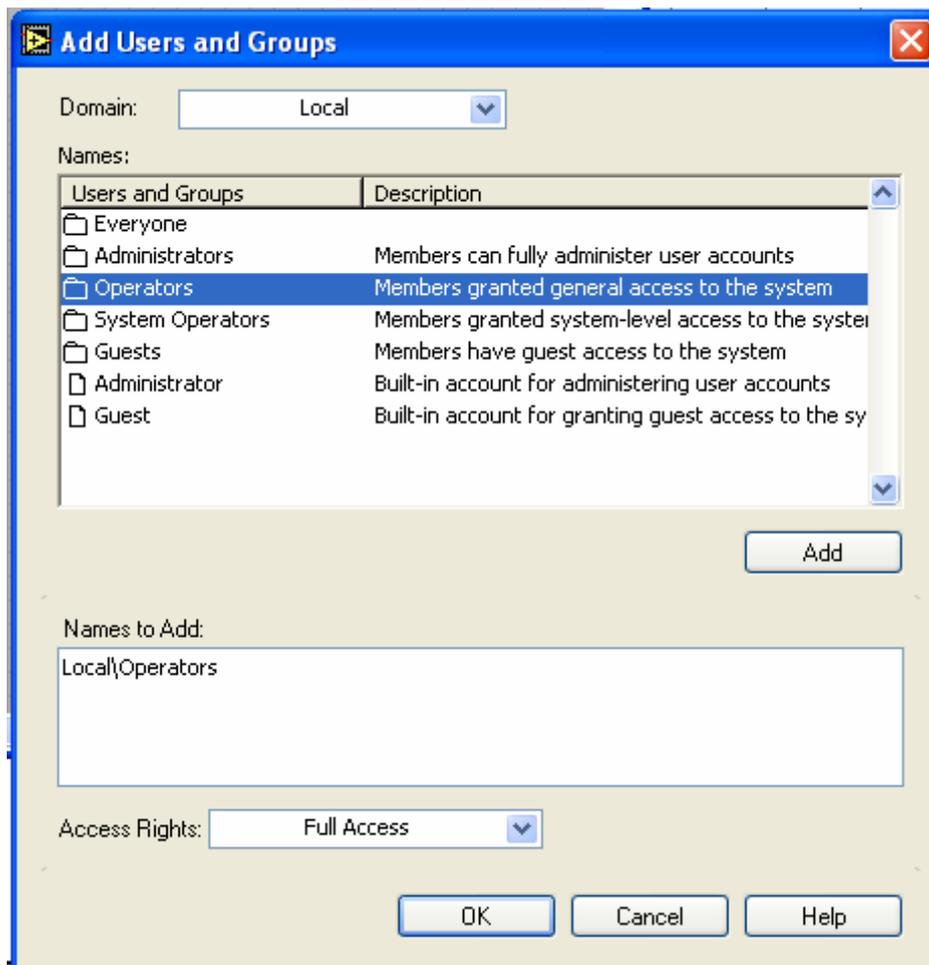
El funcionamiento de las alarmas configuradas se hará en Tag Monitor, escogiendo las etiquetas requeridas y visualizando su valor.

La figura 5.30 indica los procedimientos de configuración de alarmas y seguridad.



**Figura 5.30 selección de alarmas y operaciones.**

Para terminar, en cada servidor se deberá colocar niveles de acceso al sistema supervisorio, ésto se desarrolla comúnmente confeccionando un panel frontal pequeño que sirva para el registro de usuarios. En el menú correspondiente deberá registrarse al nuevo usuario del sistema, un nivel de seguridad específica y el password de entrada en uno de los grupos configurados; tal como se indica en la figura 5.31.



**Figura 5.31 registro de usuarios en los grupos de seguridad existentes.**

Configurados todos estos ítems hay que realizar todos los HMI para el control total de las estaciones y la particularización de cada uno de los paneles de las estaciones, las pantallas de alarmas, eventos, tendencias históricas y registros de usuarios.

La pantalla principal sirve como vínculo de navegación para las cuatro estaciones, al presionar los botones (también configurados en Panel Wizard), se abrirán y correrán todos y cada uno de los programas particulares para las opciones antes mencionadas, haciendo de esta forma que el sistema SCADA resida en un sistema de monitoreo centralizado, pero de control distribuido, con la opción de cambio de los parámetros de control desde los HMI del sistema principal.

