

**CONFIGURACIÓN, ADQUISICIÓN Y
ALMACENAMIENTO DE DATOS CON
FIELDPOINT EN LABVIEW**

CONTENIDO

	Pág.
1. CONEXIÓN ENTRE EL SISTEMA SUPERVISORIO Y EL BANCO DE FIELDPOINT	2
1.1 FIELDPOINT	2
1.2 CONEXIÓN ENTRE LABVIEW Y FIELDPOINT	5
1.2.1 Configuración de MAX	7
1.2.2 Conexión entre LABVIEW y el archivo *.IAK	11
2 CONEXIÓN ENTRE EL SUPERVISORIO Y LA HERRAMIENTA DE GESTIÓN DE DATOS	14
2.1 CONFIGURACIÓN DE ODBC	14
2.2 PROGRAMAS DE CONEXIÓN EN LABVIEW	16
2.2.1 Programa para funciones sin retorno de datos	18
2.2.2 Programa para funciones con retorno de datos	29
3 MAPEO DE CONEXIONES DE LAS VARIABLES AL BANCO DE FIELDPOINT	20

1. CONEXIÓN ENTRE EL SISTEMA SUPERVISORIO Y EL BANCO DE FIELDPOINT

1.1 FIELDPOINT

FieldPoint de *National Instruments* es un sistema modular de adquisición de datos distribuido para el monitoreo y el control de procesos industriales y aplicaciones de enlace de datos, que involucran diversos sensores y actuadores localizados central o remotamente. NI FieldPoint incluye capacidades de operaciones de red para I/O distribuidas. Este sistema incluye una gran variedad de módulos análogos, digitales, de redes de comunicación, etc. lo que permite bastante flexibilidad al momento de diseñar el sistema. Con FieldPoint se pueden mejorar las funciones en los procesos de automatización de forma confiable aun en ambientes eléctrica y físicamente hostiles.

FieldPoint permite trabajar en Bancos para implementar un sistema distribuido de recolección de datos. Un banco consta de un módulo de comunicaciones y uno o varios módulos de entrada y salida para monitoreo de las variables. Cada banco de fieldpoint puede soportar hasta 8 módulos de entrada y salida y un sistema distribuido puede soportar hasta 250 bancos en red.



Fig. 1 Banco de FieldPoint

Para nuestro proyecto se eligió FieldPoint de National Instrument porque siendo un sistema modular permite escoger los módulos de entrada y salida de acuerdo a las necesidades del proceso. En la línea de UHT de Friesland Colombia existen 22 variables instrumentadas las cuales tienen transmisores de corriente con el estándar industrial 4 – 20 mA, es por esto que elegimos un módulo de comunicaciones FP-1000, el cual tiene un puerto RS-232 y dos módulos de entradas análogas FP-AI-111 con 16 entradas analógicas de 4 – 20 mA cada uno, identificadas con los números desde 0 a 15 en la bornera de conexiones. El puerto RS-232 permite comunicarse con el computador a través de su puerto serial, además contiene un puerto RS-485 el cual permite conectar en red otros módulos de comunicación de referencia FP-1001 para formar una red de entradas y salidas distribuidas.

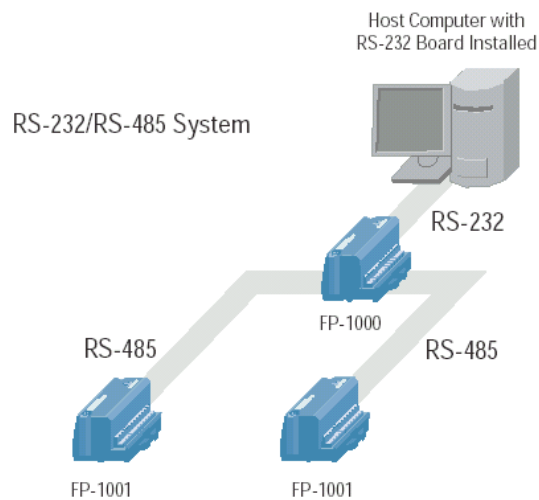


Fig. 2 Red de bancos FieldPoint

1.2. CONEXIÓN ENTRE LABVIEW Y FIELDPOINT

Para realizar la conexión entre el sistema de supervisión y el banco de Fieldpoint se requiere realizar dos procedimientos.

