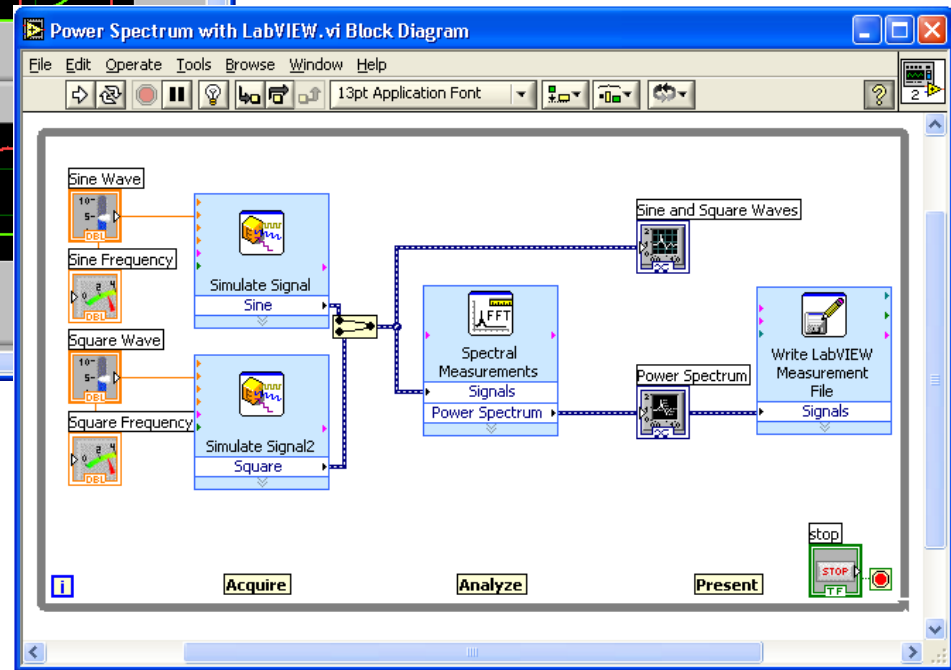


Instrumentación Virtual con LabVIEW



INSTRUMENTACIÓN AVANZADA
(Programación en Instrumentación Virtual)
Programa Analítico

•**UNIDAD TEMÁTICA I: INTRODUCCIÓN A LA PROGRAMACIÓN GRÁFICA**

Instrumentación Virtual. Entorno de desarrollo gráfico para la programación de sistemas de instrumentación y de control. Lenguaje Gráfico G en el entorno del LabVIEW. Aplicación de librerías en el entorno de Visual Basic. Equipamiento en software y hardware en el laboratorio. Menús de herramientas, controles y funciones. Conceptos de vi, subvi. Creación de un programa y depuración del código. Ejercicios prácticos.

•**UNIDAD TEMÁTICA II: PROGRAMACIÓN ESTRUCTURADA.**

Introducción: estructuras básicas. Estructuras iterativas (For Loop y While Loop). La temporización en la ejecución de código. Registros de desplazamientos. Estructuras Case y Event. Estructuras Sequence. Fórmula Node. Variables locales y globales. Ejercicios Prácticos. Laboratorio

•**UNIDAD TEMÁTICA III: LABVIEW Y MATLAB**

Aplicación de Mathscripts en aplicaciones de VI. Rutinas Matlab compatibles en LabView. Aplicaciones de programas en VI.

•**UNIDAD TEMÁTICA IV: TIPOS DE DATOS**

Tipos de Datos Estructurados. Introducción a los arrays. Funciones con arrays. Clusters. Controles e indicadores de string. Archivos de entrada/salida. Manejo de archivos. Archivos de texto. Archivos binarios. Ejercicios de Aplicación.

•**UNIDAD TEMÁTICA V: ANALISIS Y VISUALIZACIÓN DE DATOS**

Análisis y visualización de datos. Indicadores chart. Indicadores Graph. Creación de subprogramas. Configuración. Aplicaciones del LabVIEW en el campo de control de los parámetros de calidad de la energía. Espectro de frecuencias e índices THD en señales no sinusoidales. Aplicaciones.

•**UNIDAD TEMÁTICA VI: ADQUISICIÓN DE DATOS**

Análisis del hardware de laboratorio. Sistemas de adquisición y procesamiento de datos. Software de adquisición de datos NI-DAQ. Creación de tarjetas de adquisición virtuales. **Tarjetas de adquisición PCI-6221. Tarjetas USB 18200-10. Tarjeta Aduisidora con PIC18F2550.** Características. Aplicaciones monitoreadas desde una pc remota. Aplicaciones Web Server.

Requisitos para la aprobación de la asignatura:

- ❑ 80% de asistencia a las clases teóricas-prácticas.**

- ❑ Proyecto Final con el desarrollo de una Aplicación en el entorno del Programa LabVIEW.**
 - ✓ *Memoria Técnica***
 - ✓ *Archivo del desarrollo en LabVIEW***

Libros :

Entorno Gráfico de Programación (LabVIEW 8.2)

Autores: J. R. Lajara- J. Pelegrí . Ed. Marcombo.

Programación Gráfica para el Control de Instrumentación

Autores: A. M. Lázaro- J. del Río Fernández. Ed. Paraninfo

LabVIEW advanced programming techniques.

R. Bitter, T. Mohiuddin, M. Nawrocki. Ed. CRC.

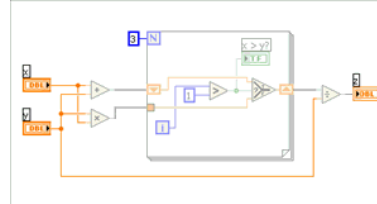
LabVIEW User Manual. National Instruments.

Measurement Manual. National Instruments.

User's Guide. Universal Library for LabVIEW (USB 18200-10).

Measurement Computing.

- Qué es LabVIEW?



¿Qué es LabVIEW?

Laboratory Virtual Instrument Engineering Workbench

LabVIEW es un entorno de desarrollo gráfico con funciones integradas para realizar ***adquisición de datos, control de instrumentos, análisis de medidas y presentaciones de datos.***

LabVIEW es un lenguaje potente en un ambiente de programación gráfico, pero mucho más sencillo que los entornos tradicionales.

Lenguaje Desarrollado para Medición, Control y Automatización

A diferencia de los lenguajes de propósito general, LabVIEW tiene funciones específicas para acelerar el desarrollo de aplicaciones de ***medición, control y automatización.***

¿Qué es LabVIEW?

Fácil Integración con Instrumentos y Dispositivos de Medida

LabVIEW se puede conectar de manera transparente con todo tipo de hardware incluyendo instrumentos, plaquetas adquirentes, controladores lógicos programables (PLCs).

LabVIEW para Investigación y Análisis

Puede utilizarse LabVIEW para analizar y registrar resultados reales para aplicaciones en amplios sectores orientados a la ingeniería mecánica, eléctrica, electrónica, biomédica, etc.

LabVIEW para Control de Procesos y Automatización en Fábricas

Puede utilizarse LabVIEW para numerosas aplicaciones de control de procesos y automatización, realizar medidas y control de alta velocidad y con muchos canales.

Proyectos Realizados por Alumnos

PROYECTOS FINALES de Carrera (Aplicando LabVIEW)

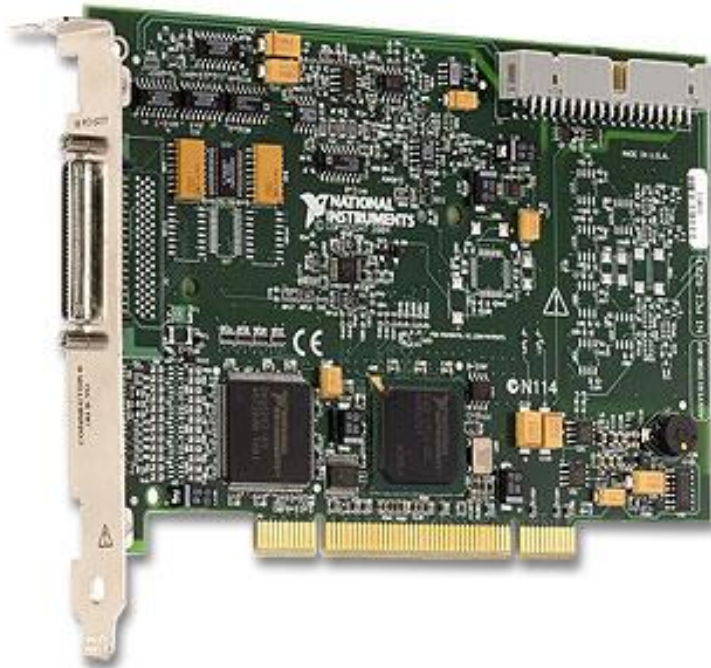
- “Estación Meteorológica” - Autor: Schnarwiler Jorge Leonardo
- “Analizador de Redes Eléctricas”- Autor: Teves, Carlos Hernán (en ejecución)
- “Análisis Modal Inverso de Vibraciones para la Detección de Entallas en Vigas Esbeltas” (Nicolás di Mauro)
- “Analizador de Armónicos inalámbrico”-Juan Simonetta
- “Adquisición de datos de un generador eólico” (Dómine-Niro)

Adquisición de datos

Adquisición de datos

NI PCI-6221

16-Bit, 250 kS/s, 16 Analog Inputs



NI PCI-6221 and Accessories

Hardware Subtotal: \$ 1,130
\$ 1,017

Adquisición de datos

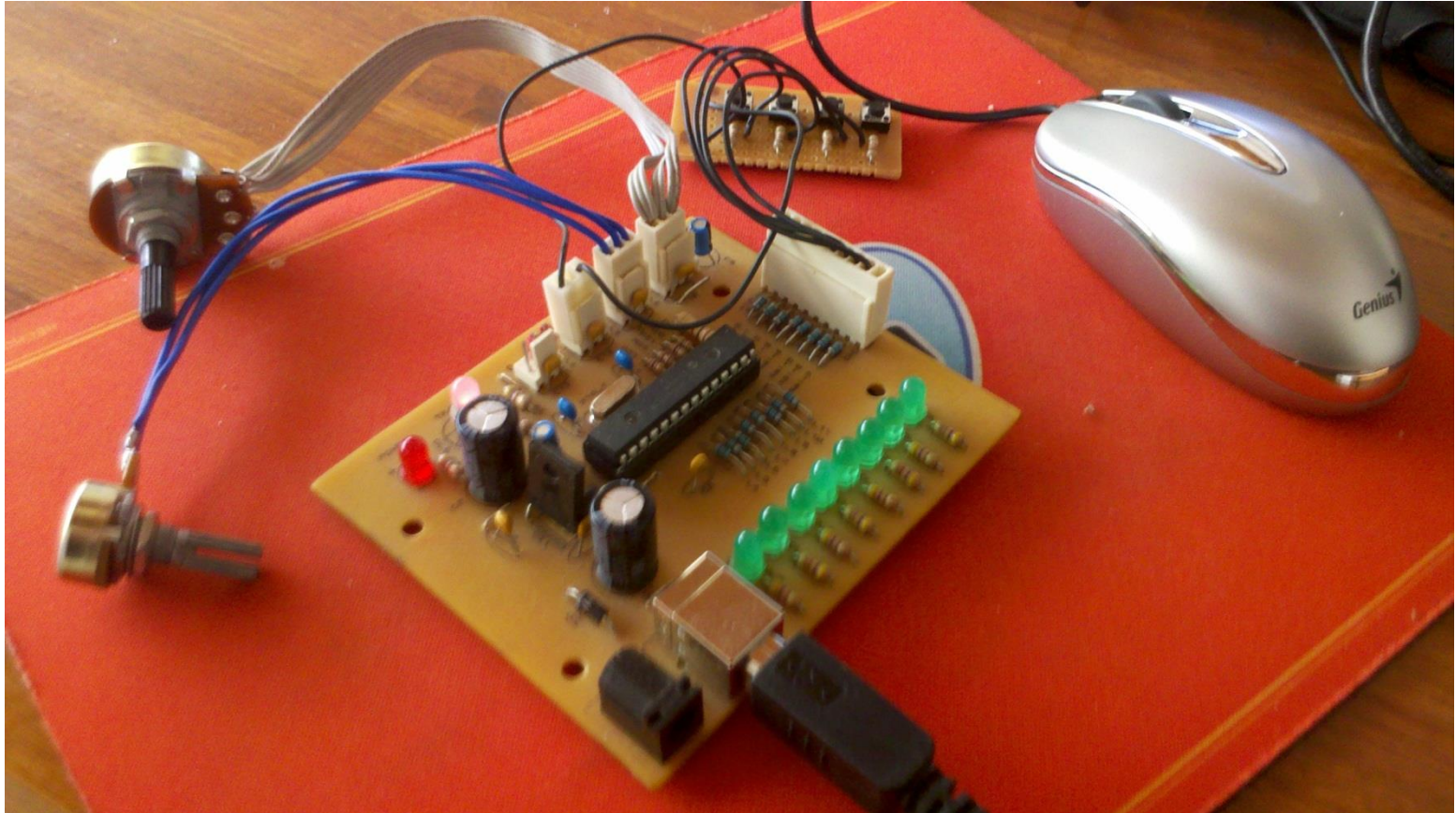


*8 canales de entrada analogica
2 canales de salida analógica
16 E/S digitales*

Specifications

Analog inputs	8 Single ended, 4 differential
Analog output	2 Channels (10 bit resolution)
Digital I/O	16 bit (bi-directional)
Counter/timer	1 (32 bit)
Operating temperature	32 to 158°F (0 to 70°C)
Input range	Diff mode: ± 1 V, ± 1.25 V, ± 2 V, ± 2.5 V, ± 4 V, ± 5 V, ± 10 V, ± 20 V
Sample rate	50 KS/s
Power	+5 volt (USB supplied from host PC)
Dimensions	3"W x 1"H x 3 1/4"D
Manufacturer number	USB-1208FS

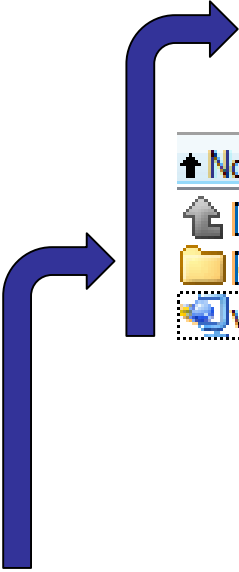
Adquisición de datos



PIC-18F2550

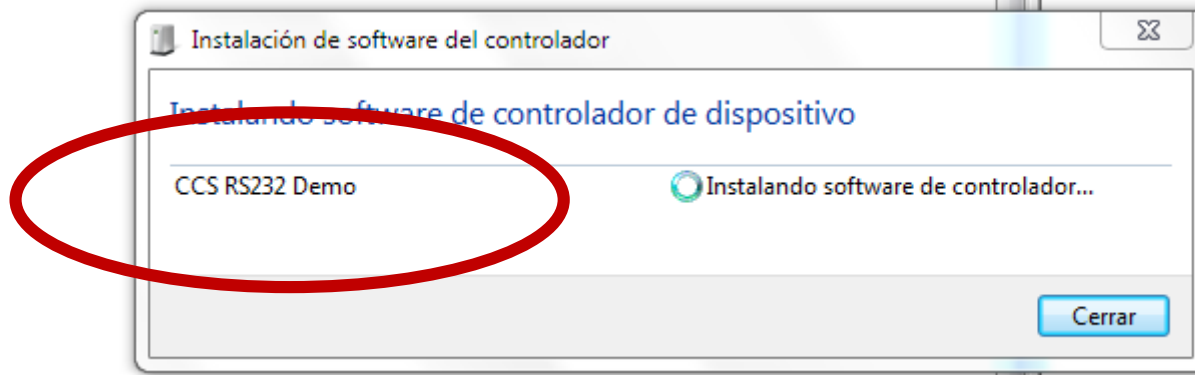
Instalación Driver PIC

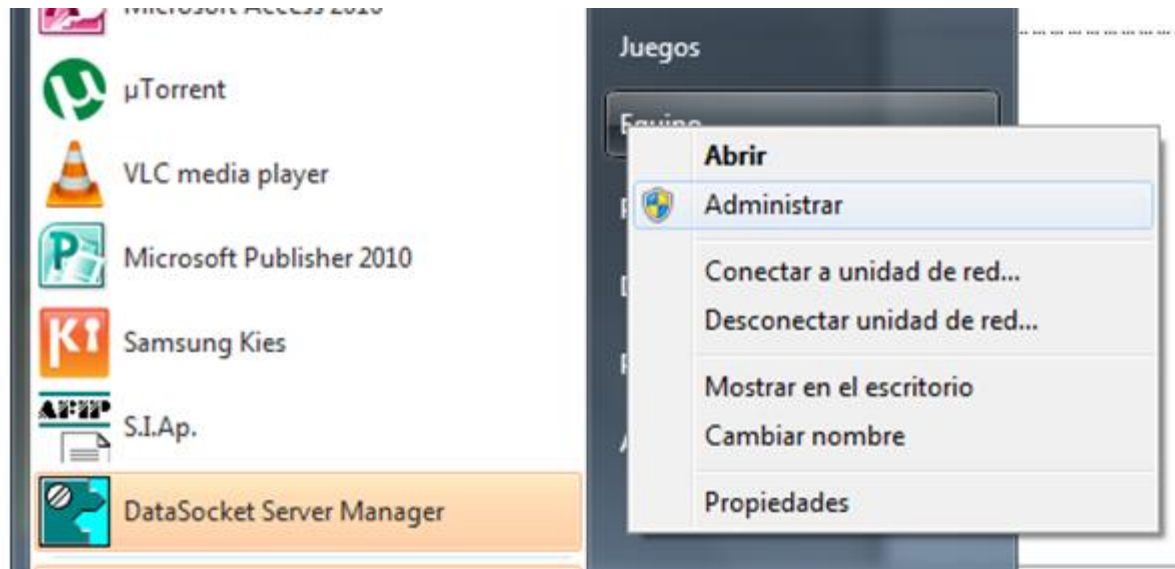
Instalar VISA v5.03 (Virtual Instrument Software Architecture)

