

# Introducción a LabVIEW™

## Curso de Seis-Horas

**September 2003 Edition**  
**Part Number 323669B-01**

### **Copyright**

© 2003 National Instruments Corporation. All rights reserved.

Universities, colleges, and other educational institutions may reproduce all or part of this publication for educational use. For all other uses, this publication may not be reproduced or transmitted in any form, electronic or mechanical, including photocopying, recording, storing in an information retrieval system, or translating, in whole or in part, without the prior written consent of National Instruments Corporation.

### **Trademarks**

LabVIEW, National Instruments, NI, and ni.com are trademarks of National Instruments Corporation. Product and company names mentioned herein are trademarks or trade names of their respective companies.

### **Patents**

For patents covering National Instruments products, refer to the appropriate location: **Help»Patents** in your software, the `patents.txt` file on your CD, or [ni.com/legal/patents](http://ni.com/legal/patents).

## **Worldwide Technical Support and Product Information**

ni.com

## **National Instruments Corporate Headquarters**

11500 North Mopac Expressway Austin, Texas 78759-3504 USA Tel: 512 683 0100

## **Worldwide Offices**

Australia 1800 300 800, Austria 43 0 662 45 79 90 0, Belgium 32 0 2 757 00 20, Brazil 55 11 3262 3599, Canada (Calgary) 403 274 9391, Canada (Montreal) 514 288 5722, Canada (Ottawa) 613 233 5949, Canada (Québec) 514 694 8521, Canada (Toronto) 905 785 0085, Canada (Vancouver) 514 685 7530, China 86 21 6555 7838, Czech Republic 420 2 2423 5774, Denmark 45 45 76 26 00, Finland 385 0 9 725 725 11, France 33 0 1 48 14 24 24, Germany 49 0 89 741 31 30, Greece 30 2 10 42 96 427, India 91 80 51190000, Israel 972 0 3 6393737, Italy 39 02 413091, Japan 81 3 5472 2970, Korea 82 02 3451 3400, Malaysia 603 9131 0918, Mexico 001 800 010 0793, Netherlands 31 0 348 433 466, New Zealand 0800 553 322, Norway 47 0 66 90 76 60, Poland 48 0 22 3390 150, Portugal 351 210 311 210, Russia 7 095 783 68 51, Singapore 65 6226 5886, Slovenia 386 3 425 4200, South Africa 27 0 11 805 8197, Spain 34 91 640 0085, Sweden 46 0 8 587 895 00, Switzerland 41 56 200 51 51, Taiwan 886 2 2528 7227, Thailand 662 992 7519, United Kingdom 44 0 1635 523545

## Curso LabVIEW de Seis Horas Notas del Instructor

Este documento zip contiene material diseñado para dar a los estudiantes un conocimiento de LabVIEW en un tiempo de 6 horas. Los contenidos son:

- Notas del Instructor.doc este documento.
- Introducción LabVIEW Seis Horas.ppt una presentación PowerPoint conteniendo notas y diapositivas de los temas cubiertos en el curso.
- Ej0- Abrir y Correr un Instrumento Virtual.doc instrucciones paso a paso para el ejercicio abrir y correr .
- Ex1- Conversión C a F.doc instrucciones paso a paso para el Ejercicio 1.
- Conversión C a F (Ej1).vi VI de Solución al ejercicio 1.
- Ej2a- Crear un SubVI.doc instrucciones paso a paso para Ejercicio 2a.
- Conversión C a F (Ej2).vi subVI de solución al Ejercicio 2.
- Ej2b- Adquisición de Datos.doc instrucciones paso a paso para el Ejercicio 2b.
- Adquisición de Datos (Ej 2b) NI-DAQ Tradicional.vi Solución para Ejercicio 2.
- Ej3-Usar un Ciclo.doc instrucciones paso a paso para el ejercicio 3.
- Ejemplo de Ciclo (Ej3).vi VI de solución al ejercicio 3.
- Ej4-Analizando y Logging Data.doc instrucciones para el Ejercicio 4.
- Logger de Temperatura (Ej4).vi subVI de solución para el ejercicio 4.
- Convertir C a F (Ej4).vi subVI de solución para el ejercicio 4.
- Logger de Temperatura (Ej4).vi . VI de solución para Ejercicio 4.
- Ej5-Usando Graficas de Forma de Onda.doc instrucciones paso a paso para el Ejercicio 5.
- Grafica Multiplot (Ej5).vi VI de solución del Ejercicio 5
- Ej6-Clusters para Error.doc . instrucciones paso a paso para el Ejercicio 6
- Raiz Cuadrada (Ex6).vi . VI de solución para Ejercicio 6.
- Ej7-Maquinas de Estado Simples.doc instrucciones paso a paso - Ejercicio 7.
- Maquina de Estado (Ej7).vi VI de solución para el Ejercicio 7.

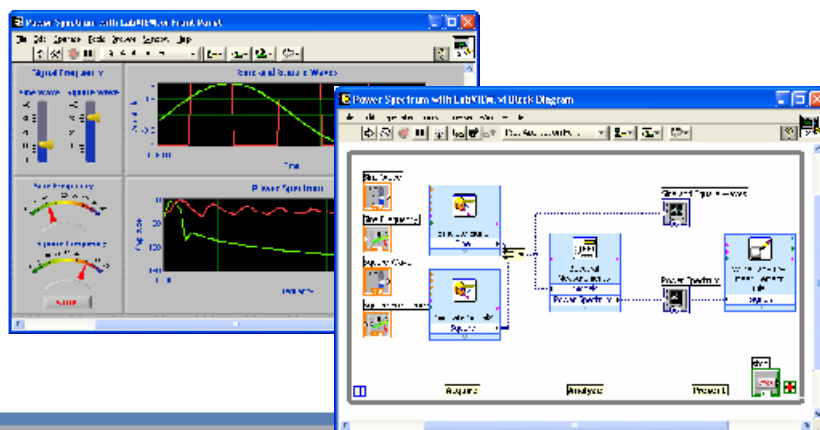
Las diapositivas pueden ser presentadas en dos laboratorios de tres horas o seis clases de una hora cada una. Dependiendo en el tiempo y los recursos disponibles en la clase, usted puede elegir si asigna o no los ejercicios como tarea para llevar o para ser hechos en clase. Si usted decide asignar los ejercicios en la clase, es mejor asignarlos en orden con la presentación. De esta manera los estudiantes pueden crear VIs mientras que la información relevante todavía esta fresca. Las notas asociadas con las diapositivas conteniendo ejercicios deberían de ser deficientes para guiar a los estudiantes a una solución. Los archivos de solución incluidos son una de las posibles soluciones, pero no significa que sean la única solución.

La instrucción paso a paso provee al estudiante con una manera fácil de completar el ejercicio, pero si usted decide asignar los ejercicios afuera del salón de clases, usted puede encontrar útil el imprimir las direcciones abreviadas de la diapositiva de ejercicio y dársela a los estudiantes como una asignación.

Los ejercicios pueden ser enviados a un evaluador via email.



# Instrumentación Virtual con LabVIEW



## Objetivos del Curso

- Comprender los componentes de un instrumento virtual
- Introducir LabVIEW y las funciones comunes de LabVIEW
- Construir una aplicación simple de adquisición de datos
- Crear una subrutina en LabVIEW
- Trabajar con Arreglos, Clusters y Estructuras
- Aprender sobre impresiones y características de documentación
- Desarrollo en arquitecturas de programación.
- Publicar VIs en el navegador

[ni.com](http://ni.com)



Esta es una lista de los objetivos del curso

Este curso le preparara para lo siguiente:

- Utilizar LabVIEW para crear aplicaciones.
- Entender paneles frontales, diagramas de bloque, íconos y paneles de conexión.
- Incorporación de funciones de LabVIEW.
- Crear y guardar programas en LabVIEW para que los pueda usar como subrutinas.
- Crear aplicaciones que usen dispositivos DAQ.

Este curso no describe lo siguiente:

- Teoría de programación
- Cada función u objeto encontrada en LabVIEW.
- Teoría analoga-a-digital (A/D)

## Sección I

- Términos de LabVIEW
- Componentes de aplicación de LabVIEW
- Herramientas de programación de LabVIEW
- Crear una aplicación en LabVIEW

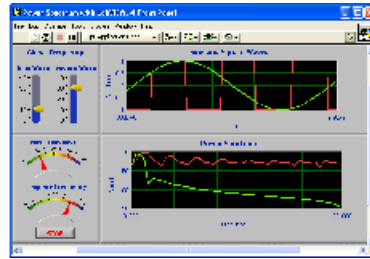
[ni.com](http://ni.com)



## Programas de LabVIEW se llaman instrumentos virtuales (VIs)

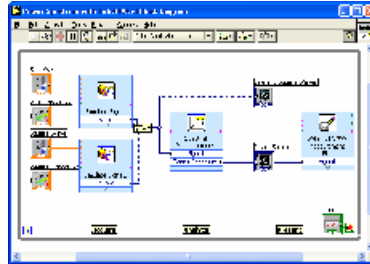
### Panel frontal

- **Controles** = entradas
- **Indicadores** = salidas



### Diagrama de bloque

- Programa de acompañamiento para el panel frontal
- Componentes “cableados” entre si



ni.com

NATIONAL  
INSTRUMENTS

Los Programas en LabVIEW son llamados instrumentos virtuales (VIs).

Haga énfasis en que controles es igual a entradas, e indicadores es igual a salidas.

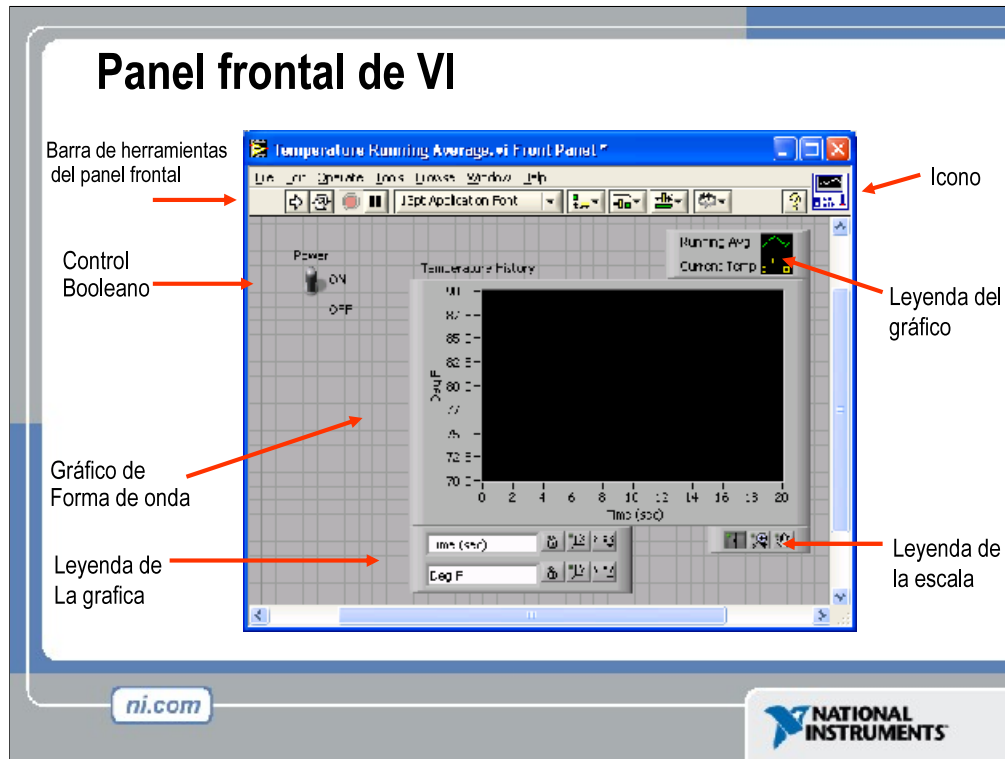
Cada VI contiene tres partes principales:

- Panel frontal– Cómo el usuario interacciona con el VI.
- Diagrama de bloque – El código que controla el programa.
- Icono/Conector – Medios para conectar un VI con otros VIs.

El panel frontal es utilizado para interaccionar con el usuario cuando el programa esta corriendo. Usuarios pueden controlar el programa, cambiar entradas, y ver datos actualizados en tiempo real. Haga énfasis en que los controles son usados como entradas - ajustando controles de deslizamiento para colocar un valor de alarma, encendiendo o apagando un switch, o parando un programa. Los indicadores son usados como salidas. Termómetros, luces, y otros indicadores indican valores del programa. Esto puede incluir datos, estados de programa y otra información.

Cada control o indicador del panel frontal tiene una terminal correspondiente en el diagrama de bloques. Cuando un VI se ejecuta, los valores de los controles fluyen a través del diagrama de bloques, en donde estos son usados en las funciones del diagrama, y los resultados son pasados a otras funciones o indicadores.

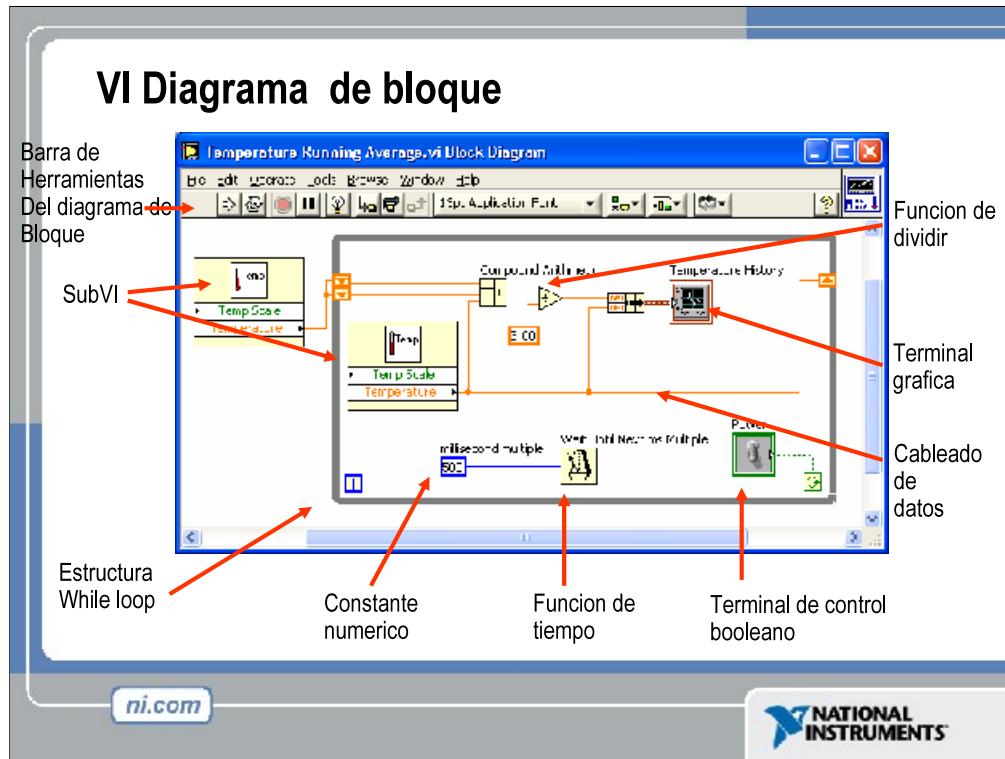




El panel frontal es la interfase del usuario con el VI. Usted construye el panel frontal con controles e indicadores, que son las entradas y salidas que interactúan con las terminales del VI, respectivamente. Los controles son botones, botones de empuje, marcadores y otros componentes de entradas. Los indicadores son las graficas, luces y otros dispositivos. Los controles simulan instrumentos de entradas de equipos y suministra datos al diagrama de bloques del VI. Los indicadores simulan salidas de instrumentos y suministra datos que el diagrama de bloques adquiere o genera.

En esta imagen, el switch de poder (encendido o apagado) es un control booleano. Un valor booleano contiene ambos un valor verdadero o falso. El valor es falso hasta que el switch se presiona. Cuando el switch se presiona, el valor se convierte en verdadero. El indicador del historial de temperatura es una grafica de forma de onda. Esta muestra valores múltiples. En este caso, la grafica señalara grados F versus tiempo (seg.).

El panel frontal también contiene una barra de herramientas, y sus funciones serán discutidas después.



El diagrama de bloque contiene el código fuente grafico. Los objetos del panel frontal aparecen como terminales en el diagrama de bloque. Adicionalmente, el diagrama de bloque contiene funciones y estructuras incorporadas en las bibliotecas de LabVIEW VI. Los cables conectan cada uno de los nodos en el diagrama de bloques, incluyendo controles e indicadores de terminal, funciones y estructuras.

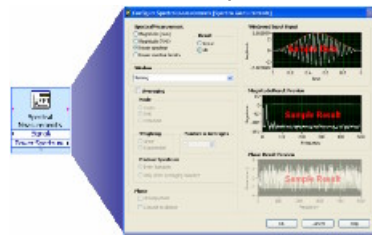
En este diagrama de bloque, el subVI **Temp** llama a la subrutina la cual obtiene una temperatura desde una tarjeta de adquisición de datos (DAQ). Esta temperatura es graficada junto con el valor average de la temperatura en la grafica de forma de onda **Temperature History**. El switch de poder (**Power**) es un control booleano en el panel frontal el cual va a detener la ejecución de la estructura mientras( While Loop). La estructura mientras (While Loop) también contiene una función de tiempo para controlar que tan frecuentemente la estructura se repite.

## VIs Expreso, VIs y Funciones

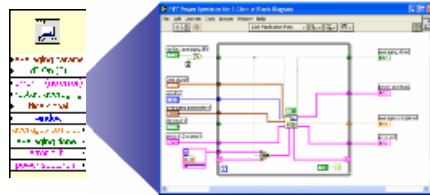
- **VIs Expreso:** VIs interactivos con pagina de dialogo configurable
- **VIs estándar:** VIs modulares y personalizables mediante cableado
- **Funciones:** Elementos fundamentales de operación de LabVIEW; no hay panel frontal o diagrama de bloque



Funcion



VI Expreso



VI Estandar

ni.com



LabVIEW 7.0 introduce un nuevo tipo de subVI llamado VIs Expreso (Express VIs). Estos son VIs interactivos que tienen una configuración de caja de dialogo que permite al usuario personalizar la funcionalidad del VI Expreso. LabVIEW entonces genera una subVI basado en estos argumentos.

VIs estándar son aquellos VIs (que consisten de un panel frontal y un diagrama de bloque) que son usados adentro de otro VI.

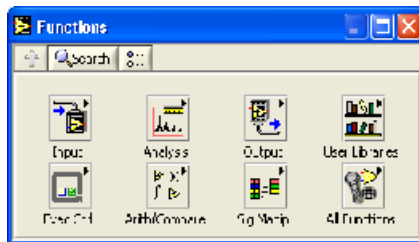
Las funciones son los bloques de construcción de todos los VIs. Las funciones no tienen un panel frontal o un diagrama de bloque.

## Paleta de controles y funciones

### Paletas de control (Ventana del panel frontal)




### Paleta de funciones (Ventana de diagrama de bloque)




Use la paleta de controles (**Controls**) para colocar los controles e indicadores en el panel frontal. La paleta de controles esta disponible solamente en el panel frontal. Seleccione **Window»Show Controls palette** o haga clic derecho en el espacio de trabajo en el panel frontal para desplegar la paleta de controles. Usted también puede desplegar la paleta de controles haciendo un clic derecho en una área abierta del panel frontal. Para desaparecer la paleta de controles presione el botón en la parte superior izquierda de la paleta.











Use la paleta de funciones (**Functions**), para construir un diagrama de bloque. La paleta de funciones esta disponible solamente en el diagrama de bloque. Seleccione **Window»Show Functions Palette** o haga un click derecho en el espacio de trabajo del diagrama de bloque para desplegar la paleta de funciones. Usted también puede desplegar la paleta de funciones dando un click derecho en una área abierta del diagrama de bloques. Para desaparecer la paleta de funciones presione el botón en la parte superior izquierda de la paleta.


