

TUTORIAL BÁSICO DE LabBIEW 6.1

EDWIN PAYARES FIGUEROA

CARLOS E. ROMERO ACOSTA

CORPORACIÓN UNIVERSITARIA TECNOLÓGICA DE BOLÍVAR

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

CARTAGENA DE INDIAS, D.T. Y C.

MAYO 2003

TUTORIAL BÁSICO DE LabVIEW 6.1

EDWIN PAYARES FIGUEROA

CARLOS E. ROMERO ACOSTA

**Monografía de grado presentada como requisito para optar al título
de Ingenieros Electricistas.**

**Director
JUAN A. CONTRERAS M.
Ingeniero Electricista**

CORPORACIÓN UNIVERSITARIA TECNOLÓGICA DE BOLÍVAR

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

CARTAGENA DE INDIAS, D.T. Y C.

MAYO 2003

Cartagena, 30 de mayo del 2003

Señores

COMITÉ DE PROYECTOS DE GRADO

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

CORPORACIÓN UNIVERSITARIA TECNOLÓGICA DE BOLÍVAR

La ciudad.

Distinguidos señores :

Por medio de la presente me permito comunicarles que la **monografía** titulada **"TUTORIAL BASICO DE LABVIEW 6.1 "** ha sido desarrollado de acuerdo con los objetivos establecidos.

Como director considero que el trabajo es satisfactorio y cumple con todos los requerimientos necesarios para ser presentado por sus autores para su evaluación. Por tal motivo hago entrega formal del proyecto.

Cordialmente,

Ing. JUAN A. CONTRERAS M.

Director del proyecto

Cartagena, 30 de mayo del 2003

Señores

COMITÉ DE PROYECTOS DE GRADO

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

CORPORACIÓN UNIVERSITARIA TECNOLÓGICA DE BOLÍVAR

La ciudad.

Distinguidos señores :

La presente tiene como objetivo presentar a su consideración para estudio y aprobación, la **monografía** titulada "**TUTORIAL BASICO DE LabVIEW 6.1**", como requisito para optar el título de Ingeniero Electricista.

Cordialmente,

Edwin payares F.

Cartagena, 30 de mayo del 2003

Señores

COMITÉ DE PROYECTOS DE GRADO

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

CORPORACIÓN UNIVERSITARIA TECNOLÓGICA DE BOLÍVAR

La ciudad.

Distinguidos señores :

La presente tiene como objetivo presentar a su consideración para estudio y aprobación, la **monografía** titulada "**TUTORIAL BASICO DE LABVIEW 6.1**", como requisito para optar el título de Ingeniero Electricista.

Cordialmente,

Carlos E Romero A.

Nota de aceptación

Presidente del Jurado

Jurado

Jurado

Ciudad y fecha (día, mes, año) : _____

CONTENIDO

	Pag.
INTRODUCCIÓN	
1. ¿ QUE ES LABVIEW?	17
2. INSTRUMENTOS VIRTUALES	18
3. EL AMBIENTE DE LabVIEW	19
3.1. MENU PRINCIPAL.	20
3.2. BARRA DE HERRAMIENTAS	21
3.3. PALETAS FLOTANTES	22
3.3.1. PALETA DE HERRAMIENTAS.	23
3.3.2. PALETA DE CONTROL.	25
3.3.3. PALETA DE FUNCIONES.	34
4. PANEL FRONTAL	46
5. DIAGRAMA DE BLOQUES	49
6. ICONO Y CRISTAL CONECTOR	51
7. LAZOS Y ESTRUCTURAS	56
8. COMO CONSTRUIR UN VI	59
9. ACTIVIDADES	64
9.1. ACTIVIDAD 1	65

9.2.	ACTIVIDAD 2	70
9.3.	ACTIVIDAD 3	72
9.4.	ACTIVIDAD 4	74
9.5.	ACTIVIDAD 5	77
9.6.	ACTIVIDAD 6	79
9.7.	ACTIVIDAD 7	82
9.8.	ACTIVIDAD 8	87
9.9.	ACTIVIDAD 9	94
	CONCLUSIONES	
	BIBLIOGRAFÍA	
	ANEXOS	

LISTA DE FIGURAS

	Pag.
Figura 1. Componentes de un VI	18
Figura 2. Menú y Barra de herramientas	19
Figura 3. Paletas flotantes	19
Figura 4. Menús desplegables	20
Figura 5. Obtención de las paletas flotantes	23
Figura 6. Obtención de la paleta de herramienta	24
Figura 7. Obtención de la paleta de controles	26
Figura 8. Numeric control	27
Figura 9. Boolean controls	27
Figura 10. String & path	28
Figura 11. Array & cluster	28
Figura 12. List & table	29
Figura 13. Graph	29
Figura 14. Ring & enum	30
Figura 15. I/O	30
Figura 16. Refrum	31
Figura 17. Dialog controls	31

Figura 18.	Classic controls	32
Figura 19.	Actives	32
Figura 20.	Decorations	33
Figura 21.	Obtención de las paletas de funciones	34
Figura 22.	Structure	35
Figura 23.	Numeric funcions	35
Figura 24.	Boolean funcions	36
Figura 25.	String	36
Figura 26.	Array	37
Figura 27.	Cluster	37
Figura 28.	Comparison	38
Figura 29.	Time & dialog	38
Figura 30.	File I/O	39
Figura 31.	Data adquisition	39
Figura 32.	Waveform	40
Figura 33.	Analyce	40
Figura 34.	Instrument I/o	41
Figura 35.	Motion & vision	41
Figura 36.	Mathematics	42
Figura 37.	Communication	42
Figura 38.	Application control	43

Figura 39.	Graphics & sound	43
Figura 40.	Tutoríal	44
Figura 41.	Report generation	44
Figura 42.	Advanced	45
Figura 43.	Ejemplos de panel frontal	46
Figura 44.	Elementos del panel frontal y sus equivalentes en el diagrama de bloques	47
Figura 45.	Ejemplo de diseño de un panel frontal	48
Figura 46.	Ejemplo de un diagrama de bloque	49
Figura 47.	Icono y cristal conector	52
Figura 48.	Menu desplegable del icono	53
Figura 49.	Editor del icono	53
Figura 50.	Ubicación del icono	55
Figura 51.	Structure	56
Figura 52.	Ventana del icono del LabView	59
Figura 53.	Ventana de LabView	60
Figura 54.	Despliegue de las ventanas a izquierda y derecha	60
Figura 55.	Ubicación de las paletas	61
Figura 56.	Organización de elementos	62
Figura 57.	Cableado	63
Figura 58.	Actividad 1. Panel frontal	66
Figura 59.	Actividad 1. Elementos del diagrama de bloques	66

Figura 60.	Organización del Diagrama A.1	67
Figura 61.	Cuadro de secuencia	68
Figura 62.	Conexión de secuencia Local	68
Figura 63.	Creación de la variable local	69
Figura 64.	Actividad 2. Panel Frontal	70
Figura 65.	Organización y cableado del Diagrama de Bloques A.2	71
Figura 66.	Actividad 3. Panel frontal	72
Figura 67.	Organización y cableado del diagrama de bloque A.3	73
Figura 68.	Actividad 4. Panel frontal	75
Figura 69.	Organización y cableado del diagrama de bloque A.4	76
Figura 70.	Actividad 5. Panel frontal	78
Figura 71.	Organización y cableado del Diagrama de Bloques A.5	79
Figura 72.	Actividad 6. Panel frontal	81
Figura 73.	Organización y cableado del diagrama de bloque. A.6	82
Figura 74	Actividad 7. Panel Frontal para envío	83
Figura 75	Organización y cableado del Diagrama de Bloques para envío	84
Figura 75 a.	Configuración del puerto serial	84
Figura 75 b.	Envío de datos	84
Figura 76	Actividad 7. Panel frontal para recepción	85
Figura 77	Organización y cableado del Diagrama de Bloques para recepción	86