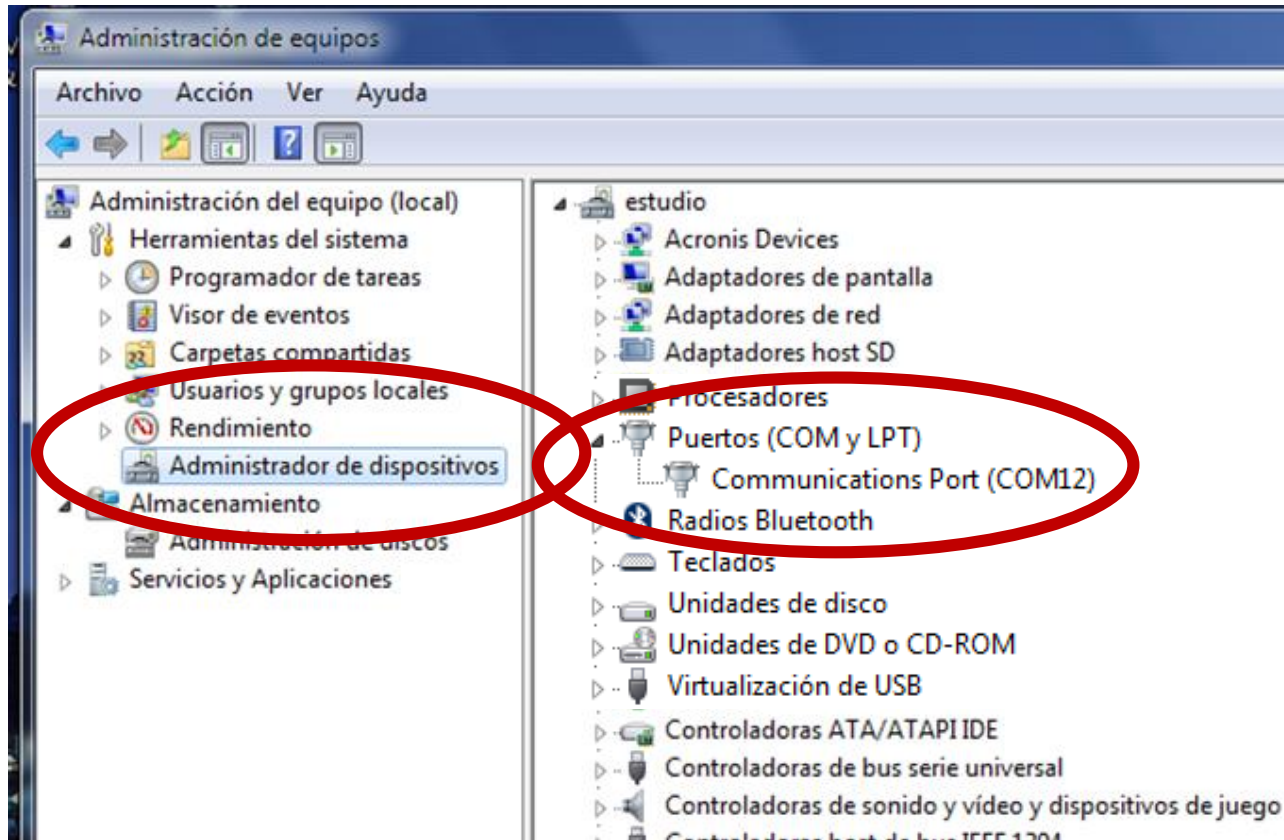


Nombre	Ext	Tamaño	Fecha	Atrib.
[.]		<DIR>	27/08/2011 16:39---	
[driver usb pic 18F2550]		<DIR>	25/08/2011 20:13---	
visa503runtime	exe	74,186,752	22/08/2011 14:06-a-	

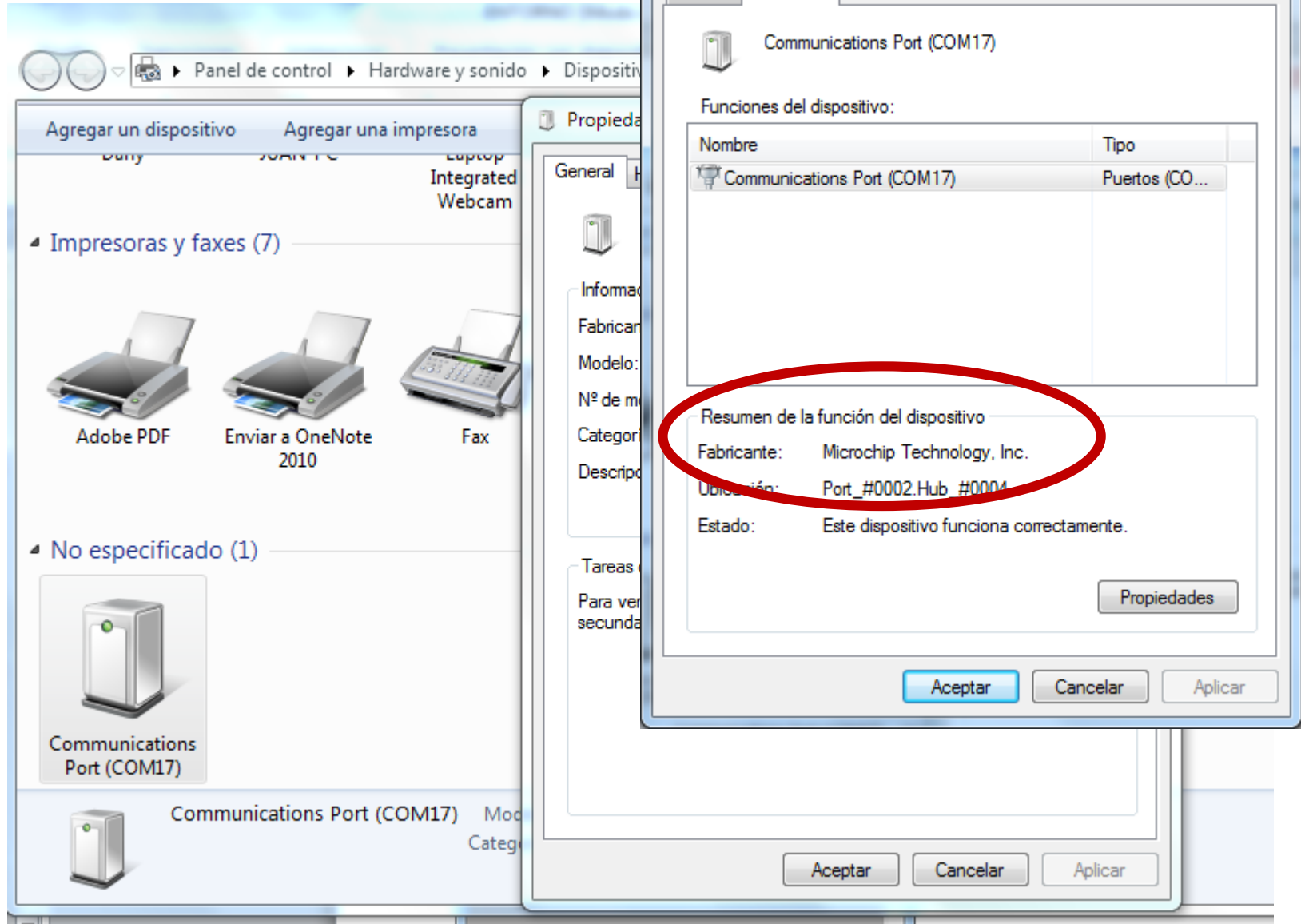
Instalar driver usb pic 18F2550

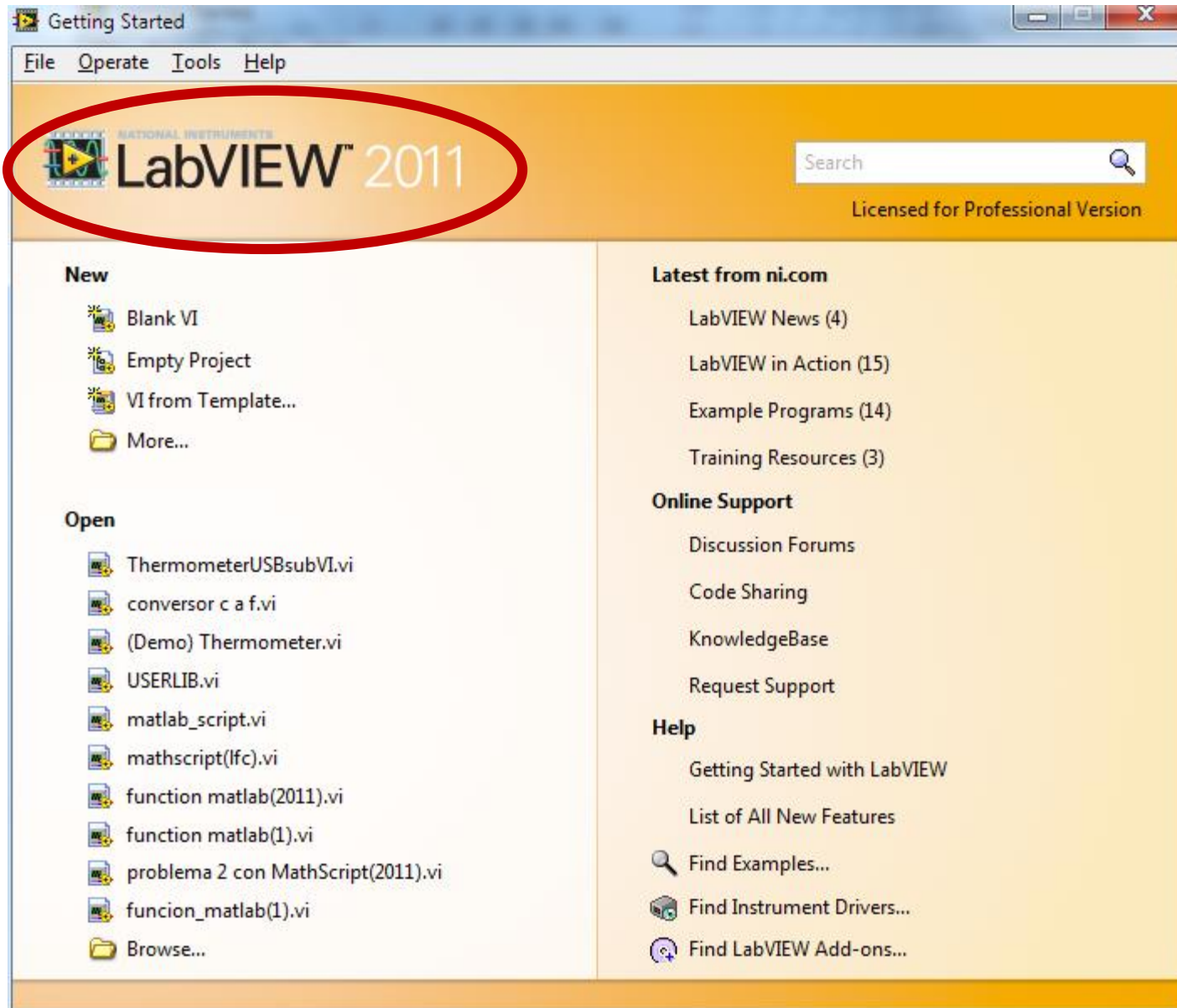






Instalar driver del PIC





Getting Started





File Operate Tools Help

 NATIONAL INSTRUMENTS
LabVIEW™ 2011












Search 

Licensed for Professional Version

New

-  Blank VI
-  Empty Project
-  VI from Template...
-  More...

Open

-  ThermometerUSBsubVI.vi
-  conversor c a f.vi
-  (Demo) Thermometer.vi
-  USERLIB.vi
-  matlab_script.vi
-  mathscript(lfc).vi
-  function matlab(2011).vi
-  function matlab(1).vi
-  problema 2 con MathScript(2011).vi
-  funcion_matlab(1).vi
-  Browse...

Latest from ni.com

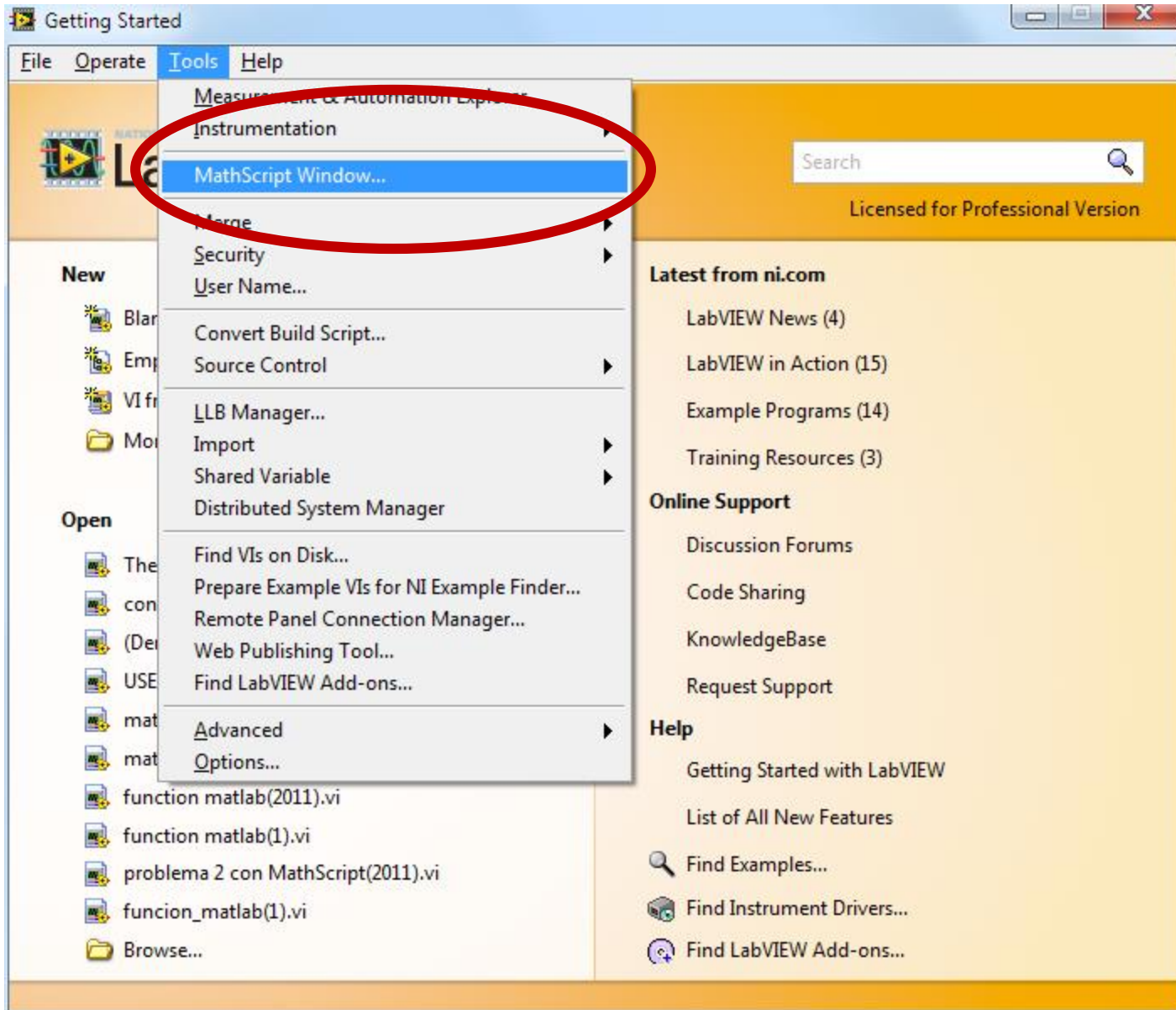
- LabVIEW News (4)
- LabVIEW in Action (15)
- Example Programs (14)
- Training Resources (3)

Online Support

- Discussion Forums
- Code Sharing
- KnowledgeBase
- Request Support

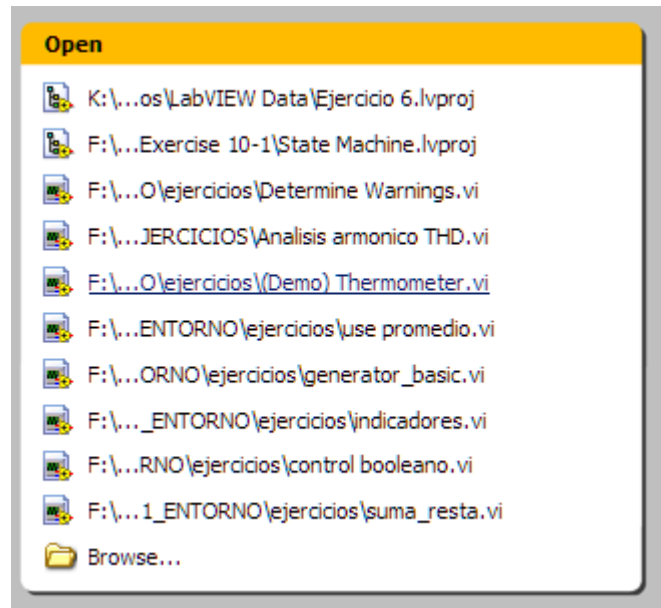
Help

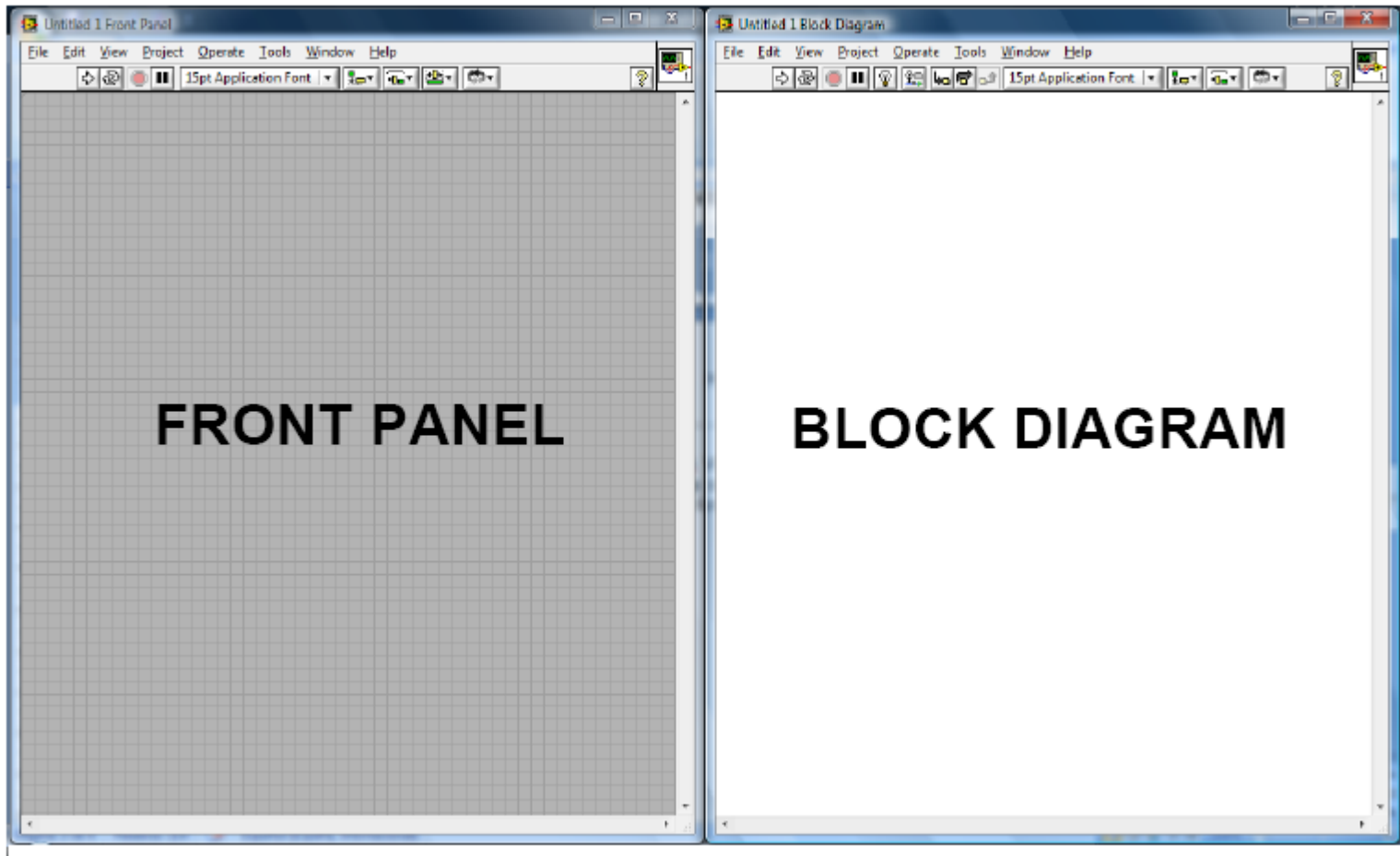
- Getting Started with LabVIEW
- List of All New Features
-  Find Examples...
-  Find Instrument Drivers...
-  Find LabVIEW Add-ons...