## 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**

 <b>Herramienta de operación</b>  <b>Herramienta de posicionamiento y redimensión</b>  <b>Herramienta de etiquetado</b>  <b>Herramienta de cableado</b>  <b>Herramienta de menú (atajo)</b>	 <b>Herramienta de desplazamiento</b>  <b>Herramienta de punto de paro</b>  <b>Herramienta de prueba</b>  <b>Herramienta para copia de color</b>  <b>Herramienta para colorear</b>
--	---





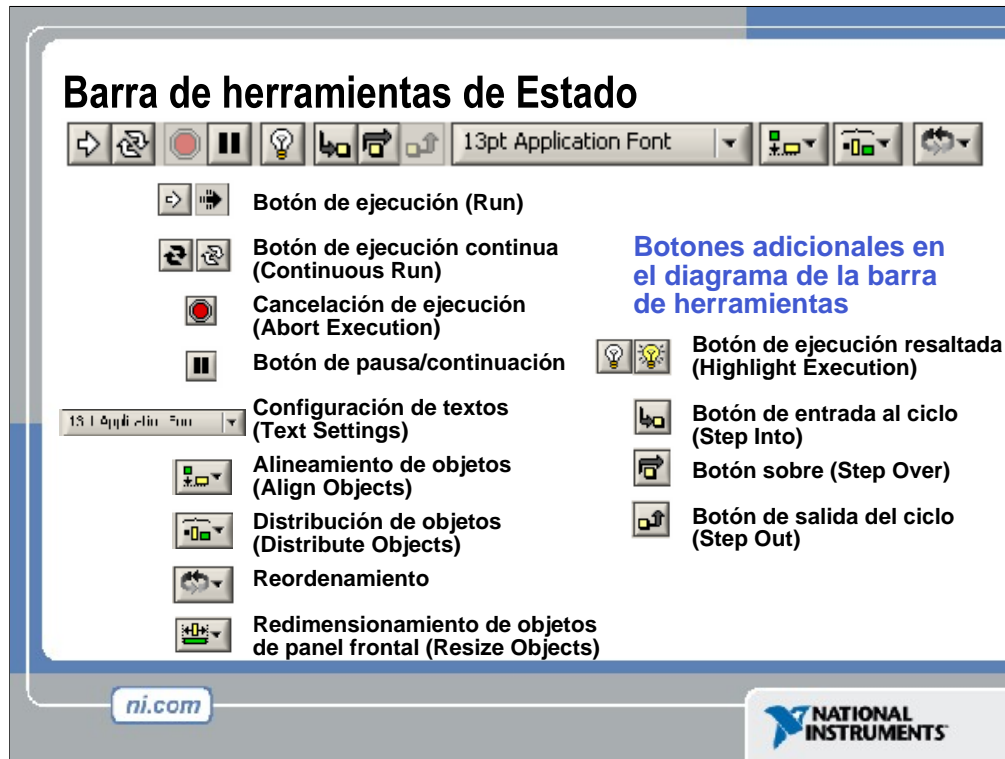
Si la herramienta de selección automática esta habilitada y usted mueve el cursor sobre un objeto en el panel frontal o en el diagrama de bloque, LabVIEW automáticamente selecciona la herramienta correspondiente de la paleta de controles. Cambie a la herramienta de selección automática tecleando el botón de Selección de Herramienta Automática en la paleta de Herramientas.

Utilice la herramienta de operación para cambiar valores o para seleccionar texto de un control.

Utilice la herramienta de posicionamiento para seleccionar, mover o redimensionar objetos. La herramienta de posicionamiento cambia las formas cuando se mueve sobre una esquina de un objeto reajutable.

Utilice la herramienta de etiquetado para editar textos y crear etiquetas libres. La herramienta de etiquetado se convierte en cursor cuando usted crea etiquetas libres.

Utilice la herramienta de cableado para cablear objetos juntos en el diagrama de bloque.



•Pulse el botón de ejecución (**Run**) para ejecutar el VI. Mientras el VI se esta ejecutando, el botón de ejecución aparecerá con una flecha negra si es que el VI es un VI principal, lo que significa que no ha sido llamado por otro VI y por lo tanto este no es un subVI.

•Pulse el botón de ejecución continua (**Continuous Run**) para ejecutar el VI hasta que el botón de cancelación de ejecución o de pausa sea presionado. Usted también puede pulsar este botón nuevamente para deshabilitar la ejecución continua.

•Mientras el VI se esta ejecutando, El botón de cancelación de ejecución (**Abort Execution**) aparece. Presione este botón para detener el VI inmediatamente.

**Nota:** Evite usar el botón de cancelación de ejecución (**Abort Execution**) para detener el VI. Ya sea deje que el VI complete su flujo de datos o diseñe un método para detener el VI programáticamente. Al hacer esto el VI se encuentra en un estado conocido. Por ejemplo, coloque un botón en el panel frontal que detenga el VI cuando sea presionado.

•Pulse el botón de pausa (**Pause**) para detener momentáneamente la ejecución de un VI. Cuando usted presiona el botón de Pausa, LabVIEW señala la posición donde usted detuvo la ejecución en el diagrama de bloque. Pulse el botón de Pausa nuevamente para que el VI continúe ejecutándose.

•Seleccione el menú **Configuración de Textos (Text Settings)** para cambiar el tipo de fuente (letra) del VI, incluyendo el tamaño, estilo y color.

•Seleccione el menú **Alineamiento de Objetos (Align Objects)** para alinear objetos con respecto a los ejes, incluyendo eje vertical, superior, izquierdo, etc.

•Seleccione el menú **Distribución de Objetos (Distribute Objects)** para espaciar objetos uniformemente, incluyendo espacios vacíos, compresiones, etc.

•Seleccione el **Redimensionamiento de Objetos (Resize Objects)** para cambiar el

- Seleccione el menú **Reordenamiento (Reorder)** cuando tenga objetos superpuestos unos con otros y usted quiera definir cual esta enfrente de quien y cual atrás. Seleccione uno de los objetos con la herramienta de posicionamiento y seleccione entre **Move Forward** (mover hacia adelante), **Move Backward** (mover hacia atrás), **Move to Front** (mover al frente) y **Move to Back** (mover hacia atrás).

*<Los siguientes puntos solo aparecen en la barra de herramientas del diagrama de bloques.>*

- Pulse en el botón de ejecución resaltada (**Highlight Execution**) para ver el flujo de información en el diagrama de bloques. Pulse el botón nuevamente para detener este tipo de ejecución resaltada.

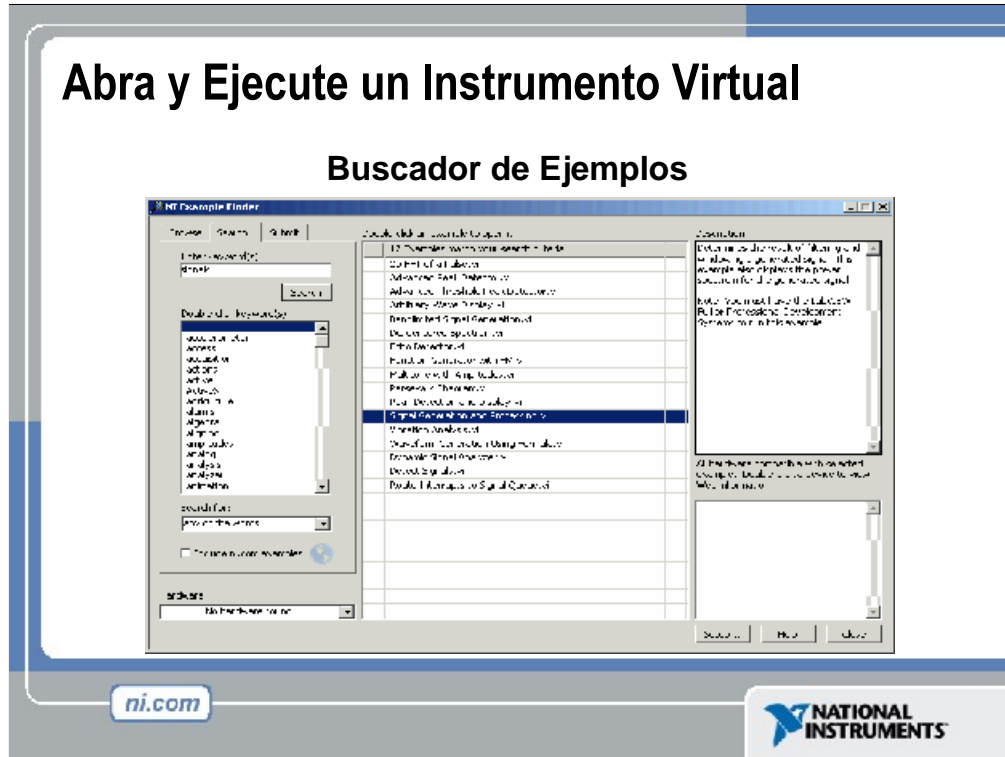
- Pulse el botón de entrada al ciclo (**Step Into**) para entrar un paso adentro de un ciclo, un SubVI, etc. Entrando un paso a la vez adentro de un VI le permite introducirse en la VI nodo a nodo. Cada nodo se resalta para señalar cuando esta listo para ejecutarse. Al entrar un paso adentro de un nodo usted esta listo para caminar paso a paso adentro del nodo.

- Pulse el botón de sobre (**Step Over**) para posicionarse encima de un ciclo, un SubVI, etc. Al posicionarse encima del nodo, usted ejecuta el nodo sin entrar paso a paso adentro del nodo.

- Pulse el botón de Salida del ciclo (**Step Over**) para salirse de un ciclo, SubVI, etc. Al posicionarse fuera del nodo, usted completa la ejecución de un nodo y puede moverse al siguiente nodo.

# Abra y Ejecute un Instrumento Virtual

## Buscador de Ejemplos

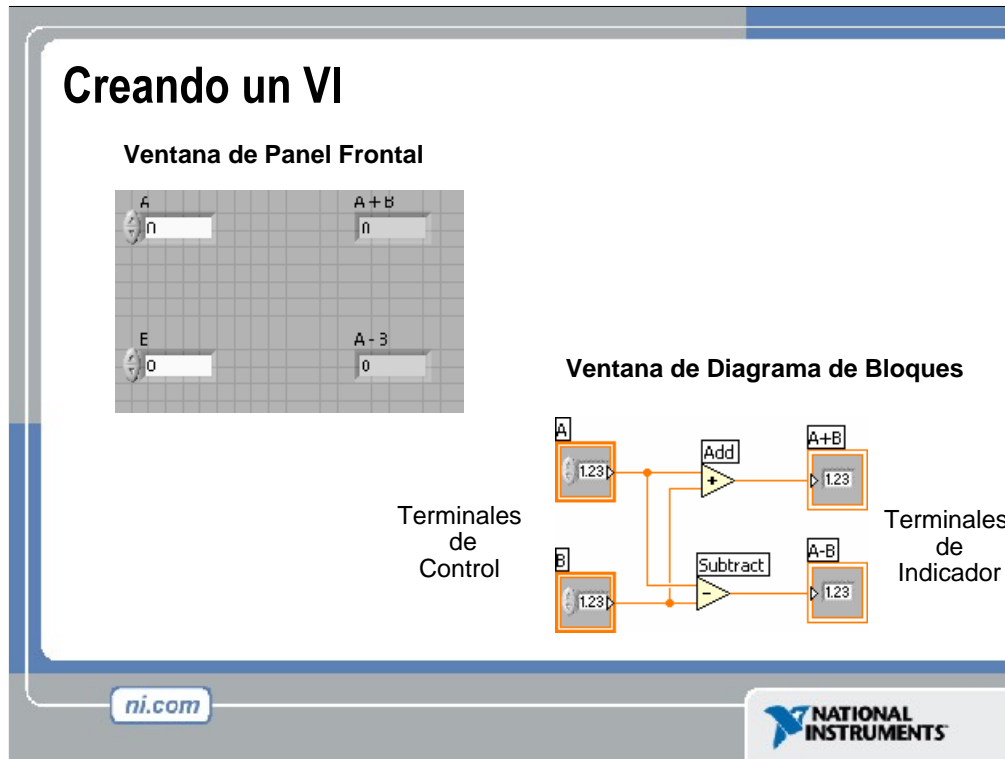


1. Seleccione **Start » Programs » National Instruments » LabVIEW 7.0 » LabVIEW** para lanzar LabVIEW. Aparece la ventana de dialogo de **LabVIEW**.
  2. Seleccione **Find Examples** del menú de ayuda (Help). La ventana de dialogo que aparece da la lista y enlaces a todos los VIs de ejemplo de LabVIEW.
  3. Usted puede buscar ejemplo por categorías, o puede utilizar palabras claves en su búsqueda. Haga clic en la viñeta de búsqueda (**Search**) para abrir el buscador de palabras claves.
  4. En la caja de **“Enter Keyword(s)”** escriba **“Signal”**
  5. Una lista de temas relacionados aparecerá en la ventana de ejemplos (**examples**). Haga doble-clic en **signals**, esto generara una lista de ejemplos en el lado derecho.
  6. Haga clic en cualquier programa para ver una descripción detallada de el ejemplo. Haga doble clic sobre **Signal Generation and Processing.vi** para lanzar el ejemplo.
- Esto abrirá el panel frontal del VI “Signal Generation and Processing.vi” .

Examine el VI y ejecútelo. Cambie las frecuencias y tipos de señales de entradas y note como el despliegue de las graficas cambia. Cambie la ventana de procesamiento de señales (Signal Processing Window) y opciones de filtrado (Filter options). Después de que haya examinado el VI y las diferentes opciones que puede cambiar, pare el VI presionando el botón de paro (Stop button).

**Nota** Usted también puede abrir un VI al hacer clic en el botón de **Open VI** y navegar a labview\examples\apps\demos.llb\Signal Generation and Processing.vi.





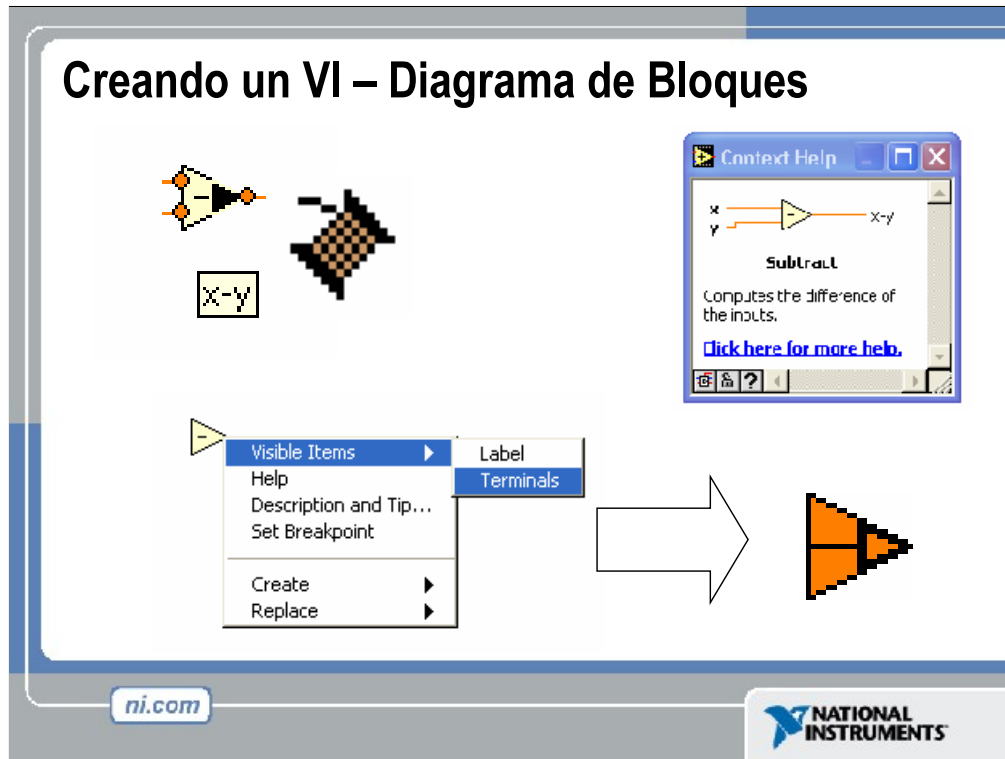
Cuando usted crea un objeto en el panel frontal, una terminal es creada en el diagrama de bloques. Estas terminales le dan acceso a los objetos del panel frontal del código creado mediante el diagrama de bloque.

Cada terminal contiene información útil referente al objeto al cual corresponde en el panel frontal. Por ejemplo, el color y los símbolos proporcionan el tipo de dato. Números de punto flotante y de doble-precisión, son representados con terminales anaranjadas y las letras DBL. Las terminales booleanas son verdes y son representadas por las letras TF.

En general, las terminales anaranjadas deben unirse (cablearse) con las terminales anaranjadas, verdes con verdes, y así sucesivamente. Esta no es una regla que no se puede romper; por ejemplo LabVIEW permitirá al usuario conectar una terminal azul (valor entero) a una terminal anaranjada (valor fraccional). Pero en la mayoría de casos, busque mejor una igualdad en colores.

Los controles tienen una flecha en el lado derecho y tienen un borde grueso. Los indicadores tienen una flecha en el lado izquierdo y un borde fino. Reglas lógicas pueden ser aplicadas al conectar en LabVIEW: Cada cable debe tener una (pero solo una) fuente (o control), y cada cable puede tener varios destinos (o indicadores).

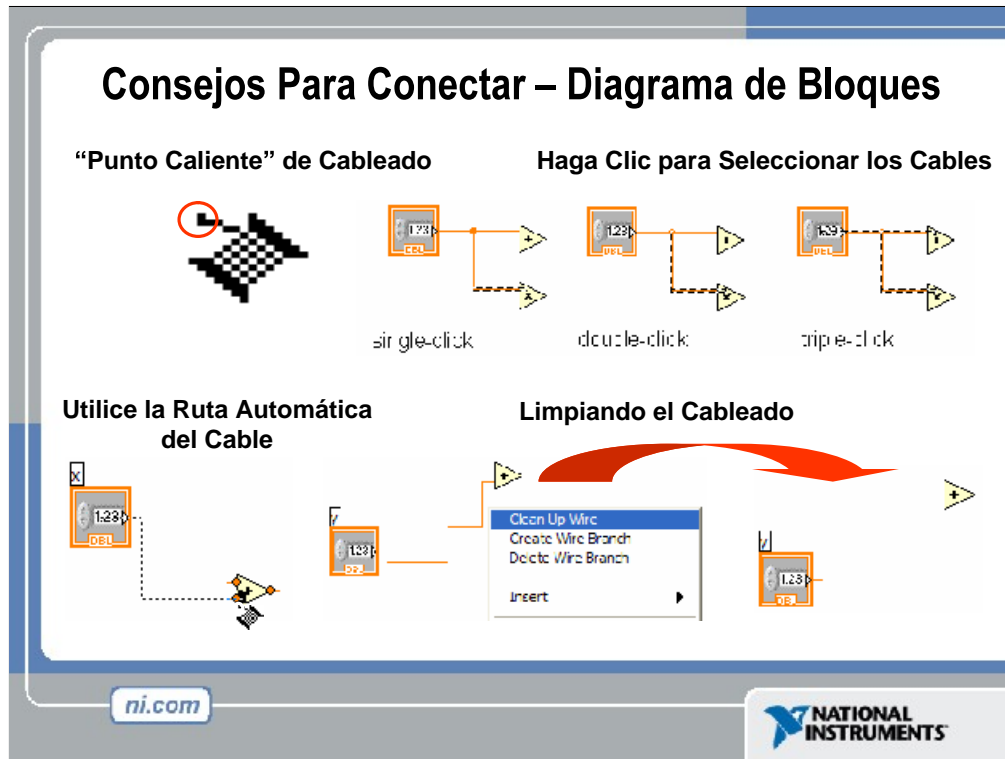
El programa en esta diapositiva toma datos de A y B y pasa valores a una función de adición y a una función de resta. Los resultados son mostrados en los indicadores apropiados.



Además de los terminales del panel frontal, el diagrama de bloques contiene funciones. Cada función puede tener múltiples terminales de entradas y salidas. La conexión de estas terminales es una parte muy importante de la programación en LabVIEW.

Una vez que usted tenga cierta experiencia programado en LabVIEW, la conexión de cables se le hará mas fácil. Primero puede que necesite ayuda. En seguida se le muestra algunas recomendaciones para comenzar:

- La herramienta para conectar o de cableado es utilizada para conectarse a los nodos de las funciones. Cuando usted “apunte” con la herramienta de cableado, apunte con el extremo del cable que cuelga del carrete. Aquí es donde el cable sera colocado.
- Mientras usted mueve la herramienta de cableado sobre las funciones, observe la viñeta amarilla que aparece. Esto le dirá el nombre de la terminal al que se esta conectando.
- Mientras usted mueva la herramienta de cableado encima de una terminal, esta va a mostrar información. Esto le ayudara a identificar donde se va a unir el cable.
- Para mas ayuda con los terminales, haga clic derecho en la función y seleccione **Visible Items o Objetos Visibles >> Terminals o Terminales**. Un dibujo de la función será colocada atrás para revelar las terminales de la conexión. Note los colores - estos corresponden a los tipos de datos utilizados por los terminales del panel frontal.
- Para ayuda adicional, seleccione **Help >> Show Context Help**, o presione **CTRL+H**. Esto mostrara la ventana de ayuda en contexto. A medida que uno mueva el raton (mouse) sobre la función, esta ventana le mostrara la función,



Además de las terminales del panel frontal. El diagrama de bloque contiene funciones. Cada función puede tener múltiples terminales de entradas y salidas. La conexión de estas terminales es una parte muy importante de la programación en LabVIEW.

Una vez que usted tenga cierta experiencia programado en LabVIEW, la conexión de cables se le hará mas fácil. Primero puede que necesite ayuda. En seguida se le muestra algunas recomendaciones para comenzar:

- La herramienta para conectar o de cableado es utilizada para conectarse a los nodos de las funciones. Cuando usted “apunte” con la herramienta de cableado, apunte con el extremo del cable que cuelga del carrito. Aquí es donde el cable sera colocado.
- Mientras usted mueve la herramienta de cableado sobre las funciones, observe la viñeta amarilla que aparece. Esto le dirá el nombre de la terminal al que se esta conectando.
- Mientras usted mueva la herramienta de cableado encima de una terminal, esta va a mostrar información. Esto le ayudara a identificar donde se va a unir el cable.
- Para mas ayuda con los terminales, haga clic derecho en la función y seleccione **Visible Items o Objetos Visibles>>Terminales o Terminales**. Un dibujo de la función será colocada atrás para revelar las terminales de la conexión. Note los colores - estos corresponden a los tipos de datos utilizados por los terminales del panel frontal.
- Para ayuda adicional, seleccione **Help >> Show Context Help**, o presione **CTRL+H**. Esto mostrara la ventana de ayuda en contexto. A medida que uno mueva el raton (mouse) sobre la función, esta ventana le mostrara la función, terminales, y una breve descripción. Utilice esto junto con otras herramientas para ayudarse mientras conecta los cables.
- Si el cableado no se mira muy bien, haga clic derecho en el cable que se desea arreglar y escoja la opción de **Clean Up Wire o Limpieza del Cable** para que automáticamente el cable haga su ruta de nuevo.