Las figura 5.32 muestra el panel frontal de la pantalla principal del SCADA, es aquella que permite la navegación a través del programa.

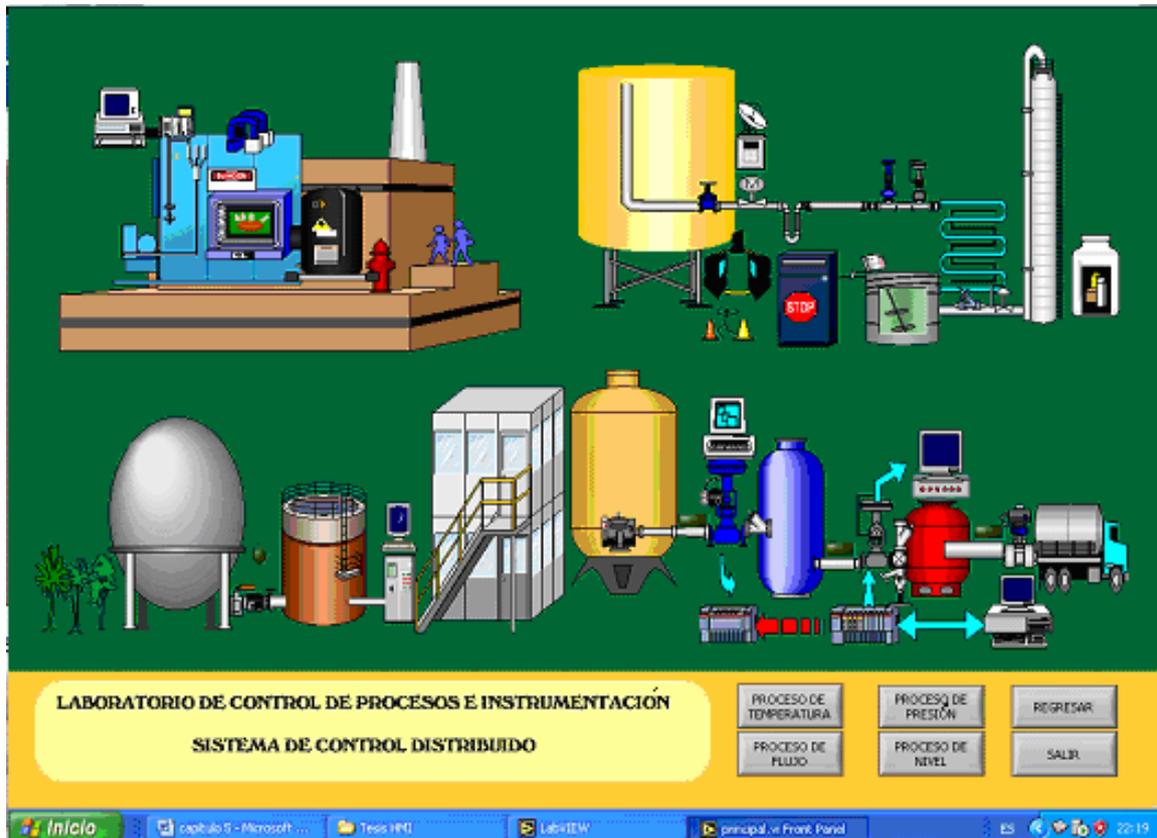


Figura 5.32 Panel frontal del sistema de monitoreo de las estaciones del laboratorio de control de procesos e instrumentación.

## 5.6 DISEÑO DE LA BASE DE DATOS ADMINISTRATIVA (BDD).

En esta sección se implementará y desarrollará una aplicación informática, la misma que se utilizará para enlazar el control DSC en LabVIEW con una Base de Datos Administrativa. La aplicación será desarrollada como FRONT-END: Visual Basic y BACK-END: SQL SERVER 2000.