El primero es configurar la conexión entre el banco de fieldpoint y la herramienta de conexión OPC llamada Measurement & Automation (MAX) que se instala automáticamente con cualquiera de los software licenciados por National Instruments y con la cual generamos un archivo de conexión con la extensión IAK. Este archivo contiene toda la información correspondiente a la configuración de los bancos conectados al puerto serial del computador. Para nuestro proyecto usamos la versión 3.0 incluida con la versión Labview 7.0.



Fig. 3 Herramienta MAX

El segundo procedimiento es realizar la conexión entre Labview y el archivo de configuración IAK. Esta conexión se crea realizando un llamado a este archivo desde un programa creado en Labview utilizando la herramienta de conexión Configure FieldPoint.

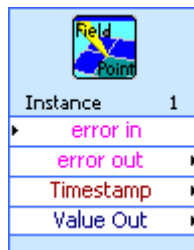


Fig. 4 Herramienta Configure FieldPoint

1.2.1 Configuración de MAX.

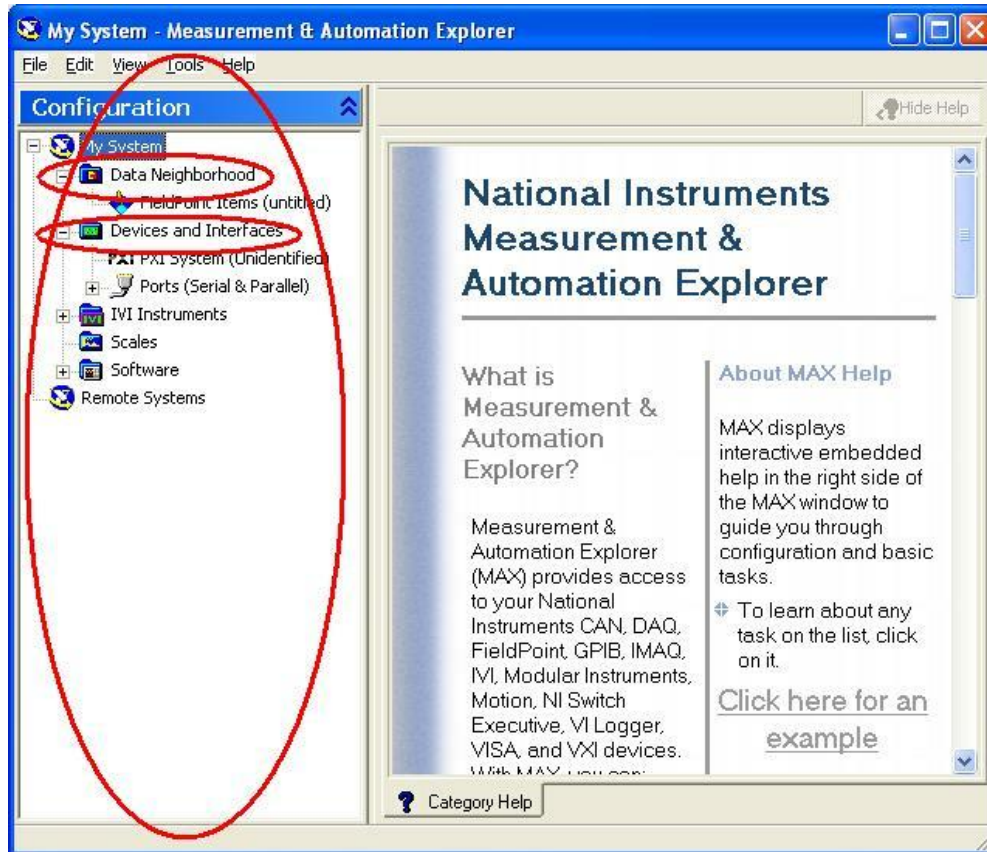


Fig 5 Ventana principal del MAX.









En esta ventana se señala el árbol de configuración del MAX y las partes más importantes del mismo

El primer paso para crear el archivo de configuración IAK es crear un recurso de comunicación serial. Un recurso de comunicación serial establece la información necesaria para que el módulo de FieldPoint se identifique y pueda transmitir sus datos al sistema.