Instrumentos Virtuales (VIs)













- Los programas desarrollados en LabVIEW son llamados instrumentos virtuales o VIs.
- En apariencia y operación imitan a instrumentos físicos, tales como osciloscopios e instrumentos analógicos y digitales.
- *Las aplicaciones son grabadas con extensión .vi*





Controls Search

Modern

 Numeric	 Boolean	 String & Path
 Array, Matrix...	 List, Table & ...	 Graph
 Ring & Enum	 Containers	 I/O
 Variant & Cl...	 Decorations	 Refnum

Express

User Controls

Select a Control...
















Tools



Functions

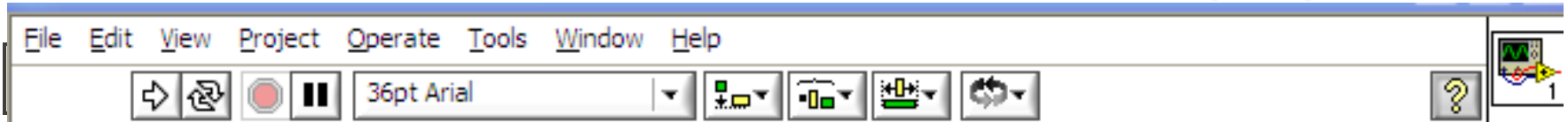
Search View

▼ Programming

 Structures	 Array	 Cluster & Vari...
 Numeric	 File I/O	 Boolean
 String	 Comparison	 Timing
 Dialog & User...	 Waveform	 Application C...
 Synchronization	 Graphics & So...	 Report Gener...

- ▶ Measurement I/O
- ▶ Instrument I/O
- ▶ Vision and Motion
- ▶ Mathematics
- ▶ Signal Processing
- ▶ Data Communication
- ▶ Connectivity
- ▶ Control Design & Simulation
- ▶ SignalExpress
- ▶ Express
- ▶ Addons
- ▶ Favorites
- ▶ User Libraries
- Select a VI...

Barra de Herramientas de Estado (PF)



  **Botón de ejecución (Run)**

  **Botón de ejecución continua (Continuous Run)**

 **Cancelación de ejecución (Abort Execution)**

 **Botón de pausa/continuación**

 **Configuración de textos (Text Settings)**

 **Alineamiento de objetos (Align Objects)**

 **Distribución de objetos (Distribute Objects)**

 **Reordenamiento**

 **Redimensionamiento de objetos de panel frontal (Resize Objects)**

Botones adicionales en el diagrama de la barra de herramientas



**Botón de ejecución resaltada
(Highlight Execution)**



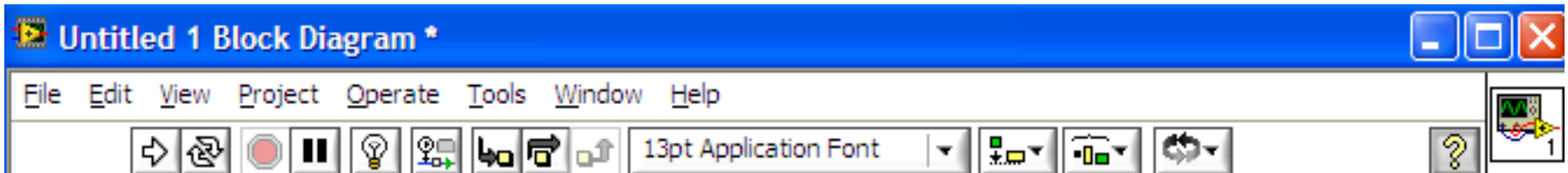
**Botón de entrada al ciclo
(Step Into)**



Botón sobre (Step Over)



**Botón de salida del ciclo
(Step Out)**



Paleta de Herramientas



- Paleta flotante
- Utilizado para operar y modificar objetos en el panel frontal y en el diagrama de bloques.



Herramienta de selección automática



Herramienta de operación



Herramienta de desplazamiento



**Herramienta de posicionamiento
y redimensión**



Herramienta de punto de paro



Herramienta de prueba



Herramienta de etiquetado



Herramienta para copia de color



Herramienta de cableado



Herramienta para colorear

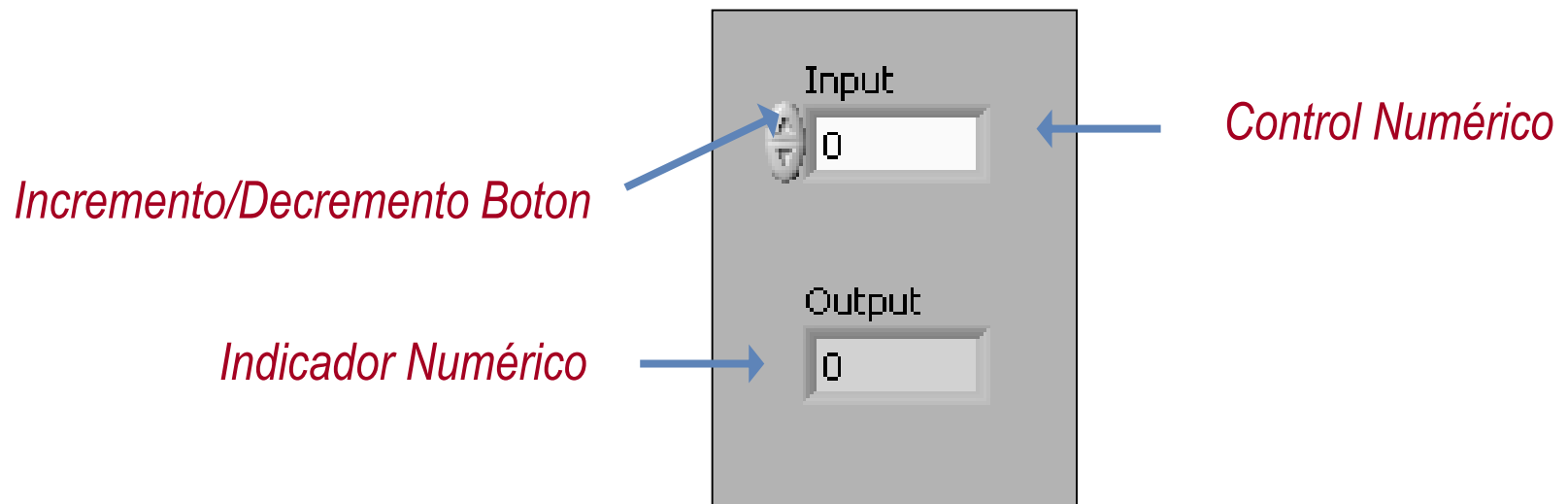


Herramienta de menú (atajo)

Panel Frontal

Controles Numéricos - Indicadores

- El tipo de dato numérico puede presentarse como números de varios tipos, tales como enteros o reales



Untitled 1 Block Diagram *

File Edit View Project Operate Tools Window Help

13pt Application Font

Tools

Functions

Search View

Programming

- Structures
- Array
- Cluster & Vari...
- Numeric
- File I/O
- Boolean
- String
- Comparison
- Timing
- Dialog & User...
- Waveform
- Application C...
- Synchronization
- Graphics & So...
- Report Gener...

Measurement I/O

Instrument I/O

Vision and Motion

Mathematics

Signal Processing

Data Communication

Connectivity

Control Design & Simulation

SignalExpress

Express

Addons

Favorites

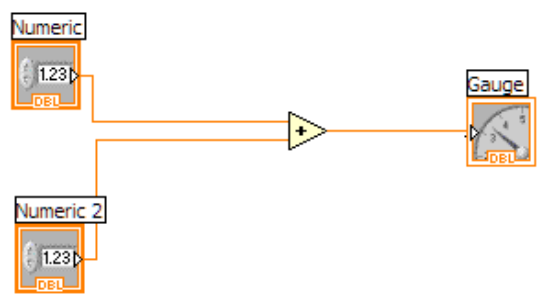
User Libraries

Select a VI...

The screenshot shows the LabVIEW software interface. At the top is a menu bar with 'File', 'Edit', 'View', 'Project', 'Operate', 'Tools', 'Window', and 'Help'. Below the menu bar is a toolbar with various icons, including a green arrow icon highlighted with a green box. The main workspace contains a block diagram with three numeric display indicators labeled 'Numeric', 'Numeric 2', and 'Gauge', each showing the value '1.23'. A connector symbol (a triangle with a plus sign) is also present. On the right side, there is a 'Functions' palette with a search bar and a 'View' dropdown. The palette is organized into categories: 'Programming' (with sub-categories like Structures, Array, Cluster & Vari...), 'Measurement I/O', 'Instrument I/O', 'Vision and Motion', 'Mathematics', 'Signal Processing', 'Data Communication', 'Connectivity', 'Control Design & Simulation', 'SignalExpress', 'Express', 'Addons', 'Favorites', 'User Libraries', and 'Select a VI...'. A 'Tools' palette is also visible in the center-right area of the workspace.



Tools



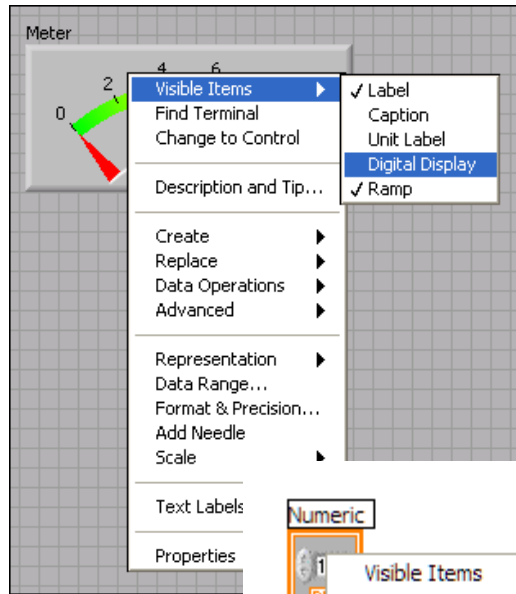
Functions

Search [View]

- Programming
 - Structures
 - Array
 - Cluster & Vari...
 - Numeric
 - File I/O
 - Boolean
 - String
 - Comparison
 - Timing
 - Dialog & User...
 - Waveform
 - Application C...
 - Synchronization
 - Graphics & So...
 - Report Gener...
- Measurement I/O
- Instrument I/O
- Vision and Motion
- Mathematics
- Signal Processing
- Data Communication
- Connectivity
- Control Design & Simulation
- SignalExpress
- Express
- Addons
- Favorites
- User Libraries
- Select a VI...

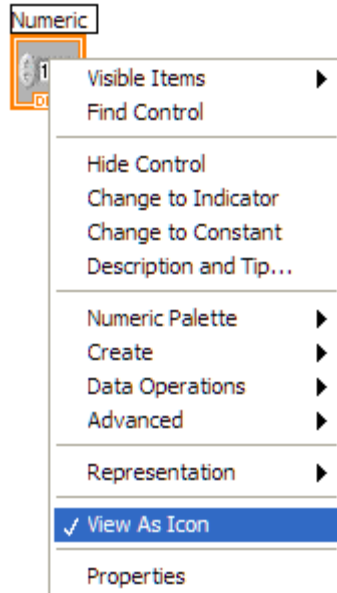
OPCIONES DE VISUALIZACIÓN DE LOS DATOS TIPO CONTROL

PF



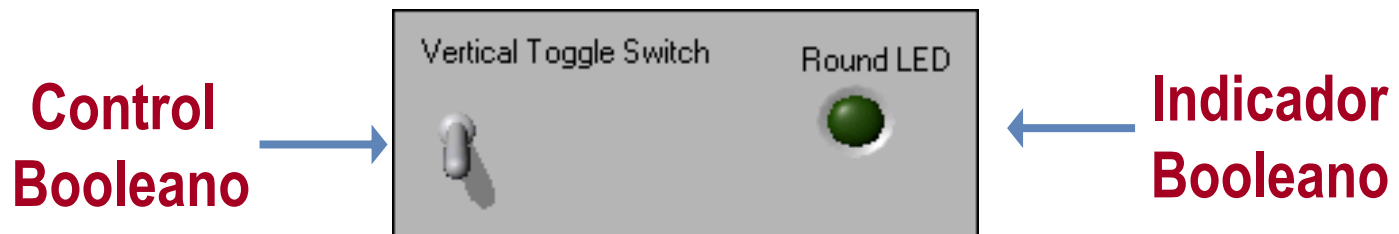
- Label/Caption
- Date Range
- Format and Precision
- Incremento/decremento
- Unit Label

DB



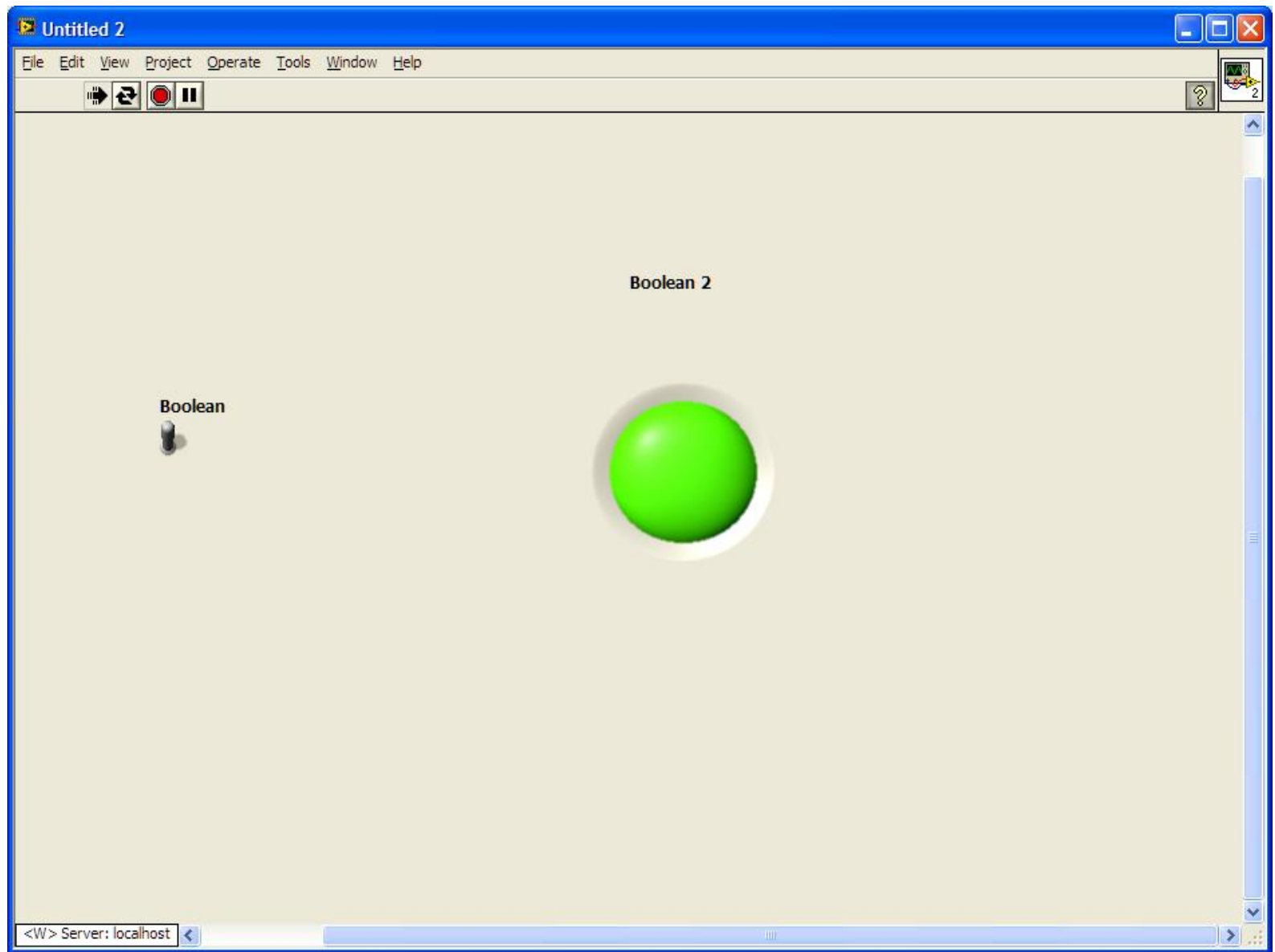
Panel Frontal –Controles/Indicadores

- Un dato Booleano puede representar únicamente dos estados Verdadero o Falso (On - Off)
- Los controles e indicadores Booleanos son usados para desplegar valores verdaderos o falsos
- Objetos Booleanos simulan interruptores y LEDs





Ejercicio



Partes de un VI –Diagrama de Bloques

- Contiene el código fuente gráfico.
- Los objetos del panel frontal aparecen como terminales en el diagrama de bloque.

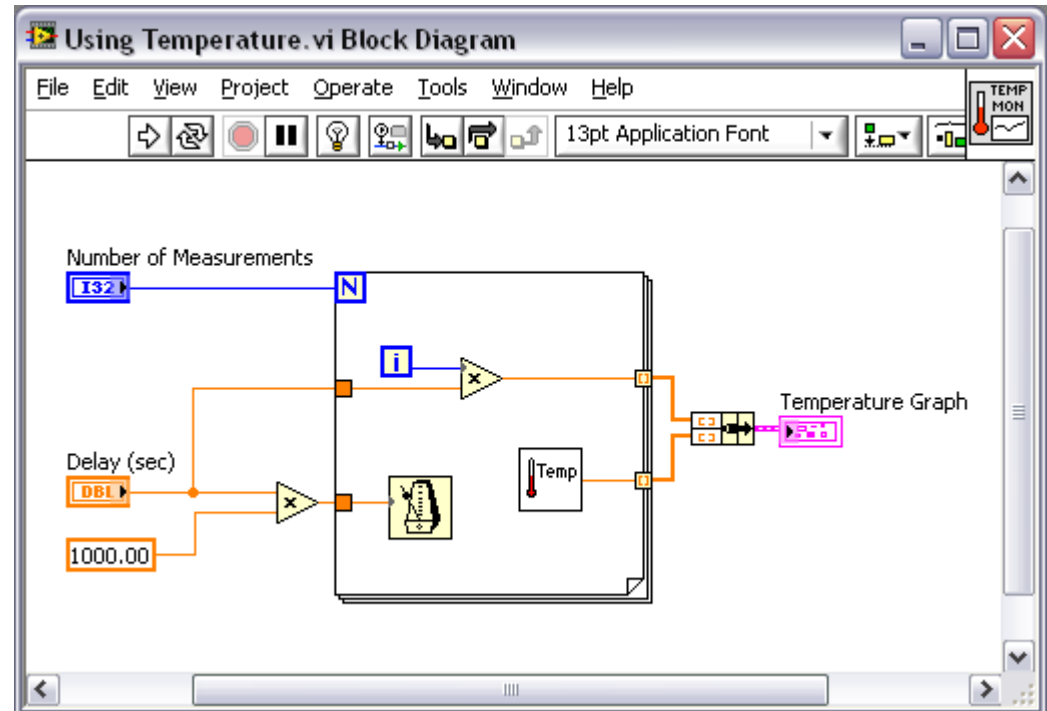


Diagrama de Bloque

- Incluye objetos como los siguientes:
 - Terminales
 - SubVIs
 - Funciones
 - Constantes
 - Estructuras
 - Cables

Diagrama de bloques – Cableado

- Transfiere datos entre objetos en el diagrama de bloques.
- Los cables tienen diferentes colores, estilos, y grosor dependiendo del tipo de datos.
- La ruptura de un cableado aparece como una línea punteada en color negro y una X en la mitad.