El cableado es muy flexible en LabVIEW. Experimente con combinaciones de clics y/o teclas cuando este cableando. Aquí hay algunas de las características mas a menudo utilizadas.

- El hacer un simple, doble, y triple clic en el cable selecciona el cable para moverlo o para borrarlo.
- El hacer un clic mientras se esta cableando hace un dobléz en el cable.
- Haciendo clic derecho o presionando el botón **Esc** mientras se esta cableando cancela la operación de cableado.

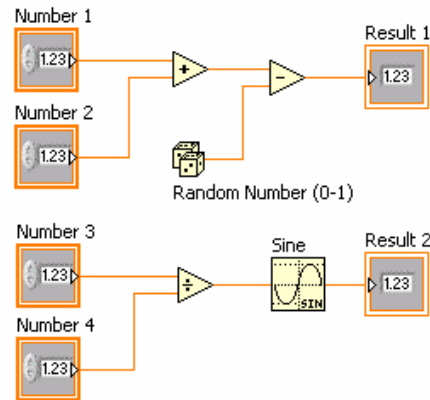
No se preocupe por el color de los cables. LabVIEW seleccionará automáticamente el cable correcto y adecuado para cada situación.

### **Cableando Objetos Automáticamente**

LabVIEW cablea automáticamente los objetos a medida que usted los coloca en el diagrama de bloque. También puede cablear automáticamente objetos que ya están colocados en el diagrama de bloque. LabVIEW conecta la terminal que mejor combina (match) y deja las terminales que no combinan (match) sin conectar. A medida que usted mueve un objeto seleccionado cerca de otros objetos en el diagrama de bloque, LabVIEW dibuja cables temporales para mostrarle conexiones validas. Cuando usted suelta el botón del mouse para poner el objeto en el diagrama de bloque, LabVIEW conecta automáticamente los cables. Cambie al cableado automático presionando el espaciador (spacebar) mientras mueve un objeto utilizando la herramienta de posicionamiento (Positioning tool). Puede ajustarlas opciones del cableado automático seleccionando **Tools >> Options** y seleccionando **Block Diagram** de el menú superior principal.

## Programando el Flujo de Datos

- El diagrama de bloque se ejecuta dependiendo del flujo de los datos; el diagrama de bloques NO se ejecuta de izquierda a derecha
- El nodo se ejecuta cuando los datos están disponibles para TODOS los terminales de entrada.
- Los nodos suministran datos a todos los terminales de salida cuando termina.



ni.com

NATIONAL  
INSTRUMENTS

LabVIEW sigue un modelo de flujo de datos para correr los VIs. Un nodo del diagrama de bloque se ejecuta cuando todas sus entradas están disponibles. Cuando un nodo completa la ejecución, suministra datos a sus terminales de salida y pasa los datos de salida al siguiente nodo en la trayectoria del flujo de datos. Visual Basic, C++, JAVA y otros lenguajes de programación basados en texto, siguen un modelo de control de flujo de la ejecución de un programa. En flujo de control, el orden secuencial de los elementos del programa determina el orden de ejecución de un programa.

Considere el diagrama de bloque que se encuentra arriba. Este suma dos números y luego resta 50 del resultado de la suma. En este caso, el diagrama de bloque se ejecuta de izquierda a derecha, no porque los objetos están puestos en ese orden, sino porque una de las entradas de la función de resta no es válida hasta que la función de suma o adición haya terminado su ejecución y pasado los datos a la función de resta. Recuerde que un nodo se ejecuta solamente cuando tiene datos disponibles en todas sus terminales de entrada, y suministra datos a sus terminales de salidas solamente cuando termina su ejecución.

En el código de la derecha, considere cual segmento del código se ejecutara primero---la suma, el numero aleatorio, o la función de división. No se puede saber porque las entradas a las funciones de suma y división están disponibles al mismo tiempo, y la función de numero aleatorio no tiene entradas. En una situación en donde un segmento del código se debe ejecutar antes que otro, y no existe dependencia de datos entre las funciones, utilice una estructura de Secuencia para forzar el orden de la ejecución.

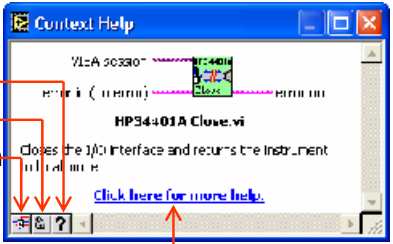
## Opciones de Ayuda

### Contexto de la Ayuda

- Ayuda en línea
- Congelar Ayuda
- Ayuda del Diagrama Simple/Completo
- Ctrl + H

### Referencias en Línea

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



ni.com

NATIONAL INSTRUMENTS

Utilice la ventana de **Context Help** (Ayuda Contextual) y **LabVIEW Help** (Ayuda de LabVIEW) para ayudarle a construir o editar los VIs. Refiérase a la ayuda de LabVIEW y a los manuales para mas información.

### Ventana de Ayuda Contextual (Context Help Window )

Para desplegar la ventana de **Context Help**, seleccione **Help>>Show Context Help** o presione las teclas <Ctrl-H>. Cuando mueva el cursor sobre el panel frontal y los objetos del diagrama de bloque, la ventana de **Context Help** despliega el icono de los subVIs, funciones, constantes, controles e indicadores, con cables adjuntos a cada una de sus terminales. Cuando mueva el cursor sobre la caja de opciones de dialogo, la ventana de **Context Help** despliega descripciones de esas opciones. En la ventana, las conexiones requeridas están en negrilla, las conexiones recomendadas en texto normal, y las conexiones opcionales están poco claras o no aparecen. Arriba se presenta un ejemplo de la ventana de **Context Help**.

Haga un clic en el botón de **Simple/Detailed Context Help** localizado en la esquina inferior izquierda de la ventana de **Context Help** para distinguir entre la ayuda contextual simple y detallada. El modo simple enfatiza las conexiones importantes. Las terminales opcionales son mostradas por segmentos de cables, informándole de la existencia de otras conexiones.

Haga clic en el botón de **Lock Context Help** para congelar el contenido actual de la ventana de la **Context Help**. Cuando los contenidos están congelados, el mover el cursor sobre otros objetos no cambia los contenidos de la ventana. Para descongelar la ventana, haga clic nuevamente en el botón. Usted también puede tener acceso a esta opción desde el menú de ayuda (**Help**).

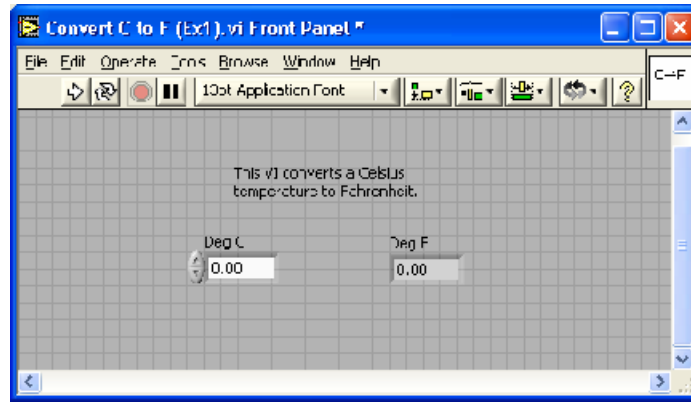
Haga clic en el botón de **More Help** (mas ayuda) para desplegar el tema correspondiente en **LabVIEW Help**, el cual describe el objeto en forma detallada.

### **LabVIEW Help** (Ayuda de LabVIEW)

Puede acceder a LabVIEW Help ya sea haciendo clic en el botón de More Help en la ventana de Context Help, seleccionando **Help»VI, Function, & How-To Help**, haciendo clic en la oración **Click here for more help** en la ventana de **Context Help** o presionando <Ctrl-?>.

La **LabVIEW Help** contiene descripciones detalladas de casi todos los menús, herramientas, VIs y funciones. También incluye instrucciones paso-a-paso para la utilización de muchas características de LabVIEW y enlaces al *Tutorial de LabVIEW*, versiones en PDF de todos los manuales y Notas de Aplicación de LabVIEW, además de recursos de soporte técnico en el sitio web de National Instruments.

## Ejercicio 1 – Convertir de °C a °F




Tiempo estimado de finalización: 20 minutos. El ejercicio es fácil, pero como será el primer VI que crearemos, es bueno permitir un tiempo amplio para explorar el ambiente de LabVIEW.

Instrucciones: Construya un VI que convierta de °C a °F. Cuando lo corra, el VI deberá de tomar un valor de entrada (°C), multiplicarlo por 1.8, sumarle 32, y desplegar el resultado (°F). El panel frontal deberá mostrar el valor de entrada y el resultado. Salve el VI como "Conversion C a F.vi"





## Técnicas para Eliminar Errores


- **Encontrando los Errores**




Haga clic en el botón de “correr” que esta roto;  
Aparece una ventana mostrando los errores
- **Resaltar la Ejecución**



Haga clic en el botón de ejecución resaltada; el  
flujo de datos es animado utilizando burbujas.  
Los valores se despliegan en los cables.
- **Herramienta de Prueba**



Haga clic con el botón derecho sobre el cable para  
exhibir la ventana de prueba y así mostrar los  
datos mientras fluyen por el segmento de cable.



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

Cuando su VI no es ejecutable, se despliega una flecha quebrada en el botón de correr en la paleta de herramientas.

**Encontrando los Errores:** Para hacer una lista de los errores, haga clic en la flecha quebrada. Para localizar el objeto malo, haga clic en el mensaje del error.

**Resaltando la Ejecucion:** Anima el diagrama y traza el flujo de datos, permitiéndole ver los valores intermedios.

Haga clic en el bombillo incandescente (**light bulb**) en la barra de herramientas.

**Probe:** Utilizado para ver los valores en los arrays (arreglos) y clusters .

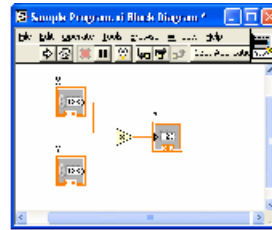
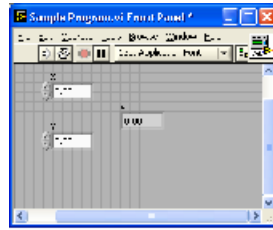
Haga clic en los cables con la herramienta **Probe** o haga clic derecho en el cable para ajustar los probes.

**Punto de Paro (Breakpoint):** Coloca pausas en diferentes lugares del diagrama.

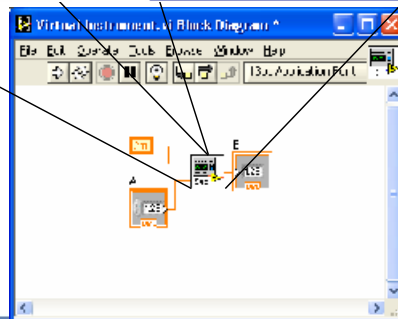
Haga clic en los cables o en los objetos con la herramienta de **Punto de Paro** para colocar los puntos de paro.

Utilice el VI **Debug Demonstrate** del BASICS.LLB para demostrar las opciones y las herramientas.

## Sección II – SubVIs



- Que es un subVI?
- Elaboración de un icono y un conector para un subVI
- Utilizando un VI como un subVI



ni.com

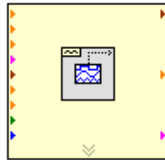
NATIONAL  
INSTRUMENTS

# Nodos del Diagrama de Bloques

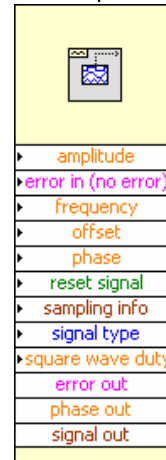
Icono



Nodo Expandible



Nodo Expandido



- VI de Generador de Funciones
- El mismo VI, visto en tres maneras diferentes.
- El campo amarillo designa un VI Estándar.
- El campo azul designa un VI Expreso

ni.com



Así como en las terminales de control o de indicador en el diagrama de bloque pueden ser vistos como un icono o como una simple terminal, los subVIs pueden ser vistos como un icono, un nodo expandible o un nodo expandido. Las diferentes vistas dependen en la preferencia del usuario y no cambia la funcionalidad del subVI.

## SubVIs

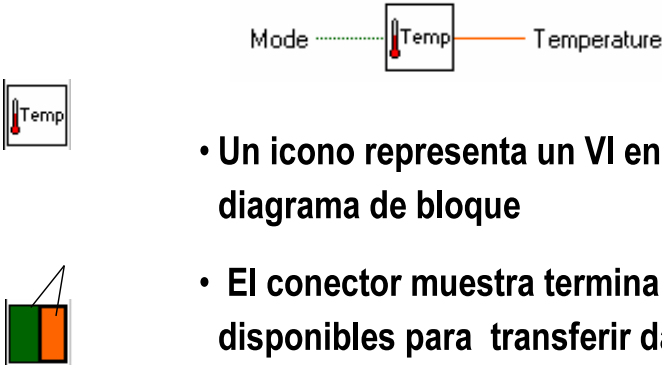
- Un SubVI es un VI que puede ser utilizado dentro de otro VI
- Similar a una subrutina
- Ventajas
  - Modular
  - Fácil para eliminar errores
  - No tiene que crear códigos
  - Requiere menos memoria

ni.com

 NATIONAL  
INSTRUMENTS

Después de haber construido un VI, y creado su propio icono y panel de conexión, este puede ser utilizado en otro VI. Un VI dentro de otro VI se llama un subVI. Un subVI corresponde a una subrutina en lenguajes de programación basados en texto. La utilización de subVIs le ayuda a manejar cambios y a eliminar errores del diagrama de bloque rápidamente.

## Icono y Conector



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

ni.com

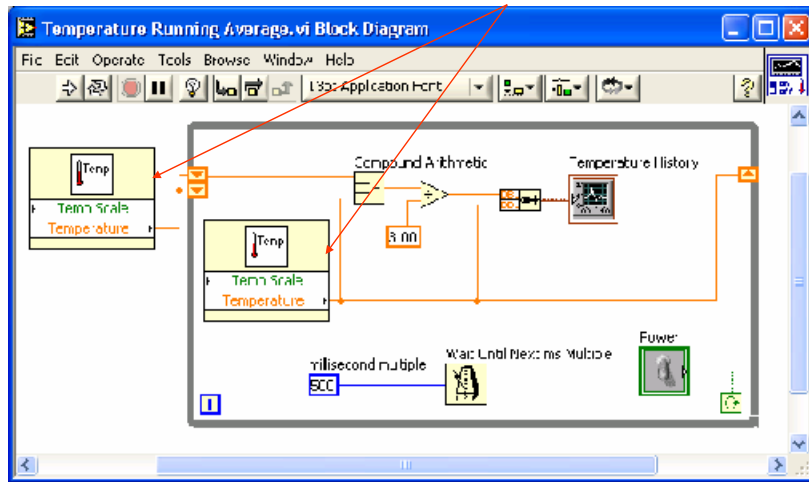
NATIONAL INSTRUMENTS

Cada VI muestra un icono, mostrado arriba, en la esquina superior derecha de las ventanas del panel frontal y del diagrama de bloque. Un icono es una representación grafica de un VI. Puede contener texto, imágenes, o una combinación de ambos. Si usted utiliza un VI como subVI, el icono identifica el subVI en el diagrama de bloque del VI.

El conector muestra terminales disponibles para la transferencia de datos hacia y desde el subVI. Hay varios patrones de conectores para elegir. Haga un clic-derecho sobre el conector y seleccione el patrón del menú de patrones. A partir de ahí usted puede asignar controles e indicadores en el panel frontal al conector terminal, como veremos mas adelante.

## SubVIs

Sub VIs



El diagrama de bloque mostrado en la parte superior contiene dos subVIs. Para ver el panel frontal de un subVI, simplemente haga un doble clic sobre el subVI. Usted también puede ver la jerarquía de subVIs dentro de un VI principal haciendo un clic en Browse>> Show VI Hierarchy.

## Pasos para crear un SubVI

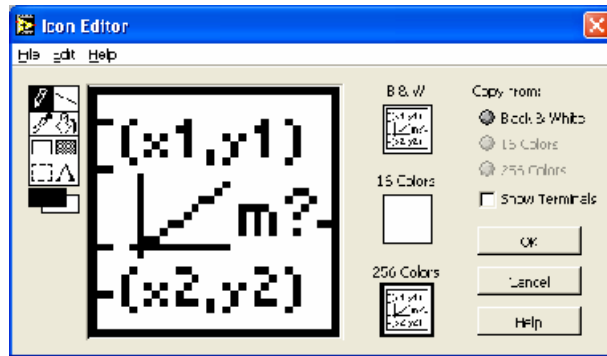
- Crear el icono
- Crear el conector
- Asignar terminales
- Salvar el VI
- Insertar el VI dentro del VI principal

[ni.com](http://ni.com)



## Crear el Icono

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



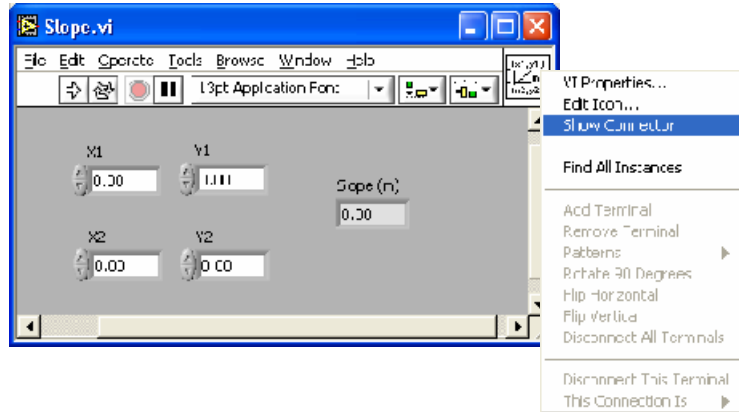
Cree iconos personalizados para reemplazar los iconos predeterminados al hacer clic-derecho en la esquina superior derecha del panel frontal o del diagrama de bloques y seleccionando el **Edit Icon** del menú de atajo o haciendo un doble clic en el icono en la esquina superior derecha del panel frontal. Usted también puede corregir iconos seleccionando **File >> VI Properties**, seleccionando **General** del menú **Category**, y haciendo clic en el botón de editar (**Edit Icon**). Utilice las herramientas del lado izquierdo de la casilla de dialogo del **Icon Editor** para crear el diseño del icono en el área de edición. El tamaño normal de la imagen del icono aparece en la casilla apropiada en la parte derecha del área de edición.

Usted también puede arrastrar un gráfico desde cualquier archivo y colocarlo en la esquina superior derecha del panel frontal o del diagrama de bloque. LabVIEW convierte el gráfico a un icono de 32 × 32 píxeles.



## Crear el Conector

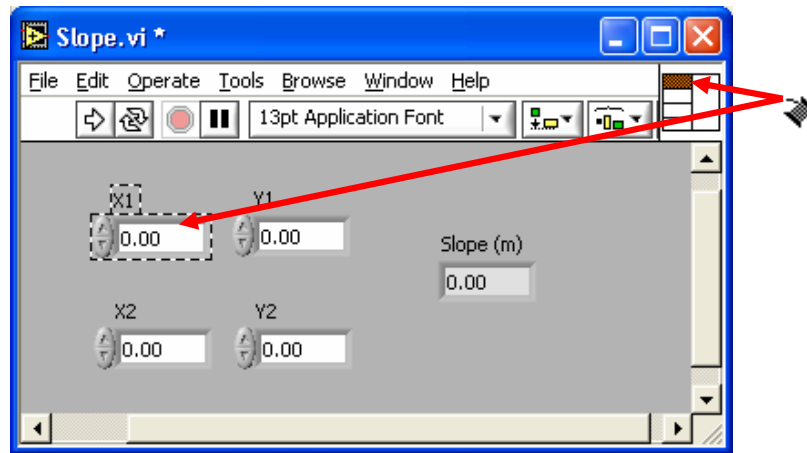
Haga click derecho sobre el icono (solamente en el panel frontal)



Para utilizar un VI como un subVI, usted necesita crear un conector. El conector es una serie de terminales que corresponden a los controles y a los indicadores de ese VI, similar a la lista de parámetros de una función basado en un lenguaje de programación de texto. El conector define las entradas y las salidas que usted puede cablear al VI para que pueda utilizarlo como subVI.

Defina las conexiones asignando un control del panel frontal o un indicador a cada una de las terminales del conector. Para definir un conector, de un clic-derecho sobre el icono en la esquina superior derecha de la ventana del panel frontal y seleccione **Show Connector** del menú. El conector substituye el icono. Cada rectángulo en el conector representa una terminal. Utilice los rectángulos para asignar entradas y salidas. El número de terminales que LabVIEW muestra en el conector depende del número de controles e indicadores en el panel frontal. El panel frontal de arriba tiene cuatro controles y un indicador, así que LabVIEW muestra cuatro terminales de entradas y una terminal de salida en el conector.