◆ SOFTWARE:

- Base de Datos CITADEL
- Base de Datos SQL SERVER 2000
- Instalación del ODBC (OPEN DATA BASE CONECTION)

### 5.6.1 Configuración de la comunicación del ODBC y CITADEL

Los datos almacenados en la BDD CITADEL, son accesible a través de una conexión ODBC, una vez que se instala el LabVIEW y se han generado los TAG'S necesarios para realizar el control, se genera automáticamente el ODBC del LabVIEW (National Instruments CITADEL 4 Data Base) como se indica en la figura 5.33:

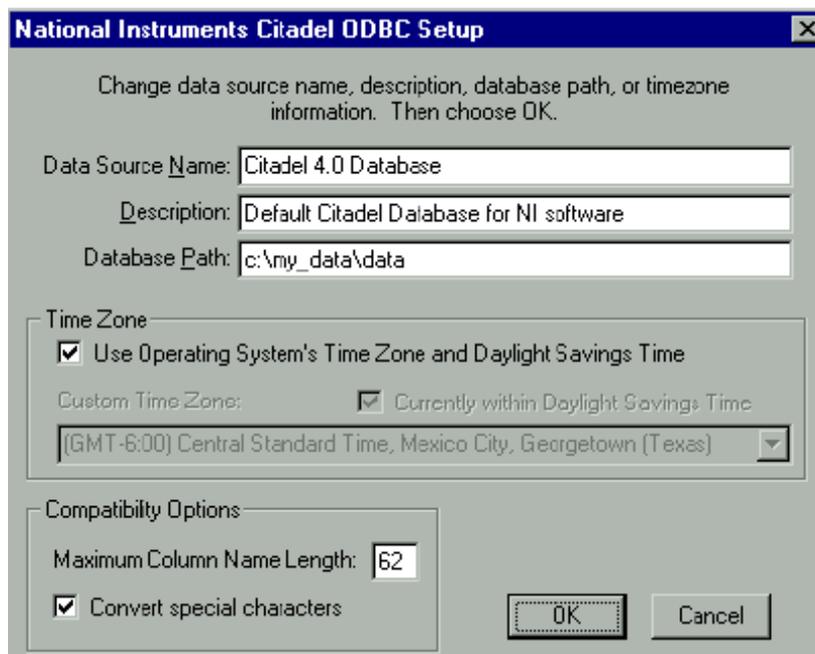
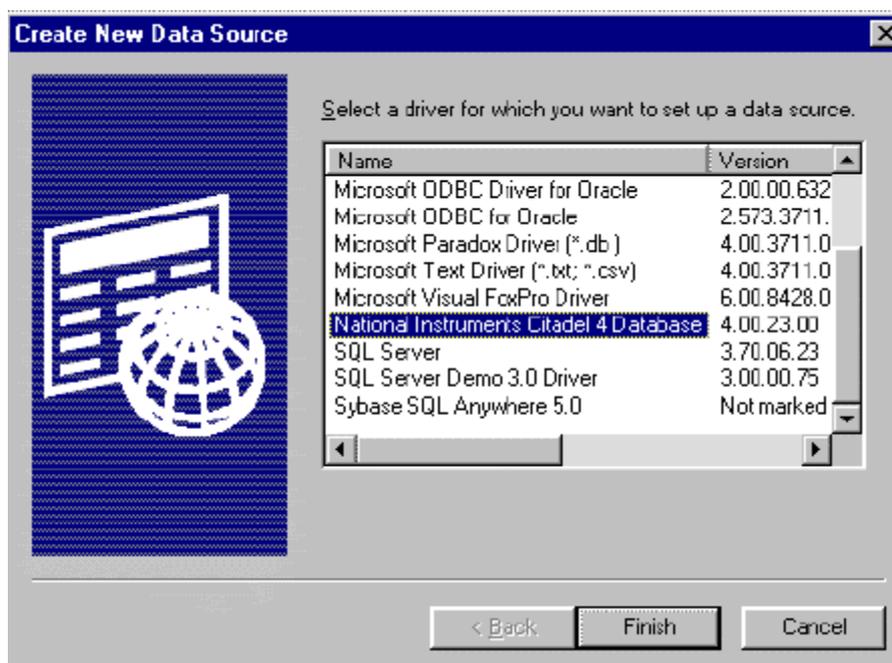


Figura 5.33 Comunicación CITADEL con ODBC

Identificado el ODBC respectivo, se procederá a configurar la comunicación de la Base CITADEL con la Base Administrativa en SQL SERVER 2000 con la finalidad de obtener la información generada por el control DSC; cabe recalcar que la estructura de la Base de Datos Administrativa no es la misma que la BDD CITADEL.

En la figura 5.34 se presenta la caja de diálogo para la comunicación con el SQL Server.



**Figura 5.34 Comunicación con SQL SERVER**

La aplicación en Visual Basic se conectará a la Base de datos SQL SERVER 2000 a través del ODBC respectivo para poder procesar la información y obtener los reportes necesarios para la administración correspondiente por parte del Administrador del Sistema y los usuarios finales, la misma que se realizará jerárquicamente.