Figura 77 a.	Configuración del puerto serial	86
Figura 77 b.	Recepción de datos	86
Figura 78.	Actividad 8. Panel frontal actuador	88
Figura 79	Organización y cableado del diagrama de bloque actuador	89-90
Figura 79 a.	Configuración del puerto serial	89
Figura 79 b.	Envío de datos	89
Figura 79 c.	Recepción de datos	90
Figura 80	Actividad 9. Panel frontal controlador	91
Figura 81	Organización y cableado del diagrama de bloque del controlador	91-93
Figura 81 a.	Configuración del puerto paralelo	91
Figura 81 b.	Recepción de datos	92
Figura 81 c.	Ecuaciones de control	92
Figura 81 d.	Almacenamiento de error	93
Figura 81 e.	Envío de datos	93
Figura 82	Icono para software de configuración de tarjeta	94
Figura 83	Caja de dialogo para identificación de la tarjeta	95
Figura 84	Ventana para abrir el asistente de creación de un nuevo canal	95
Figura 85	Asistente de creación de un nuevo canal	96
Figura 86	Actividad 8. Panel frontal	96

Figura 87 Organización y cableado del diagrama de bloques A.8 97

LISTA DE ANEXOS

	Pag.
Anexo A. Comunicación por el puerto paralelo	100
Anexo B. Comunicación a través del puerto serial	104
Anexo C. Comunicación con tarjetas DAQ	105

INTRODUCCIÓN

Bienvenido al Tutorial básico de LabVIEW. Este archivo de ayuda interactivo lo introducirá a los conceptos básicos de LabVIEW y lo guiará a través de muchas actividades para familiarizarlo con la programación gráfica.

Aquí encontrará la información básica sobre el uso del software, con ilustraciones que le ayudaran a comprender fácilmente, de igual forma usted encontrará una serie de actividades que lo conducirán de forma sencilla a la programación en LabVIEW. En estas actividades aprenderá a realizar simulaciones, implementar aplicaciones con el uso del puerto paralelo, puerto serial y tarjetas de adquisición de datos.

El fin de este Tutorial es que usted en una forma sencilla empiece a interactuar con el software, mas no convertirlo en un experto, por lo tanto, es menester de cada quien practicar e instruirse aún mas para adquirir un mejor dominio del software.

1. ¿ QUE ES LABVIEW?

LabVIEW (*Laboratory Virtual Instrument Engineering Workbench*) es un lenguaje de programación gráfico que usa los iconos en lugar de las líneas de texto para crear las aplicaciones. En contraste con los lenguajes de programación basados en texto, donde las instrucciones determinan la ejecución del programa, LabVIEW usa programación de Flujo de datos, donde el flujo de datos determina la ejecución.

En LabVIEW, usted construye una interfaz del usuario usando un juego de herramientas y objetos. La interfaz del usuario es conocida como el panel frontal (**FRONT PANEL**). Usted entonces agrega código usando representaciones gráficas de funciones para controlar los objetos del panel frontal. El Diagrama de Bloques (**BLOCK DIAGRAM**) contiene este código. De alguna manera, el diagrama de bloques se parece a un diagrama de flujo.

2. INSTRUMENTOS VIRTUALES

Los programas de LabVIEW son llamados instrumentos virtuales, o VIs, porque su apariencia y funcionamiento imitan instrumentos físicos, como los osciloscopios y multímetros. Cada VI usa funciones que manipulan la entrada desde la interfaz del usuario u otras fuentes y despliegan esta información o la mueve a otros archivos u otras computadoras.

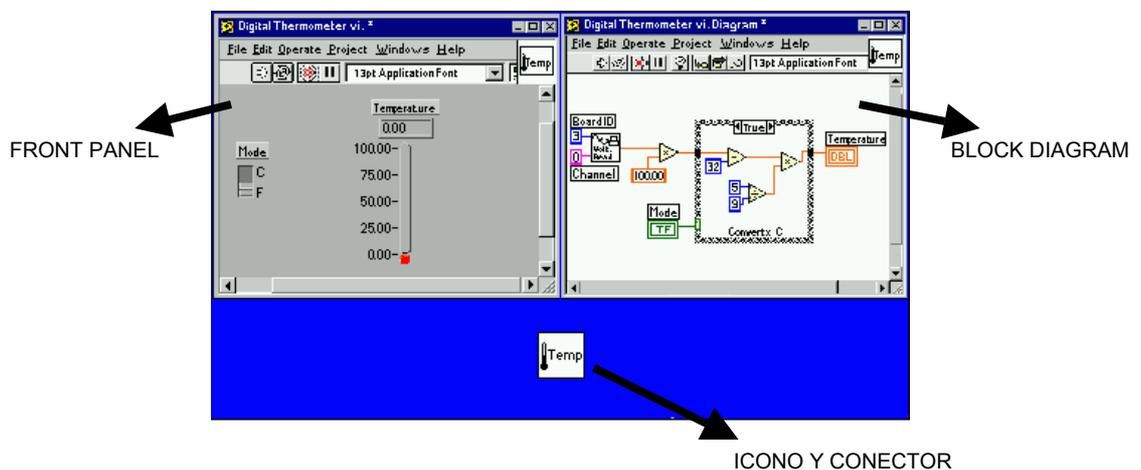


Figura 1. Componentes de un VI.

Un VI contiene tres componentes que son:

- **PANEL FRONTAL** (*front panel*) – Sirve como la interfaz del usuario.
- **DIAGRAMA DE BLOQUE** (*Block diagram*) -- contiene el código fuente gráfico que define la funcionalidad de los VI.
- **EI ICONO Y CRISTAL CONECTOR** (*Icon and conector*). -- Identifica los VI para que usted pueda usarlos en otro VI. Un VI dentro de otro VI es llamado un subVI.

3. EL AMBIENTE DE LabVIEW

Los elementos básicos del ambiente de LabVIEW son el **menú** , la **barra de herramientas**, y las **paletas flotantes** (paletas de control, de funciones y de herramientas) que usted puede poner en cualquier parte de la pantalla.