## Asignar Terminales



Después que seleccione un patrón para utilizar en su conector, debe definir las conexiones asignando un control del panel frontal o indicador a cada uno de los terminales del conector. Cuando usted hace vínculos entre controles e indicadores y el conector, coloque las entradas en la izquierda y las salidas a la derecha para evitar patrones complicados y poco claros en su VI. Para asignar una terminal a un control de panel frontal o indicador, presione una terminal del conector. Presione el control del panel frontal o indicador que usted desea asignar a la terminal. Haga clic en un área abierta del panel frontal. La terminal cambia al color del tipo de datos del control para indicar que usted conectó la terminal. Usted también puede seleccionar el control o el indicador primeramente y después seleccionar la terminal.

Asegurase que salvó el VI después de haber hecho las asignaciones de las terminales.

## Salvar el VI

- Escoja un sitio o un folder fácil de recordar
- Organizar por funcionalidad
  - salvar los VIs similares en un mismo directorio (Ej. Utilidades matemáticas)
- Organizar por aplicación
  - Salvar todos los VIs Usados para una Aplicación Especifica 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

ni.com

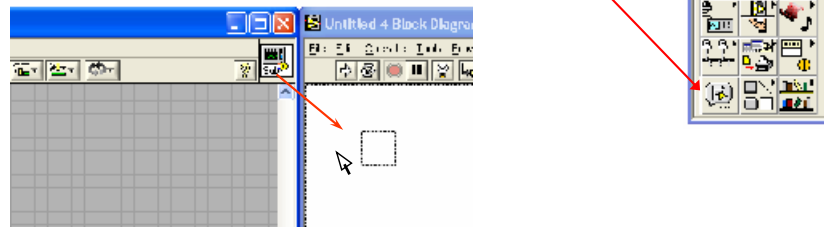


Hay varias maneras de organizar sus subVIs. La manera más común es organizarlos por aplicación. En este caso, son salvados en el mismo directorio o dentro de un archivo de librería de VI todos los VI's para una aplicación en particular. Salvarlos en un archivo de librería permite que usted transporte una aplicación entera dentro de un solo archivo.

Salvar dentro de una librería es simple. Después de presionar Save As... (salvar como), presione New VI Library (nueva librería de VI). Esto permitirá que usted nombre la librería, y entonces salvar su VI en él. Para agregar VI subsecuentes, simplemente haga un doble-clic al archivo .llb de la ventana estándar de Salvar, y dar un nombre al VI.

## Insertar el SubVI dentro de un VI Principal

Acceder subVIs creados por el usuario  
Funciones >> All Functions >> Seleccione  
un VI o  
arrastre el icono hasta el diagrama deseado



ni.com

NATIONAL  
INSTRUMENTS

Después de construir un VI y crear su icono y conector, usted puede utilizarlo como un subVI. Para colocar un subVI en el diagrama de bloque, seleccione **Functions>> Select VI**. Navegue y haga un doble-clic en el VI que usted desea utilizar como subVI y colóquelo en el diagrama de bloques.

También puede colocar un VI abierto en el diagrama de bloque de otro VI abierto usando la herramienta de colocación (Positioning Tool) para hacer clic en el icono en la esquina superior derecha del panel frontal o diagrama de bloque de el VI que usted desea utilizar como subVI y arrastrar el icono al diagrama de bloque del otro VI.

## Tips 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.

ni.com



LabVIEW tiene muchas teclas de atajo que hacen el trabajo mas fácil. Las mas comunes se listan en la diapositiva.

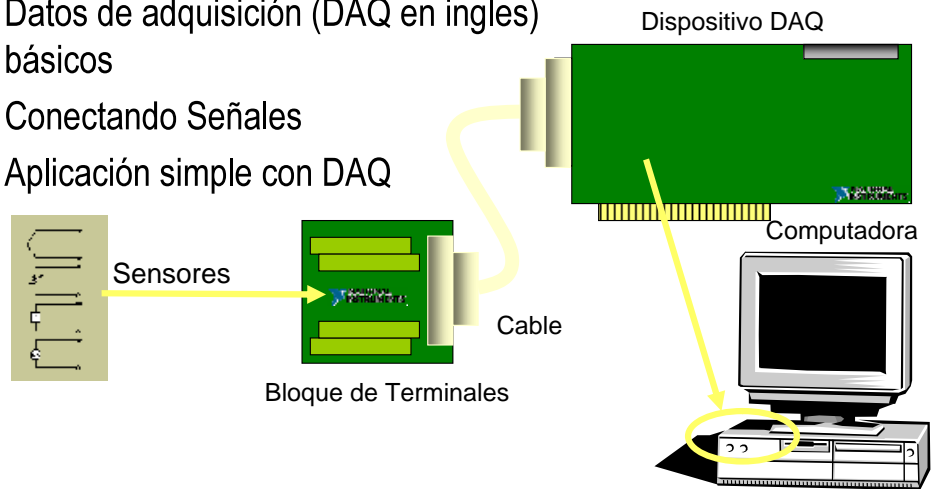
Mientras la Herramienta de Selección Automática (Automatic Selection Tool) es genial para escoger la herramienta que usted desearía utilizar en LabVIEW, a veces hay casos cuando usted desea controles manuales. Utilice la tecla Tab para cambiar entre las cuatro herramientas mas comunes (Operate Value (valor de operacion), Position/Size/Select, (Posición/Tamaño/Selección), Edit Text (editar texto), Set Color on Front Panel (establecer color en panel frontal), Connect wire on Block Diagram (conectar cable en el diagrama de bloque)). Una vez que se ha terminado con la selección de herramientas, puede presionar <Shift-Tab> para activar la Herramienta de Selección Automática.

En el dialogo de Herramientas >> Opciones... , hay muchas opciones configurables para el Panel Frontal, Diagrama de Bloque, Colores, Impresión y mucho mas.

Similar a las opciones de LabVIEW, se pueden configurar propiedades especificas del VI al ir a File >> VI Properties... Ahí se puede documentar el VI, cambiar la apariencia de la ventana, y personalizarlos de varias maneras.

### Sección III – Adquisición de Datos

- Datos de adquisición (DAQ en ingles) básicos
- Conectando Señales
- Aplicación simple con DAQ



ni.com

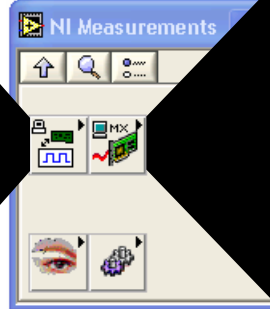
NATIONAL  
INSTRUMENTS

# Adquisición de Datos en LabVIEW

## NI-DAQ Tradicional

Vis específicos para realizar:

- Entrada Análoga
- Salida Análoga
- I/O (entrada/salida) Digital
- Operaciones de conteo



## NI-DAQmx

Controlador (Driver) de siguiente generación:

- Vis para ejecutar una tarea
- Un serie de Vis para todos los tipos de mediciones

ni.com



La plataforma de Adquisición de Datos en LabVIEW contiene una plataforma para la NI-DAQ tradicional y una para NI-DAQmx.

Los VIs tradicionales están divididos por el tipo de medición, los VIs DAQmx están divididos por el tipo de tarea.

Antes de utilizar la Adquisición de Datos de VIs se debe completar varios pasos. Los dispositivos deben ser configurados para las computadoras en esta clase.

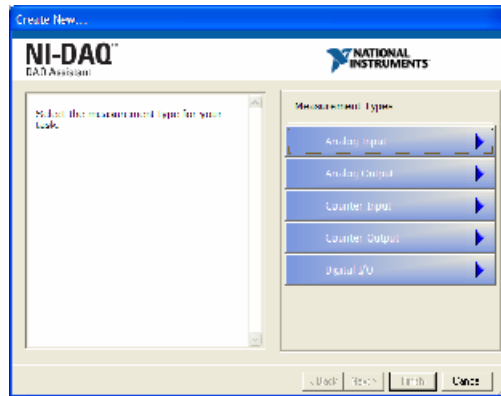
1. El software NI-DAQ debe estar instalado en la computadora.
2. Debe tener instalado una tarjeta E-series DAQ y esta debe ser configurada usando el Explorador de Automatización & Mediación (Measurement & Automation Explorer (MAX)).

Para mas información sobre la instalación y configuración de hardware de National Instruments, consultar el DAQ Quick Start Guide:

[http://digital.ni.com/manuals.nsf/websearch/E502277FE33ED60686256B3B0056AEDF?OpenDocument&node=132100\\_US](http://digital.ni.com/manuals.nsf/websearch/E502277FE33ED60686256B3B0056AEDF?OpenDocument&node=132100_US)

## DAQ – Adquisición De Datos

Adquisición de temperatura utilizando el Asistente de DAQ  
(DAQ Assistant)



Arriba esta la ventana de ayuda del DAQ (DAQ Assistant) que puede ser configurada rápidamente para leer temperaturas de una tarjeta de Adquisición de Datos (DAQ).



## Terminología De la Adquisición De Datos

- **Resolución** – Determina cuantos diferentes cambios de voltajes pueden ser medidos
  - Resolución mas grande → Una representación mas exacta de la señal
- **Rango** – Voltajes mínimos y máximos
  - Rango mas pequeño → Una representación mas precisa de la señal
- **Gain (ganancia)** – Amplifica o atenúa la señal para un mejor ajuste del rango

ni.com



**Resolución:** Al adquirir datos a una computadora, un convertidor de Análogo-a-Digital (ADC) toma una señal análoga y la convierte a un número binario. Por lo tanto, cada número binario del ADC representa cierto nivel de voltaje. El ADC devuelve el nivel más alto posible sin pasar el nivel de voltaje real de la señal análoga. La resolución se refiere al número de niveles binarios que el ADC puede utilizar para representar una señal. Para encontrar el número de niveles de voltaje disponibles basados en la resolución usted simplemente toma  $2^{\text{Resolución}}$ . Por lo tanto, cuanto más alta es la resolución, más son los niveles que usted tendrá para representar su señal. Por ejemplo, un ADC con 3-bits de resolución puede medir  $2^3$  o 8 niveles de voltajes, mientras que un ADC con 12-bit de resolución puede medir  $2^{12}$  o 4096 niveles de voltajes.

**Rango:** A diferencia de la resolución del ADC, el rango del ADC es seleccionable. La mayoría de los dispositivos DAQ ofrecen un rango desde 0 - +10 o -10 a +10. Se elige el rango cuando usted configura su dispositivo en NI-DAQ. Tenga presente que la resolución del ADC será extendida por cualquier rango que usted elija. Cuanto más grande sea el rango, mas esparcida será su resolución, y tendrá una peor representación de su señal. Entonces es importante escoger su rango para apropiadamente encajar su señal de entrada.

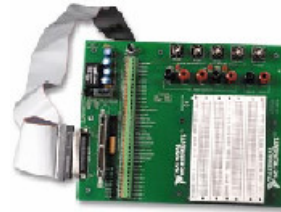
**Ganancia:** Escogiendo apropiadamente el rango de su ADC es una manera de asegurarse de que usted esta maximizando la resolución de su ADC. Otra manera de ayudar a su señal a maximizar la resolución del ADS es aplicando una ganancia. Ganancia se refiere a cualquier amplificación o atenuación de una señal. El ajuste de ganancia (gain setting) es un factor de escala. Cada nivel de voltaje en su señal entrante es multiplicado por el ajuste de ganancia para obtener la señal amplificada o atenuada. A diferencia de la resolución que es un ajuste fijo (fixed setting) del ADC, y el rango que es escogido cuando el dispositivo DAQ es configurado, la ganancia es especificada indirectamente a través de un ajuste llamado límites de entrada. Los límites de entrada se refieren a los valores máximos y mínimos de su señal de entrada analógica actual. Basado en los límites de entrada que usted defina, la ganancia posible más grande que mantendrá la señal entre el rango escogido del ADC es aplicada a su señal. Así que en lugar de necesitar calcular la mejor ganancia basada en su señal y en el rango escogido, todo lo que usted necesita saber son los valores máximos y mínimos de su señal.

## Conexiones de Hardware

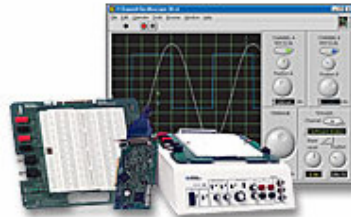
**BNC-2120**



**SC-2075**



**NI-ELVIS**



**SCB-68**



ni.com

NATIONAL  
INSTRUMENTS

Hay muchos arreglos diferentes posibles de hardware para adquirir datos. Todos los sistemas de la Adquisición de Datos requieren algún tipo de conexión terminal que acepta una señal de su transductor y lo transmite a la tarjeta DAQ. Cuatro de estos bloques de terminales son el BNC-2120, SC-2075, SCB-68, y NI-ELVIS.

El BNC-2120 es un bloque de conexiones aislado (shielded) con conectores BNC de señal-etiquetada para la conectividad fácil a su dispositivo DAQ. También proporciona un generador de funciones, encoder (codificador) de cuadratura, referencia de la temperatura, conector de termocupla, y LED para que usted pueda probar la funcionalidad de su hardware.

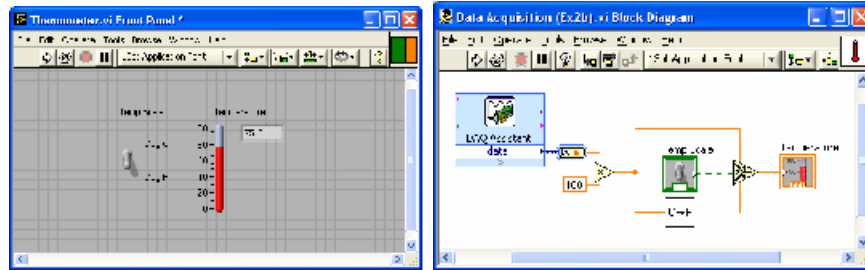
El SC-2075 proporciona áreas para conexión (breadboard areas) para hacer prototipos y conectividad BNC y de terminales de resorte (spring terminal). Los  $\pm 15$  V de alimentación fija así como la alimentación ajustable de 0 a 5 V y LEDs hacen del SC-2075 un instrumento ideal para laboratorios académicos.

El SCB-68 es un bloque de conexión de E/S (I/O) aislado (shielded) para señales resistentes y de bajo-ruido. Incluye dos áreas de conexión (breadboard areas) de propósito-general así como un sensor de temperatura IC para la compensación de empalme frío en mediciones de temperatura.

NI-ELVIS (Sistema de Laboratorio Educacional de Instrumentación Virtual) es un ambiente de prototipo y diseño basado en LabVIEW y consiste de instrumentos virtuales basados en LabVIEW, un dispositivo de adquisición de datos multifuncional, y en una estación de trabajo con un banco y tablero de diseño y

## Ejercicio 2 – Adquisición de Datos Simples

Complete el VI de conversión de C a F (Conversion C a F.vi), y después cree el VI Termómetro (Thermometer.vi).



Nota: Para completar este ejercicio, usted necesitará el sensor de la temperatura IC disponible en el BNC-2120, SCB-68 o Accesorio de Señal DAQ.

El tiempo estimado para completarse: 30 minutos.

Instrucciones: Este ejercicio tiene tres partes.

Primero, cree un icono y conector para Conversion C a F.vi (Ejercicio 1). El icono debe recordarle la funcionalidad del VI (e.g.  $C \rightarrow F$  o C a F). El conector debe tener una entrada y una salida, permitiendo una terminal para °C de entrada, y °F de salida.

Segundo, cree un VI principal que adquiera un punto de datos del canal 0 (el sensor de la temperatura) de su tarjeta DAQ y permita al usuario mostrar la temperatura en Centígrados o Fahrenheit. Para hacer esto usted necesitará adquirir un solo punto de datos de su tarjeta DAQ y escalarlo por un factor de 100. Esto le dará °C. Usted debe tener un interruptor o botón Booleano que le permita al usuario escoger Centígrados o Fahrenheit. Si el usuario escoje Centígrados, el valor escalado se debe mostrar en un indicador de termómetro. Si el usuario escoje Fahrenheit, el valor en Centígrados se debe de pasar por el Conversion C a F.vi (utilizado como un subVI), y el valor de salida Fahrenheit debe ser desplegado.

*Sugerencia:* Utilice la función **Select** (seleccionar) en la paleta de **Comparison** (comparación).

Finalmente, cree un Icono y Conector para Termómetro.vi. Un posible Icono seria una imagen de un termómetro. El conector debería tener dos terminales. Uno para la entrada Booleana (°C or °F), y el segundo para la salida de la temperatura escalada. Guarde el VI como termómetro.vi

## Sección IV – Ciclos y Gráficas

- For Loop (Ciclo Para)
- While Loop (Ciclo Mientras)
- Gráficas
- Multiplots

[ni.com](http://ni.com)



Dele clic para agregar notas

# Ciclos

- **Ciclo Mientras**
  - Tiene una Terminal de Iteración
  - Siempre corre al menos una vez
  - Corre de acuerdo a la Terminal Condicionante
- **Ciclos Para**
  - Tiene una Terminal de Iteración
  - Corre de acuerdo a la entrada N de la Terminal de Conteo

ni.com

NATIONAL INSTRUMENTS

Ambos los ciclos Mientras (While) y Para (For) están localizados en la paleta **Functions»Structures**. El ciclo Para difiere del ciclo Mientras en que el ciclo Para ejecuta una serie (set) de veces predeterminada. El ciclo Mientras deja de ejecutar el sub diagrama solo si el valor en la condición terminal existe.

## Ciclo Mientras

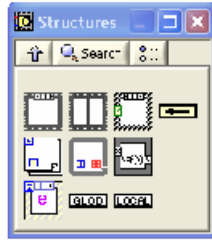
Similar al ciclo Haga (Do) o al ciclo Repita-hasta (Repeat-Until) en lenguajes de programación basados en texto, un ciclo Mientras, mostrado arriba a la derecha, ejecuta un sub diagrama hasta que la condición sea cumplida. El ciclo Mientras ejecuta el sub diagrama hasta que la terminal dependiente, recibe un valor Booleano específico. El comportamiento incumplido y la apariencia de la terminal dependiente (valor por defecto) es **Continue if True** (continúe si es Verdadero), mostrado a la izquierda. Cuando una terminal dependiente es **Continue if True**, el ciclo Mientras ejecuta su sub diagrama hasta que la terminal dependiente recibe un valor FALSO. La terminal de iteración (una terminal de salida), mostrada a la izquierda, contiene el número de iteraciones completas. El conteo de iteraciones siempre empieza en cero. Durante la primera iteración, la terminal de iteración regresa a cero.

## Ciclo Para

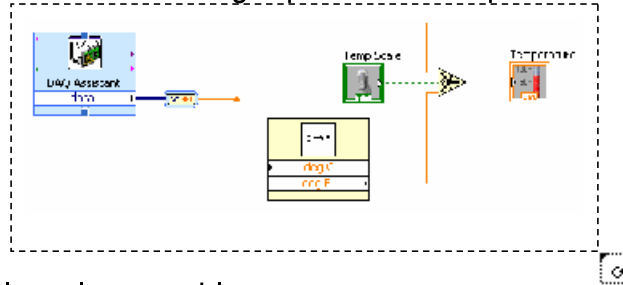
Un ciclo Para, mostrado a la izquierda, ejecuta una serie varias veces. El valor en la terminal de conteo (una terminal de entrada) representada por la N, indica cuantas veces repetir el sub diagrama. La terminal de iteración (una entrada de salida), mostrada a la izquierda, contiene el número de iteraciones completas. El conteo de iteraciones siempre empieza en cero. Durante la primera iteración, la terminal de iteración regresa a cero.

## Ciclos (cont.)

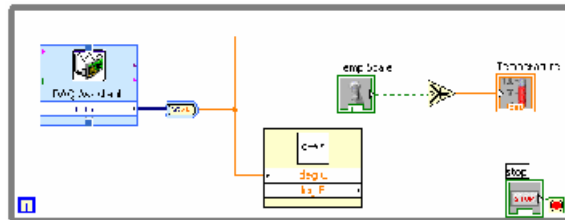
1. Seleccionar el ciclo



2. Encerrar código que va a ser repetido



3. Arrastrar nodos adicionales y luego cablear



Coloque ciclos en su diagrama seleccionándolos desde la paleta de Estructuras de la paleta de Funciones (demostrar)

Cuando este seleccionado, el cursor del ratón se convierte en un puntero especial que usted usa para

encerrar la sección del código que desea repetir.

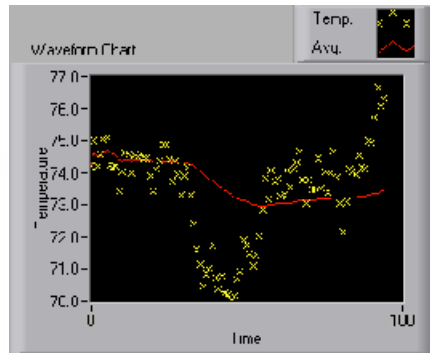
Déle clic al botón del ratón para definir la esquina superior-izquierda, déle clic otra vez al botón del ratón en

la esquina inferior izquierda, y la frontera del ciclo Mientras es creada alrededor del código seleccionado.