### 5.6.2 Implementación de la Base de Datos (BACK-END)

La base de Datos Administrativa SQL SERVER 2000 está diseñada de la siguiente manera:

- SEGURIDADES

Con la finalidad de mantener una seguridad de la información se diseñó un modelo de seguridad, el mismo que esta conformado de cuatro tablas: REF\_MENU, REF\_MENU\_PERFIL, REF\_PERFIL, REF\_USUARIO.

Este modelo permite al administrador del sistema dar permisos a los usuarios según el Perfil asignado (Administrador, Operador, Técnico, etc.)

- PROCESO ADQUISICIÓN DE DATOS

Para efectos de almacenar la información de la Bases del CITADEL, se han creado dos tablas: ESTADOMAQ, TMP\_ESTADOMAQ; las mismas que serán utilizadas por la aplicación informática con el fin de almacenar la información que generen cada una de las máquinas con los procesos de control de las estaciones.

### 5.6.3 Implementación de la Aplicación (FRONT-END)

Para utilizar el sistema, se debe ingresar el nombre de **usuario** y **contraseña** asignados para cada usuario por el administrador del Sistema. Como se observa en la figura 5.35



Figura 5.35 Pantalla de usuarios de la Base de Datos

Si los datos ingresados son correctos, aparecerá la pantalla principal de la figura 5.36:

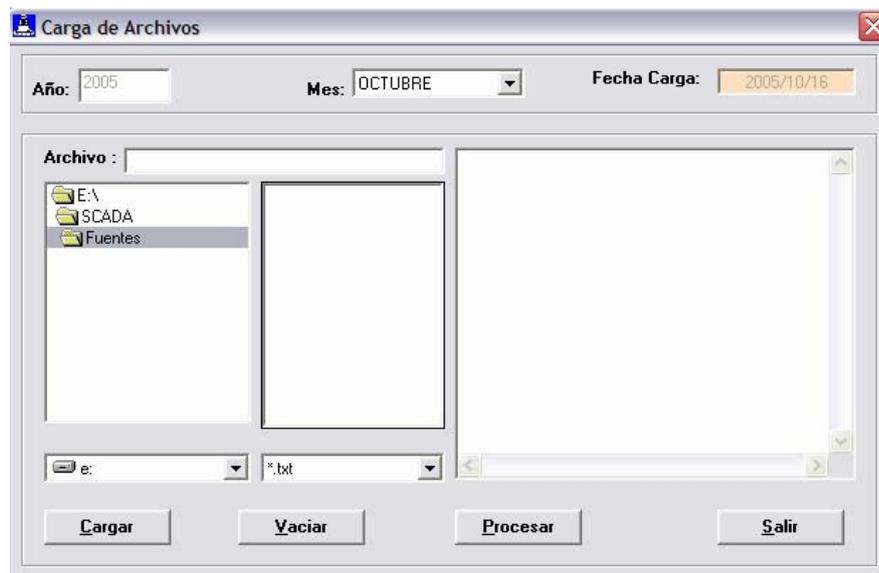


Figura 5.36 Menús de la Base de Datos

En esta pantalla se encuentra el menú principal, el mismo que servirá para poder manejar el sistema; el Menú se divide en tres opciones:

- Base de datos.- Esta opción permite realizar la carga de la información desde la Base de Datos CITADEL a la Base de Datos Administrativa.
- Mantenimiento.- Esta opción permite administrar al sistema creando usuarios, perfiles y accesos al sistema
- Reportes.- Esta opción permite obtener reportes como: Alarmas, Reportes Históricos, eventos.

A continuación se realiza la descripción del proceso de carga de información del CITADEL a la base de Datos Administrativa; para esto se debe ubicar en la opción Base de Datos \ carga, tal como se indica en la figura 5.37:



**Figura 5.37** Ventana para la Carga de la información a la Base de Datos

Esta pantalla presenta tres botones:

- Cargar.- Permite seleccionar el archivo a cargar, para esto se selecciona el archivo del CITADEL, al presionar CARGAR el sistema empieza a importar los datos a la base de datos administrativa a una Tabla Temporal, el proceso realiza validaciones internas entre la tabla temporal y la tabla real con la finalidad de evitar inconsistencia y duplicidad de información.