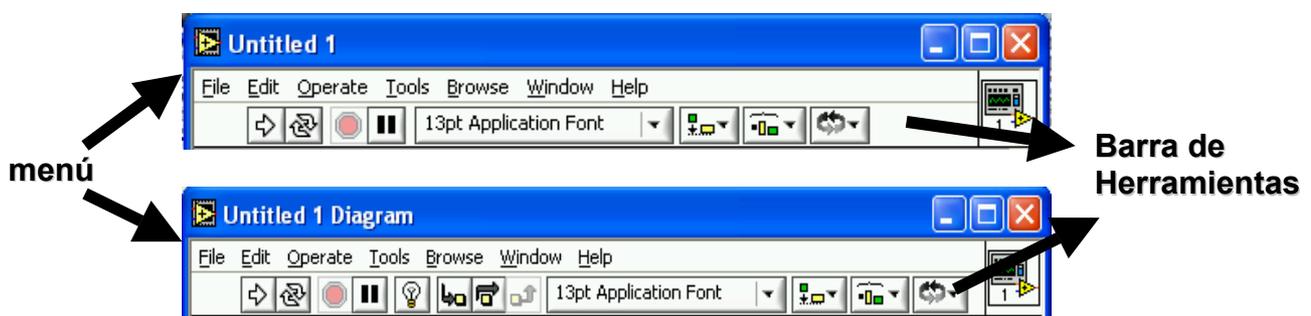


Figura 2. Menú y Barra de Herramienta

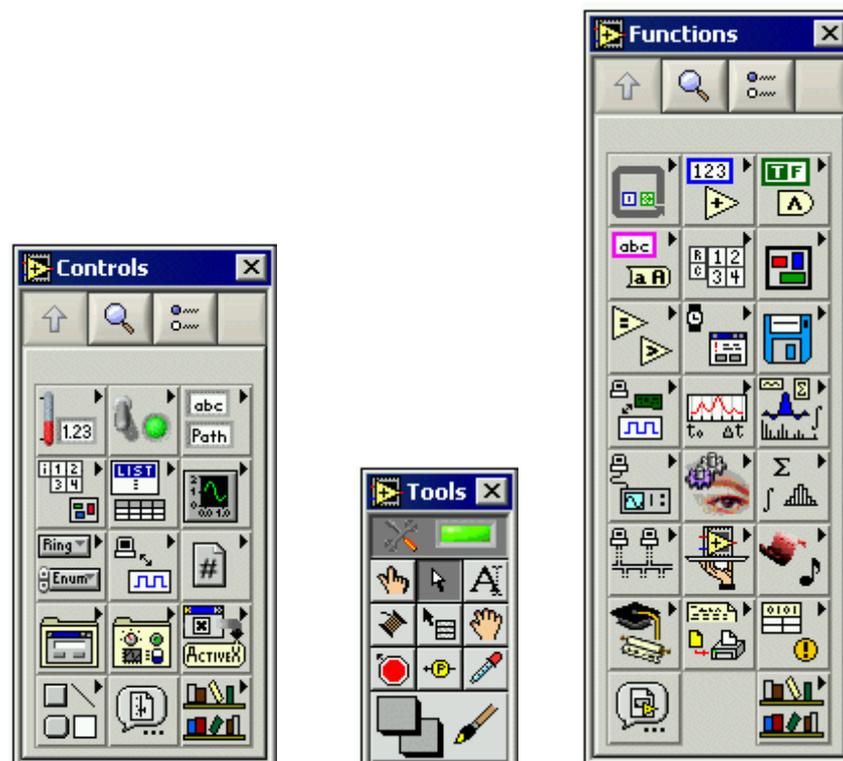


Figura 3. Paletas Flotantes

3.1. MENU PRINCIPAL.

Estas se encuentran en la parte superior de las ventanas del panel frontal y del diagrama de bloques.

La **barra de menús** contiene diversos menús *pull-down* (Desplegables). Cuando se hace clic sobre un ítem o elemento de esta barra, aparece un menú por debajo de ella. Dicho menú contiene elementos comunes a otras aplicaciones Windows, como **Open** (Abrir), **Save** (Guardar) y **Paste** (Pegar), y muchas otras particulares de LabVIEW, como se muestra en la siguiente figura.

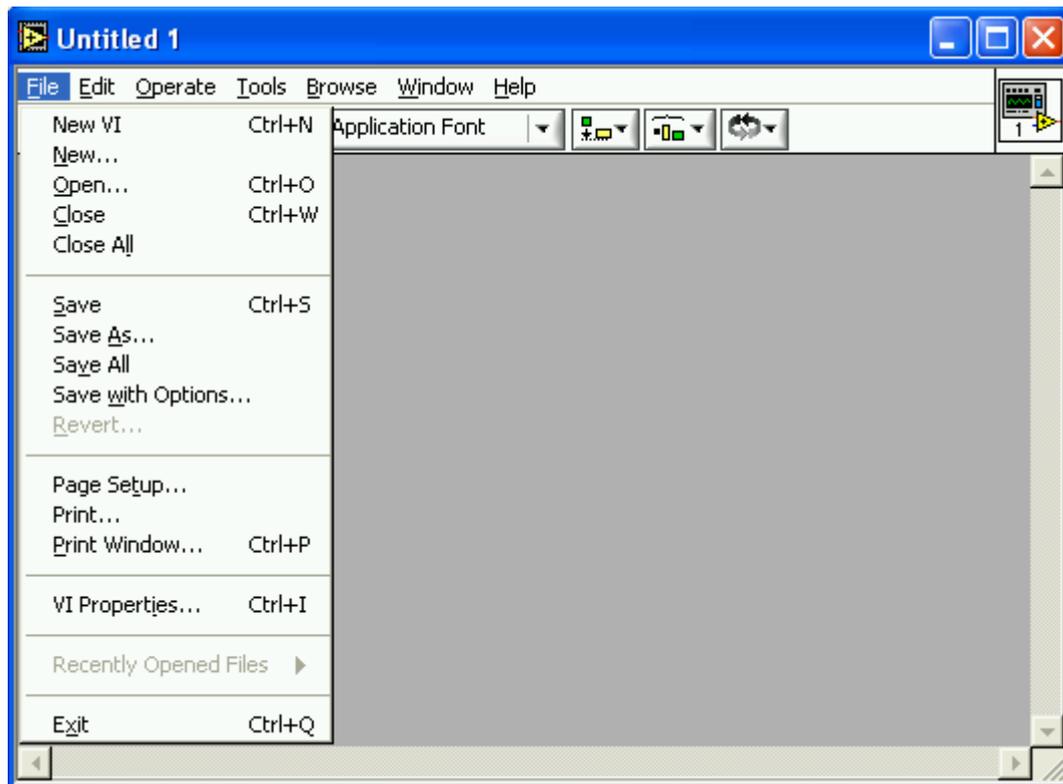
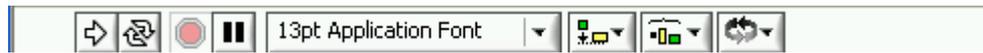


Figura 4. Menús Desplegables

3.2. BARRA DE HERRAMIENTAS.

La **Barra de Herramientas** contiene los comandos necesarios para la ejecución de un VI. Esta se describe a continuación.



El botón ***RUN*** ejecuta un VI. LabVIEW compila los VI, si es necesario. Usted puede ejecutar un VI si el botón RUN aparece como una flecha blanca sólida.



Mientras el VI se ejecuta, el botón ***RUN*** aparece como se muestra a la izquierda.



Si el VI que se está ejecutando es un subVI, el botón ***RUN*** aparece como se muestra a la izquierda.



El botón ***RUN*** aparece a menudo roto, como se muestra a la izquierda, cuando usted crea o revisa un VI. Si todavía está roto cuando usted termina la instalación eléctrica del diagrama de bloques, el VI tiene errores y no se ejecutara.



El botón ***Run Continuously*** ejecuta los VI hasta que usted interrumpa o pause la ejecución.



EL botón ***Abort Execution*** interrumpe la ejecución del VI de máximo nivel.



El botón ***Pause*** pausa la ejecución del VI. Cuando usted pulsa este botón, LabVIEW resalta en el diagrama de bloques la situación dónde usted hizo una pausa en la ejecución. Pulse el botón para continuar nuevamente el funcionamiento del VI.



Highlight Execution despliega una animación de la ejecución del diagrama de bloques.



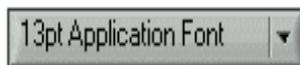
Step Into abre un nodo y pausa. Cuando usted pulsa el botón de nuevo, ejecuta la primera acción y pausa a la próxima acción del subVI o estructura.



Step Over ejecuta un nodo y pausa en el próximo nodo.



Step Out Finaliza la ejecución del nodo actual y pausa. Cuando el VI termina la ejecución, el botón **Step Out** se oscurece.



Cambia la fuente para los VI.



Align Objects encuadra los objetos a lo largo de los ejes.



Distribute Objects Distribuye los objetos uniformemente.



Reorder. Mueve los objetos relativos unos a otros. Seleccione el menú desplegable cuando usted tiene objetos que se traslapan entre si y usted quiere definir cuál estará delante o detrás del otro.

3.3. PALETAS FLOTANTES

Existen 3 tipos de paletas, la paleta de herramientas, la paleta de control y la paleta de funciones. Estas paletas contienen los elementos, funciones y herramientas necesarias para la elaboración de un VI.

Estas paletas se obtienen de la opción *Window* de la barra de menú principal, como se muestra en la figura 5.