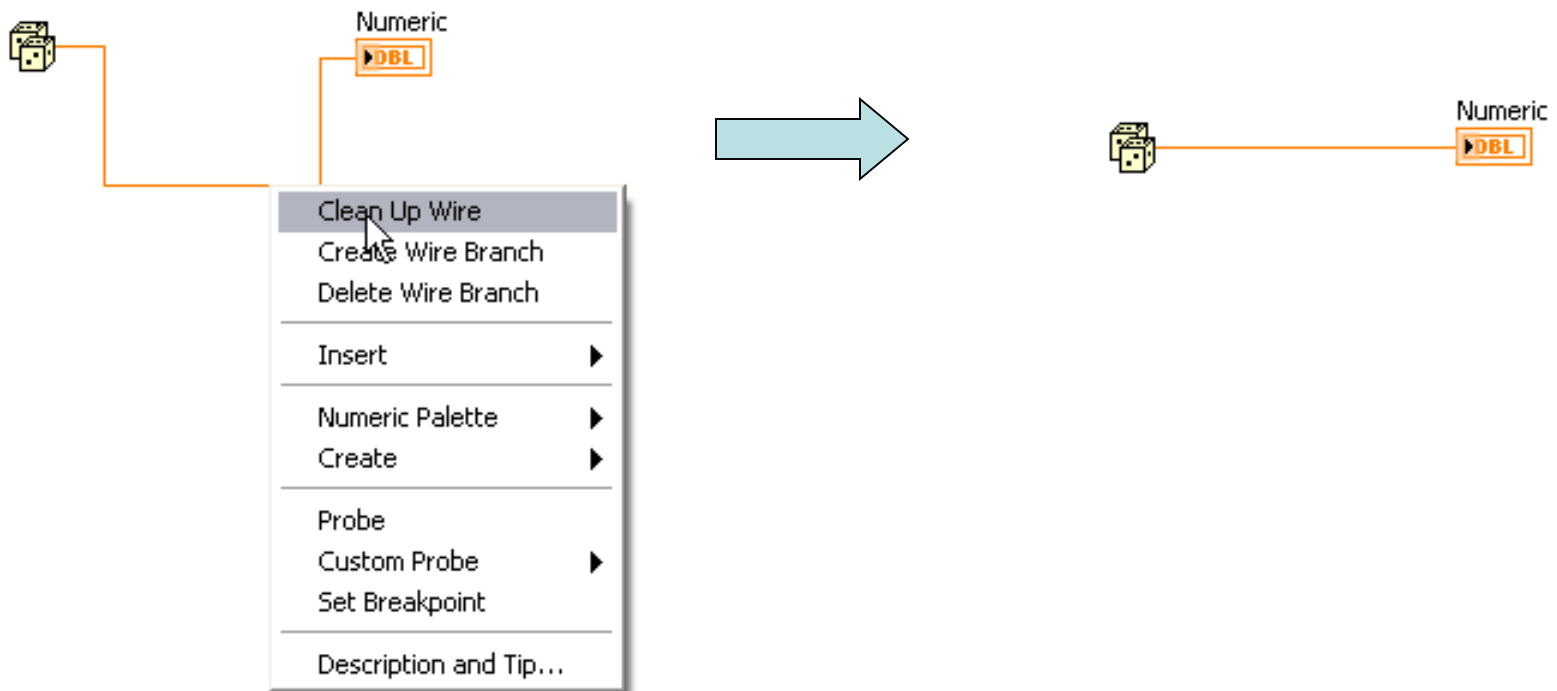
	DBL Numeric	Integer Numeric	String
Scalar			
1D Array			
2D Array			

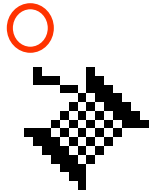
Diagrama de bloques – Consejos en el cableado.

- Presionar <Ctrl>-B para borrar cableado cortado.
- Click Derecho y seleccione “Clean Up Wire” para limpiar el cableado

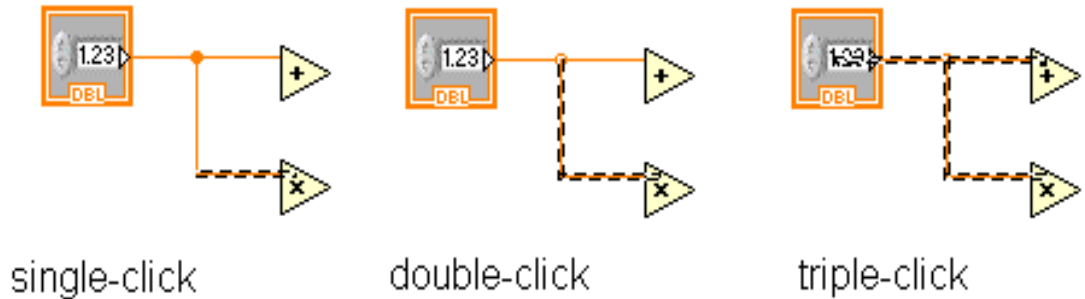


Consejos Para Conectar – Diagrama de Bloques

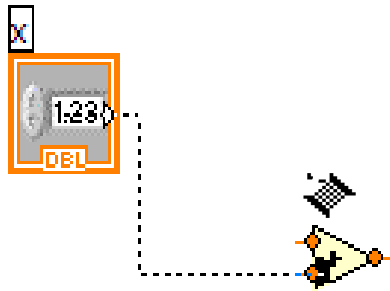
“Punto Contacto” de Cableado



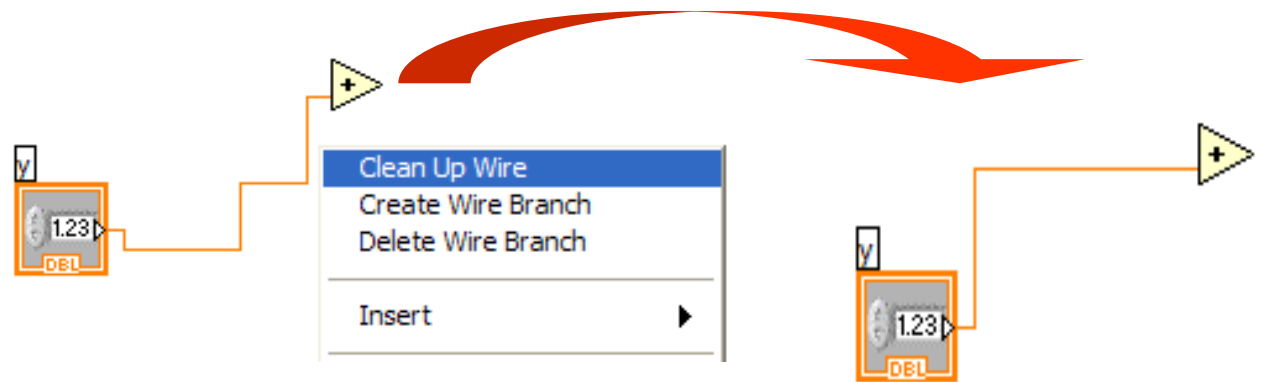
Haga Clic para Seleccionar los Cables



Utilice la Ruta Automática del Cable



Limpiando el Cableado



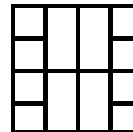
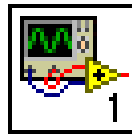
Seleccionando una herramienta

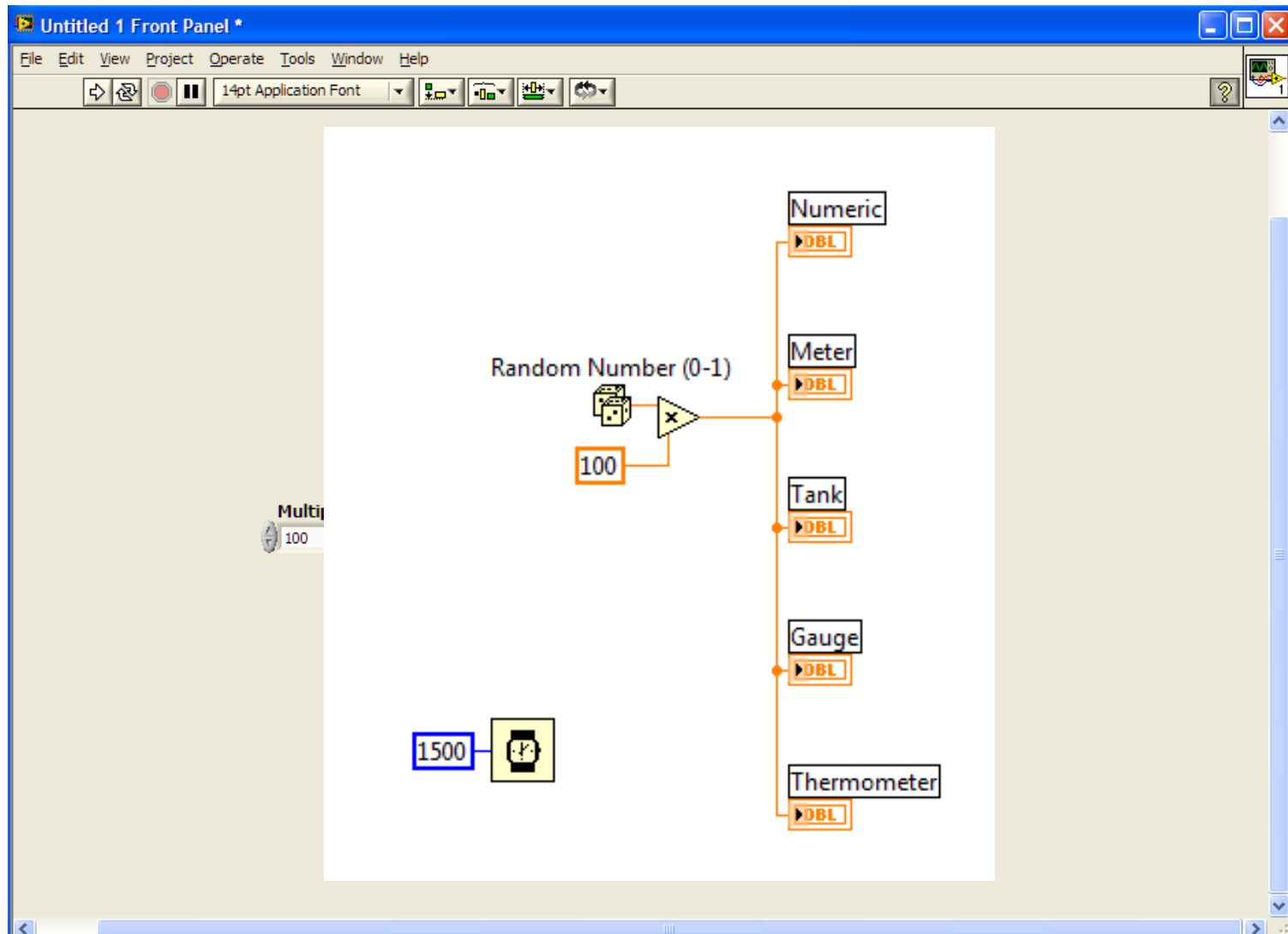
- Creando, modificar, y depurando VIs usando herramientas provistas por LabVIEW
- Una herramienta es un modo especial de activación del cursor del mouse.
- Cuando usamos “Automatic Tool Selection”, LabVIEW selecciona la herramienta correcta en función de la ubicación del mouse sobre los objetos.



Partes de VI – Icono/Conector

- Icono: representación gráfica de un VI
- Panel Conector: mapa de entradas y salidas VI
- Iconos y Panel Conector son necesarios para usar un VI como un subVI
 - Un subVI es un VI que está dentro de otro VI
 - Similar a una función en un lenguaje de programación convencional de texto
 - Icono Connector Panel



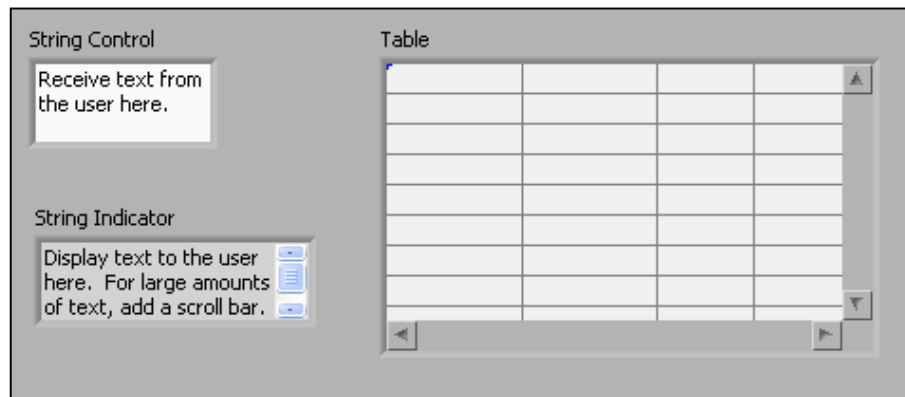


Ejemplo

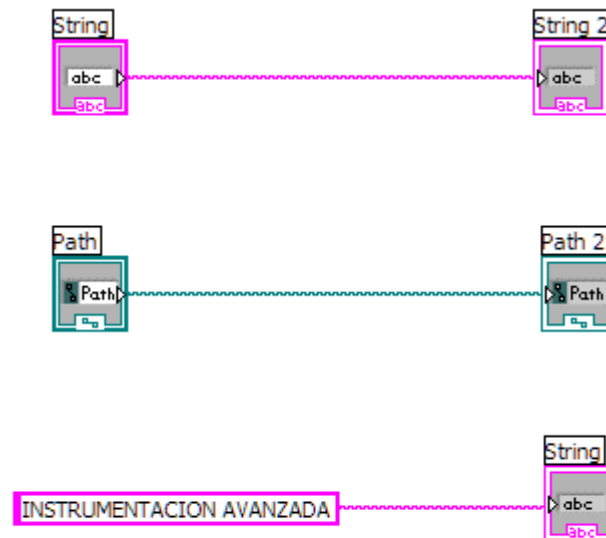
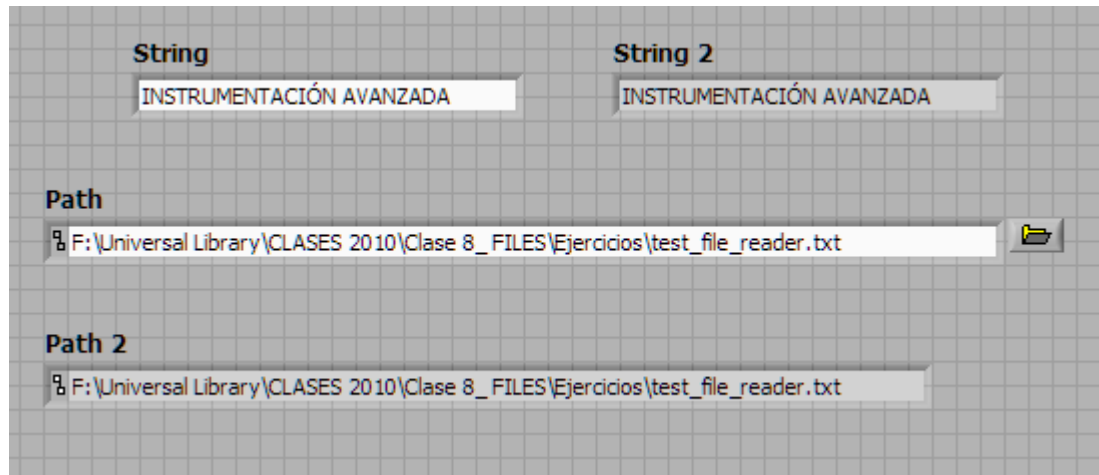
Generar un número aleatorio entre los valores 0 y 100 para ser leído por distintos indicadores de LabVIEW

Panel Frontal – Strings (cadena)

- Un “string” es una secuencia de caracteres ASCII.
- Los controles “strings” sirven para recibir texto desde el usuario p.e. nombre y password para abrir un programa.
- Los indicadores “strings” se usan para desplegar texto para el usuario.



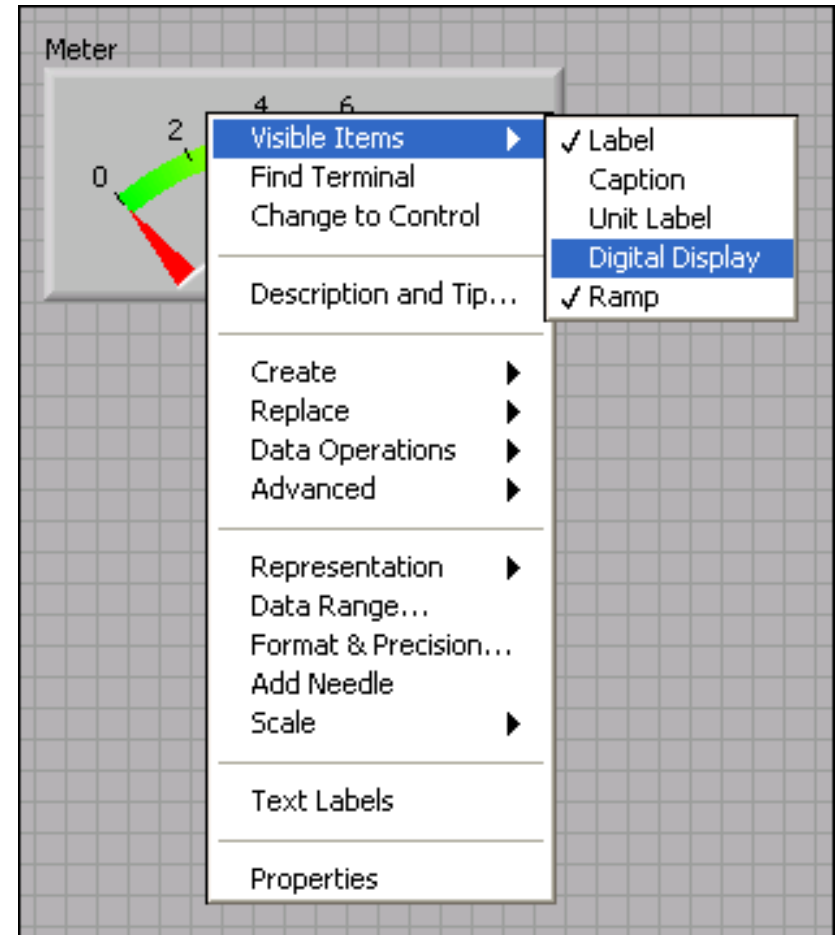
Panel Frontal – Strings (cadena) y Path



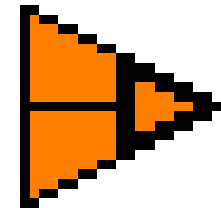
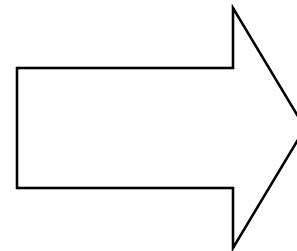
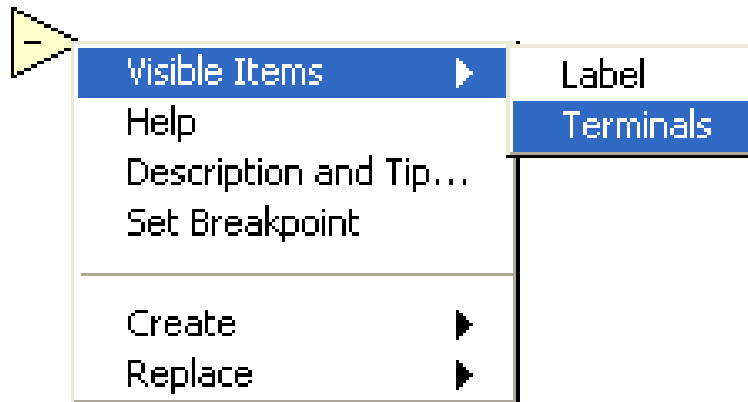
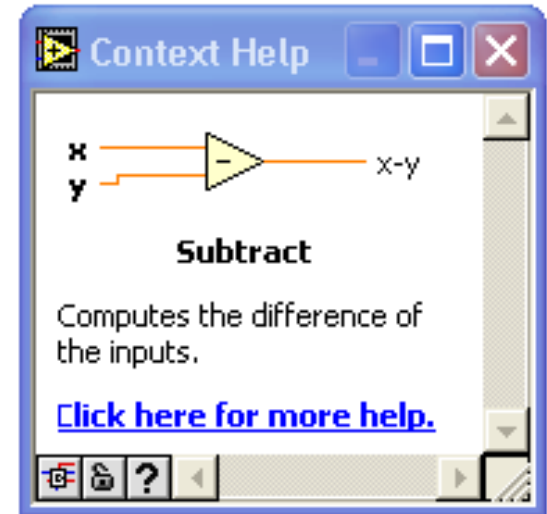
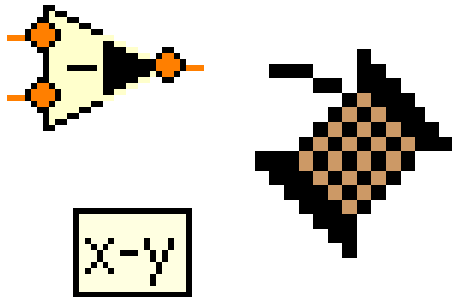
Ejemplo

Panel Frontal

- Todos los objetos de LabVIEW tienen asociados “menús flotantes”.
- Para crear un VI, usamos “menú flotante” para cambiar la vista o comportamiento de objetos en PF o en DdB
- Para acceder al “Menú Flotante” hacemos clic derecho sobre el objeto.