Para adicionar un recurso de comunicación serial se realizan los siguientes pasos:

- Haga clic derecho en el menú My System>>Device and Interfaces del árbol de configuración y seleccione “Create new...”.
- En la ventana de dialogo que aparece seleccione FieldPoint>>FieldPoint Serial y presione Next.
- Luego de presionar Next aparece una nueva ventana de dialogo en la cual se configura la comunicación serial entre el computador y el banco FieldPoint asociado a esta aplicación. En esta ventana haga los siguientes pasos:
 - Seleccione el puerto serial COM al que fue conectado el sistema FieldPoint.
 - Especifique la velocidad de transmisión Baudios “Baud Rate”. Debe tener en cuenta que la velocidad de transmisión que especifique debe ser la misma en el módulo de red.

La velocidad de transmisión en el módulo de red se configura estableciendo la posición de cada uno de los switches:

Posición switches 6 a 8	Velocidad en Baudios	Posición switches 6 a 8	Velocidad en Baudios	Posición switches 6 a 8	Velocidad en Baudios
	300		9600		57600
	1200		19200		115200
	2400		38400		

- En la parte inferior de la ventana presione Finish.
- Luego de presionar Finish aparece una ventana de dialogo en la cual se escribe el nombre que se le desea dar al recurso de comunicación creado. Presione OK.
- Al presionar OK, MAX busca los dispositivos conectados. Al finalizar la búsqueda se crea un recurso de comunicación serial bajo la ruta My System>>Devices and Interfaces con el nombre FP@COMx, donde x indica el puerto serial del computador al que fue conectado el sistema FieldPoint. Este recurso de comunicación muestra las interfaces o módulos de comunicaciones y los módulos de entrada y salida conectados, además se crea un folder bajo My System>>Data Neighborhood el cual contiene los datos de cada una de las entradas y salidas de los módulos.

El aspecto del árbol de configuración de MAX quedará de la siguiente forma:

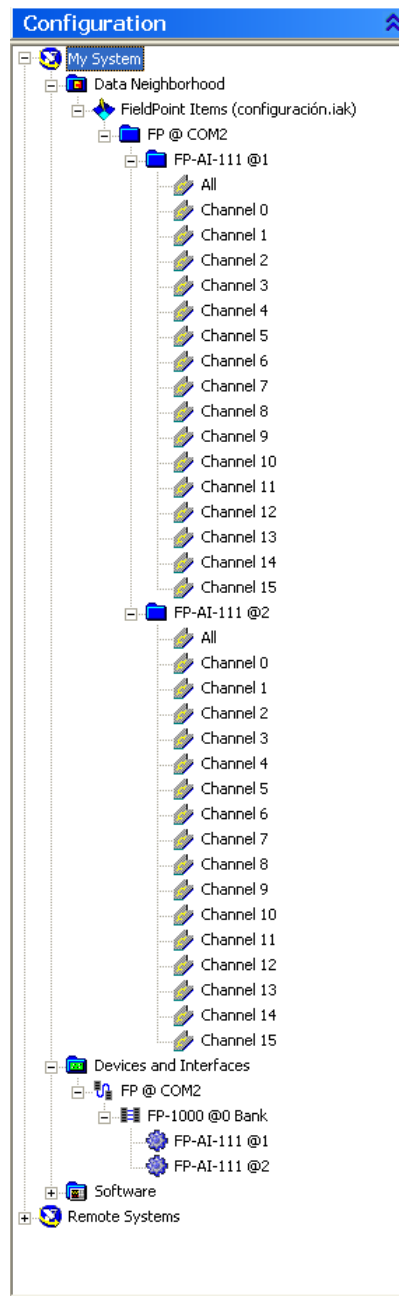


Fig. 6 Árbol de configuración terminado

Antes de cerrar el MAX guardamos la configuración creando de esta manera el archivo de configuración con el nombre *configuracion.iak* el cual utilizaremos en Labview para poder adquirir los datos en el sistema supervisorio.