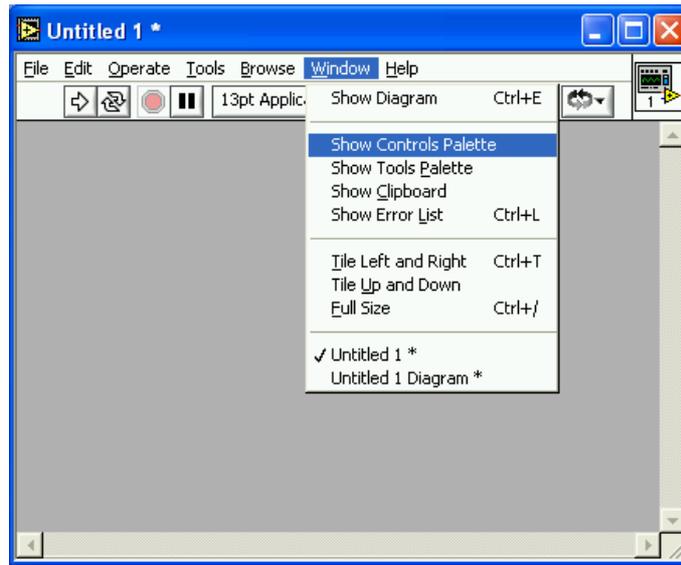


Figura 5. Obtención de las Paletas flotantes

3.3.1. PALETA DE HERRAMIENTAS. La Paleta de Herramientas (*Tools Palette*) Se emplea tanto en el panel frontal como en el diagrama de bloques. Contiene las herramientas necesarias para editar y depurar los objetos tanto del panel frontal como del diagrama de bloques.

Para obtener esta paleta haga clic en el menú Windows y luego en la opción *Tools Palette* como se indica en la figura 6.

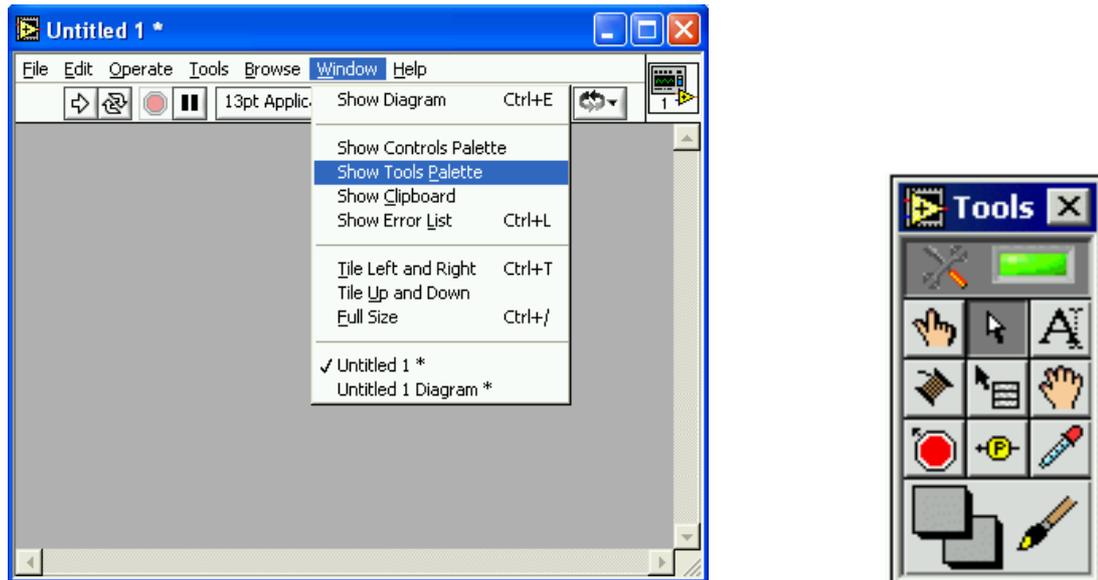


Figura 6. Obtención de la Paleta de Herramientas

Las opciones que se presentan en esta paleta son las siguientes:



Automatic Tool Selection – Esta herramienta automática de selección está habilitada y usted mueve el cursor encima de los objetos en el panel frontal o en el diagrama de bloques, LabVIEW selecciona la herramienta correspondiente automáticamente de la paleta de Herramientas. Usted puede desactivar la selección de la herramienta automática y puede seleccionar una herramienta manualmente.



Operating – Esta herramienta cambia el valor de los controles.



Positioning – Posiciona, cambia de tamaño y selecciona objetos.



Labeling – Edita y crea etiquetas libres.



Wiring – Cablea los objetos entre sí en el diagrama de bloques.



Object Shortcut Menu – Abre el menú abreviado de un objeto.



Scrolling – Desplaza las ventanas sin utilizar la barra de desplazamiento.



Breakpoint – Ajusta los puntos de interrupción en los VIs, funciones, nodos, cables, y estructuras para hacer una pausa en la ejecución del programa.



Probe – Crea puntos de prueba en los cables. Use esta herramienta para chequear valores intermedios en VI que produce resultados cuestionables o inesperados.



Color Copying – Copia el color para después establecerlo mediante la siguiente herramienta.



Coloring – Establece el color de los objetos y del fondo.

3.3.2. PALETA DE CONTROL. La **Paleta de Control** se utiliza únicamente en el *panel frontal*. Contiene todos los *controles* e *indicadores* que se emplearán para crear la interfaz del VI con el usuario.

Para obtener esta paleta haga clic en el botón derecho del *mouse* sobre el panel frontal o de clic en el menú Window, y seleccione la opción *show control palette*. Usted puede colocar la paleta en cualquier parte de la pantalla.

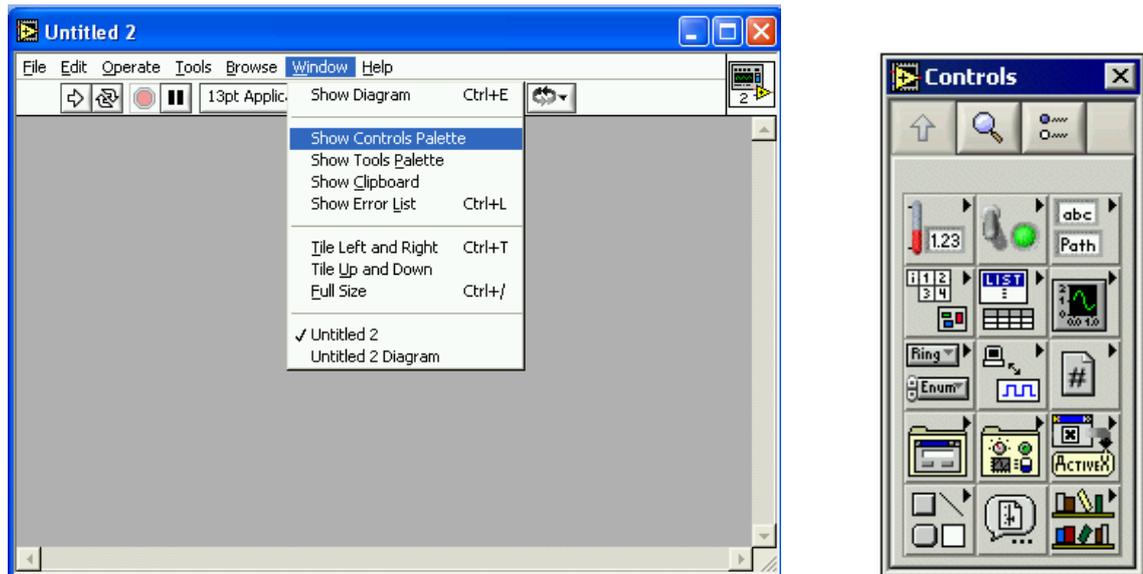


Figura 7. Obtención de la Paleta de Controles

En esta paleta se encuentran las siguientes opciones:



Up – Sube un nivel en la jerarquía de las paletas.



Search – Cambia la paleta a modo de búsqueda. En este modo usted puede desarrollar búsquedas basadas en texto para localizar controles, VIs, o funciones en las paletas.