Arrastre nodos adicionales adentro del ciclo Mientras si es necesario.



## Gráficas



Gráfica de forma de onda –  
indicador numérico especial  
que puede mostrar una  
historia de valores

**Controles >> Indicadores de  
Gráficas >> Gráfica de  
forma de onda**

ni.com

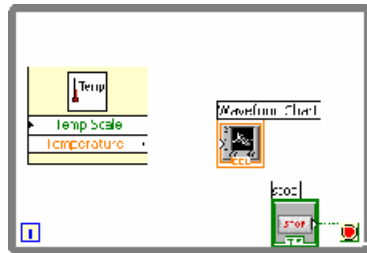
NATIONAL  
INSTRUMENTS

La grafica de forma de onda es un indicador numérico especial que muestra una o más graficas (plots). La grafica de forma de onda esta localizada en la paleta **Controls>Graph Indicators**. Las graficas de forma de onda pueden mostrar graficas simples o múltiples. El siguiente panel frontal muestra un ejemplo de una grafica de forma de onda múltiple.

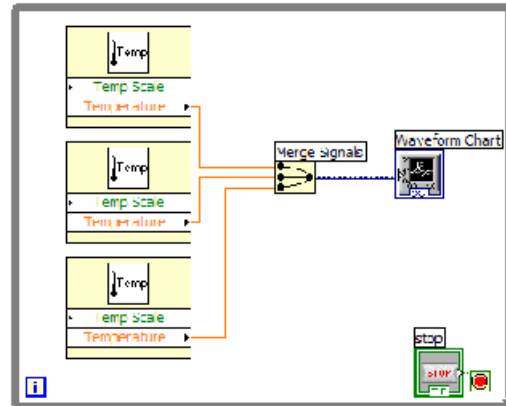
Usted puede cambiar los valores min y max del eje x y y dándole doble clic al valor con la herramientas de etiquetado y escribiendo el nuevo valor. Similarmente, usted puede cambiar la etiqueta del eje. También puede darle clic derecho a la leyenda de la grafica y cambiar el estilo, forma y color del dibujo que se muestra en la gráfica.

## Cableando Datos a las Gráficas

Grafica de diagrama simple



Grafica de diagrama múltiple



ni.com

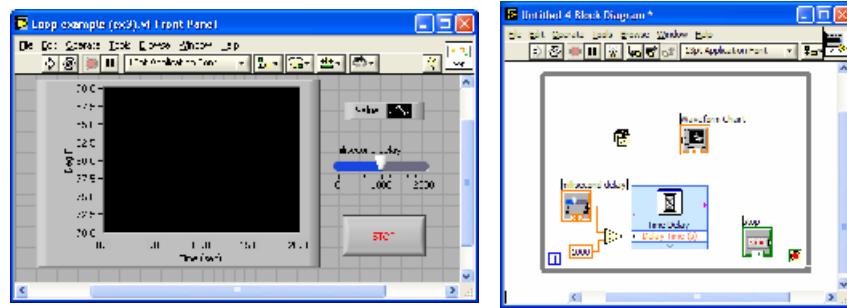
NATIONAL  
INSTRUMENTS

Usted puede cablear una salida escalar directamente a una grafica de forma de onda para mostrar un diagrama. Para mostrar múltiples diagramas en una grafica, use la función **Merge Signals** encontrada en la paleta **Functions >> Signal Manipulation**. La función **Merge Signal** junta múltiples salidas para diagramar en la gráfica de forma de onda. Para agregar mas diagramas, utilice la herramienta de Posicionamiento para cambiar el tamaño de la función **Merge Signal**.

La ayuda de contexto contiene muy buena información en como cablear de diferentes maneras los datos dentro de las graficas.

## Ejercicio 3 – Usando ciclos

Estudiantes construyen el VI Ejemplo de Ciclo.vi.



Este ejercicio debe tomar de 15-20 minutos.

Instrucciones:

Cree un VI que genere un número aleatorio a una velocidad específica y muestre las lecturas en una grafica de forma de onda hasta que el usuario lo pare. Conecte la terminal de terminación a un botón de paro en el panel frontal, y agregue un control de deslizamiento (slider control) al panel de control. Este control de deslizamiento debe recorrer de 0 a 2000 en valor, y será conectado a la función Express VI **Time Delay** (demora de tiempo) dentro de su ciclo Mientras.

Guarde el VI como Use a Loop.vi

## Sección V – Arreglos & Manejo de Archivo (File I/O)

- Construir arreglos manualmente
- Dejar que LabVIEW construya arreglos automáticamente
- Escribir a un archivo de hoja de cálculo (spreadsheet)
- Leer desde un archivo de hoja de cálculo (spreadsheet)

ni.com



Los arreglos agrupan elementos de datos del mismo tipo. Un arreglo consiste de elementos y dimensiones. Los elementos son los datos que componen el arreglo. Una dimensión es la longitud, altura, o profundidad de un arreglo. Un arreglo puede tener una o mas dimensiones y hasta  $2^{31} - 1$  elementos por dimensión, si lo permite la memoria.

Usted puede construir arreglos de tipos de datos numéricos, Booleanos, rutas, cadenas, formas de onda y cluster data. Considere usar arreglos cuando trabaje con una colección de datos similares y cuando realice computaciones repetitivas. Los arreglos son ideales para almacenar información que se obtiene de formas de ondas o información generada en los ciclos, donde cada iteración de un ciclo produce un elemento del arreglo.

Los elementos del arreglo son ordenados. Un arreglo utiliza un índice para que usted pueda tener acceso rápido a cualquier elemento en particular. El índice es basado en cero, lo que significa que tiene un rango de 0 a  $n - 1$ , donde  $n$  es el numero de elementos en el arreglo. Por ejemplo,  $n = 9$  para los nueve planetas, así que el índice recorre de 0 a 8. La Tierra es el tercer planeta, así que tiene un índice de 2.

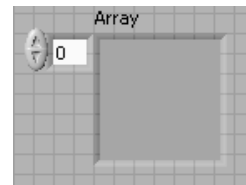
Las operaciones de Entrada/Salida de archivo (I/O) pasan los datos desde y a los archivos. Utilice los VI de File I/O y las funciones localizadas en la paleta **Functions » File I/O** para manejar todos aspectos de I/O de archivo. En esta clase nosotros cubriremos la lectura y escritura de archivos de hoja de cálculo (spreadsheet) utilizando el Express VIs para File I/O.

## Agregando un Arreglo al Panel Frontal

Desde la sub-paleta de **Controls >> All Controls >> Array and Cluster**, seleccione el **Array Shell**



Colóquelo en la pantalla.



ni.com

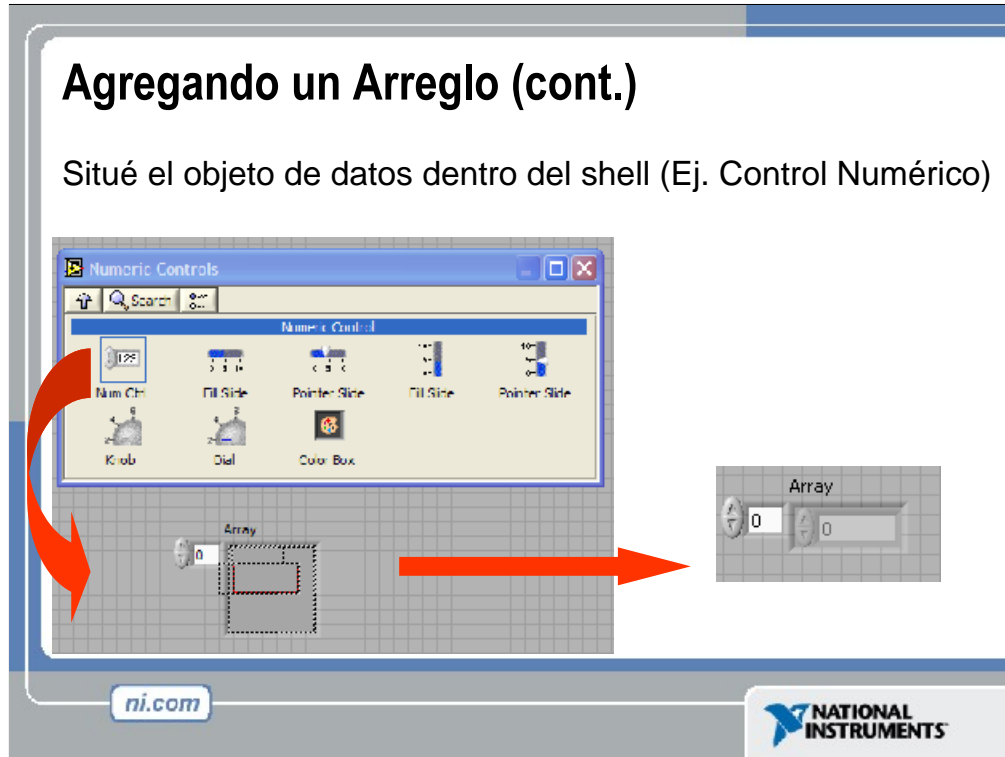
NATIONAL  
INSTRUMENTS

Para crear un control o indicador de un arreglo como el mostrado, escoja un arreglo (array) en la paleta **Controls » All Controls » Arrays & Cluster**, colóquelo en el panel frontal, y arrastre un control o un indicador adentro de la estructura (shell) del arreglo. Si usted procura arrastrar un control o indicador inválidos tales como una gráfica XY adentro de la estructura del arreglo, usted será incapaz de dejar caer el control o el indicador en la estructura del arreglo.

Usted debe insertar un objeto en la estructura del arreglo antes de que usted utilice el arreglo en el diagrama de bloques. De otro modo, la terminal del arreglo aparece negra con un paréntesis vacío.

## Agregando un Arreglo (cont.)

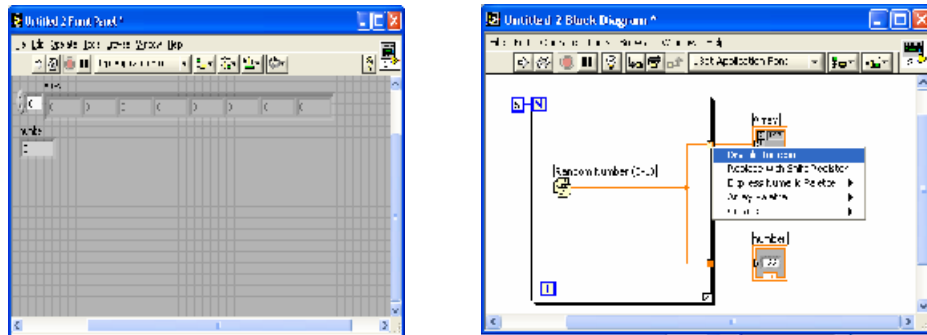
Situé el objeto de datos dentro del shell (Ej. Control Numérico)



Para agregar las dimensiones a un arreglo de uno en uno, hace clic derecho en el despliegue del índice y seleccione **Add Dimension** del menú de atajo. Usted puede utilizar también la herramienta que Posicionamiento para cambiar el tamaño del despliegue del índice hasta que usted tenga las dimensiones que usted quiera.

## Creando un Arreglo con un Ciclo

- Los ciclos acumulan arreglos en sus limites.



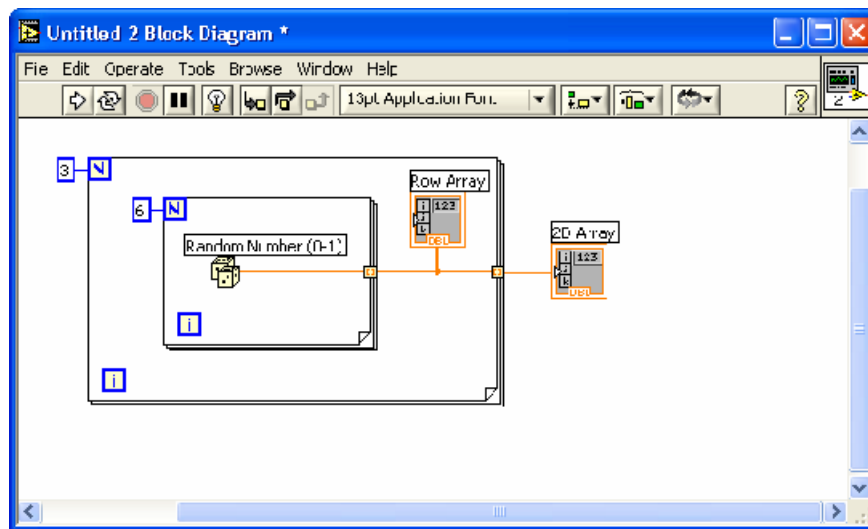
Si se cablea un arreglo a un tunel de entrada de un ciclo Para (For Loop) o un ciclo Mientras (While Loop), usted puede leer y procesar cada elemento en ese arreglo al activar el auto-indexing (auto-indicación). Cuando usted utiliza el auto-índice en un arreglo de túnel de salida el arreglo de salida recibe un nuevo elemento por cada iteración del ciclo. El cable que va del túnel de salida al arreglo indicador se vuelve mas grueso al cambiar a un arreglo en el borde del ciclo, y el túnel de salida contiene corchetes cuadrados representando un arreglo, como se muestra en la siguiente ilustración.

Desactive el auto-indicador al hacer un clic derecho en el tunel y seleccionar **Disable Indexing** (desactivar Indicación) desde el menu de atajo. Por ejemplo, desactive el auto-indicador si necesita únicamente el ultimo valor pasado al tunel en ejemplo anterior, sin crear un arreglo.

**Nota:** Debido a que usted puede utilizar ciclos Para (For Loops) para procesar arreglos y elemento a la misma vez, LabVIEW, activa el auto-indicador como opción por defecto para cada arreglo que se cablee a un Ciclo Para. El Auto-Indicador para Ciclos Mientras (While Loops) esta desactivados por defecto. Para activar el auto-indicador, haga clic derecho en un tunel y seleccione **Enable Indexing** del menu de atajo.

Si usted activa el auto-indicador en un arreglo cableado a una terminal de entrada de un Ciclo Para, LabVIEW arregla el contador de la terminal al tamaño del arreglo para que así usted no tenga que cablear la terminal del

## Creando Arreglos Bidimensionales (2D)



ni.com

NATIONAL  
INSTRUMENTS

Se pueden utilizar dos Ciclos Para, uno dentro del otro, para crear arreglos bidimensionales. El Ciclo Para externo crea las filas de elementos y el Ciclo Para interno crea las columnas de elementos.



## Manejo de Archivos (File I/O)

**Manejo de Archivos** – transferir datos a y desde archivos

- Los Archivos pueden ser binarios, texto u hojas de calculo
- Escritura/Lectura de Archivos de Medición de LabVIEW (LabVIEW Measurements – \*.lvm)

Escritura en Archivo LVM

Lectura de un Archivo LVM



ni.com

NATIONAL  
INSTRUMENTS

Operaciones de Entradas y Salidas de archivo (File I/O) pasan datos desde y hacia archivos. En LabVIEW, usted puede utilizar funciones de archivo I/O para:

Abrir y cerrar archivos de datos.

Lectura de datos desde y escribir datos a archivos

Lectura y escritura de datos a documentos en formato de hoja de calculo (spreadsheet)

Mover y retitular los archivos y directorios

Cambiar las características del archivo

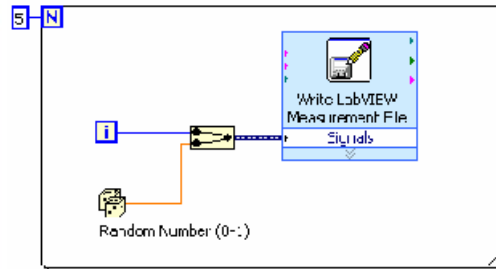
Crear, modificar, y leer archivos de configuración

Escribir a o leer en archivos de Medición LabVIEW (LabVIEW Measurement Files).

En este curso examinaremos cómo escribir a o leer desde archivos de Medición LabVIEW (archivos \*.lvm).

## Escritura de Archivos de Medición LabVIEW (LabVIEW Measurement)

- Incluye las funciones de apertura, escritura, cierre y manejo de errores
- Maneja el formateo de cadenas de caracteres (strings) ya sea con delimitación por medio de tab o de coma
- La función de concatenar señales es usada para combinar datos en datos de tipo dinámico



	A	B	C	D
1		0	J 100.000	
2		1	C 23516	
3		2	D 985184	
4		3	E 1177303	
5		4	F 550310	
6				
7				

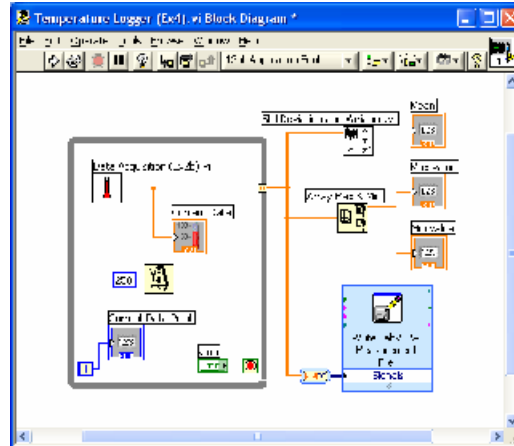
ni.com

NATIONAL  
INSTRUMENTS

El archivo de escritura LVM puede escribir a archivos de hoja de calculo (spreadsheet). Sin embargo, su propósito principal es el de introducir datos, que luego serán utilizados en LabVIEW. Este VI crea un archivo .lvm que se puede abrir en una aplicación de hoja de calculo. Para los archivos de hoja de calculo simple, utilice el VIs expreso: **Write LVM y Read LVM**.

## Ejercicio 4 – Analizando e Introduciendo Datos

Los Estudiantes construyen el VI Logger de Temperatura.vi



ni.com

NATIONAL  
INSTRUMENTS

Tiempo de completación estimado: 30-45 minutos.

### Instrucciones:

Cree un VI que adquiera y exhiba datos de temperatura a una velocidad fija hasta que el usuario lo detenga. Si usted ha terminado el ejercicio 2 y tiene una tarjeta DAQ, utilice Thermometer.vi para obtener sus datos. Si usted no ha terminado el ejercicio ni tiene una tarjeta de DAQ, usted puede utilizar el Digital Thermometer.vi de la subpaleta Tutorial de la paleta de funciones.

Una vez que esté parado, el VI debe realizar análisis en los datos que recogió mientras estaba corriendo. Acumule un arreglo de puntos y de valores en la frontera del túnel del Ciclo Mientras (While Loop). Encuentre el máximo, el mínimo, y el valor medio de los datos de la temperatura y exíbalos en indicadores numéricos (la función **Mean** (media) puede ser encontrada en **Functions» Analyze» Mathematics» Probability and Statistics**, y la función **Array Max & Min** (máximo y mínimo de un arreglo) puede ser encontrada en **Functions» Array**). Utilice el Write LabVIEW Measurements File Express VI (VI de escritura expreso de documentos de medición LabVIEW) que puede ser encontrado en **Functions» Output**. Una vez que se corra, verifique que el archivo fue creado correctamente abriéndolo en Notepad o creando un VI que lo lea de nuevo utilizando el Read LabVIEW Measurements File. Una vez que haya terminado el ejercicio, guarde su VI como Logger de Temperatura.vi.

## Sección VI – Funciones de Arreglos y Graficos

- Funciones Básicas de los Arreglos
- Uso de gráficos
- Crear Diagramas Múltiples con Gráficos

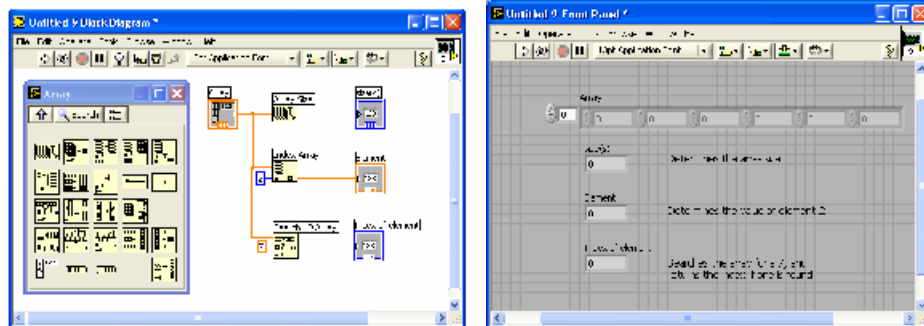
[ni.com](http://ni.com)



[Haga click para adicionar notas.](#)

# Funciones de los Arreglos – Lo Básico

Functions >> All Functions >> Array

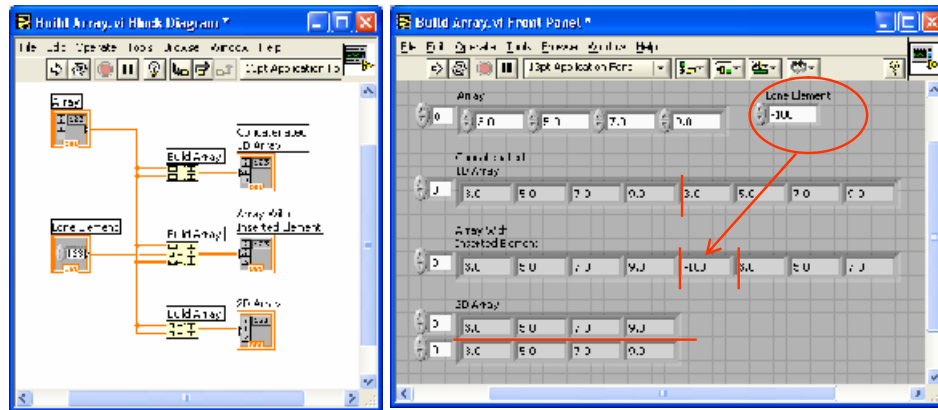


Utilice las funciones de los arreglos situadas en la paleta **Functions» All Functions» Array** para crear y para manipular arreglos.