1.2.2. Conexión entre LABVIEW y el archivo *.IAK. A continuación se explicará la conexión básica para poder obtener en Labview los valores de las variables conectadas a los módulos de entrada y salida del banco de Fieldpoint.

Se inicia abriendo un programa (*.VI) en blanco. En la ventana de Diagrama de Bloques (Block Diagram) se inserta el bloque Configure FieldPoint mostrado en la figura 4. Al hacer doble clic en esta herramienta se abre la siguiente ventana de configuración:

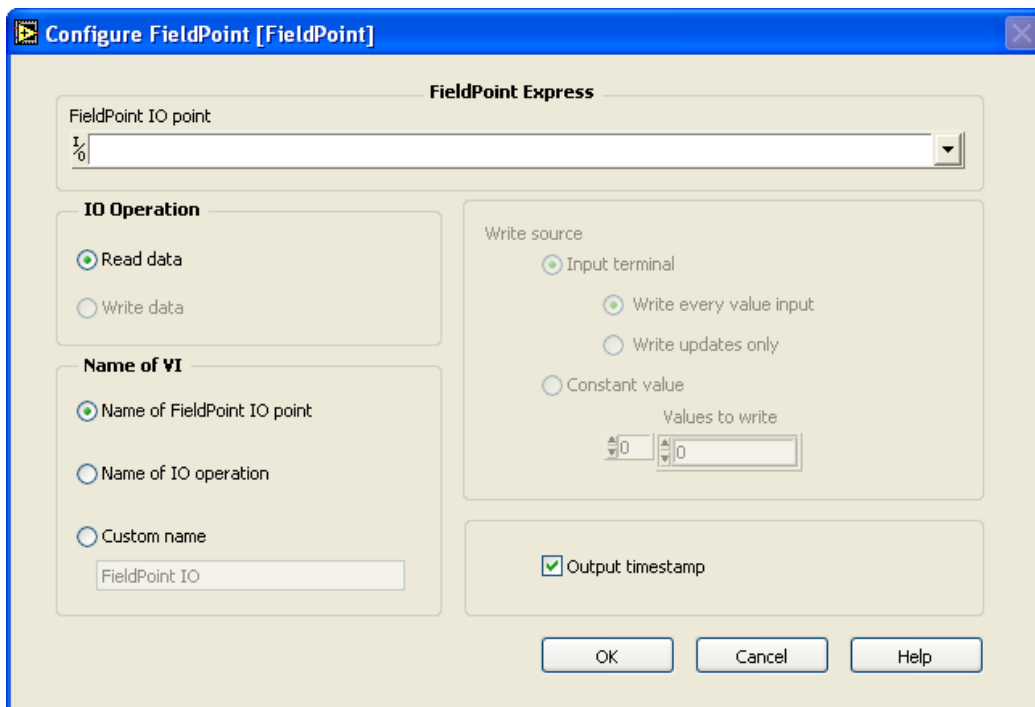


Fig. 7 Ventana de configuración de la herramienta Configure Fieldpoint

En esta ventana escogemos y configuramos el módulo o las entradas que deseamos obtener su lectura. Primero expandimos el campo *FieldPoint IO* y

escogemos “Browse”. Luego de realizar esta operación se despliega la siguiente ventana:

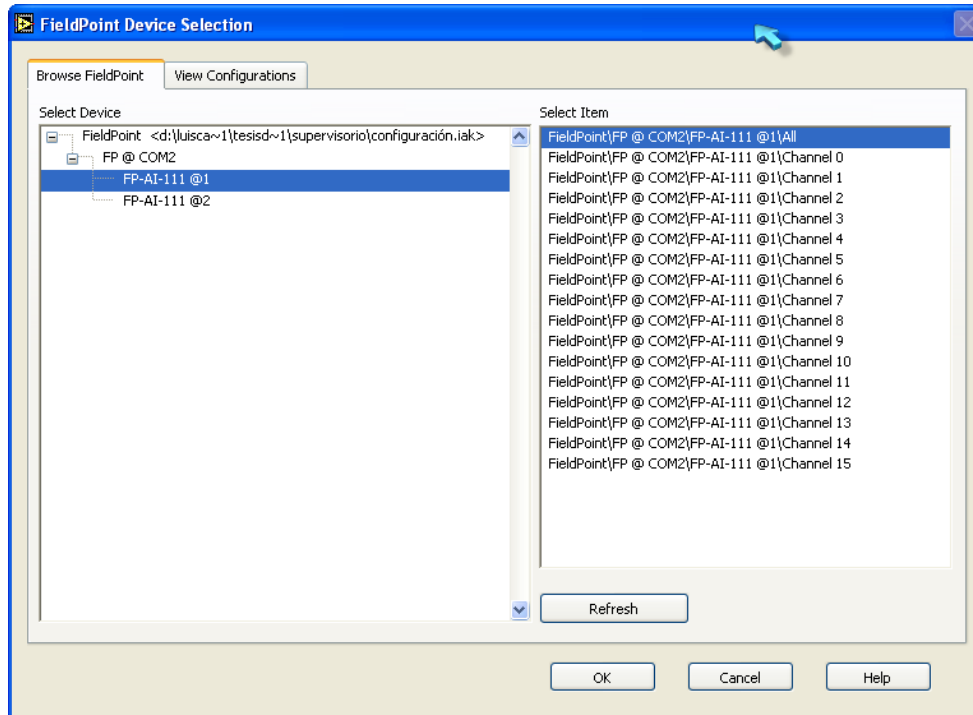


Fig. 8 Ventana de selección de dispositivo de lectura

En el lado izquierdo de la ventana seleccionamos el módulo de entradas analógicas del que deseamos obtener sus datos. En el lado derecho seleccionamos las entradas o simplemente seleccionamos el ítem *All* para leer las 16 entradas del módulo seleccionado. Posteriormente damos OK y cerramos la ventana *Configure Fieldpoint* presionando OK nuevamente.

Para obtener los datos del segundo módulo de entradas análogas cargamos otro bloque de la herramienta *Configure Fieldpoint* y repetimos los pasos anteriores.

Es importante tener en cuenta que LabVIEW es un software de programación gráfica que ejecuta varias instrucciones a la vez, por lo que no se hace conveniente tener las lecturas de los dos módulos al mismo tiempo ya que se pueden presentar conflictos de datos o timeouts de lectura. Para solucionar este problema se hace necesario usar la estructura *Stacked sequence structure* la cual nos permite ejecutar códigos de manera secuencial.

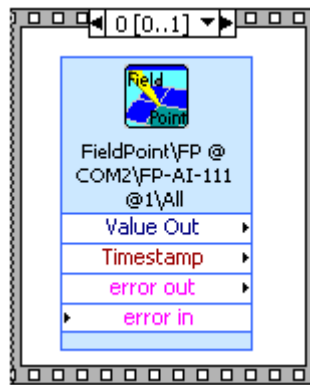


Fig. 9 Estructura Stacked sequence structure

Luego de ingresar los bloques en la estructura podemos obtener las lecturas seleccionadas en el campo *Value Out* de la herramienta *Configure Fieldpoint* la cual se puede apreciar en la figura 9.

2. CONEXIÓN ENTRE EL SUPERVISORIO Y LA HERRAMIENTA DE GESTIÓN DE DATOS

La conexión entre la base de datos creada en Microsoft Access y Labview se realiza por medio de la herramienta administrativa de Windows ODBC, el cual es un programa de interface de aplicaciones para acceder a datos en sistemas de manejadores de bases de datos utilizando para ello SQL. Este programa almacena información acerca de cómo conectarse al proveedor de datos, en nuestro caso, la base de datos creada en Access.

2.1. CONFIGURACIÓN DE ODBC

Iniciamos cargando el programa ODBC desde el Panel de Control de Windows.