Options – Abre las opciones de la caja de diálogo del navegador de funciones, desde la cual usted puede configurar la apariencia de las paletas.



Numeric – Muestra la subpaleta para seleccionar controladores e indicadores como se muestra en la figura.

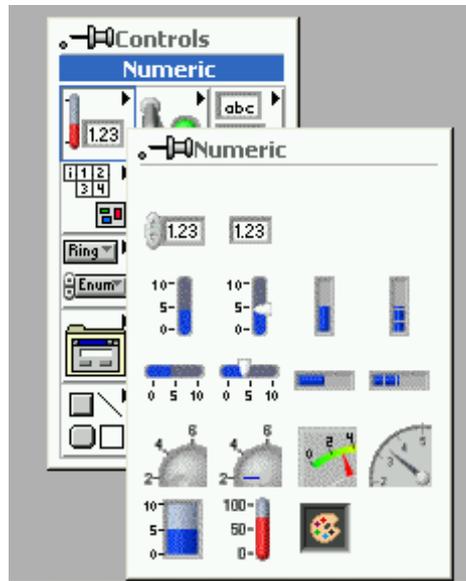


Figura 8. *Numeric Controls*



Boolean – Muestra la subpaleta para la selección de elementos booleanos como switches y leds.

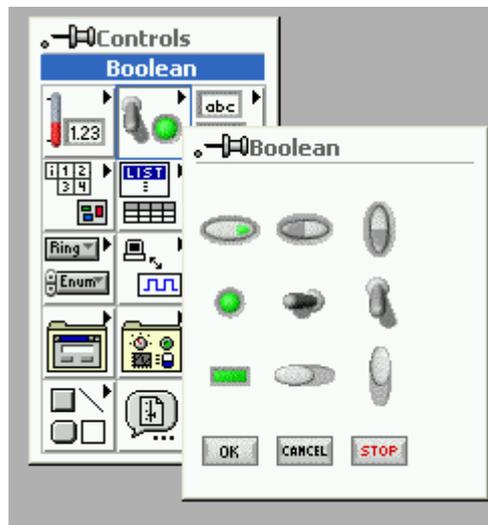


Figura 9. *Boolean Controls*



String & path – Muestra la subpaleta para seleccionar las cajas de entrada de texto y entrar o retornar la localización de un archivo o directorio.

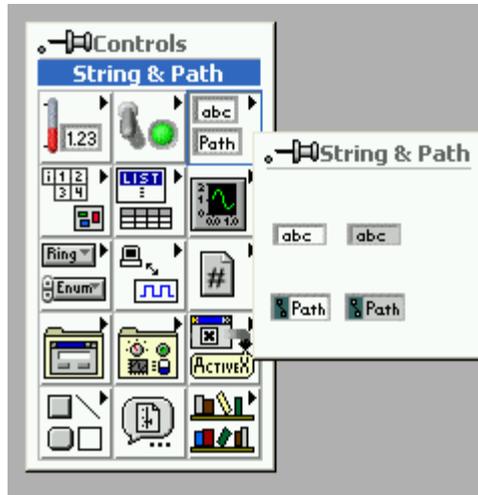


Figura 10. *String & Path*



Array & Cluster – Muestra la subpaleta en la cual se puede elegir las diferentes herramientas para agrupar elementos.

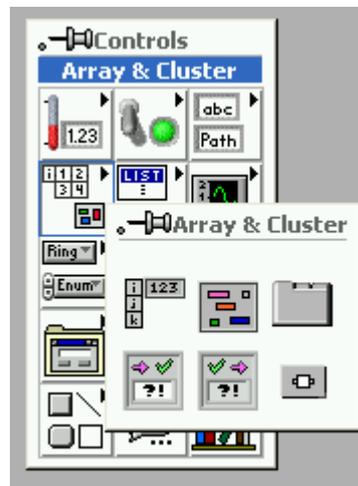


Figura 11. *Array & Cluster*



List & table – Muestra la subpaleta en la cual se pueden seleccionar elementos para visualizar y/o ingresar una lista de opciones.



Figura 12. List & Table



Graph – Proporciona una lista de controladores e indicadores de gráfico para imprimir datos numéricos en forma de gráfico o de mapa.

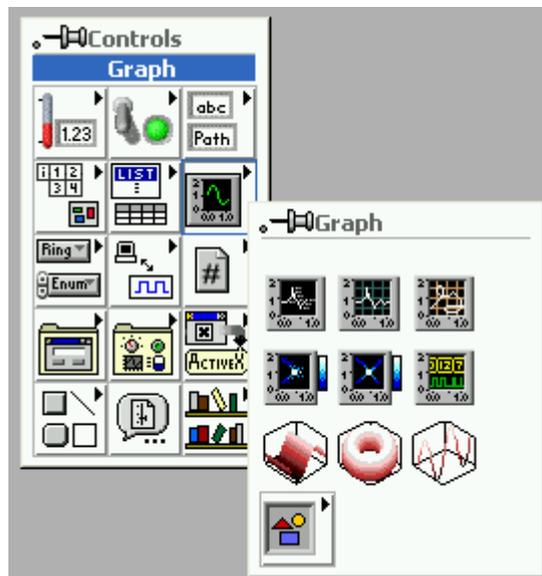


Figura 13. Graph



Ring & Enum – Proporciona controladores e indicadores para manipular cadenas de caracteres.

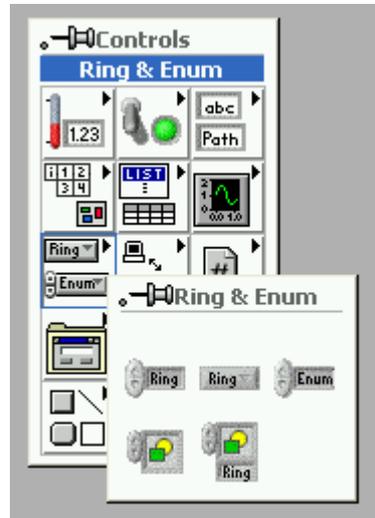


Figura 14. Ring & Enum



I/O – Proporciona las herramientas necesarias para la comunicación con un instrumento o con un dispositivo de adquisición de datos.

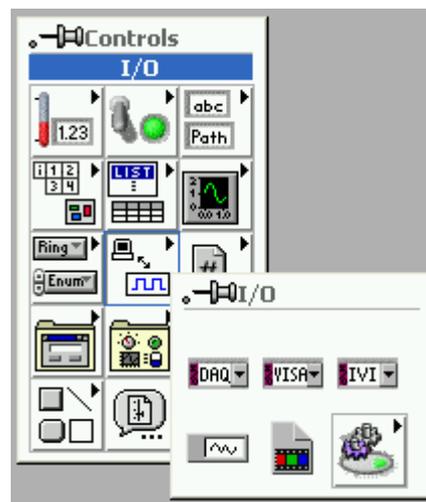


Figura 15. I/O



Refnum – Use esta paleta de referencia de numero para trabajar con archivos, directorios, dispositivos y conexiones de red.

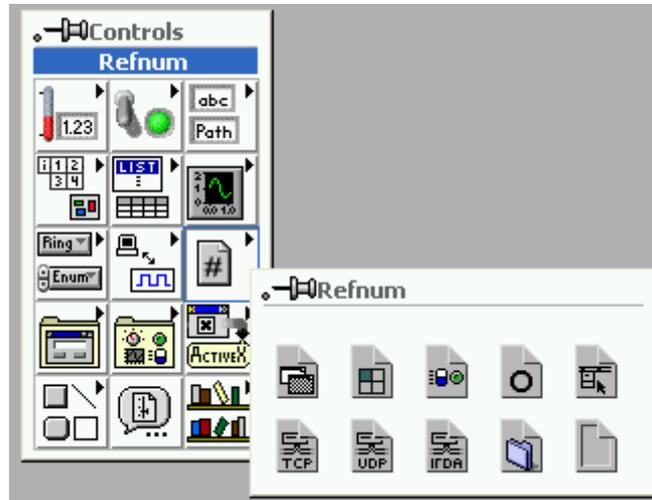


Figura 16. *Refnum*



Dialog Controls – Contiene funciones para trabajar con cuadros de diálogo.