Las funciones de los arreglos incluyen las siguientes:

- **Array Size** (Tamaño del Arreglo) – devuelve el número de elementos en cada dimensión de un arreglo. Si el arreglo es de n-dimensiones, el tamaño (**size**) de la salida es un arreglo de n elementos.
- **Initialize Array** (Inicializar el Arreglo) – crea un arreglo de n dimensiones en donde cada elemento ha sido inicializado al valor del elemento (**element**). Cambie el tamaño de la función para incrementar el número de dimensiones del arreglo de salida.
- **Build Array** (Construcción del Arreglo) -- concatena arreglos múltiples o añade elementos a un arreglo de n dimensiones. Cambie el tamaño de la función para incrementar el número de dimensiones del arreglo de salida.
- **Array Subset** (Subconjunto de Arreglo) – devuelve parte de un arreglo que comienza en el índice y que contiene elementos de la longitud.
- **Index Array** (Índice del Arreglo) – devuelve un elemento de un arreglo en el índice. Usted también puede utilizar la función Índice del Arreglo para extraer una fila o una columna de un arreglo bidimensional para así crear un subarreglo del original. Para hacer esto cablee un arreglo bidimensional a la entrada de la función. Dos terminales de índice están disponibles. La terminal del índice superior indica la fila, y la segunda terminal indica la columna. Usted puede cablear entradas a ambas terminales de índice para indicar un solo elemento, o usted puede cablear solamente una terminal para extraer una fila o una columna de datos. Por ejemplo, cablee el arreglo

## Funciones de los Arreglos – Construcción de un Arreglo



Build Array (Construcción de Arreglos) puede realizar dos funciones distintas. Concatena múltiples arreglos o añade elementos a un arreglo de  $n$  dimensiones. Cambie el tamaño de la función para incrementar el número de dimensiones del arreglo de salida. Para concatenar las entradas en un arreglo más largo de la misma dimensión según lo demostrado en el siguiente arreglo, haga click derecho en el nodo de la función y seleccione **Concatenate Inputs** (Concatenar Entradas) desde el menú de atajo.

## Gráficos

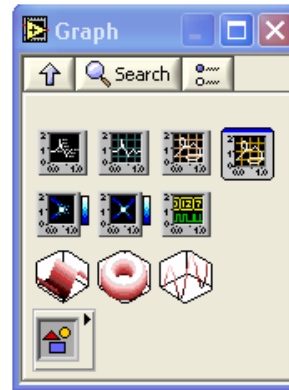
- Seleccionados desde la Paleta de Controles del Menu **Controls >> All Controls >> Graphs**

### Grafico de Forma de Onda

(Waveform Graph) – Grafica un arreglo de numeros en contra de sus indices

Grafico XY Expreso (Express XY Graph) – Grafica un array en contra de otro

Grafico de Forma de Onda Digital (Digital Waveform Graph) – Grafica bits de datos binarios



ni.com

NATIONAL INSTRUMENTS

VIs con gráficos usualmente recolectan los datos en un arreglo y después dibujan estos datos en una gráfica.

Los gráficos situados en la paleta **Controls >> All Controls >> Graph** incluyen el gráfico de forma de onda y el gráfico XY. El diagrama del gráfico de forma de onda traza solamente funciones de una sola variable, como en  $y = f(x)$ , con puntos distribuidos uniformemente a lo largo del eje x, tal como formas de onda adquiridas con tiempo variable. Los gráficos expresos XY exhiben cualquier set de puntos, ya sea que estos sean muestreados uniformemente o no. Cambie el tamaño de la leyenda de la grafica para exhibir múltiples diagramas. Utilice diagramas múltiples para ahorrar espacio en el panel frontal y para hacer hacer comparaciones entre diagramas. Los gráficos XY y de forma de onda se adaptan automáticamente a los diagramas múltiples.

### Grafica de Forma de Onda de Trazo-Simple (Single-Plot Waveform Graph)

Los gráficos de forma de onda aceptan un solo arreglo de valores e interpreta los datos como puntos en el gráfico e incrementa el indice x por uno comenzando en  $x = 0$ . El gráfico también acepta un cluster de un valor inicial x, un .x, y de un arreglo de datos y. Refierase al VI del gráfico de forma de onda (Waveform Graph VI) en `examples\general\graphs\gengraph.llb` para encontrar ejemplos de los tipos de datos que los gráficos de forma de onda de trazo-simple aceptan.

## Grafico de Forma de Onda de Trazo- Múltiple

Los gráficos de forma de onda de trazo múltiple aceptan un arreglo bidimensional de valores, en donde cada fila del arreglo es una grafica sencilla. La gráfica interpreta los datos como puntos en la gráfica e incrementa el indice x por uno, comenzando en  $x = 0$ . Cablee un arreglo de datos bidimensionales a la gráfica, haga clic derecho en la gráfica, y seleccione **Transpose Array** (arreglo transpuesto) del menú de atajo para manejar cada columna del arreglo como un diagrama aparte. Refiérase al (Y) Multi Plot 1 graph en el VI de Waveform Graph en `examples\general\graphs\gengraph.llb` para encontrar un ejemplo de un gráfico que acepte este tipo de datos.

Un gráfico de forma de onda de diagramas múltiples también acepta un cluster de un valor  $x$ , un valor  $\Delta x$ , y de un arreglo bidimensional de datos  $y$ . El gráfico interpreta los datos de  $y$  como los puntos en el gráfico e incrementa el indice  $x$  por  $\Delta x$ , comenzando en  $x = 0$ . Refierase a  $(X_0, \Delta x, Y)$  Multi Plot 3 graph en el VI de Waveform Graph que se encuentra en `examples\general\graphs\gengraph.llb` para encontrar un ejemplo de un gráfico que acepte este tipo de datos.

Un gráfico de forma de onda de diagramas múltiples acepta un cluster de un valor inicial  $x$ , de un valor  $\Delta x$ , y de un arreglo que contenga clusters. Cada cluster contiene un arreglo de puntos que contiene los datos de  $y$ . Usted utiliza la función de Bundle (empaquetado) para empaquetar los arreglos en clusters, y usted utiliza la función de Build Array (construir arreglo) para construir los clusters resultantes en un arreglo. Usted también puede utilizar el Build Cluster Array (construir arreglo de clusters), el cual crea arreglos de clusters que contienen entradas que usted especifica. Refiérase a  $(X_0, \Delta x, Y)$  Multi Plot 2 en el VI de Waveform Graph que se encuentra en `examples\general\graphs\gengraph.llb` para encontrar un ejemplo de un gráfico que acepte este tipo de datos.

## Single-Plot XY Graph (Grafica XY de Trazo Simple)

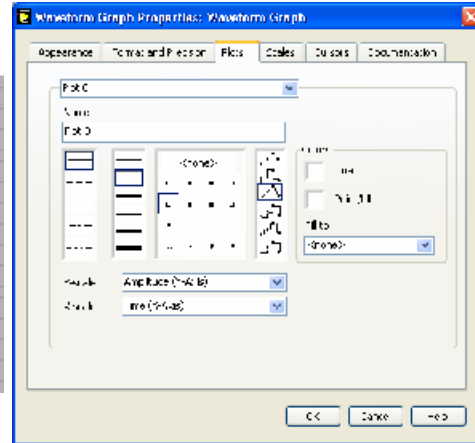
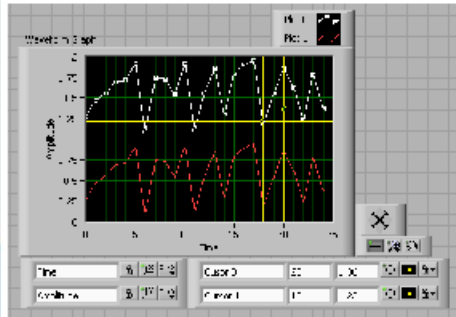
Los gráficos XY de trazo simple aceptan un cluster que contenga un arreglo  $x$  y un arreglo  $y$ . El gráfico de XY también acepta un arreglo de puntos, en donde un punto es un cluster que contiene un valor de  $x$  y un valor de  $y$ . Refiérase al VI de XY Graph en `examples\general\graph\gengraph.llb` para encontrar un ejemplo del tipo de datos de los gráficos de XY de trazo simple.

## MultiPlot XY Graph (Grafica XY de diagramas múltiples)

La Grafica XY de diagramas múltiples acepta un arreglo de diagramas, en donde un diagrama es un cluster que contiene un arreglo  $x$  y un arreglo  $y$ . La Grafica XY de diagramas múltiples también acepta un arreglo de clusters de diagramas, en donde una grafica es un arreglo de puntos. Un punto es un cluster que contiene un valor  $x$  y un valor  $y$ . Refiérase al VI del XY Graph encontrado en `examples\general\graph\gengraph.llb` para encontrar un ejemplo de los tipos de datos del Grafico XY de diagramas múltiples.



## Gráficos



Haga Clic Derecho en la grafica y seleccione Properties (propiedades) para poder personalizar interactivamente

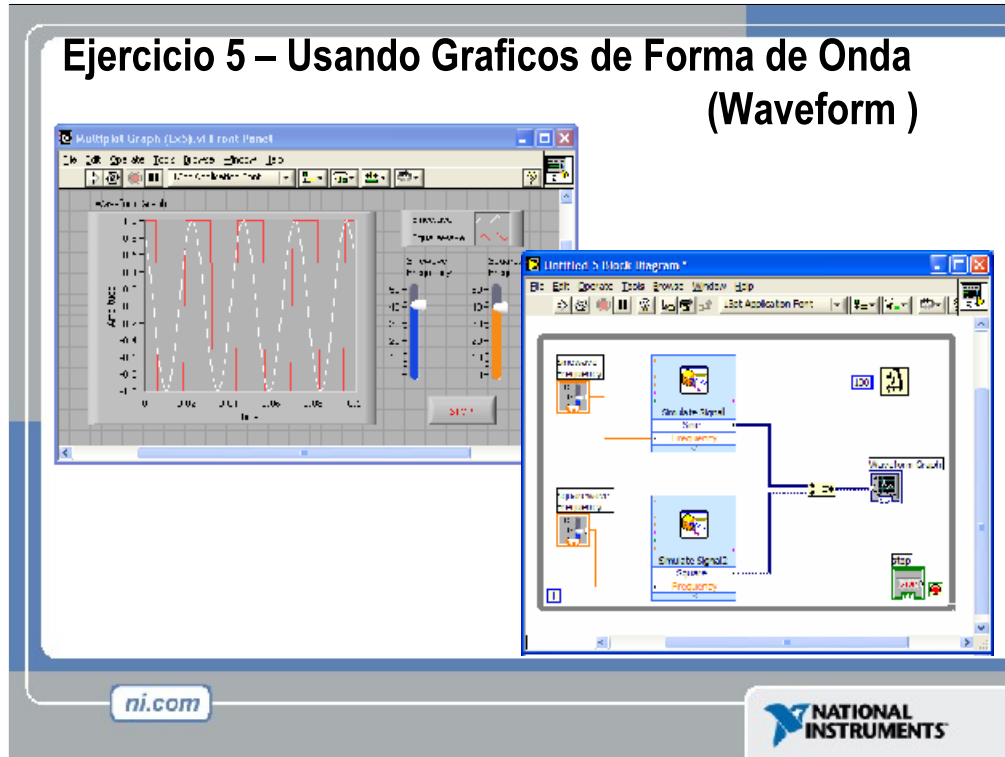
ni.com

NATIONAL  
INSTRUMENTS

Los Gráficos son indicadores muy poderosos en LabVIEW. Pueden ser altamente adaptables y pueden ser usados para mostrar de forma concisa una gran cantidad de información.

La pagina de propiedades de las Graficas le permite mostrar las opciones de configuración, escala, cursores y muchas otras características de las gráficas.

## Ejercicio 5 – Usando Graficos de Forma de Onda (Waveform )



Tiempo De Completación Estimado: 20 minutos.

Cree un VI como el que esta representado arriba. El VI debe utilizar un Ciclo Mientras (While Loop) con 100 milisegundos de retraso para generar continuamente formas de onda seno y cuadradas y mostrarlas en un gráfico de forma de onda. Utilice el VI expreso Simulate Signal Express VI de la paleta **Functions» Input** para generar las señales. La entrada de la frecuencia para cada función es elegida por el usuario.

Cambie los colores, los artículos visibles, y los estilos de diagramado del gráfico. Experimente con algunas de las opciones de cursor y de zoom disponibles.

## **Sección VII – Cadenas de Caracteres, Clusters, & Manejo de Errores**

- **Cadenas de caracteres**
- **Creando Clusters**
- **Funciones de los Clusters**
- **I/O Error**

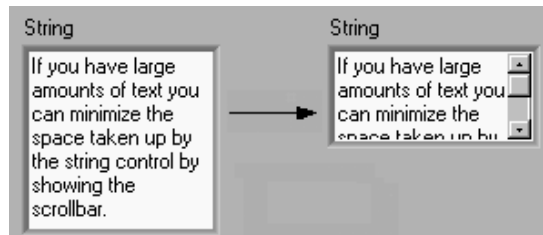
[ni.com](http://ni.com)



[Haga click para adicionar notas.](#)

## Cadenas de caracteres

- Una cadena es una secuencia de caracteres despleables o no despleables (ASCII)
- Tienen muchos usos – despliegue de mensajes, Control de instrumentos, Archivos de entrada y Salida (I/O)
- El controlador/indicador de cadenas se encuentra en **Controls »Text Control o Text Indicator**



ni.com

NATIONAL  
INSTRUMENTS

Una cadena es una secuencia de caracteres despleables o no despleables (ASCII). Las Cadenas se usan frecuentemente para enviar comandos a instrumentos, para proveer información acerca de pruebas (tales como el nombre del operador y la fecha), o para desplegar resultados a el usuario.

Los Controles e indicadores de Cadena se encuentran en la subpaleta **Text Control** o **Text Indicator** de la paleta de **Controles**.

- Introduzca o cambie texto utilizando la herramienta de texto (Text Tool) y presionado con el mouse el control de cadena (String Control).
- Las Cadenas pueden variar en su tamaño.
- Las Cadenas de Indicadores y Controles pueden tener barras de movimiento (Scroll Bars): Haga clic con el botón derecho del mouse y seleccione **Visible Items»Scrollbar**. La barra de movimiento no se activara si el Control o el Indicador no son lo suficientemente grandes.

## Clusters

- Estructura de Datos que agrupa datos juntos.
- Los Datos pueden ser de diferentes tipos.
- Semejante a una estructura (struct) en C++
- Los elementos deben de ser ya sea, todos Controles o todos Indicadores
- Se puede considerar como un conjunto de alambres agrupados en un cable.



ni.com

NATIONAL  
INSTRUMENTS

Los Clusters, agrupan componentes iguales o desiguales juntos.

Es equivalente a un Record en Pascal o una Struct en C.

Los Componentes de los Clusters pueden ser de diferentes tipos de datos.

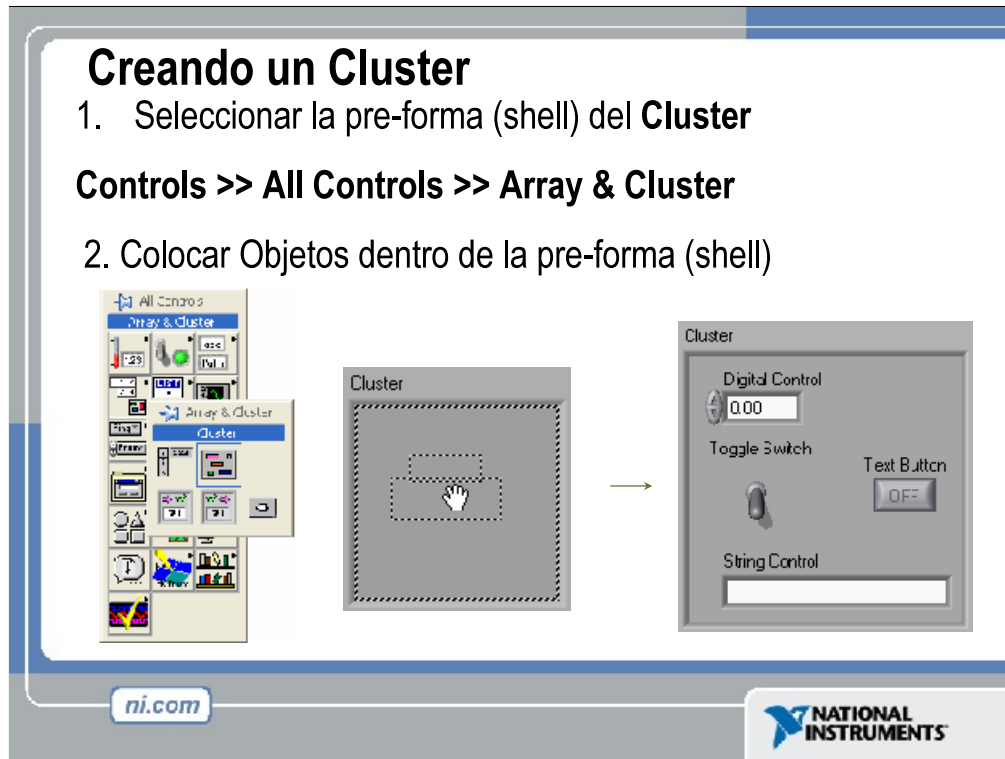
Ejemplos:

Información de Error—Agrupando una señal de error Booleano, un código de error numérico, y una cadena de origen de un error para especificar el error exacto.

Información de Usuario—Agrupando una cadena indicando el nombre del usuario y su número de identificación (ID) especificando el código de seguridad.

Todos los elementos de un Cluster deben ser ya sea Indicadores o Controles. No se puede tener un Control de Cadena y un Indicador Booleano.

Los Clusters se pueden considerar como un grupo de alambres individuales (datos) juntos en un cable (cluster).



Demostrar como crear un cluster en el panel frontal escogiendo **Cluster** desde la paleta **Controls » All Controls » Array & Cluster**.

- Esta opción le brinda una pre-forma (shell) parecida a la pre-forma obtenida al crear arreglos.
- Uno puede cambiarle el tamaño a la pre-forma una vez esta se ha seleccionado.
- Haga clic derecho adentro de la pre-forma para añadir objetos de cualquier tipo.

***Nota: Usted inclusive puede tener un cluster adentro de un cluster.***

El cluster se convierte en un Indicador o Control basado en el primer objeto que fue ingresado dentro del Grupo.

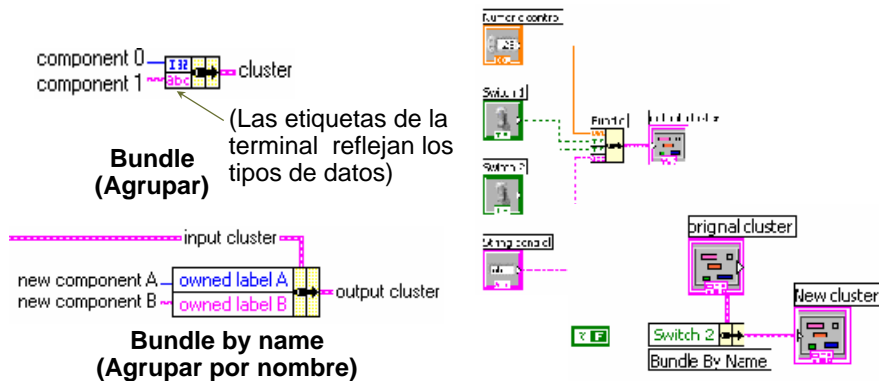
Usted también puede crear una constante de cluster en el diagrama de bloque escogiendo **Cluster Constant** de la paleta **Cluster**.

- Esto le brinda una pre-forma (shell) de cluster vacía.
- Usted puede darle cualquier tamaño al cluster una vez ingresado.
- Puede Introducir otras constantes dentro de la pre-forma (Shell).

***Nota: Usted no puede colocar terminales para los objetos del panel frontal en una constante de cluster en el diagrama de bloques, ni tampoco puede colocar constantes especiales como el Tab o una Cadena de caracteres vacías dentro de una pre-forma de cluster vacía.***

## Funciones del Cluster

- Están ubicadas en la subpaleta de **Cluster** de la paleta **Functions>>All functions**.
- También puede ser accesada haciendo clic con el botón derecho del mouse en la terminal del Grupo.