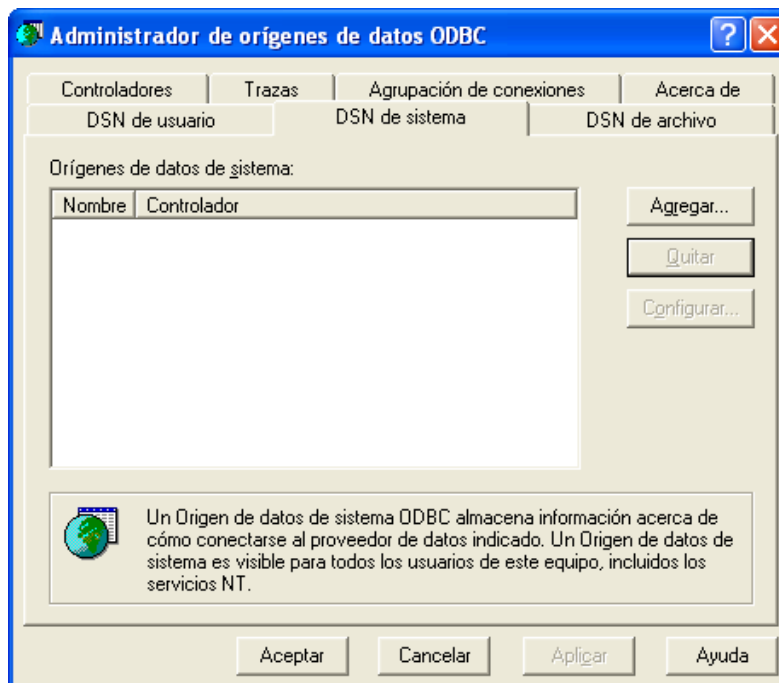


Fig. 10 Ventana principal ODBC

Al abrir el programa seleccionamos la pestaña *DNS de sistema* como se muestra en la anterior figura. En esta pestaña damos clic en *Agregar* presentándose una ventana con el listado de los posibles proveedores de datos. Seleccionamos el proveedor de datos *Microsoft Access Driver (*.mdb)* y damos clic en finalizar. Posteriormente aparecerá la siguiente ventana

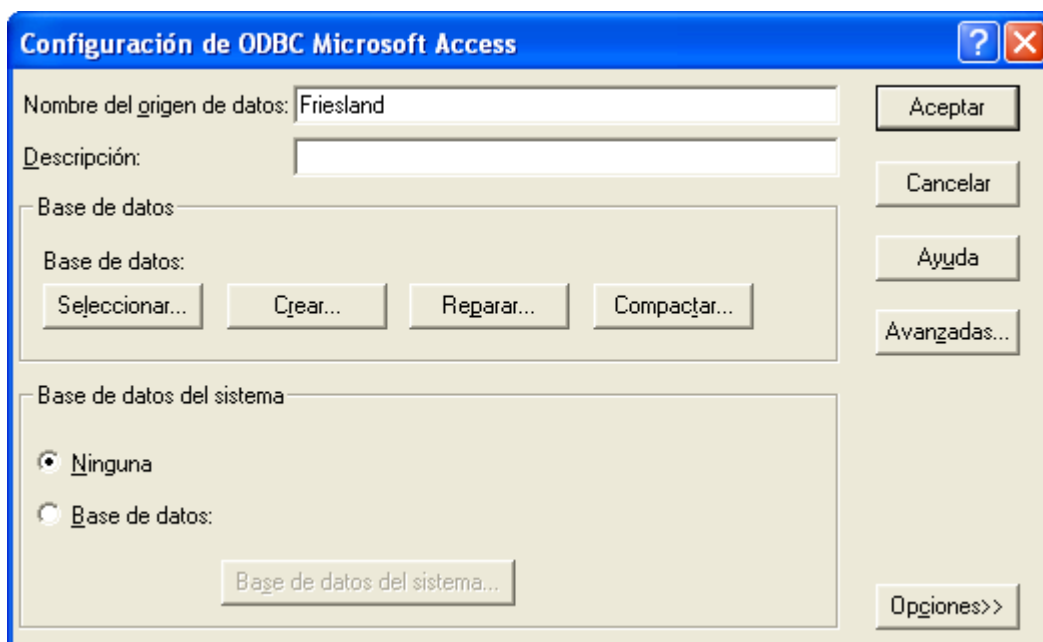


Fig. 11 Selección del proveedor de datos.

En la anterior ventana damos como nombre al Origen de datos *Friesland* y en el ícono *seleccionar* ubicamos la ruta del archivo creado en Microsoft Access. Finalmente damos aceptar a las ventanas y tenemos el sistema preparado para realizar la comunicación entre Labview y la base de datos creada en Access.