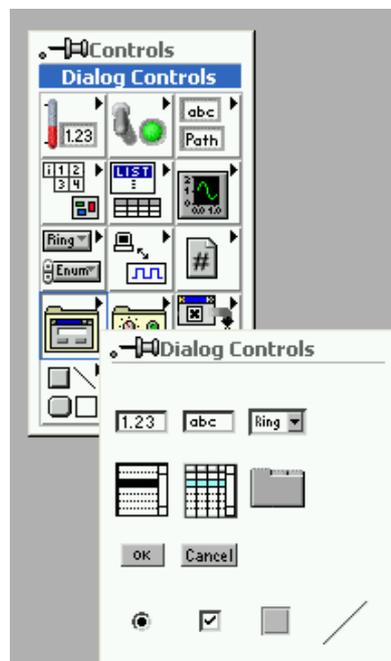


Figura 17. *Dialog Controls*



Classic Controls – Muestra la subpaleta en la cual se encuentran los elementos de versiones anteriores.

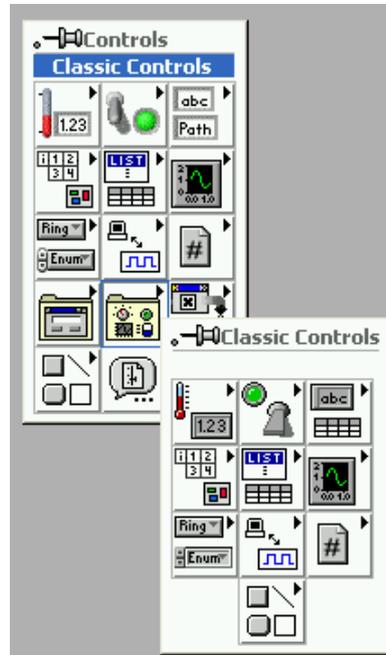


Figura 18. **Classic Controls**



ActiveX – Para transferir datos y programas de unas aplicaciones a otras dentro de Windows.

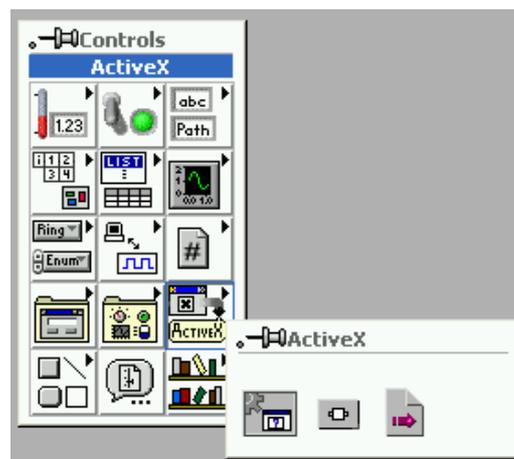


Figura 19. **ActiveX**



Decorations – Use esta paleta para agrupar o separar objetos en el panel frontal con cajas, líneas o flechas. Estos son para decoraciones únicamente y no para mostrar datos.

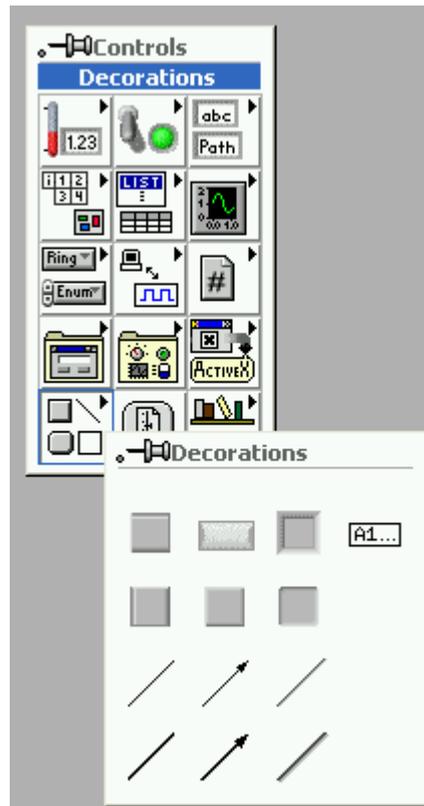


Figura 20. Decorations



Select a Control – Use este botón para seleccionar un control personalizado diseñado por usted mismo.



User Controls – Use esta subpaleta para seleccionar controles automáticamente de la librería.

3.3.3. PALETA DE FUNCIONES. La **Paleta de Funciones** se utiliza únicamente en el diagrama de bloques. Contiene todas las funciones y VIs que se emplearán para crear el diagrama de bloques.

Para obtener esta paleta haga clic en el botón derecho del mouse sobre el diagrama de bloques o de clic en el menú Window, y seleccione la opción **Show Functions Palette**. Usted puede colocar la paleta en cualquier parte de la pantalla.

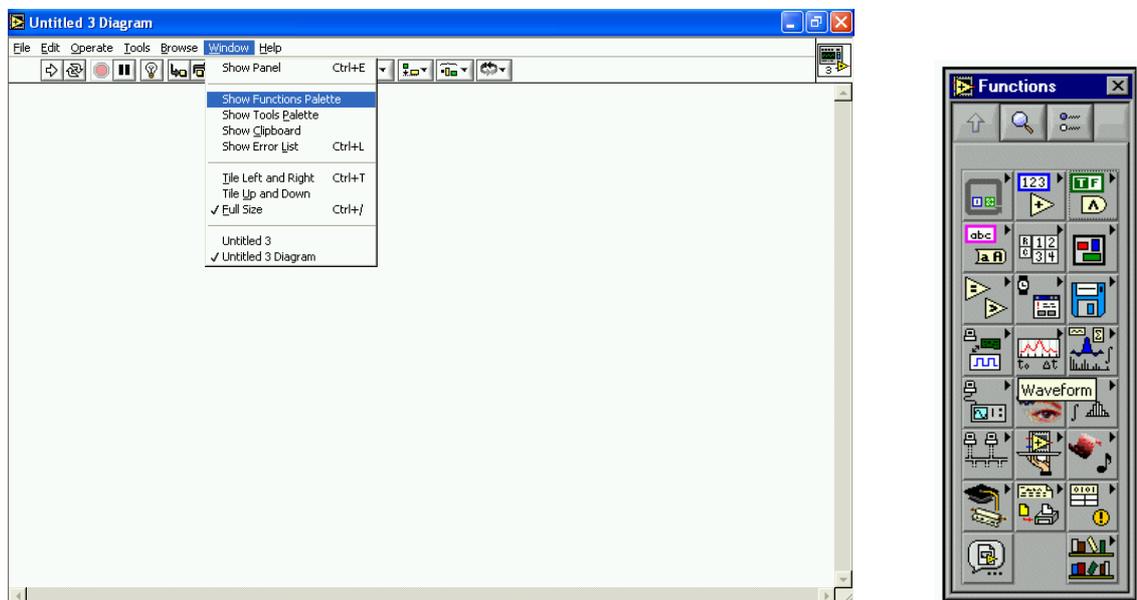


Figura 21. Obtención de la Paleta de Funciones

En esta paleta se encuentran las siguientes opciones:



Up – Sube un nivel en la jerarquía de las paletas.



Search – Cambia la paleta a modo de búsqueda. En este modo usted puede desarrollar búsquedas basadas en texto para localizar controles, VIs, o funciones en las paletas.



Options – Abre las opciones de la caja de diálogo del navegador de funciones, desde la cual usted puede configurar la apariencia de las paletas.



Structures – Use esta subpaleta para localizar los diferentes tipos de estructuras para la programación lógica.