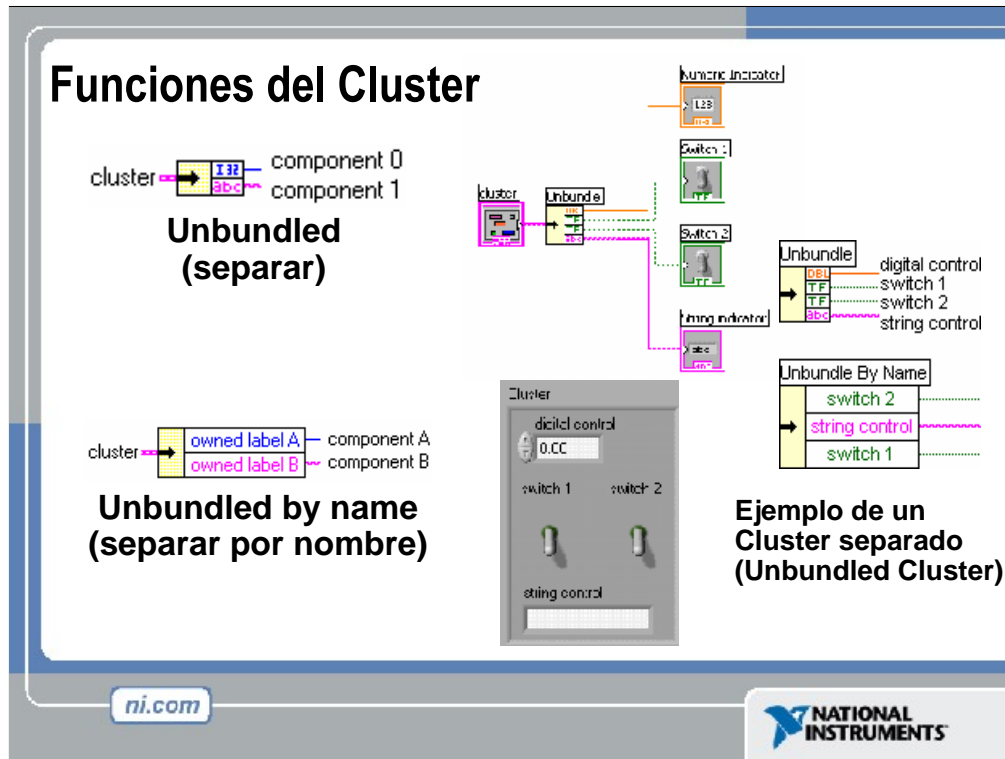
ni.com

NATIONAL  
INSTRUMENTS

**Función de Bundle (agrupación)**—Forma un cluster conteniendo los objetos dados (explique el ejemplo).

**Función Bundle by Name (agrupe por nombre)** —Actualiza valores de objetos localizados en grupos específicos (los objetos deben tener su propia clasificación).

***Nota:** Usted debe de tener un cluster existente cableado a la terminal de en medio de la función para poder utilizar Bundle by Name.*



Función **Unbundle (desagrupar)**—Se utiliza para acceder a todos los objetos del cluster.

Función **Unbundle by Name (desagrupar por nombre)**—Se utiliza para acceder objetos específicos (uno o mas) en el cluster.

*Nota: Solamente los objetos en el cluster que tengan su propia marca (Label) pueden ser accesados. Cuando se desagrupe por nombre, haga clic en la terminal con la herramienta de operación para escoger el elemento que quiere acceder.*

La función **Unbundle** (desagrupar) debe tener exactamente el mismo numero de terminales como se tienen elementos en el cluster. Añadiendo o Removiendo elementos en el cluster rompe cables en el diagrama.

Usted también puede obtener las funciones de **Bundle** (agrupar), **Unbundle** (desagrupar), **Bundle by Name** (agrupar por nombre), y **Unbundle by Name** (Desagrupar por Nombre) al hacer clic derecho con el mouse en la terminal del cluster en el diagrama de bloques y eligiendo **Cluster Tools** (herramientas de cluster) del Menú. Cuando usted escoge **Cluster Tools** las funciones **Bundle** (agrupar) y **Unbundle** (desagrupar) automáticamente contienen el numero correcto de terminales. Las funciones **Bundle by Name** (agrupar por nombre) y **Unbundle by Name** (desagrupar por nombre) aparecen con el primer elemento del Cluster.



## Errores de Cluster

- Los Errores de Cluster contienen la siguiente información:
  - **Booleano** (Boolean) Para reportar si surgió algún error
  - **Enteros** (Integer) Para reportar errores específicos de códigos
  - **Cadena** (String) Para brindar información acerca del error

The screenshot shows a LabVIEW cluster control titled "error in (no error)". It contains three fields: "status" with a green checkmark icon, "code" with a numeric value of 0, and "source" with an empty text box. The control has a standard LabVIEW look with a grey border and a small "OK" button at the bottom right.

The screenshot shows a LabVIEW cluster control titled "error out". It contains three fields: "status" with a green checkmark icon, "code" with a numeric value of 0, and "source" with an empty text box. The control has a standard LabVIEW look with a grey border and a small "OK" button at the bottom right.

ni.com

NATIONAL  
INSTRUMENTS

Los clusters para errores son un medio poderoso para manejar errores.

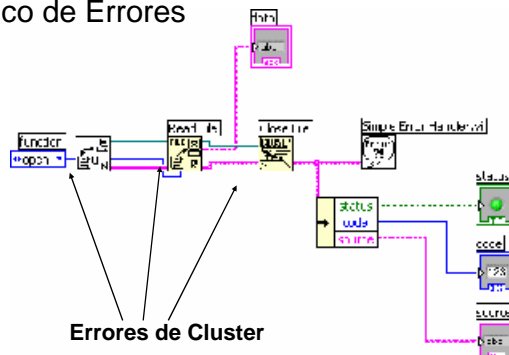
Los VI de adquisición de datos (DAQ VIs) las funciones de manejo de entradas y salidas de archivo (File I/O), los VI de networking, y muchos otros VI utilizan este método para pasar información acerca de errores entre nodos.

El cluster para errores contiene los siguientes elementos:

- **status**, Un Booleano al que se le asigna un valor verdadero si ocurre algún error.
- **code**, Un valor numerico al que se le asigna un numero de código correspondiente al error que ha ocurrido.
- **source**, Una cadena de caracteres que identifica el VI en donde ocurrió el error.

## Técnicas de Manejo de Errores

- La información de un Error es transmitida de un subVI al siguiente
- Si un Error ocurre en un subVI, todos los subVI subsecuentes no serán ejecutados de la manera usual
- Los errores de Cluster contienen todas las condiciones del Error
- Manejo automático de Errores



Errores de Cluster

ni.com

NATIONAL  
INSTRUMENTS

Los clusters para errores son útiles para determinar la ejecución de un subVI cuando se encuentra un error.

Note también que los clusters para errores pueden ser útiles para determinar el flujo del programa debido al paradigma de programación del flujo de datos. Esto puede ser útil especialmente al configurar un muestreo en mas de una tarjeta de adquisición de datos (DAQ board) simultáneamente.

La función de **Unbundle by Name** (desagrupar por nombre) muestra los componentes de un cluster de error.

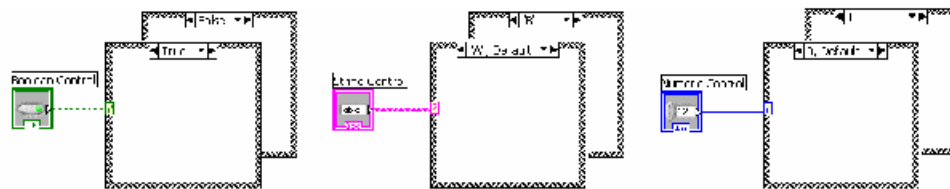
## Sección VIII – Estructuras de Caso y Secuenciales, Nodos de Formula

[ni.com](http://ni.com)



## Estructuras Caso

- En la subpaleta de estructura de las paleta de funciones.
- Encerrar los nodos o arrastrarlos hacia adentro de la estructura.
- Colocados como un juego de cartas (naipes), solo un caso es visible a la vez.



ni.com

NATIONAL  
INSTRUMENTS

Las estructuras de caso nos permiten elegir un curso de acción dependiendo del valor de entrada.

En la subpaleta de **Execution Control** (control de ejecución) de la paleta de **Functions** (funciones).

Es semejante a los enunciados if-then-else (si-entonces-de otro modo) en otros lenguajes.

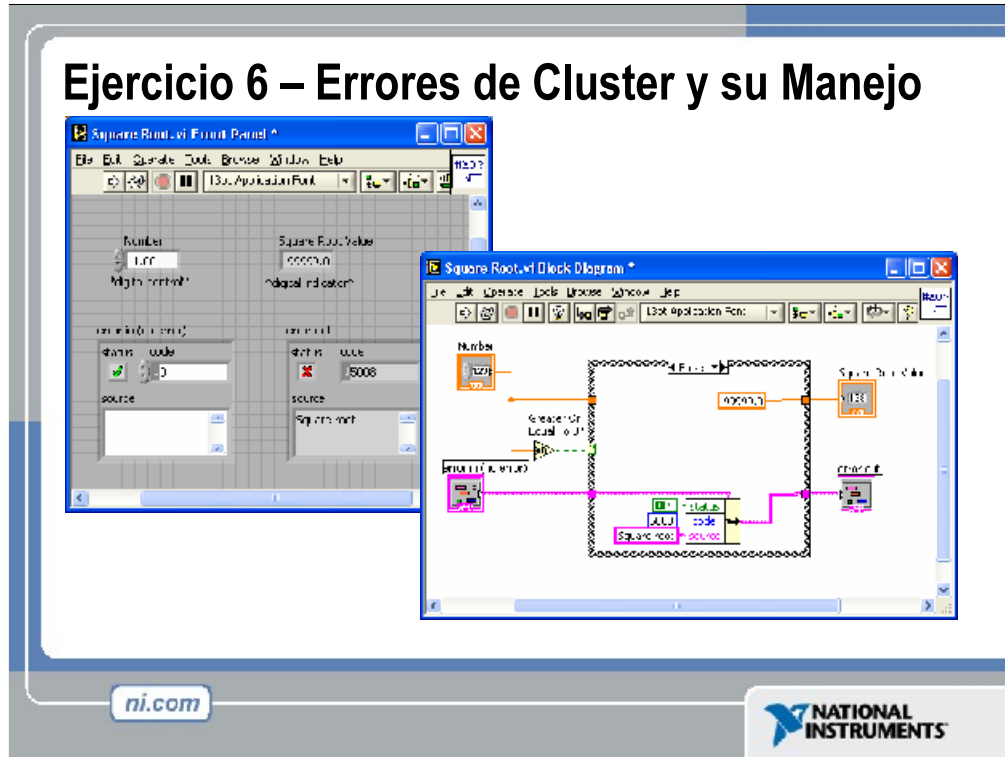
Como una baraja de naipes (cartas). Solo se puede ver un caso a la vez.

- Ejemplo 1: Entrada Booleana: Caso simple if-then (si-entonces). Si el valor booleano de la entrada es verdadero (TRUE), el caso verdadero se ejecutará; de otra manera el caso falso (FALSE) se ejecutará.

- Ejemplo 2: Valor numérico de entrada. El valor de entrada determina cual caso (caja) se ejecutará. Si esta fuera de rango de los casos LabVIEW escogerá el caso por omisión (DEFAULT).

- Ejemplo 3: Entrada de cadena de caracteres. Como el caso de los valores numéricos de entrada, el valor de la cadena de caracteres determina cual caso (caja) se ejecutará. Haga hincapié en que el valor debe ajustarse exactamente al de los casos o sino la estructura ejecutará el caso por omisión.

## Ejercicio 6 – Errores de Cluster y su Manejo



Tiempo estimado de completación: 20 minutos.

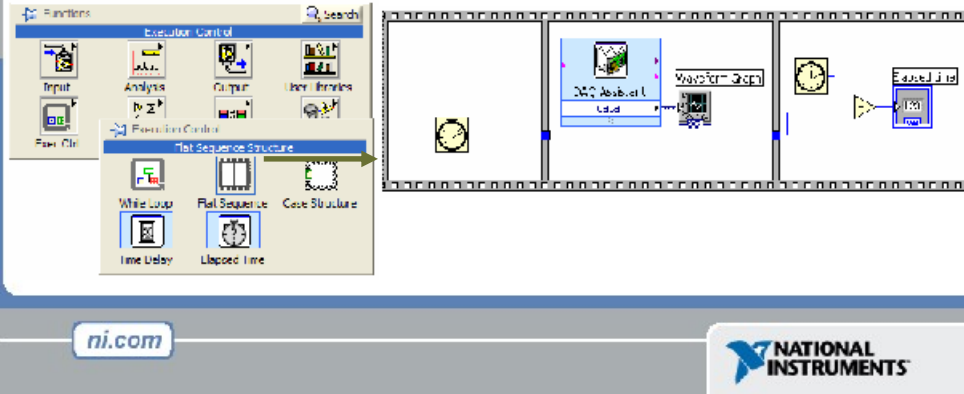
Cree un VI que calcule la raíz cuadrada de un numero. Si el numero es mayor o igual a cero, el VI debe devolver el valor de la raíz cuadrada y no generar ningún error. Si el numero es menor que cero el programa debe devolver un resultado con valor de -9999.90 e insertar un error dentro del cluster para errores.

Utilice un **Case Structure** (estructura de casos) desde la función **Greater or Equal To 0** (mayor o igual a cero) de la paleta numérica para determinar si el VI va a calcular la raíz cuadrada o a generara un error. El Caso “Falso” mostrado arriba es el caso de error. Utilice una función **Bundle by Name** de la paleta de cluster para insertar una constante Booleana, Numerica, y de Cadena de Carateres en los objetos de estado (status), codigo (code) y origen (source) respectivamente. Los valores de las constantes deberan ser Verdaderos (True), 5008( un codigo de error definido arbitrariamente), y “Raiz Cuadrada.vi” ( El nombre del programa que genero el error ). Cablee el nuevo cluster en el indicador de Error de Salida (Error Out), y un valor constante de -9999.90 al indicador de la raíz cuadrada. En el caso “Verdadero”, que no se esta mostrando, deberia de cablearse simplemente el control de error de entrada directamente al indicador de error de salida a traves del caso. El valor de entrada de la raíz cuadrada debe ir cableado hacia la funcion **Raiz Cuadrada (Square Root)** de la paleta numerica, y el resultado deberia ser cableado afuera del caso hacia el indicador de la raíz cuadrada.

Indique que este VI puede ser fácilmente configurado como un subVI para un código de programación mas grande, y que la búsqueda de problemas (troubleshooting) y depuración es mas fácil cuando los clusters manejan errores con

## Estructuras de Secuencia

- En la subpaleta de control de ejecución (**Execution Control**) de la paleta de funciones.
- Ejecuta diagramas secuencialmente.
- Presionar botón derecho del mouse para agregar un nuevo cuadro (frame).



En un lenguaje basado en texto, las declaraciones del programa se ejecutan en el orden en que aparecen. En un flujo de datos, un nodo se ejecuta cuando los datos están disponibles en todas sus terminales de entrada.

Algunas veces es difícil determinar el orden exacto de ejecución. A menudo, ciertos eventos deben suceder antes que otros. Cuando usted necesite controlar el orden de la ejecución del código en su diagrama de bloques, usted puede utilizar una **Sequence Structure** (estructura de secuencia).

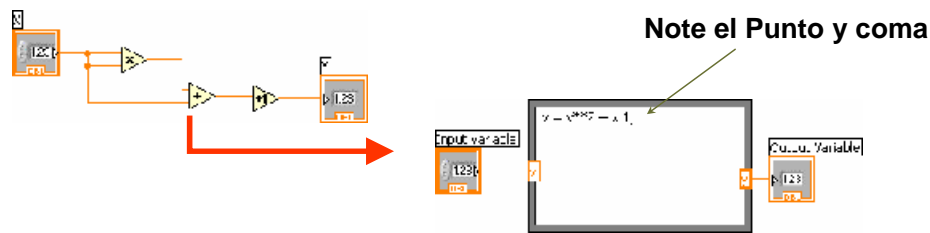
**Estructura de Secuencia:** Usada para controlar el orden en que los nodos en un diagrama se ejecutaran.

En la paleta de **Execution Control** (control de ejecución).

- Luce como un pedazo de rollo de película.
- Utilizada para ejecutar diagramas secuencialmente.
- Hacer clic derecho con el mouse en los bordes para crear un nuevo cuadro.

## Nodos de Formula

- En la subpaleta de Estructuras (Structures).
- Implementa ecuaciones complicadas.
- Variables creadas en los márgenes
- Nombres de variables son sensibles (mayúsculas/minúsculas)
- Cada declaración debe de terminar con un punto y coma(;) )
- La ventana de ayuda muestra las funciones disponibles.



Algunas veces es preferible programar expresiones matemáticas con funciones basadas en texto, en lugar de hacerlo con iconos (los que pueden tomar mucho espacio en el diagrama).

**Nodo de Formula:** Nos permite implementar ecuaciones complicadas usando instrucciones basadas en texto.

- Localizada en la subpaleta de **Structures** (estructuras).
- Las cajas pueden cambiar de tamaño para introducir formulas algebraicas directamente a los diagramas de bloque.
- Para agregar variables, se presiona el botón derecho del mouse y se escoge la opción de agregar entrada (**Add Input**) o agregar salida (**Add Output**). Hay que nombrar las variables de la misma manera en que serán usadas en la formula. (Los nombres son sensibles a capitalización)
- Las declaraciones deben terminar con un punto y coma (;)
- Al usar varias formulas en un solo nodo de formula, cada variable asignada (las que aparecen en la parte izquierda de cada formula) deben de tener una terminal de salida en el nodo de formula. Sin embargo, estas terminales de salida no necesitan ser cableadas.

Compare los ejemplos en la diapositiva.

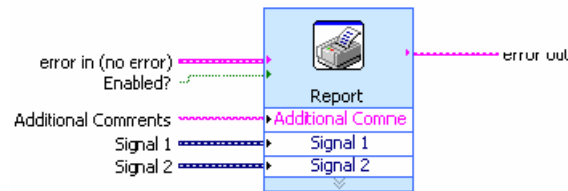
## Sección IX – Documentacion e Impresion

- Imprimir desde el Menú de Archivo (File Menu) hacia: la Impresora, HTML, Archivos de Texto.
- Imprimir programáticamente Graficas o Imágenes del Panel Frontal
- Documentar los VIs en *VI Properties » Documentation Dialog*
- Agregar Comentarios usando etiquetas libres dentro del panel frontal y diagramas de bloques.



## Impresión

- **File » Print...** Da muchas opciones de impresion
  - Seleccione la Impresión del Icono, el Panel Frotal, el Diagrama de Bloques, La Jerarquía del VI, SubVIs Incluidos, Historia del VI
- **Print Panel.vi** (Programaticamente imprime un Panel Frontal)
  - **Functions » All Functions » Application Control**
- **Generar e Imprimir Reportes (Functions » Output » Report)**



ni.com

NATIONAL  
INSTRUMENTS

LabVIEW ofrece muchas opciones para imprimir los VIs. Desde la forma estándar **File » Print...** menu, el usuario puede imprimir una copia de su VI, o puede generar una copia para guardarla en un archivo o publicarla.

Utilizando el VI Print Panel (Impresión de Panel) en LabVIEW le permite al usuario imprimir programáticamente los resultados de una prueba. Los VIs también pueden ser configurados para imprimir automáticamente inmediatamente después de cada ejecución. Esta opción es colocada en **VI Properties » Print Options**.

Para aplicaciones mas avanzadas, LabVIEW tiene herramientas que generan reportes que le permiten al usuario crear reportes personalizados para aplicación individuales.

LabVIEW 7.0 incluye un VI Express llamado Report. Este VI genera un reporte preformado que contiene la documentación del VI, datos que el VI regresa y reportes de propiedades, tales como el autor, la compañía y el numero de paginas.

## Documentando VIs

- VI Properties » Documentation
  - Provee una Descripción e Información de Ayuda para el VI
- VI Properties » Revision History
  - Monitorea cambios entre diferentes Versiones de un VI
- Individual Controls » Description and Tip...
  - Haga clic con el botón derecho del mouse para proveer una descripción y ayuda (Description and Tip Strip)
- Utilice la herramienta de etiquetado (Labeling Tool) para documentar paneles frontales y diagramas de bloques

ni.com

NATIONAL INSTRUMENTS

Al abrir VI's properties (propiedades del VI) un desarrollador puede añadir documentación a su VI. La documentación colocada en el Description field (espacio para la descripción) de la ventana de documentación del VI es mostrada en Context Help (ayuda en contexto), y se imprime con el VI.

Si un usuario tiene una aplicación que esta siendo desarrollada continuamente, el o ella puede seguir los cambios en el desarrollo con Revision History (historial de revisiones), también encontrado en VI Properties.

Cualquier control o indicador en el Panel Frontal puede ser documentado al hacer clic derecho con el mouse y elegir Description and Tip... (descripción y sugerencia...) La información de descripción aparece en el menu Context Help (ayuda en contexto) cuando un usuario pasa el cursor sobre el objeto, y la información de sugerencia es mostrada en una pequeña ventana pegada al cursor cuando el usuario pausa el movimiento del cursor sobre el objeto.

Al igual que los comentarios en un lenguaje basado en texto, el desarrollador podría querer explicar una porción de su código, o proveer direcciones en el panel de frontal. Cualquiera de estas necesidades puede ser cumplida al usar la herramienta de etiquetado (labeling tool) para crear una caja de texto libre con tantas instrucciones o explicaciones como sean necesarias.