2.2. PROGRAMAS DE CONEXIÓN EN LABVIEW.

Como se pudo observar en el capítulo 2 en la mayoría de secciones y partes del programa se realizan concurrencias a la base de datos desde el supervisorio. Estas concurrencias se realizan con cinco subprogramas básicos los cuales tienen una configuración de interconexión predeterminada, expuesta en el punto 3.2.2.1, y a los cuales se modifican sus entradas y/o salidas dependiendo del tipo de función SQL que se desea ejecutar (escribir, leer, borrar o actualizar). Estos subprogramas vienen con una herramienta llamada LABSQL.

A continuación se describen de forma general los cinco subprogramas.

SubVI Create Connection.



Este subprograma crea una conexión con el servidor de datos ODBC.

SubVI Open Connection.



Este subprograma abre una conexión con un origen de datos. En nuestro caso el origen de datos se denomina *Friesland* el cual fue establecido en la configuración

de ODBC en el punto 3.2.1. Este subprograma tiene como entrada el nombre del origen de datos.

SubVI SQL Execute.



Este subprograma ejecuta una instrucción SQL la cual es establecida de acuerdo a los requerimientos de la concurrencia. Este subprograma tiene como entrada el código SQL de la concurrencia y una entrada booleana estableciendo si se requiere retorno o no de datos.

SubVI Close Connection.



Este subprograma cierra la conexión con el origen de datos. Este paso se debe hacer previo a la destrucción de la conexión con ODBC.

SubVI Destroy Connection.



Este subprograma destruye la conexión con el servidor ODBC.

Es importante tener en cuenta que para realizar una nueva concurrencia se requiere crear y abrir otra conexión para hacer una diferenciación entre las consultas.

Los anteriores subVI's se acondicionan de la siguiente manera para ejecutar las diferentes funciones de SQL.

2.2.1. Programa para funciones sin retorno de datos. Las funciones que no requieren retorno en nuestro proyecto son aquellas que realizan la tarea de inserción (Insert Into), borrado (Delete) y actualización de datos (Update).

En la siguiente figura podemos observar la conexión de los programas en el diagrama de bloques de LabView para poder ejecutar las anteriores funciones.

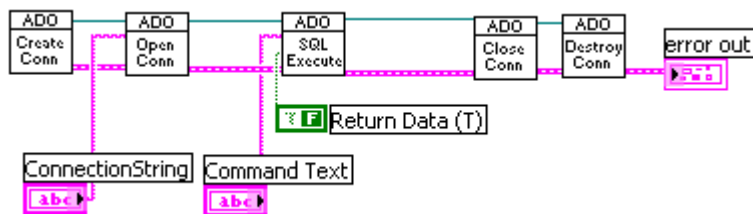


Fig. 12. Conexión de subprogramas LabSQL para funciones de no retorno de datos.

Las entradas y salidas para este programa son:

- **Open Connection.**

Entradas

- Nombre del Origen de datos (*ConnectionString*).

- **SQL Execute.**

Entradas

- Código SQL (*Command Text*).
- Entrada booleana en Falso, lo cual indica que no tenemos retorno de datos (*Return Data*).

2.2.2. Programa para funciones con retorno de datos. Las funciones que requieren retorno en nuestro proyecto son aquellas que realizan las tareas de búsqueda (SELECT).

En la siguiente figura podemos observar la conexión de los programas en el diagrama de bloques de LabView.

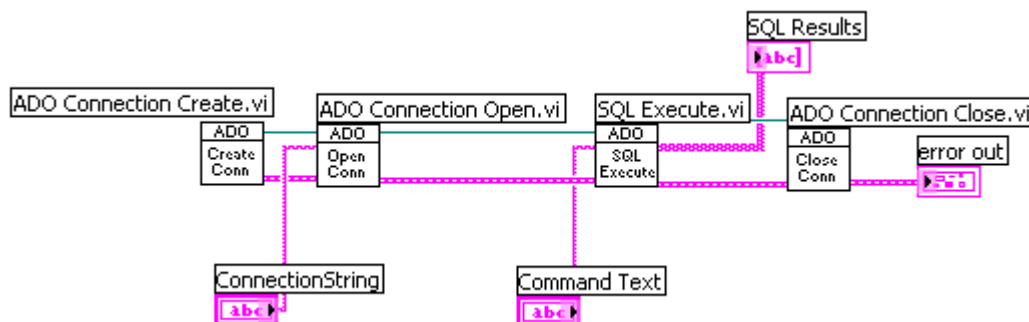


Fig. 13 Conexión de subprogramas LabSQL para funciones con retorno de datos.