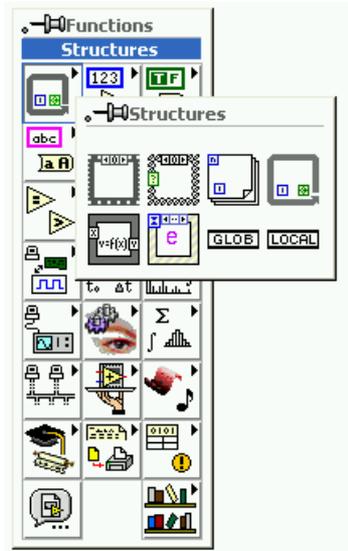


Figura 22. *Structure*



Numeric – Use esta subpaleta para crear o desarrollar operaciones matemáticas y para convertir números de un tipo de datos a otro.

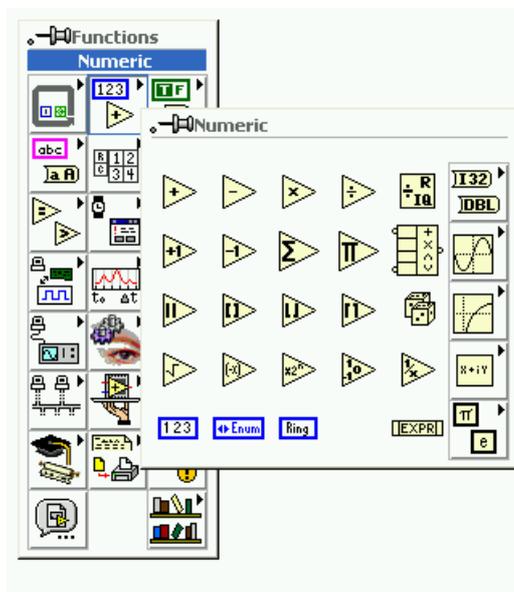


Figura 23. *Numeric Funtions*



Boolean – Use esta subpaleta para desarrollar operaciones lógicas en valores boléanos simples o en series.

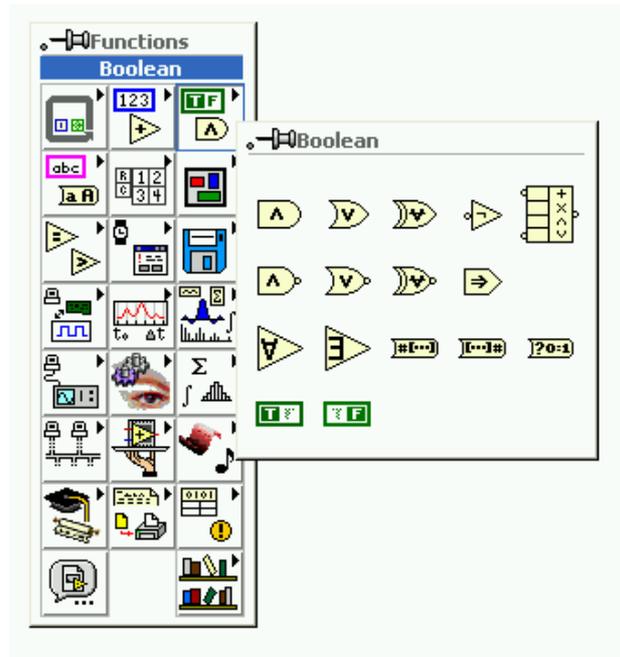


Figura 24. *Bolean Funtions*



String – Use esta subpaleta para concatenar dos o más cadenas y convertir datos a cadenas.

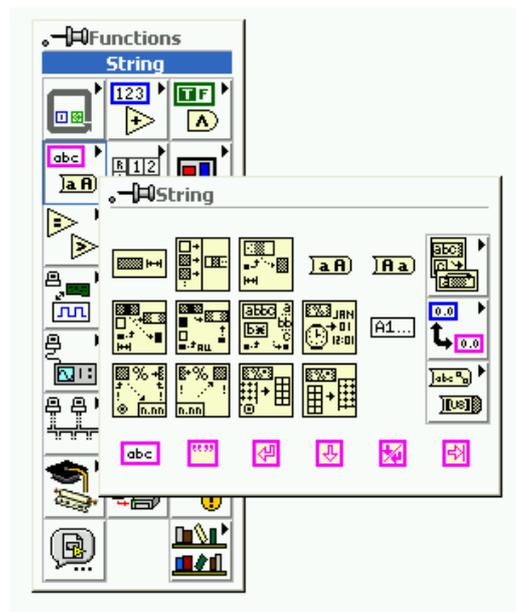


Figura 25. *String*



Array – Use esta subpaleta para manipular y crear datos en forma vectorial.

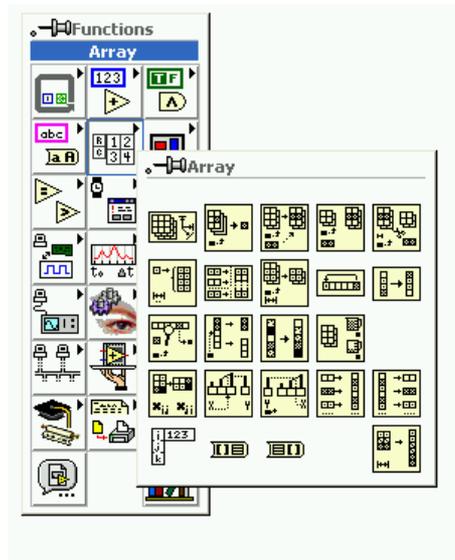


Figura 26. *Array*



Cluster – Contiene funciones útiles para procesar datos procedentes de gráficas y destinados a ser representados en ellas, así como las correspondientes constantes.

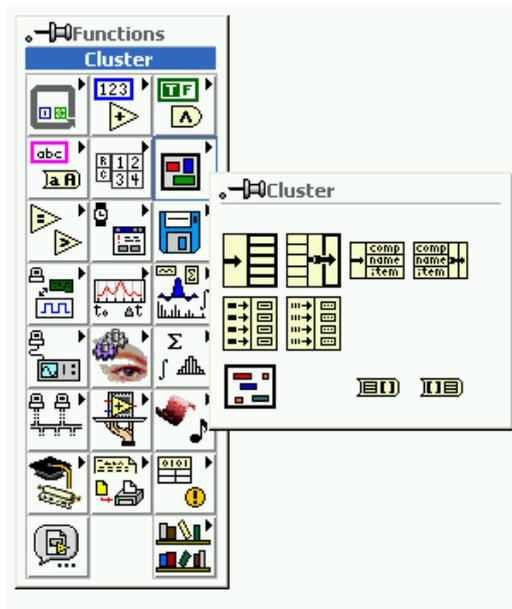


Figura 27. *Cluster*



Comparison – contiene funciones que sirven para comparar números, valores booleanos o cadenas de caracteres.

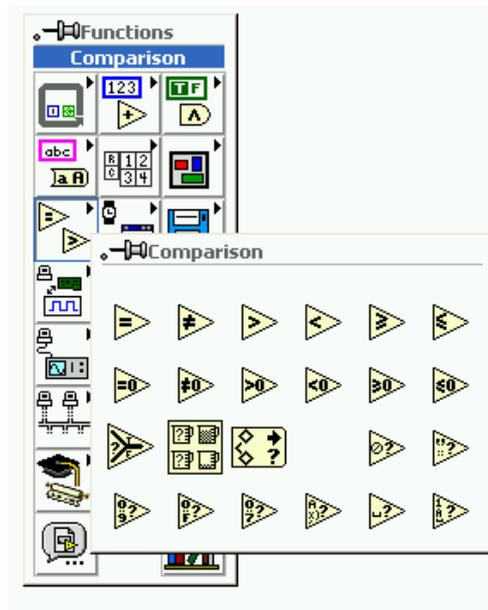


Figura 28. **Comparison**



Time & Dialog – Use esta subpaleta para manipular la velocidad a la cual una operación se ejecuta y obtener información de el reloj de su computadora como fecha y hora.