## Sección X – Arquitectura Básica de Programación

- Arquitectura VI de un simple
- Arquitectura de un VI General
- Arquitectura de maquina de estado

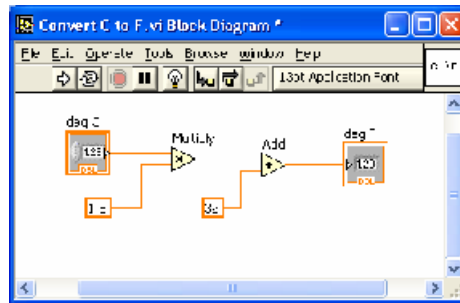
ni.com

NATIONAL  
INSTRUMENTS

Usted puede estructurar los VIs dependiendo de la funcionalidad que desea que tengan. En esta sección discutiremos algunas de las arquitecturas de VI mas comunes, a si como sus ventajas/desventajas: *simple*, *general*, *maquina de estado* (*state machine*).

## Arquitectura de un VI simple

- VI funcional que produce resultados al ser ejecutado
  - No tiene opciones para “comienzo” o “fin”
  - Adecuado: para pruebas de laboratorio, cálculos
- Ejemplo: Convertir C a F.vi



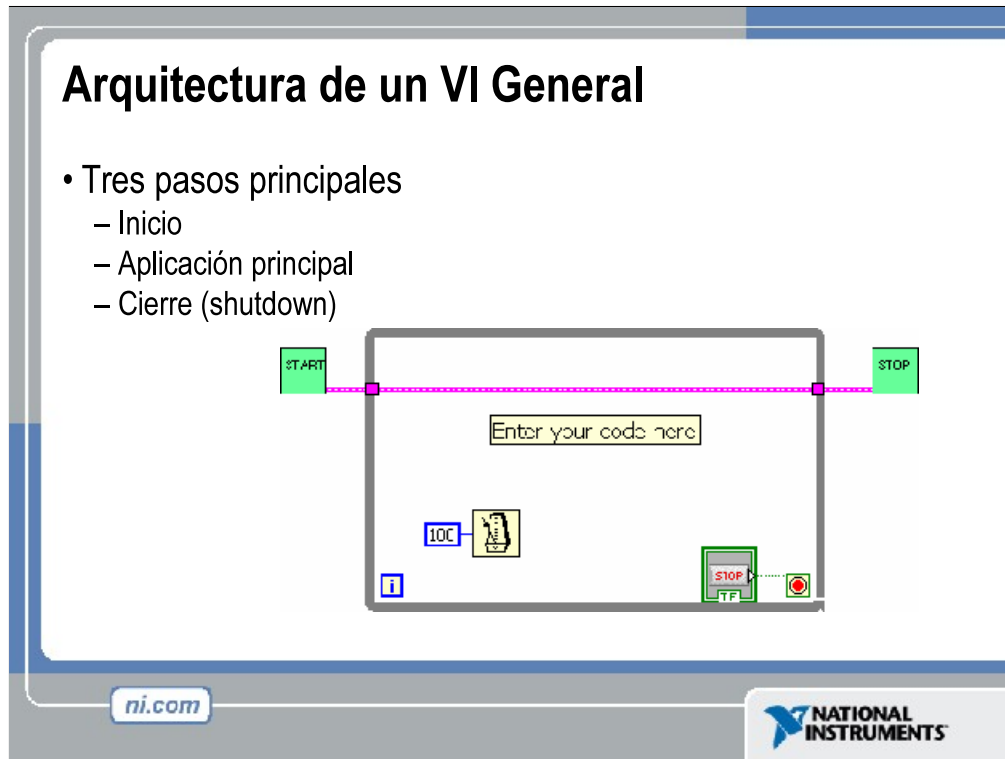
ni.com

NATIONAL  
INSTRUMENTS

### Arquitectura de VI Simple

Cuando se hacen pruebas de laboratorio rápidas, no se necesita una arquitectura complicada. Su programa puede consistir de un solo VI que toma una medición, hace cálculos, y ya sea despliega los resultados o los guarda en el disco. La medición puede ser iniciada cuando el usuario le da clic a la flecha de inicio (run arrow).

Aparte de ser usada para aplicaciones simples, esta arquitectura es usada para componentes “funcionales” dentro de aplicaciones mas grandes. Usted puede convertir estos VIs simples en subVIs que son utilizados como componentes de una aplicación mas grande.



### Arquitectura de un VI General

Generalmente al diseñar una aplicación se siguen tres pasos:

- *Inicio (startup)*—Esta area se usa para inicializar el hardware, leer información de configuración de archivos o preguntar al usuario la localización de datos de archivos.
- *Aplicación Principal (main application)*—Generalmente consiste de por lo menos un ciclo que se repite hasta que el usuario decida salir del programa, o el programa termina debido a otras razones como la completacion de una entrada/salida (I/O)
- *Apagado (shutdown)*—Esta seccion usualmente se encarga de cerrar los archivos, escribir datos de configuracion al disco o reestablecer los datos de entrada/salida (I/O) a su estado predeterminado.

Para aplicaciones simples, el ciclo de la aplicación principal puede ser bastante directo. Cuado se tiene interfaces de usuario complicadas o eventos múltiples (acción de usuario, accionamientos de entradas/salidas, etc.), esta sección puede volverse mas complicada.

## Arquitectura de Maquina de Estado

- Ventajas
  - Puede ir de cualquier estado a otro
  - Fácil de modificar y depurar
- Desventajas
  - Se pierden eventos si dos ocurren al mismo tiempo

### Pasos:

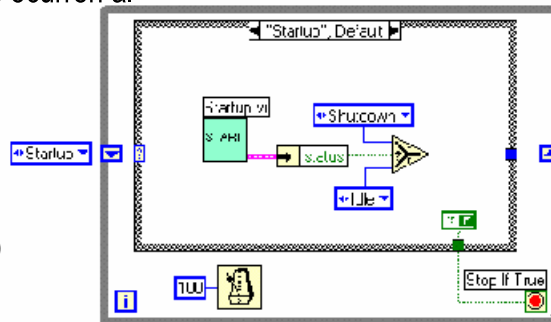
0: Inicio (Startup)

1: Ocio (Idle)

2: Evento 1

3: Evento 2

4: Cierre (Shutdown)



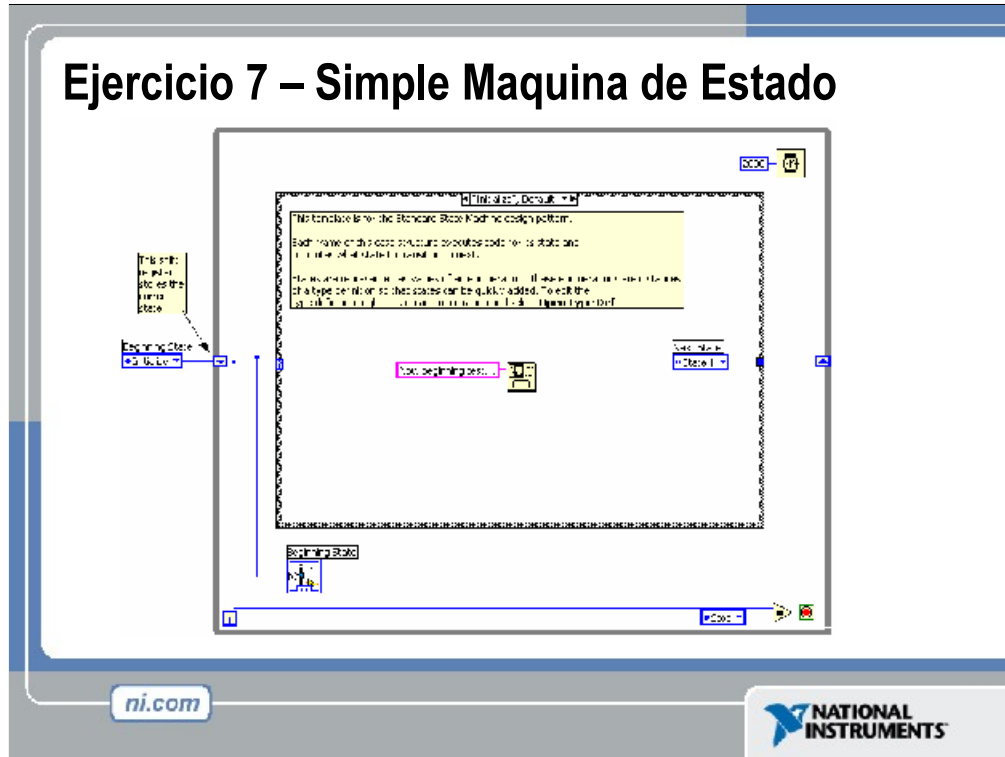
### Arquitectura de un VI de Maquina de Estado

Usted puede hacer sus diagramas mas compactos con el uso de una estructura de caso simple para manejar todos sus eventos. En este modelo usted escanea (scan) la lista de posibles eventos, o estados, y luego correlaciona (map) estos a un caso. Para el VI mostrado arriba, los estados posibles son *inicio (startup)*, *ocio (idle)*, *evento 1*, *evento2* y *apagado (shutdown)*. Estos estados se encuentran guardados en una constante enumerada. Cada estado tiene su propio caso en el lugar que usted coloque los nodos apropiados. Mientras se ejecuta un caso, el siguiente caso es determinado por el resultado actual. El siguiente caso que se ejecutara es guardado en el registro de desplazamiento (shift register). Si ocurre un error en cualquier estado, el caso de apagado (shutdown) el llamado.

La ventaja de este modelo es que su diagrama se puede reducirse significativamente (izquierda a derecha), haciendo que sea mas fácil de leer y depurar. Una desventaja de la estructura de secuencia es que no puede saltarse o salirse de un caso. Este método resuelve ese problema ya que cada caso determina cual será el próximo caso que se ejecutara.

Una desventaja de esta técnica es que con la arquitectura simple se pueden perder eventos. Si dos eventos ocurren al mismo tiempo, este modelo maneja solo el primero, y el segundo se pierde. Esto puede conducir a errores que son difíciles de depurar porque solo ocurren ocasionalmente. Existen versiones mas complejas de la Arquitectura del VI de la Maquina de Estado que contiene código extra para construir un queue de eventos (estados) de manera que usted no pierda ningún evento.

## Ejercicio 7 – Simple Maquina de Estado



Tiempo estimado de duración: 25 minutos.

En este ejercicio crearemos un VI usando la arquitectura de Maquina de Estado. El VI tendrá un estado de ocio (idle), en donde espera por entradas. Cuando el usuario presiona un botón, el VI se ira al estado 1 (State 1). El estado 1 genera una caja de dialogo que le permite al usuario proceder a ir al estado 2 (State 2) o volver a empezar. Desde el estado 2 el usuario puede decidir cerrar el programa o iniciar nuevamente.

En lugar de empezar desde cero, utilizaremos una Plantilla de VI para crear nuestra maquina de estado. Desde el File menu, en lugar de seleccionar **New VI**, seleccione **New...**, y haga clic en Start from template. Busque ...\\Program Files\\National Instruments\\LabVIEW 6.1\\templates\\State Machine.vit, Y abralo. Esto abrira una plantilla simple de maquina de estado utilizando cadenas de caracteres para controlar el estado. Examine la plantilla, y guardela en otro directorio antes de empezar a trabajar en ella.

El primer estado que haremos es el de inicialización. Note que este estado ya ha sido creado para nosotros, y es el estado predeterminado (default). En el caso de inicialización, coloque un **One Button Dialog** (Functions » Time & Dialog), y cablee “Initialized. Clic OK para proceder.” en la terminal de mensajes. Reemplace el texto “Next State” con “Idle”.

Haga clic derecho en la estructura de casos, escoja Add Case After, y déle el nombre de “Idle”. Dentro del caso Idle, coloque una segunda estructura de caso. Cablee un Booleano llamado “Begin Test” a la terminal de selección de la estructura de caso anidada, y cablee una constante de cadena de caracteres con el valor “Test” de el caso verdadero a el registro de desplazamiento (Shift register). Esta configuracion mantendra la maquina de estados en el estado Idle hasta que el usuario proceda a presionar el boton para iniciar la prueba. Haga clic derecho en el tunel de salida del Booleano y escoja Create » Constant. Cambie el valor constante a True (verdadero).

Haga clic derecho en la estructura principal y adicione otro caso. Llame a este caso “Test”. En el caso Test, coloque un **Two Button Dialog**, un boton doble de dialogo, (Functions » Time & Dialog). Conecte el Texto “The test has run. Click OK to quit, Cancel to test again.” con la terminal de mensaje. Coloque otra estructura de caso adentro del caso Test, y conecte “T button?” la salida de el Two Button Dialog con la terminal seleccionada. Haga una conexión del valor “Idle” desde el caso falso hasta el registro de desplazamiento (Shift register), y una conexion con constante Booleana Verdadera con la terminal del ciclo mientras (while loop). El caso Verdadero debera de contener una cadena de caracteres constante vacia, y un valor Booleano Falso conectado con la condicion de terminacion. Esto terminara el programa si el usuario presiona OK, y regresara al estado idle (ocio) si el usuario presiona Cancel.



## Sección XI – Panel Frontal Remoto

- Observar y Controlar Paneles Frontales de LabVIEW desde un Navegador en la Web
- No Requiere Programación
- Usuarios en lugares Remotos pueden ver en Tiempo Real las Actualizaciones del Panel Frontal
- Múltiples clientes pueden Observar el Panel Frontal Simultáneamente
- Solo un Cliente Puede Controlar el Panel Frontal a la vez

ni.com



En LabVIEW, una opción llamada panel frontal remoto le permite al usuario observar y controlar el panel frontal de un VI remotamente desde LabVIEW o un navegador sin necesidad de programación. El usuario puede correr una aplicación o revisar el progreso desde cualquier lugar. Además diferentes usuarios en diferentes lugares pueden tener acceso simultaneo al VI.

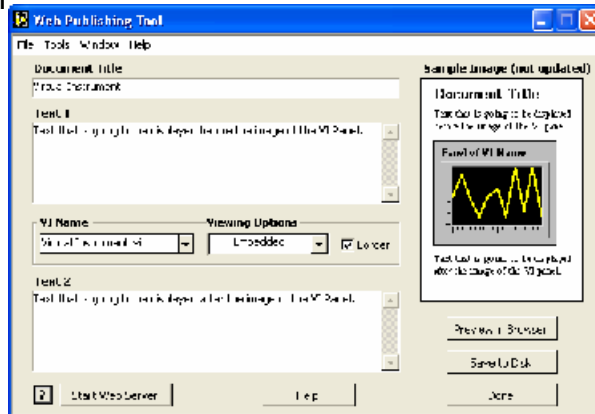
Los usuarios remotos pueden observar el panel frontal en “vivo” (en tiempo real), en lugar de imágenes estáticas. Pueden existir múltiples clientes observando el mismo panel frontal. Sin embargo, solo un cliente puede controlar el panel frontal a la vez.

## Herramienta para la Publicación de un Panel Remoto

- **Tools » Web Publishing Tool...**

- Clic Salvar a Disco y el VI es embebido a un archivo HTML

- Después que el archivo a sido grabado, puede ser reabierto y personalizado en cualquier editor HTML

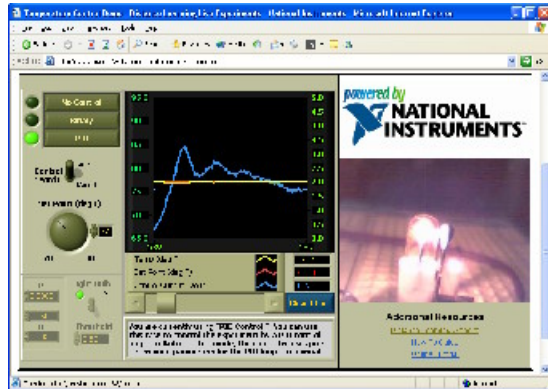


Para incrustar su VI en una pagina web, simplemente abra la herramienta de publicación web (Web Publishing Tool) en herramientas del menú. Web Publishing Tool crea un simple archivo html con un panel frontal incrustado en el. Con esta herramienta usted puede crear un titulo y colocar texto antes y después de colocar el panel frontal. Una vez creado el archivo html, se puede observar en un navegador web y ser grabado al disco. Salvar el archivo coloca el html en su directorio www de LabVIEW, que es el predeterminado para el servidor web de LabVIEW. Usted puede iniciar su servidor web desde Web Publishing Tool o desde Tools » Options... » Web Server : Configuration. (Nota: en Tools » Options... menu usted tambien puede configurar el derecho de acceso y VIs expuestos para su servidor web.

Si usted desea personalizar su pagina web mas allá de lo que provee la herramienta de publicación web (Web Publishing Tool), simplemente abra el archivo html en otro editor de html, y personalice alrededor del código del panel embebido para LabVIEW.

## Panel Frontal Remoto - Recursos

- NI Developer Zone  
([zone.ni.com](http://zone.ni.com))
  - Busque Remote Front Panel
  - Tutoriales e Instrucciones están disponibles para descargar (download).
  - Información para poder Incorporar una cámara Web en aplicaciones con panel remoto.



[ni.com](http://ni.com)

NATIONAL  
INSTRUMENTS

Para mas información en cuanto a paneles frontales remotos, visite NI Developer Zone (zona de desarrollo de NI), [zone.ni.com](http://zone.ni.com).

Remote Panels in LabVIEW -- Distributed Application Development (Panel Remoto en LabVIEW – Desarrollo de Aplicaciones Distribuidas)

<http://zone.ni.com/devzone/insights.nsf/2d17d611efb58b22862567a9006ffe76/8bc90a5f744a482886256ba000681713?OpenDocument>

Developing Remote Front Panel LabVIEW Applications (Desarrollando aplicaciones de panel frontal remoto en LabVIEW)

<http://zone.ni.com/devzone/conceptd.nsf/2d17d611efb58b22862567a9006ffe76/e789515b9976253786256b1f007e039b?OpenDocument>

## Sección XII – Temas Adicionales

- Nodo de Propiedad
- Variables Locales
- Variables Globales
- DataSocket
- Archivos Binarios I/O

ni.com




LabVIEW tiene muchas otras herramientas disponibles para su desarrollo. Si bien es cierto que las restricciones de tiempo no nos permiten entrar en muchos detalles, es una buena idea indicar algunas de las capacidades avanzadas ofrecidas por LabVIEW. Los estudiantes que están interesados en temas específicos pueden ir a LabVIEW Example Finder o pueden visitar la zona de desarrollo NI (NI's Developer Zone) para mas información.

- Property Nodes (nodos de propiedad) – Las propiedades son todas las cualidades de un objeto en el panel frontal. Con *property nodes*, se crean o leen dichas características tales como color de foreground y background, formato y precision de datos, visibilidad, texto descriptivo, tamaño y posicionamiento del panel frontal, etc. Busque property node en la ventana de Find Examples, alli se encuentran 24 programas con este tema que vienen en LabVIEW.
- Local Variables (variables locales) – Las variables locales rompen con el paradigma de programacion de flujo de datos, permitiendo que los datos sean pasados sin necesidad de cables.
- Global Variables (variable global) – Un tipo especial de VI, usado para guardar datos en la objetos del panel frontal con el proposito de intercambiar datos entre VIs.
- DataSocket – Una plataforma independiente de intercambio de datos entre computadoras y aplicaciones, basada en TCP-IP, pero sin la complejidad de la programacion en bajos niveles.
- Archivos Binarios I/O—permite al usuario transferir datos a un disco en aplicaciones de alta velocidad.

## Qué hago Ahora?

- Ejemplos de Programas (Help» Find Examples...)
- LabVIEW Edición de Estudiantes  
([www.ni.com/labviewse](http://www.ni.com/labviewse))
- Recursos Web ([ni.com](http://ni.com))
  - Zona de Desarrollo NI ([zone.ni.com](http://zone.ni.com))
  - Notas de Aplicaciones
  - grupo de noticias labview ([www.info-labview.org/](http://www.info-labview.org/))
  - Instrument Driver Library ([www.ni.com/idnet](http://www.ni.com/idnet))

[ni.com](http://ni.com)

 NATIONAL INSTRUMENTS

### Que hago ahora?

- National Instruments ofrece una amplio rango de cursos de enseñanza para expandir tu conocimiento. Por favor visite [ni.com/academic](http://ni.com/academic) para obtener información de programas y recursos disponibles.
- LabVIEW Edición de Estudiantes esta disponible en nuestra pagina Web. Incluye *aprendiendo con LabVIEW*, un libro de texto escrito por el Dr. Bob Bishop de la universidad de Austin Texas.
  - La web es el mejor lugar para buscar información sobre LabVIEW. NI.com ha sido diseñado para ser el recurso principal para encontrar información.
  - La zona de desarrollo NI (“NIDZ”) es un lugar para que los desarrolladores se reúnen, discutan temas de diseño y comenten sobre diversos temas.
  - Notas de aplicación pueden ser descargadas de [ni.com/support](http://ni.com/support) en una gran cantidad de temas.
  - Grupo de Noticias LabVIEW (Info-labview) es un grupo manejado por terceras personas.
  - Existe una extensa librería de drivers de instrumentos de LabVIEW disponibles para ser descargados de NIDZ