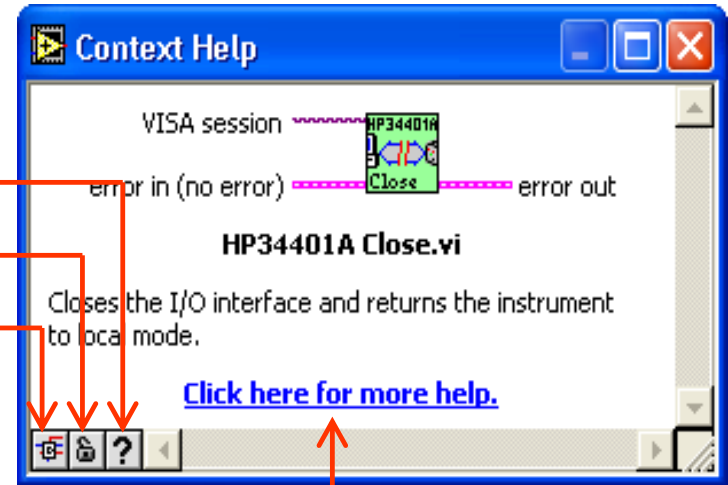
Creando un VI – Diagrama de Bloques



Opciones de Ayuda

Contexto de la Ayuda

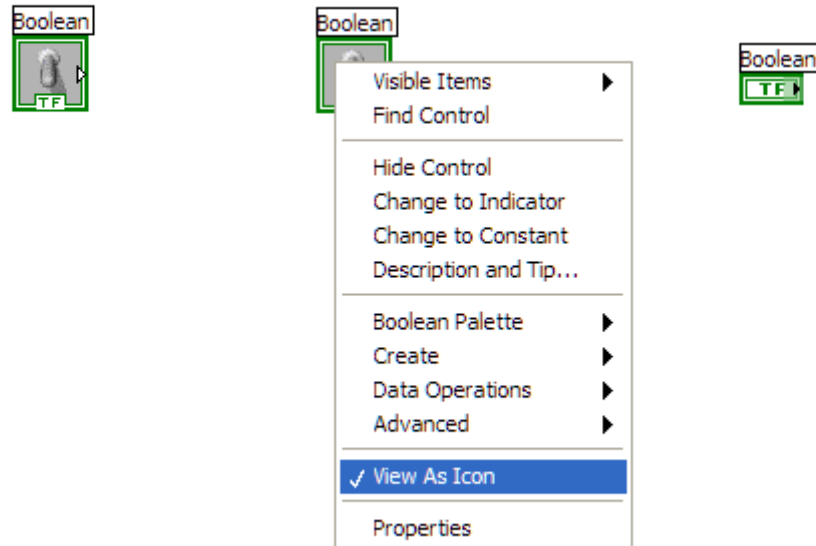
- Ayuda en línea
- Congelar Ayuda
- Path ubicación
- Ctrl + H



Referencias en Línea

- Clic en las funciones del diagrama para tener acceso directo a la información en línea.

DB – Icons/Expandible Nodes



Técnicas para Eliminar Errores

- **Encontrando los Errores**



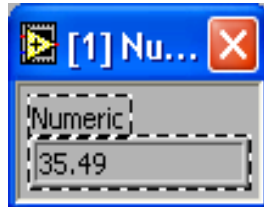
Haciendo clic en el botón de “RUN” que esta roto; aparece una ventana mostrando los errores

- **Resaltar la Ejecución**



Con clic en el botón de ejecución resaltada; el flujo de datos aparece animado utilizando burbujas desplegando valores en los cables.

- **Herramienta de Prueba**



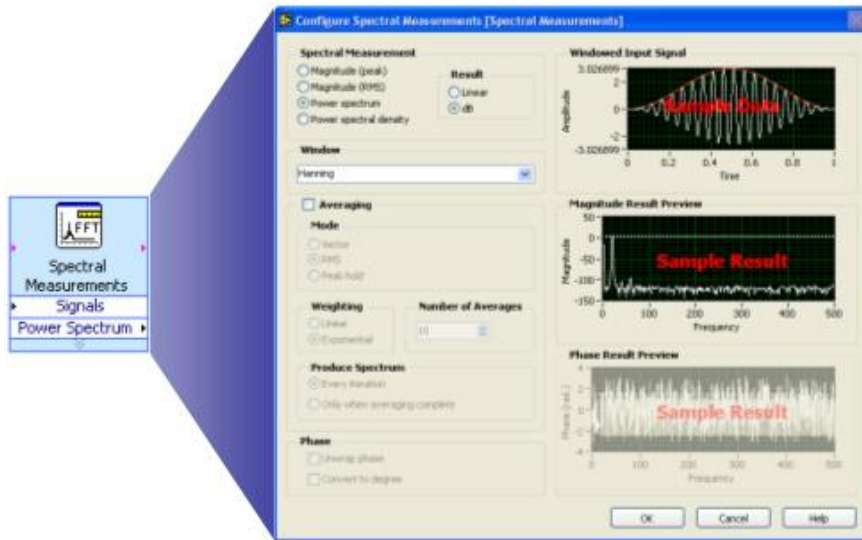
Con clic en el botón derecho sobre el cable se exhibe la ventana de prueba y mostrando los datos mientras fluyen por el segmento de cable.



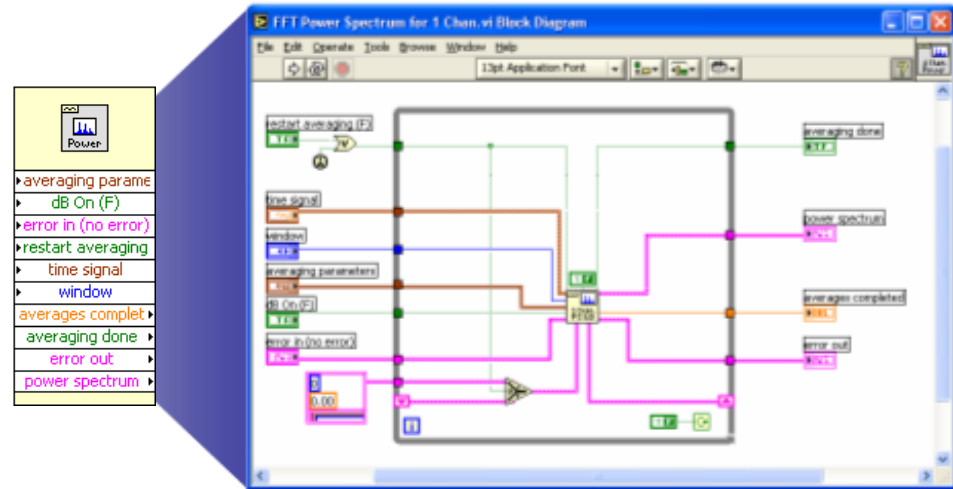
También se puede seleccionar la herramienta de prueba desde la paleta de herramientas y hacer un clic en el cable.

VIs Express, Vis Standard y Funciones

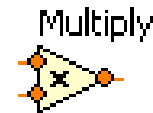
- VIs Express: VIs interactivos con página de diálogo configurable
- VIs estándar: VIs modulares y personalizables mediante cableado
- Funciones: Elementos fundamentales de operación de LabVIEW. No quedan representado en el panel frontal.



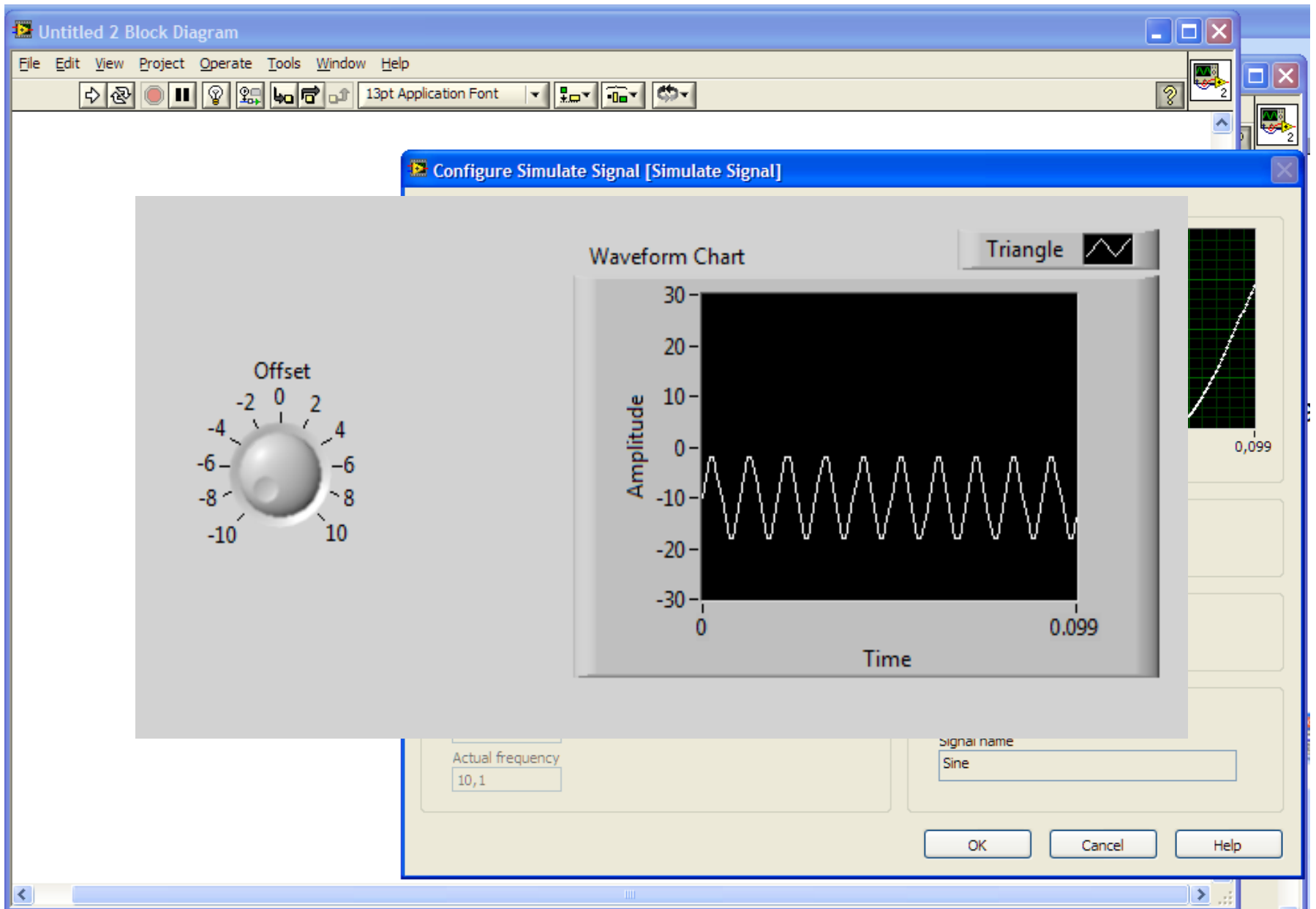
VI Express



VI Estandar



Función



Ejemplo

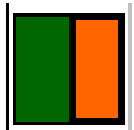
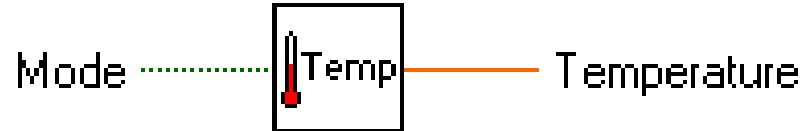
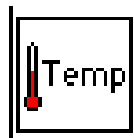
SubVIs

- Un SubVI es un VI que puede ser utilizado dentro de otro VI
- Similar a una subrutina
- Ventajas
 - Modular
 - Fácil para eliminar errores
 - No tiene que crear códigos
 - Requiere menos memoria

Pasos para crear un SubVI

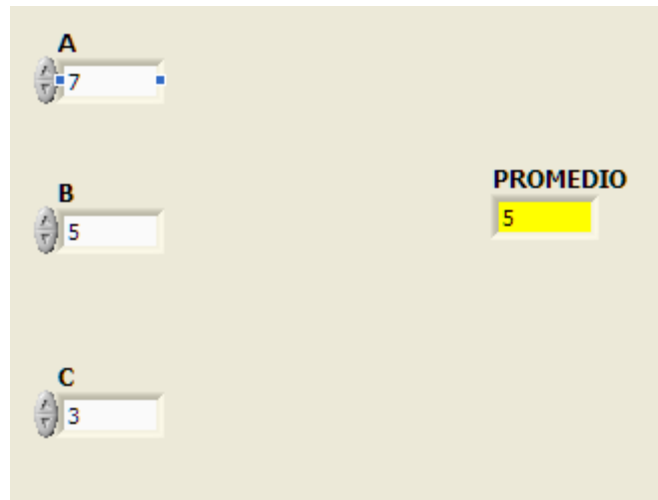
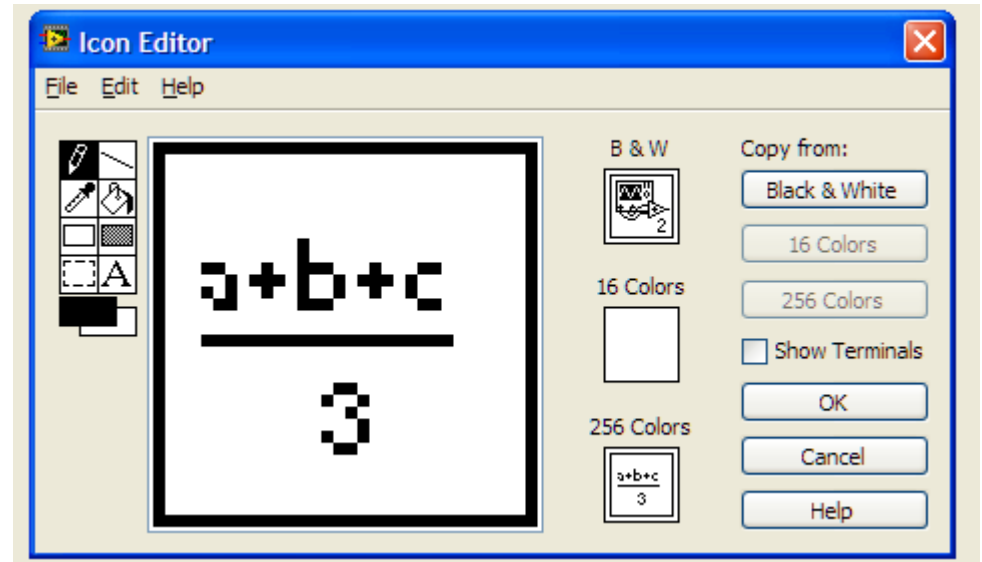
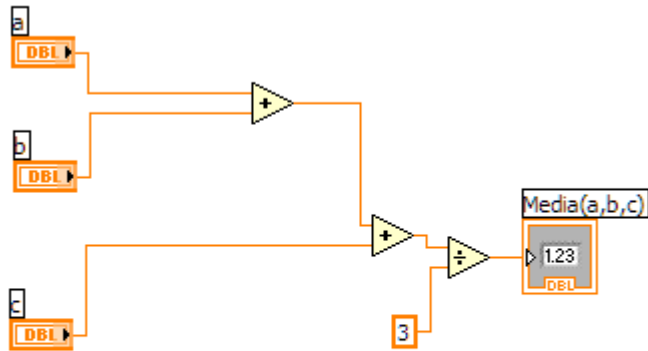
- *Crear el icono*
- *Crear el conector*
- *Asignar terminales*
- *Guardar el VI*
- *Insertar el VI dentro del VI principal*

Icono y Conector



- Un icono representa un VI en otro diagrama de bloque
- El conector muestra terminales disponibles para transferir datos