Al terminar el proceso de carga, se genera un archivo .LOG en el que se da un resumen de la carga.

- Vaciar.- Este botón permite borrar la información cargada en la tabla Temporal.
- Procesar.-Una vez cargada la información, se debe presionar el Procesar y el sistema automáticamente almacena la información de la tabla temporal a la tabla real.

## **CAPÍTULO VI**

### **CONCLUSIONES Y RECOMENDACIONES**

#### **CONCLUSIONES.**

1. Se han diseñado e implementado los programas para el control de procesos de Nivel, Flujo, Presión y Temperatura, ayudándose para ello del software LabVIEW de National Instruments y de los siguientes dispositivos de hardware: Field Point FP1000, con módulos AI210, AO200, tarjeta DAQ NI-USB 6009, tarjeta DAQ NI-PCI 6014, módulo de acondicionamiento de señales SCC AI20 y SCC AO20, con módulo portador de SCC, SCXI 2345, PLC Siemens Simatic S7-200, CPU 224 y módulo adicional de entradas y salidas análogas 235, obteniéndose de esta forma el sistema esperado conforme a los objetivos propuestos en este proyecto.
2. Se ha implementado una red de trabajo en grupo para conectar las cuatro computadoras que realizan el control de cada proceso. Una computadora que actúa como concentrador de la red industrial y servidor de base de datos para la red corporativa, y finalmente un computador que simula la red corporativa que hace uso de los datos de la red industrial para la administración y definición de los procedimientos operativos.
3. Los sistema DCS desarrollados para cada estación de procesos sustituyen al controlador Foxboro 761C y lo reemplazan por cuatro diferentes tipos de controladores, tomando en cuenta que la selección de cada dispositivo de adquisición de datos se adapta a las características de cada uno de los sistemas.

4. Se ha diseñado e implementado un sistema SCADA para los cuatro procesos anteriormente mencionados, supervisados desde un computador principal que actúa como interfase entre la red industrial y la red corporativa. Para ello, se ha utilizado el software LabVIEW DSC, y los siguientes servidores de OPC: NI-DAQ-OPC Server, NI-FieldPoint-OPC Server, NI-LookOut Protocol Drivers y SW Top Server.
5. Se ha diseñado e implementado una base de datos relacional con motor SQL que simula la operación y gestión administrativa de la red corporativa que se sirve de la base de datos generada en el módulo DSC Citadel Universe 5.0 para el manejo administrativo de los datos de producción.
6. El levantar una base de datos a tiempo real, permite realizar monitoreo de las variables principales como son los tags generados, reportes de alarmas y usuarios. Además el motor de generación de la base de datos SQL es una herramienta poderosa que facilita la lectura del archivo plano que se genera en la base propia de Citadel ya que la lectura de los parámetros son definidos por el usuario.
7. Se ha elegido en la mayoría de los casos una integración entre software y hardware de National Instruments por la dificultad que presentan los paradigmas de los sistemas cerrados, pues la comunicación con el tipo de dispositivos que manejan esos paradigmas se vuelve demasiado costosa y es un valor adicional facturado por las empresas por el desarrollo de los drivers que no son de su marca. Lo que a la postre resulta más complicado y costoso al momento de montar un sistema de este tipo.
8. El integrar hardware y software de un mismo fabricante permite mejorar la velocidad de conexión entre los dispositivos físicos y el sistema de control computacional, es por esta razón que la mayoría de empresas desarrolladoras de software de SCADA lo hacen como software dedicado para una marca exclusiva de hardware y viceversa.

9. La mayoría del software de SCADA utiliza Scripts para la programación de sus algoritmos lógicos y de control; estos son para muchas aplicaciones realmente limitados. Es mucho más eficiente realizar sistemas DCS y SCADA con LabVIEW DSC ya que la programación en este tipo de software se la hace en base al contexto real de un lenguaje de programación como es LabVIEW.
  
10. El diseño de la red industrial a diferencia de lo que sucede con una red corporativa estándar se la realizó sin servidor DHCP y sin DNS, simplemente cada computador posee una dirección IP fija establecida con un solo grupo de trabajo que se denomina SCADA y con una máscara de subred válida como 255.255.255.0. Ésta configuración se la llevó a cabo con el módulo DSC.
  