Las entradas y salidas para este programa son:

- **Open Connection.**

- **Entradas**

- Nombre del Origen de datos (*ConnectionString*).

- **SQL Execute.**

- **Entradas**

- Código SQL (*Command Text*).
 - Entrada booleana no conectada, lo cual indica que tenemos retorno de datos ya que esta configurada por defecto en Verdadero.

- **Salidas**

- Matriz de datos tipo Cadena con los resultados de la búsqueda (*SQL Results*).

3. MAPEO DE CONEXIONES DE LAS VARIABLES AL BANCO DE FIELDPOINT

En la siguiente tabla se indica en que ubicación de la bornera debe ser conectada cada variable.

VARIABLE	PUNTO DE BORNERA
MODULO ANALÓGICO 1	
TT 2	0
TT 4	1
TT 5	2
TT 8	3
TT 9	4
TT 10	5
TSL 42	6
TIC 44	7
TT 45	8
TIC 64	9
TSL 71	10
U 110	11
U 140	12
U 160	13
U 170	14
FM_HOM	15
MÓDULO ANALÓGICO 2	
P_HOM	0
AM_HOM	1
PT 130	2
SILO 1	3
SILO 2	4
SILO 3	5

Es importante tener en cuenta la distribución de las variables en la bornera de conexiones de cada módulo de entradas analógicas, ya que los datos en LabView se recolectan en forma de paquetes organizados según las conexiones asignadas. Cada módulo entrega un paquete el cual es una matriz de datos 1x16. La siguiente figura muestra el segmento de programa con el cual se realiza la extracción de los datos del módulo analógico 1.

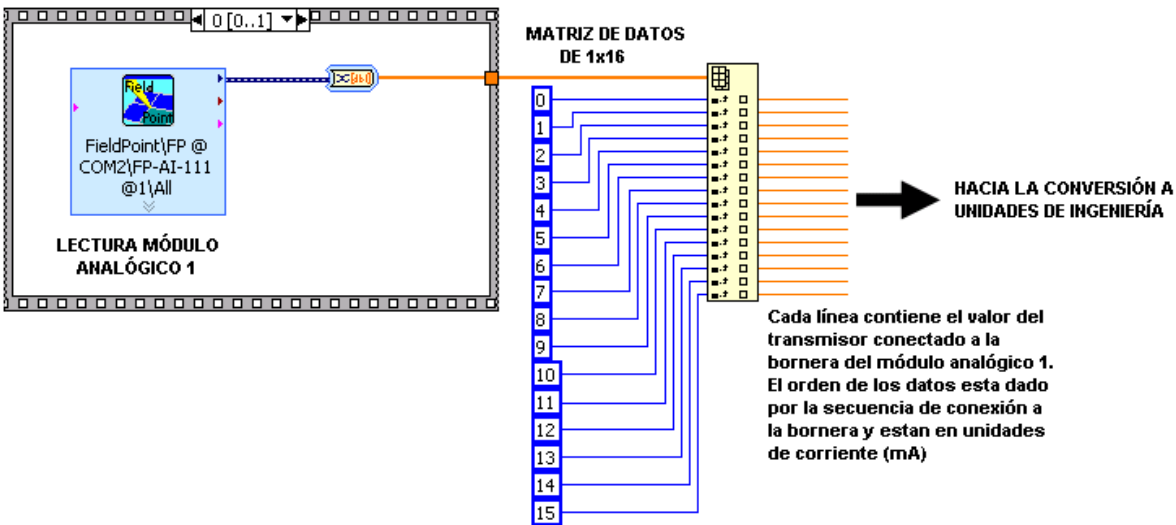


Fig. 14 Adquisición de valores de transmisores desde LabView