Pasos para crear un SubVI

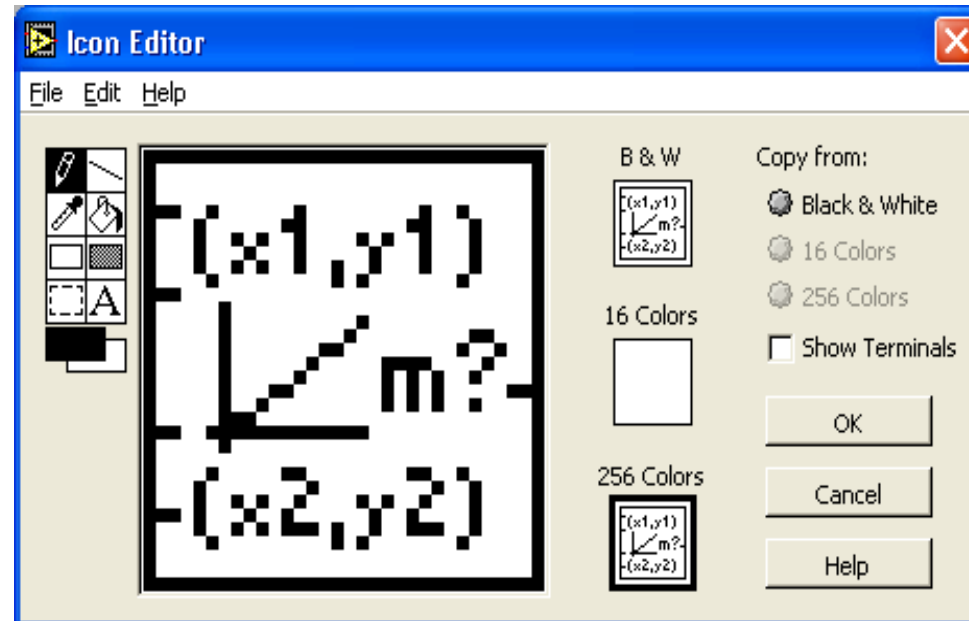


Ejercicio subVI

Ejercicio usando subVI

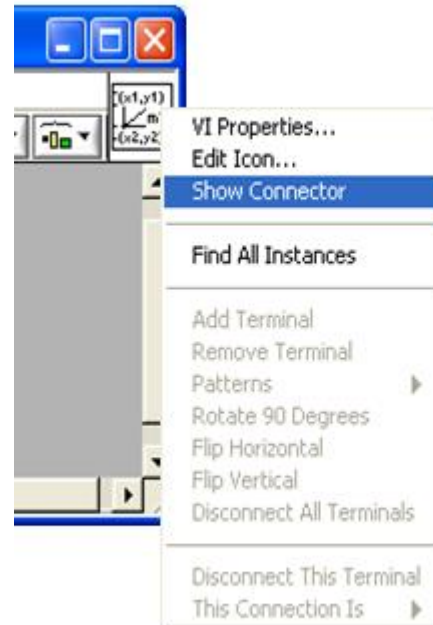
Crear el Icono

- Haga click derecho sobre el icono en el diagrama de bloque o panel frontal

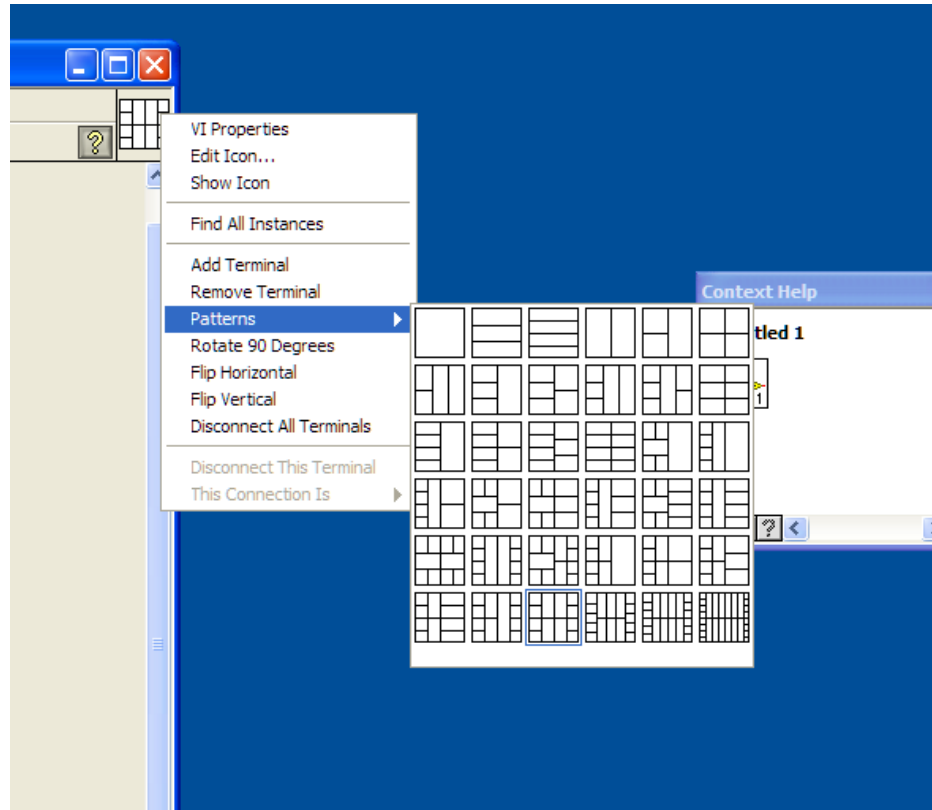


Crear el Conector

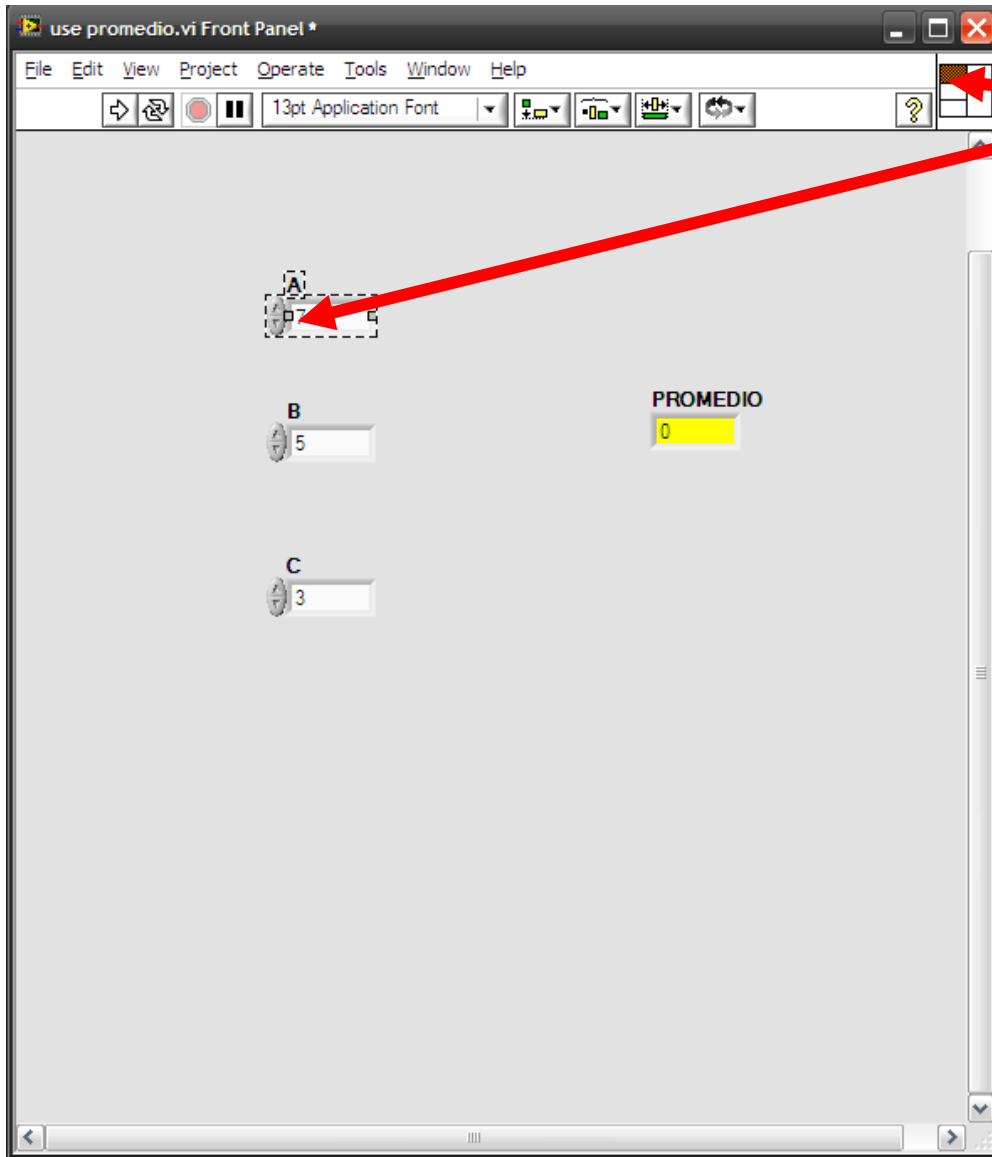
Haga click derecho sobre el icono (solamente en el panel frontal)



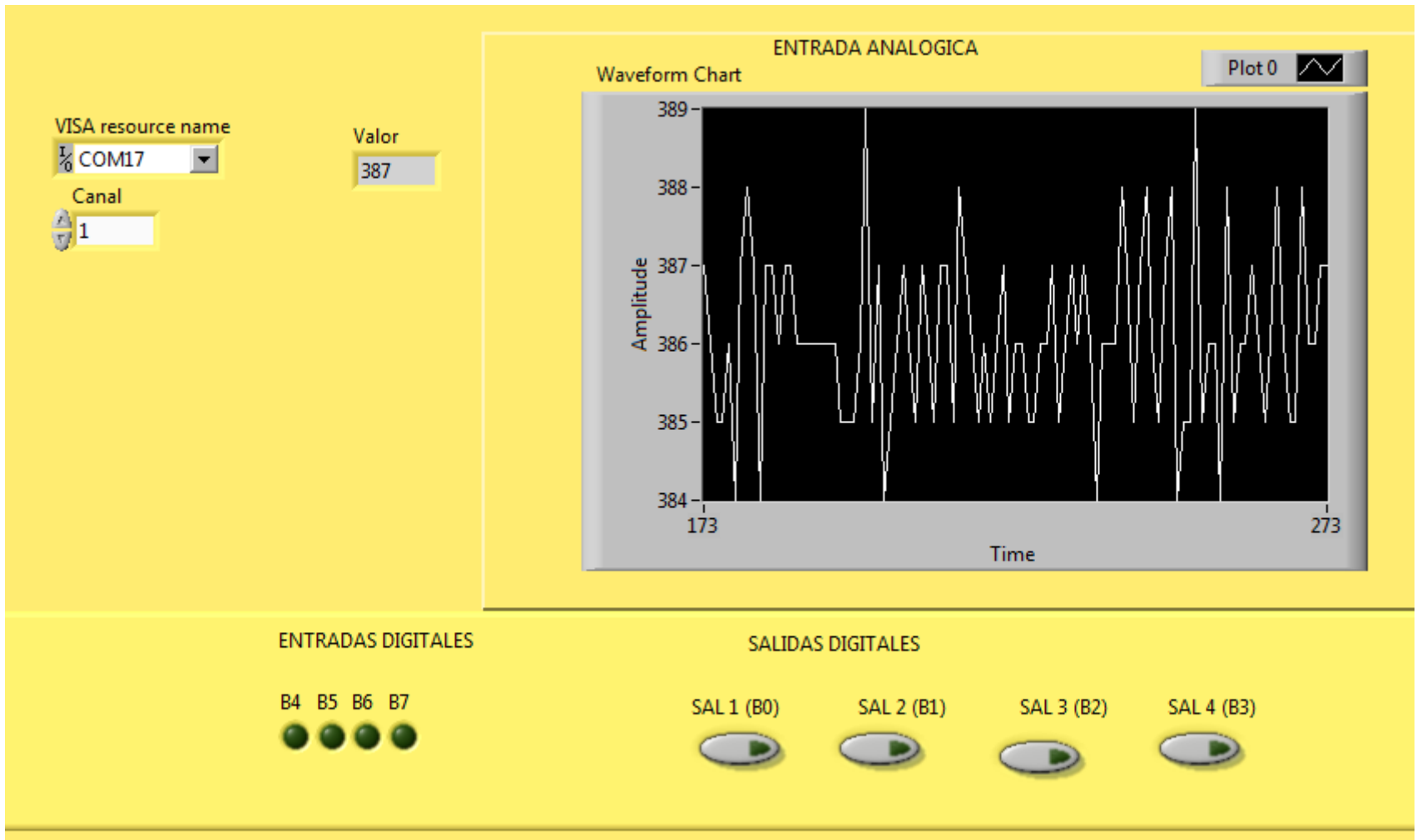
Partes de VI – Icono/Conector



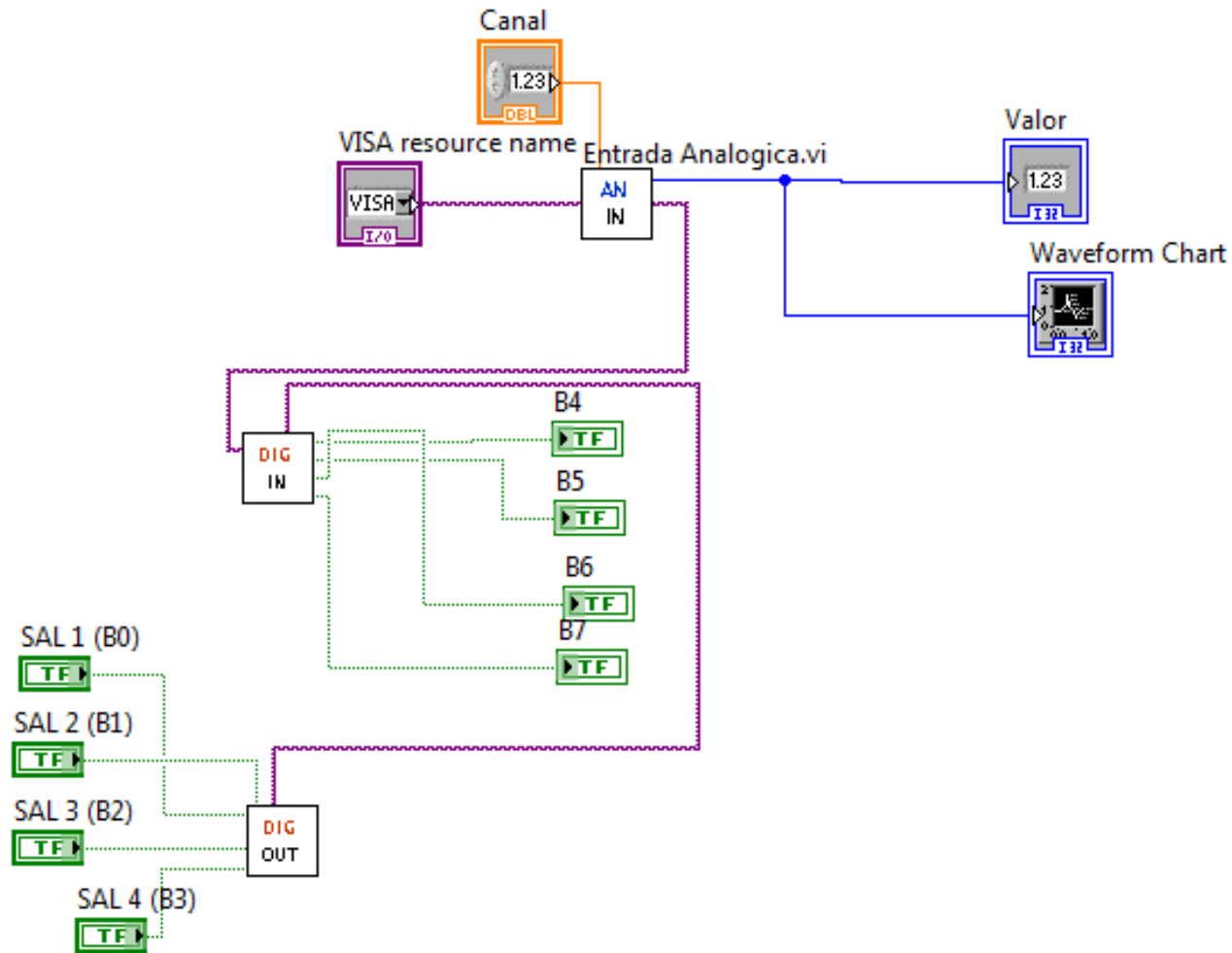
Asignar Terminales



USO DE LA PLACA USB



Ejemplo

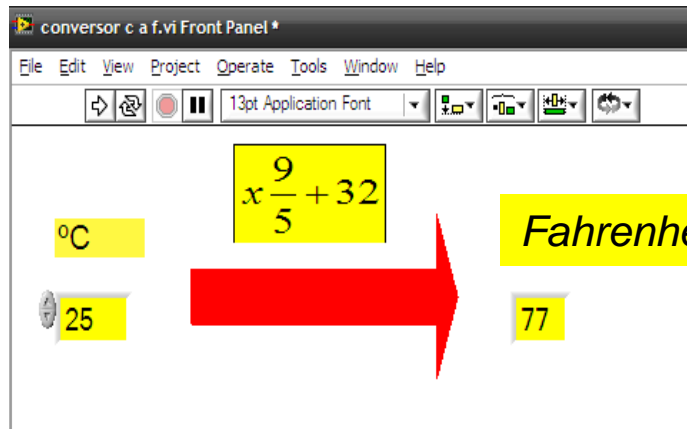


Ejemplo

Ejercicios

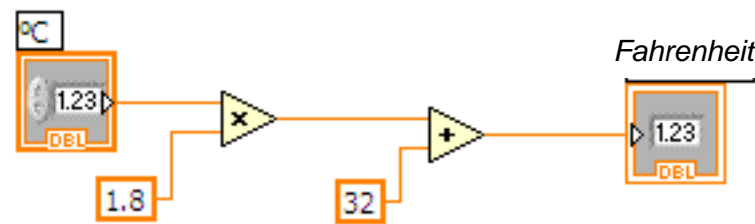
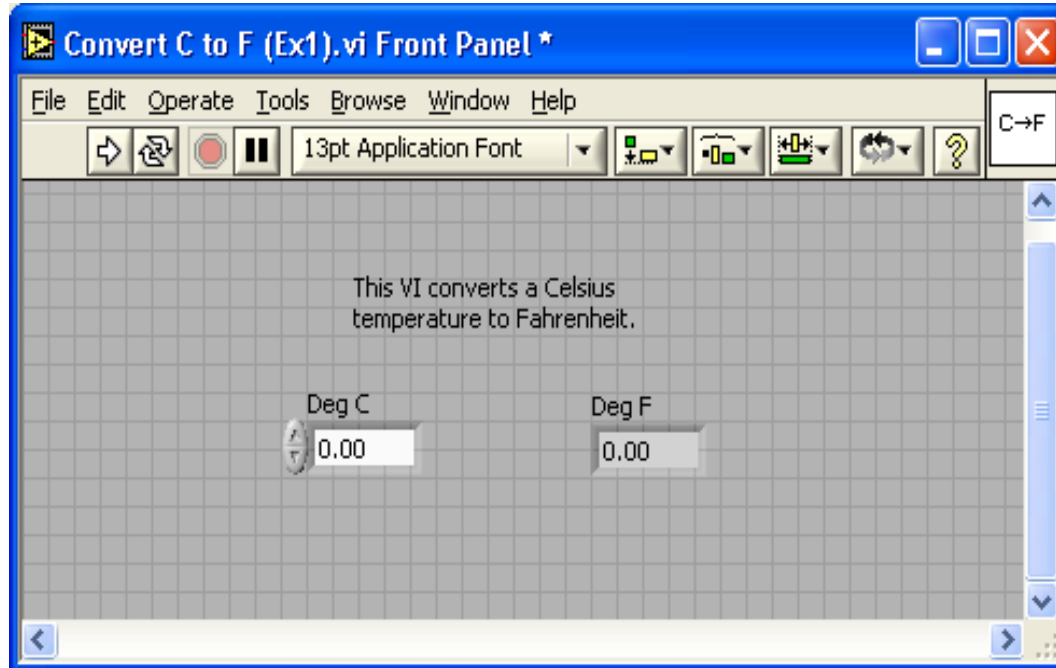
Ejercicio 1.1 – Convertir de °C a °F