11. El sistema SCADA se lo implementó en base a LabVIEW DSC por la conveniencia del acoplamiento entre el software y hardware antes mencionado; pero es sumamente importante destacar que la configuración de los dispositivos de hardware en su totalidad (independientemente de la marca) se la realizó para que puedan ser utilizados por otro software cliente de OPC, pudiendo ser usuarios de esta configuración los siguientes sistemas: Intouch, RSVIEW, Cx-Supervisor, LookOut, Aimax for Windows, Wintag Monitor, Ellipse, FoxVIEW, etc. Para la implementación del sistema con uno de ellos se necesitará que, se disponga de los drivers de National Instruments para realizar el control.
  
12. Si bien es cierto que la implementación del presente proyecto está estrechamente ligado a la enseñanza y la labor docente, es necesario aclarar que en el campo industrial si se desea realizar un sistema de este tipo y con el software LabVIEW DSC. La configuración de los dispositivos de hardware y la programación son exactamente iguales. Hay que recordar para este efecto que las estaciones de trabajo LabVolt, a pesar de ser también diseñadas para enseñanza, manejan sistemas de monitoreo y control absolutamente reales y en muchos casos son los mismos existentes en la empresa.
  
13. Se ha desarrollado una herramienta muy importante para el Laboratorio de Control de Procesos e Instrumentación, pues se ha implementado un sistema

de control distribuido completo, con todos los equipos existentes para el efecto, y un sistema supervisorio de todos los procesos disponibles. Además de que esta herramienta puede ampliarse según la ampliación y crecimiento del laboratorio lo amerite.

## **RECOMENDACIONES.**

1. El 99% de las respuestas a los problemas técnicos que se puedan presentar en el diseño e implementación de sistemas como el desarrollado, está en los manuales técnicos y de operación del software y el hardware a utilizar. Por esta razón es importante poseer dichos documentos; de esto se deriva que la adquisición de las licencias correspondientes es muy importante, porque solamente de esta forma la documentación requerida será completa. El 1% restante se puede obtener del respaldo técnico de la empresa vendedora de la herramienta hardware o software que se maneje.
2. El uso del Internet es completamente importante en cualquier investigación, sea cual sea su fase de desarrollo, toda la información adicional que se necesite se encuentra en él. Es una herramienta fundamental que es necesario utilizarla y difundirla.
3. Es muy importante configurar y calibrar todos los dispositivos utilizados para el desarrollo del sistema de control implementado, ya que los rangos de calibración que presenta cada uno de ellos dependerá única y exclusivamente del requerimiento del proceso a ser controlado.
4. Se debe manejar adecuadamente los manuales y hojas de datos de los dispositivos de adquisición, ya que estos datos técnicos dados por el fabricante garantizarán el buen funcionamiento de los mismos, y permitirán una vida útil de los equipos.
5. Para el uso y manejo de este sistema SCADA, se debe tener un previo conocimiento de los temas relacionados tales como: sistemas de control, adquisición de datos, instrumentación, control de procesos e instrumentación

virtual, para facilitar la comprensión de las variables controladas y controladores utilizados en la misma.

6. El trabajo de investigación y desarrollo que se ha realizado, debería ser utilizado no solo para la obtención de un título académico, sino como una herramienta que aporte en la formación de los estudiantes de Ingeniería de la Escuela Politécnica del Ejército sede Latacunga, como un complemento de laboratorio en las asignatura que así lo amerite; ya que en muchas de las ocasiones suele suceder que dichos trabajos no se utilizan como herramienta de apoyo académico.
7. A la Escuela Politécnica del Ejército sede Latacunga, que aproveche el hecho que la mayoría de los estudiantes realiza tesis de mucha investigación, que se sirva de ellas para actualizar los laboratorios, y los conocimientos de los docentes y dicentes modificando los planes analíticos según el tiempo y la tecnología así lo ameriten.
8. Todo proyecto de trabajo científico es susceptible a ampliaciones, modificaciones y cambios, es así como éste proyecto se puede modificar en el futuro introduciendo sistemas de control más actuales como: transmisores inteligentes que se conecten directamente a la red Ethernet; módulos de DAQ y Field Point RT y con dirección IP propia, uso de computadores industriales que manejen módulos PXI y SCXI, RTU's de última generación, pantallas táctiles que manejan su propio HMI y por supuesto, que todos los elementos mencionados que puedan "colgarse" en redes como: Seriplex, HART, Canbus, Industrial Ethernet, DeviceNet, etc.