Crear un VI que acepta la entrada de una variable «x» (temperatura) desde un control numérico para convertirla en °C a °F y mostrarla en un indicador numérico



$$x \frac{9}{5} + 32$$

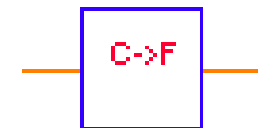
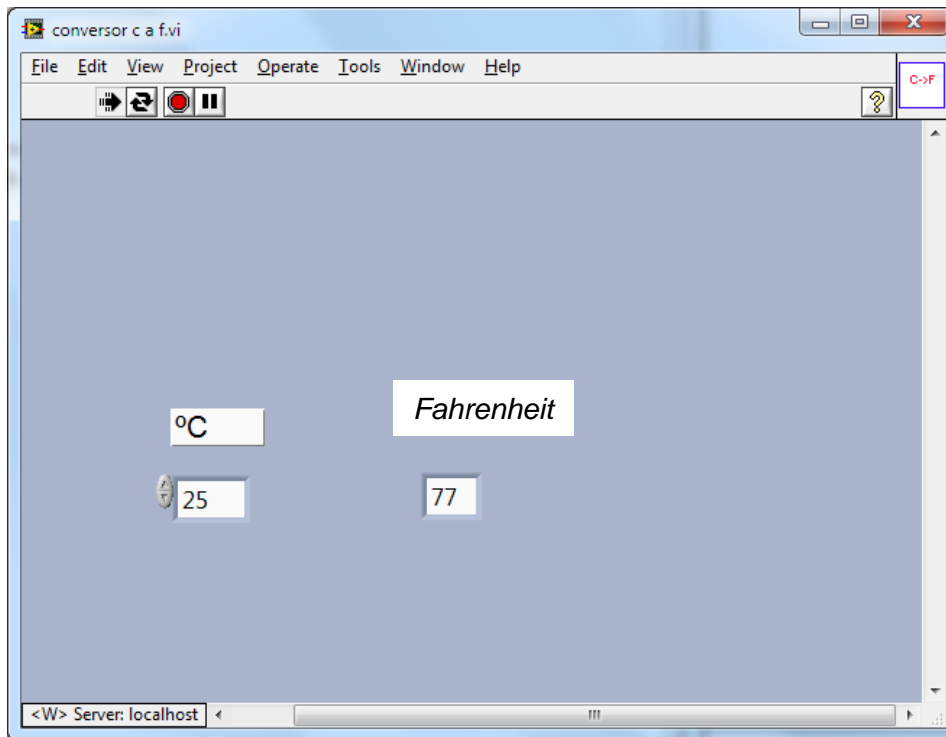
Ejercicio 1.1 – Convertir de °C a °F



Ejercicio 1.1

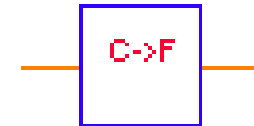
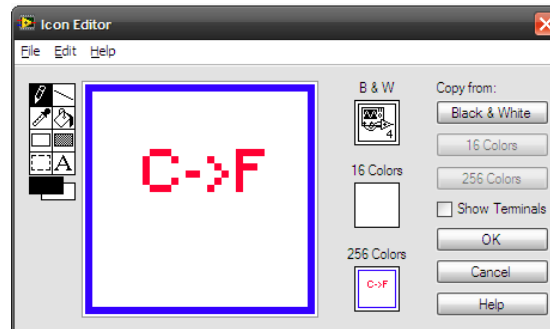
Ejercicio 1.2 – Convertir de °C a °F

Crear un sub-VI a partir de la aplicación anterior.



Ejercicio 1.2 – Convertir de °C a °F

Editar Icon

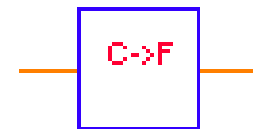
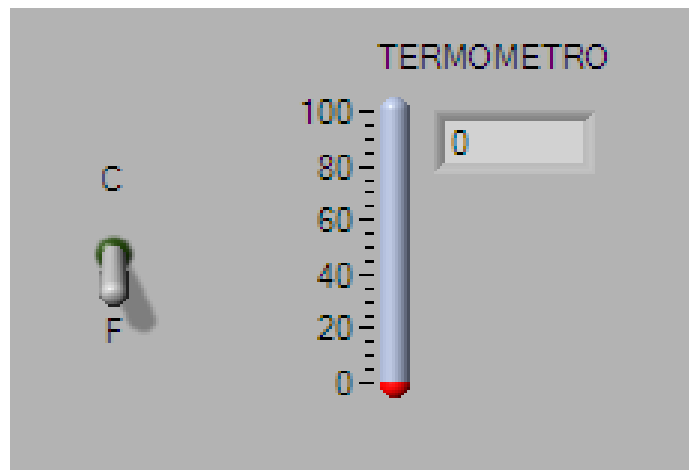


Asignar terminales de conexión



Ejercicio 1.3 – Convertir de °C a °F

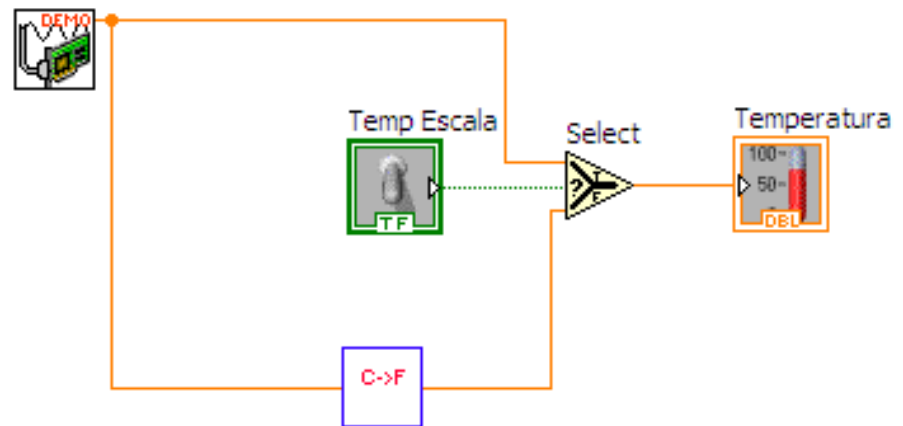
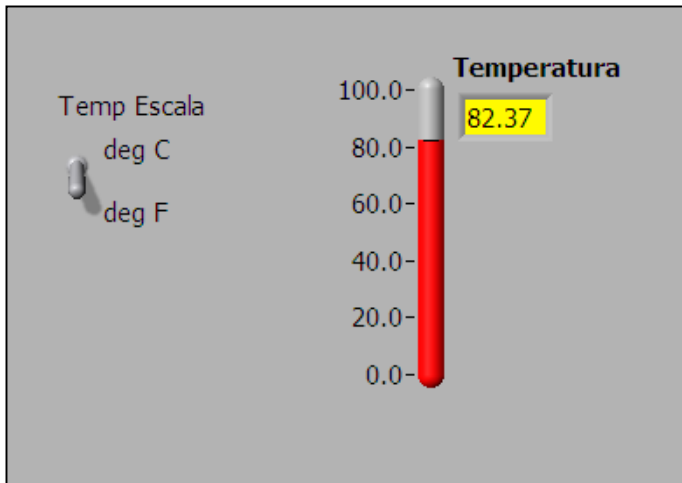
Utilizar el sub-Vi creado en el paso anterior para construir una aplicación que muestre la temperatura resultante en un termómetro, seleccionando con una llave la indicación en °C o °F.



Ejercicio 1.3 – Convertir de °C a °F

The screenshot shows the LabVIEW software interface. The main workspace displays a 'Select' function block with three inputs labeled 't', 's', and 'f', and an output labeled 's? t:f'. A tooltip is visible over the function, stating: 'Returns the value wired to the t input or f input, depending on the value of s. If s is TRUE, this function returns the value wired to t. If s is FALSE, this function returns the value wired to f.' The 'Comparison' palette is open, showing various comparison functions such as 'Equal To?', 'Not Equal To...', 'Greater Than...', 'Less Than?', 'Greater Or Equal To?', 'Less Or Equal To?', 'Select', 'Max & Min', 'In Range and...', 'Not A Number?', 'Empty Array?', 'Empty String?', 'Decimal Digit?', 'Hex Digit?', 'Octal Digit?', 'Printable?', 'White Space?', 'Lexical Class', and 'Fixed-Point ...'. The Windows taskbar at the bottom shows the system clock as 20:03 on 25/08/2011.

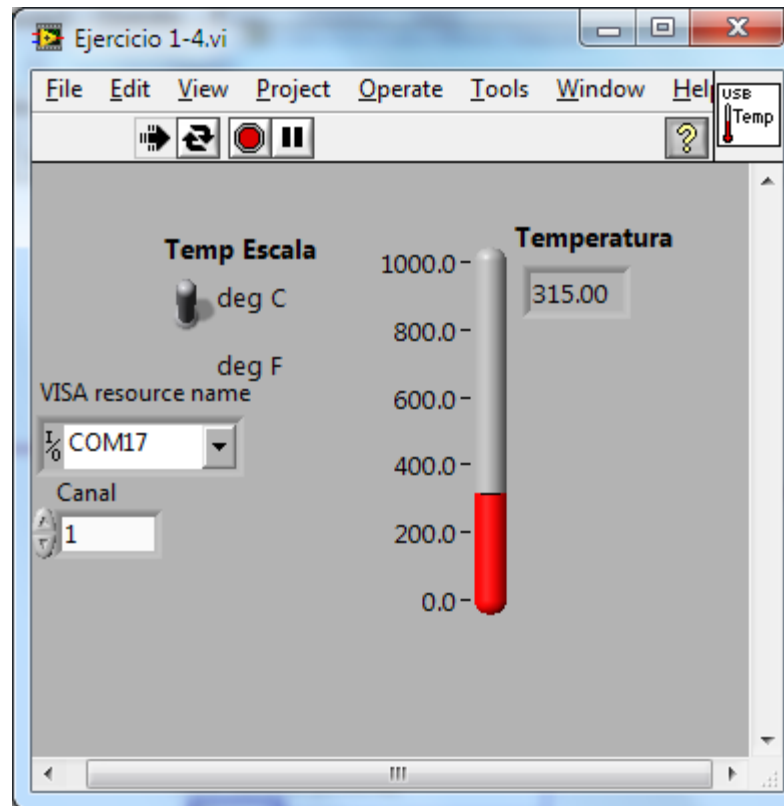
Ejercicio 1.3 – Termómetro



Ejercicio 1.3

Ejercicio 1.4 – Termómetro con entrada analógica

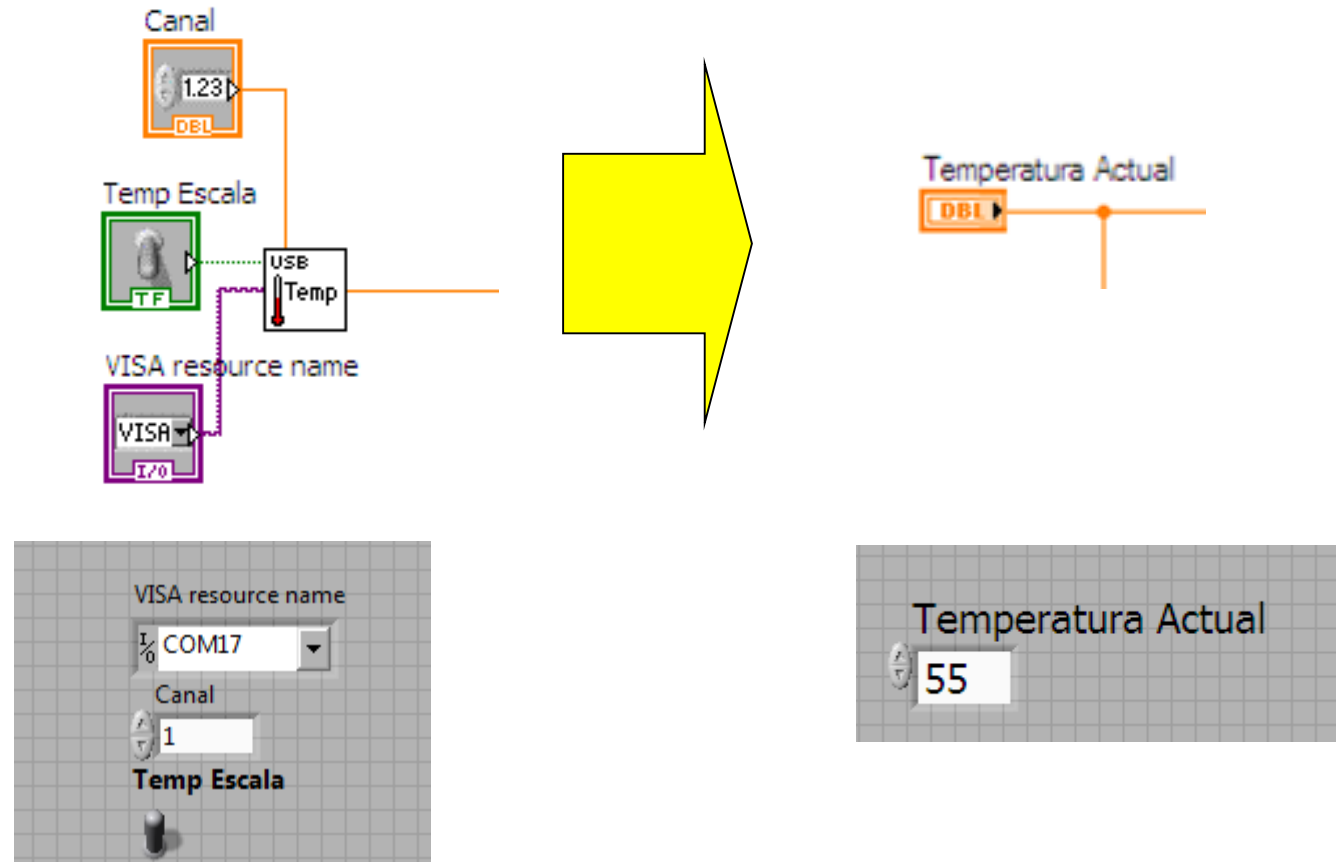
Utilizar el sub-VI “ImputAnalog.vi”, para capturar una señal analógica en el canal 1 de la placa USB adquisidora de datos (reemplazando a “Demo Read Voltage”).



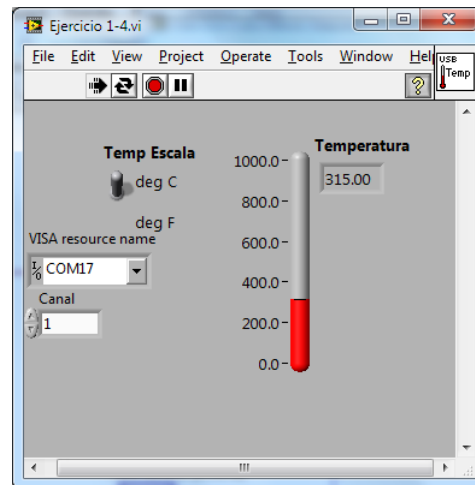
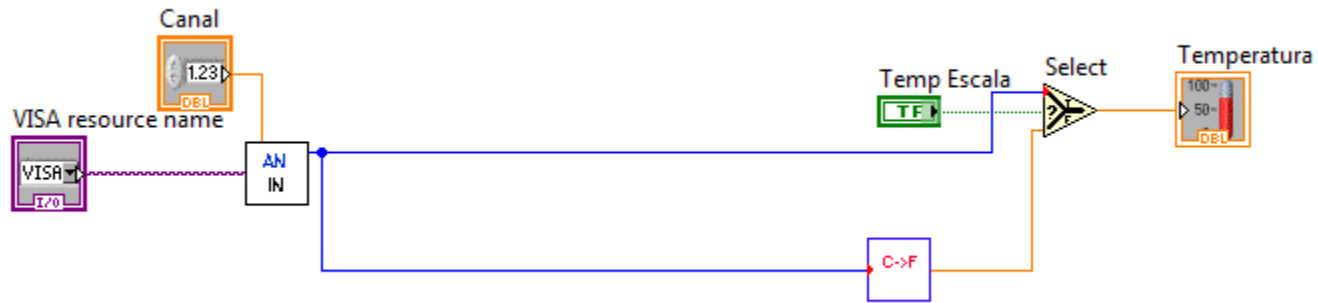
Ejercicio

Ejercicio 1.4 – Termómetro con entrada analógica

Adquisición de una variable analógica usando la placa USB PIC



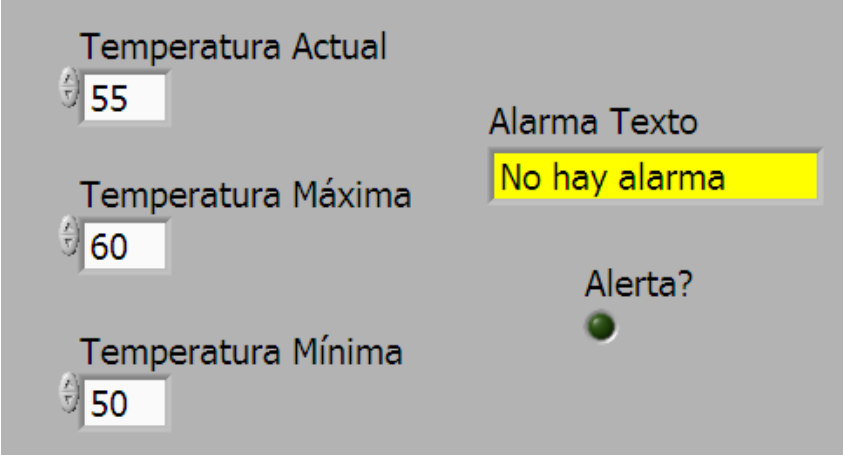
Ejercicio 1.4 – Termómetro con entrada analógica



Ejercicio 1.5 – Termómetro con alarmas

Construir una aplicación VI que a partir de dos temperaturas ingresadas máxima y mínima se comparen con otra actual indicando en pantalla el alerta (con un LED y texto) cuando la máxima o la mínima son excedidas.

Caso a) no hay alarma, temperatura entre los límites Máx y Min ingresados:



The screenshot shows a graphical user interface for a thermometer application. It features three input fields for temperature values: 'Temperatura Actual' (55), 'Temperatura Máxima' (60), and 'Temperatura Mínima' (50). To the right, there is a text display area labeled 'Alarma Texto' showing 'No hay alarma' in a yellow box, and a green LED indicator labeled 'Alerta?' which is currently lit.

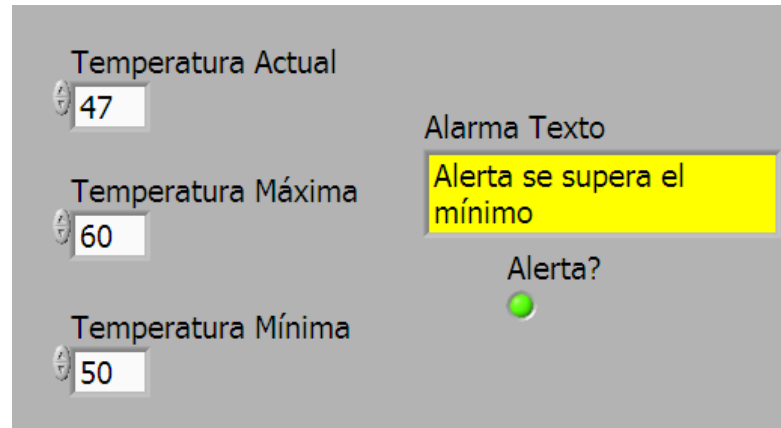
Label	Value
Temperatura Actual	55
Temperatura Máxima	60
Temperatura Mínima	50

Alarma Texto: No hay alarma

Alerta?

Ejercicio 1.5 – Termómetro con alarmas

Caso b) se excede el mínimo



Temperatura Actual
47

Temperatura Máxima
60

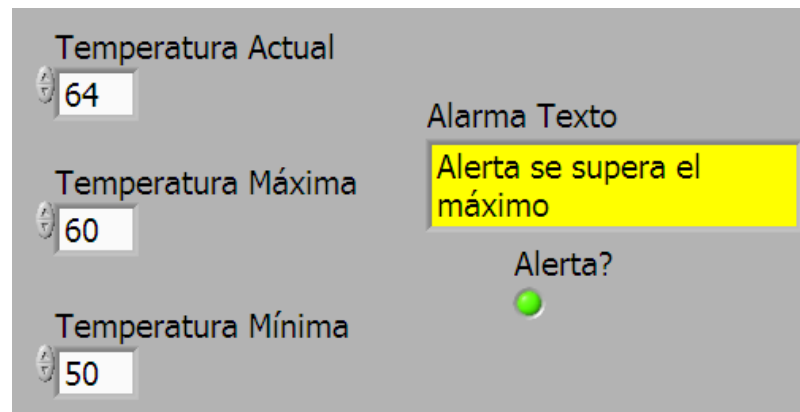
Temperatura Mínima
50

Alarma Texto
Alerta se supera el mínimo

Alerta?

Detailed description: This screenshot shows a control panel for a temperature system. On the left, three input fields are stacked vertically. The top field is labeled 'Temperatura Actual' and contains the value '47'. The middle field is labeled 'Temperatura Máxima' and contains '60'. The bottom field is labeled 'Temperatura Mínima' and contains '50'. On the right side, there is a section titled 'Alarma Texto' with a yellow highlighted box containing the text 'Alerta se supera el mínimo'. Below this, the label 'Alerta?' is followed by a green radio button that is selected, indicating that an alarm is active.

Caso c) se excede el máximo



Temperatura Actual
64

Temperatura Máxima
60

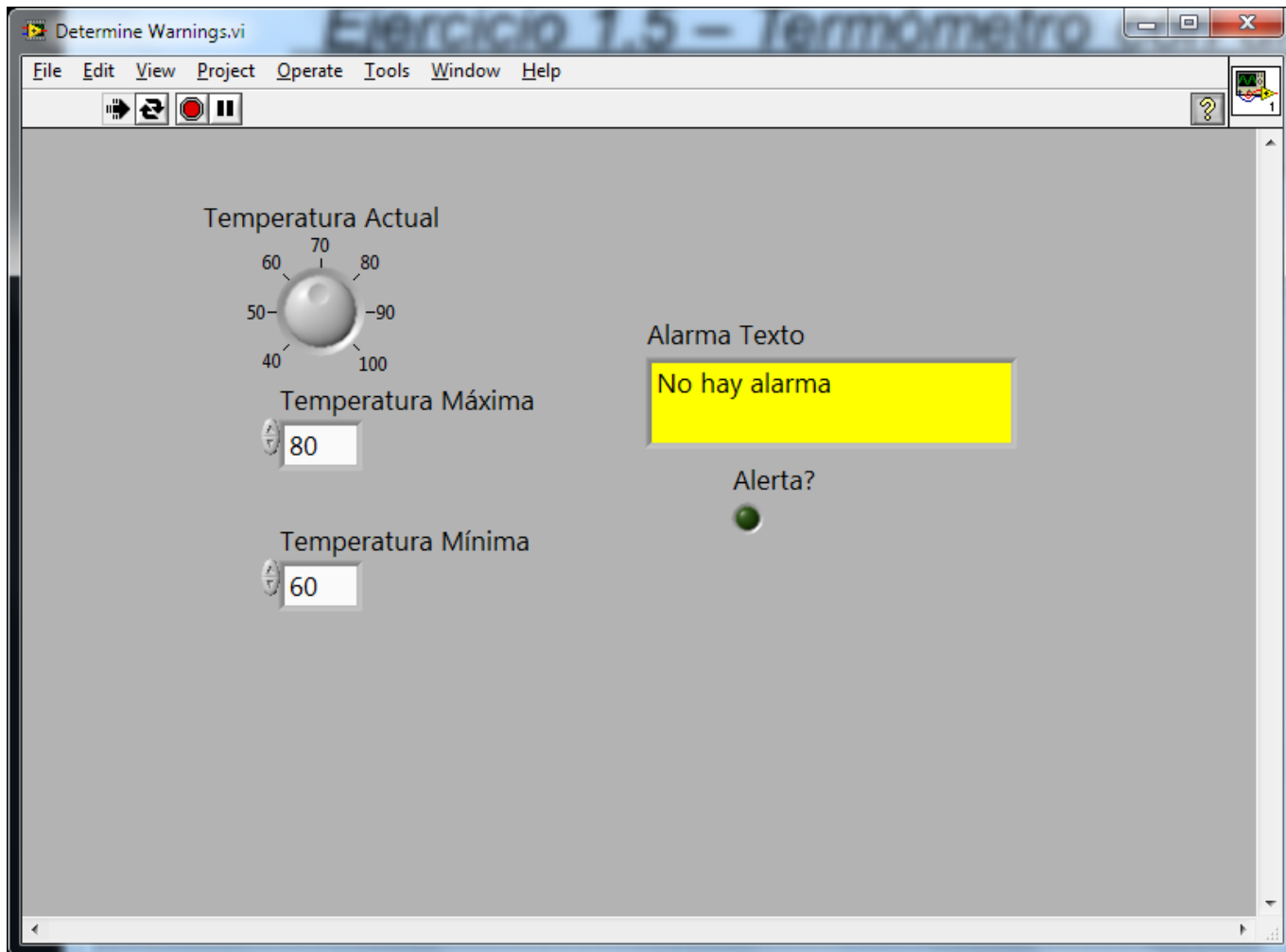
Temperatura Mínima
50

Alarma Texto
Alerta se supera el máximo

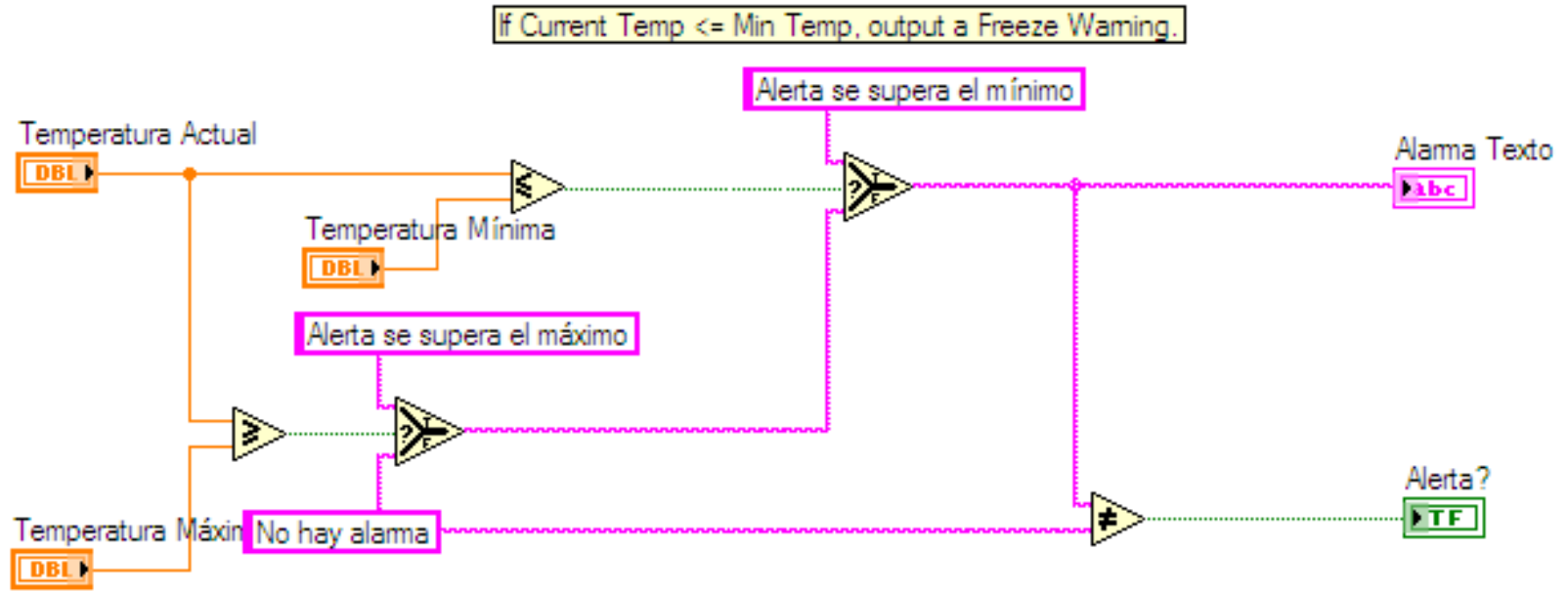
Alerta?

Detailed description: This screenshot shows the same control panel as in case b. The 'Temperatura Actual' field now contains '64', which is above the 'Temperatura Máxima' of '60'. The 'Alarma Texto' box now displays 'Alerta se supera el máximo'. The 'Alerta?' radio button remains selected (green), indicating the alarm is still active.

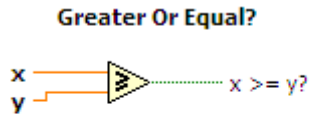
Ejemplo



Ejercicio 1.5 – Termómetro con alarmas



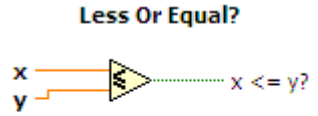
If Max Temp >= Current Temp, output a Heatstroke Warning.



Returns TRUE if **x** is greater than or equal to **y**. Otherwise, this function returns FALSE. You can change the comparison mode of this function.



Returns the value wired to the **t** input or **f** input, depending on the value of **s**. If **s** is TRUE, this function returns the value wired to **t**. If **s** is FALSE, this function returns the value wired to **f**.



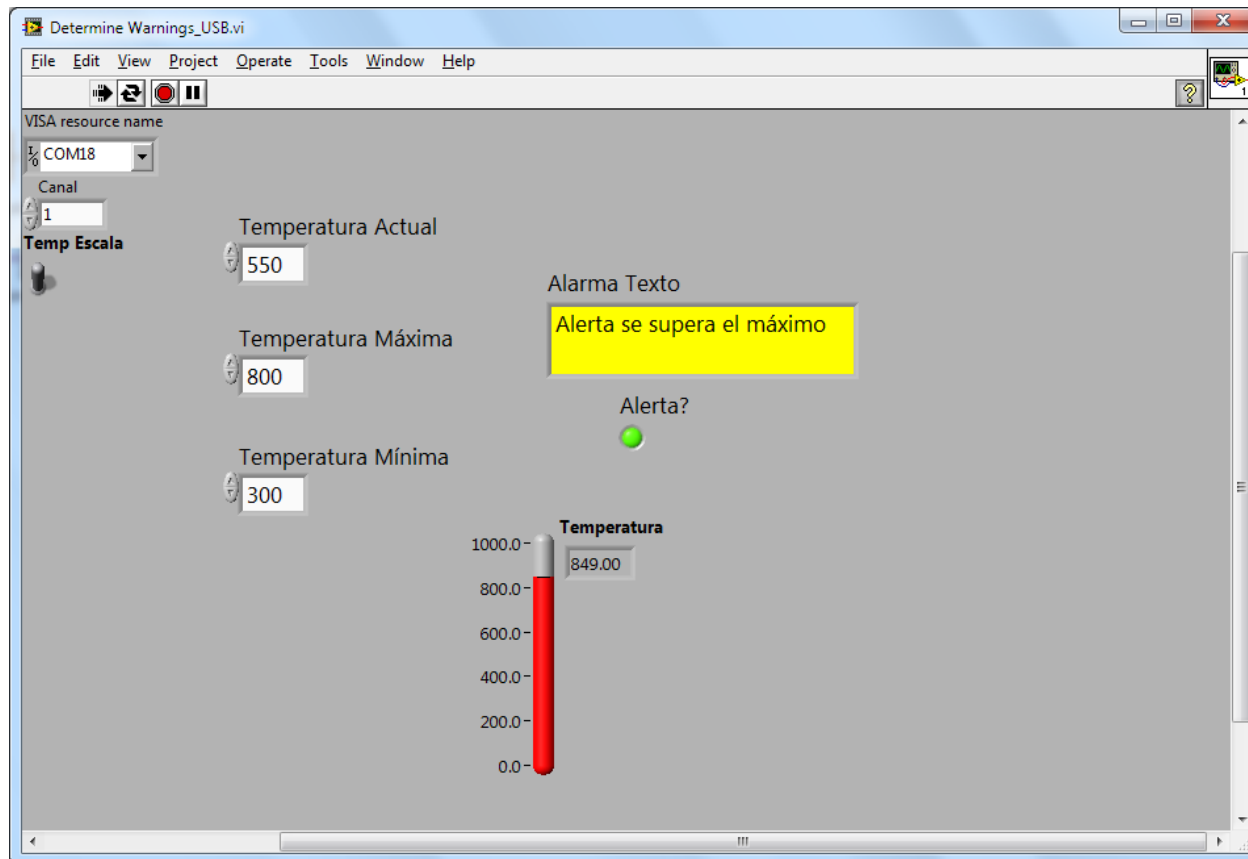
Returns TRUE if **x** is less than or equal to **y**. Otherwise, this function returns FALSE. You can change the comparison mode of this function.

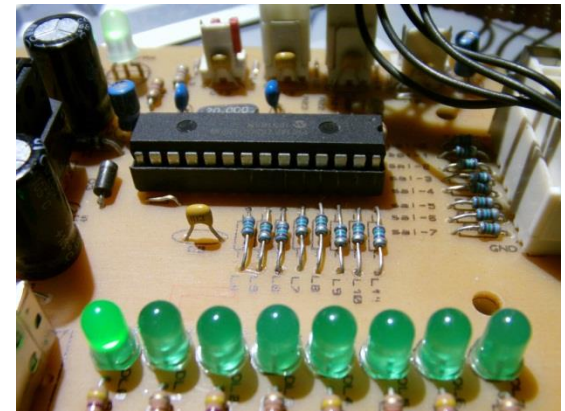
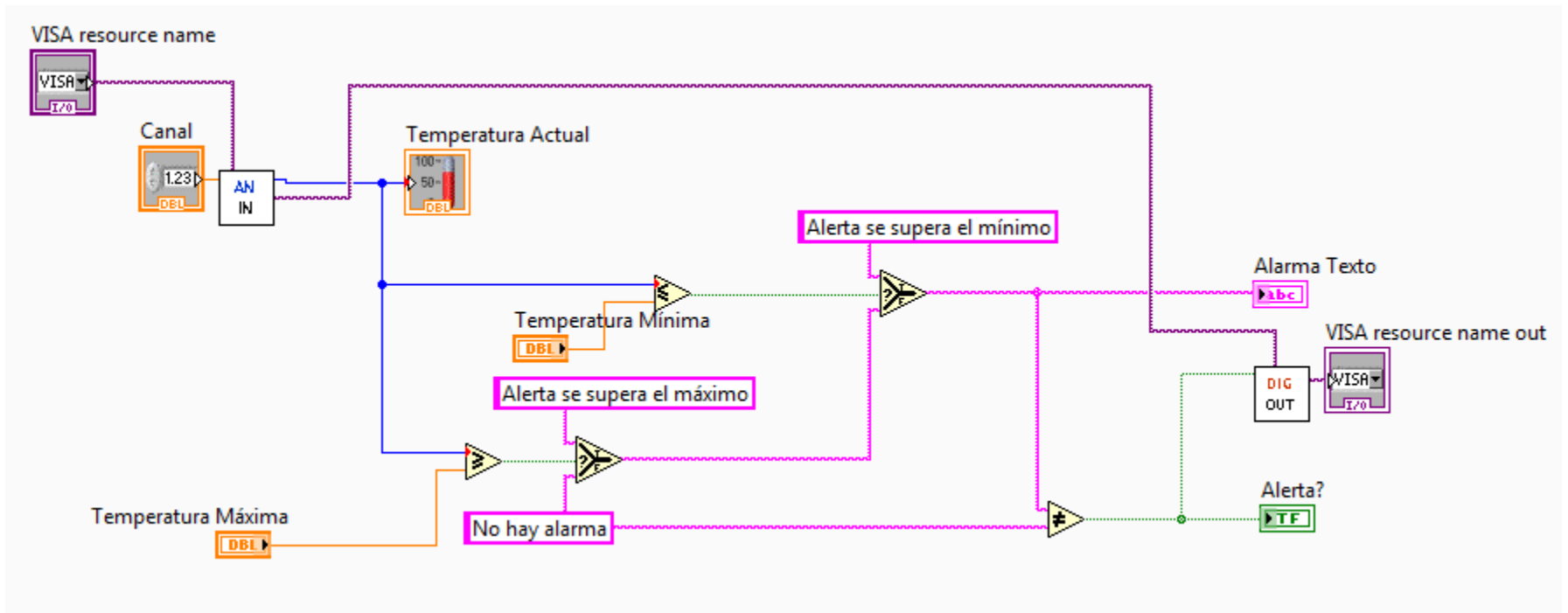
Ejercicio

Ejercicio 1.6 – Termómetro con alarmas

Reemplazar el control “Temperatura Actual”, por el sub-vi “InputAnalog.vi”, ingresando la entrada analógica desde la placa USB.

El LED de alerta en el panel frontal acompañarlo con uno de los LEDs de la placa, utilizando una de las salidas digital.





Consejos para trabajar en LabVIEW

Consejos para trabajar en LabVIEW

- Atajos desde el teclado
 - **<Ctrl-H>** – Activa/Desactiva la Ventana de Ayuda
 - **<Ctrl-B>** – Remueve todos los cables rotos del diagrama de bloques
 - **<Ctrl-E>** – Cambiar entre el Panel Frontal y el Diagrama de Bloques
 - **<Ctrl-Z>** – Deshacer cambios – Undo (también disponible en el menu de edición)
- **Herramientas » Opciones...** – Establecer Preferencias en LabVIEW
- **Propiedades del VI** – Configurar la Apariencia del VI, Documentación, etc.

Guardar el VI

- Elija un sitio o carpeta fácil de recordar
- Organizar por funcionalidad
- Guardar los VIs similares en un mismo directorio (Ej. Utilidades matemáticas)
- Organizar por aplicación
 - Guardar todos los VIs Usados para una Aplicación Especificar dentro de un directorio o un archivo de librería (Ej. Lab 1 – Frecuencia de respuesta)
 - Archivo de librería (.llbs) combina todos los VIs en un solo archivo, ideal para transferir aplicaciones enteras a través de computadoras