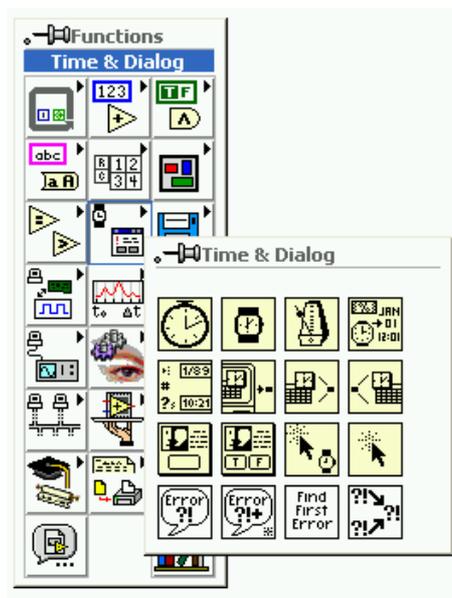


Figura 29. **Time & Dialog**



File I/O – Muestra funciones para operar con ficheros.

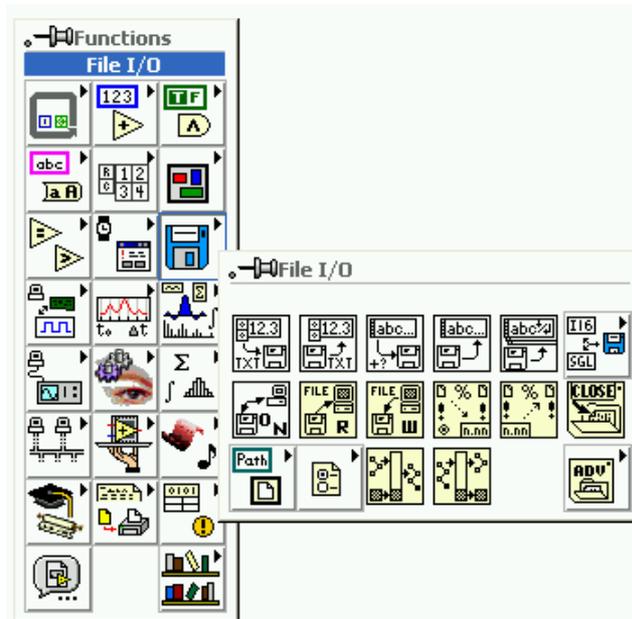


Figura 30. File I/O



Data Acquisition – Use esta subpaleta para elegir todo lo referentes a la adquisición de datos.

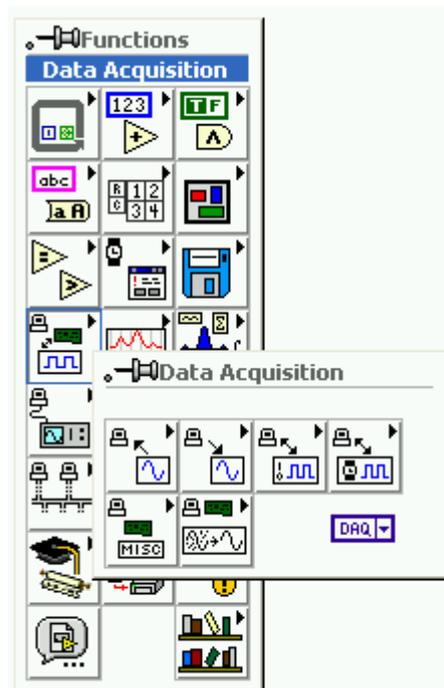


Figura 31. Data Acquisition



Waveform – Use esta subpaleta para construir diferentes formas de ondas que incluyan valores de la señal, información de canales e información de tiempo.

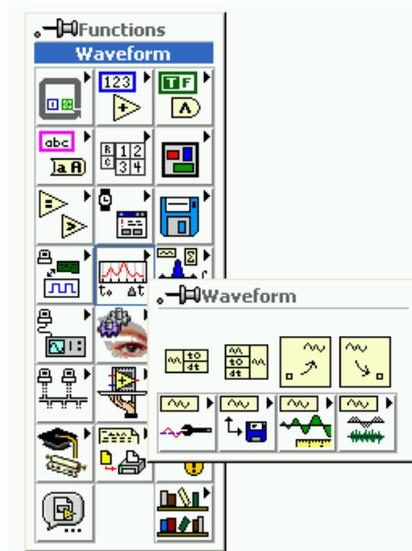


Figura 32. *Waveform*



Analyze – Use los VIs localizados en esta subpaleta para efectuar mediciones en dominio del tiempo y de la frecuencia, medición de armónicos, distorsión de ruidos y otros.

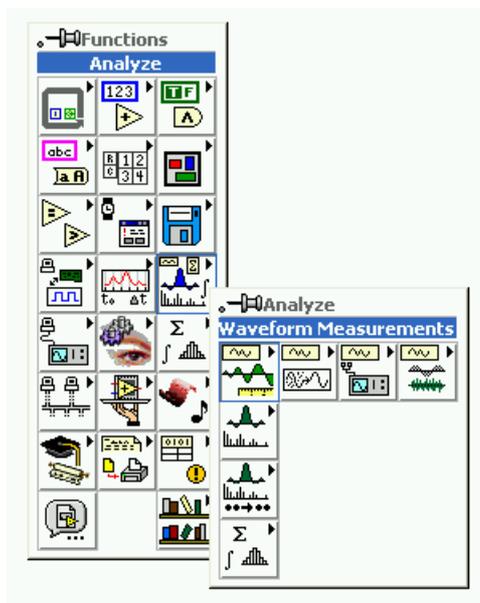


Figura 33. *Analyze*



Instrument I/O – Use esta subpaleta como interfase con dispositivos incluyendo GPIB, serial e instrumentos VXI.

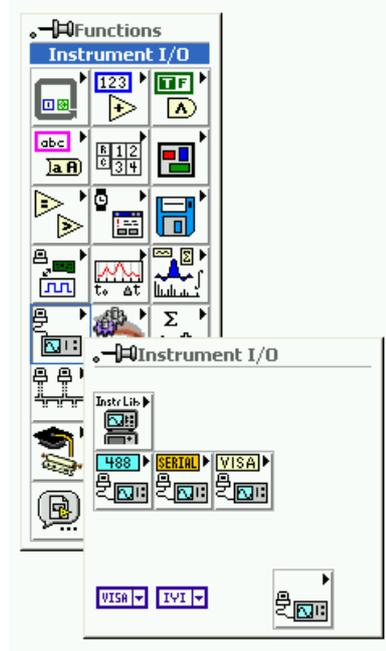


Figura 34. **Instrument I/O**



Motion & Vision – Use esta opción para analizar imágenes con una variedad de técnicas y para entregar aplicaciones del alto desempeño en movimiento.

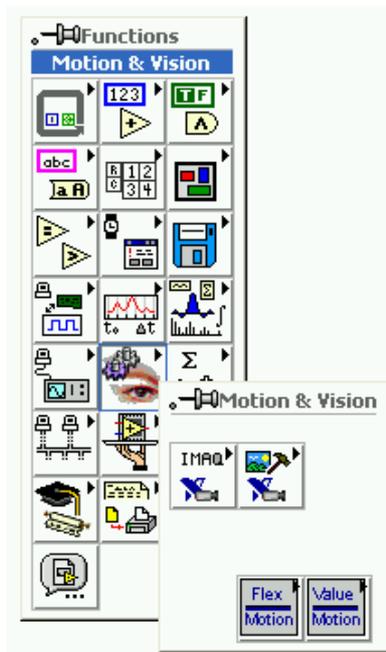


Figura 35. **Motion & Vision**



Mathematics – Use esta subpaleta para desarrollar diferentes tipos de cálculos matemáticos.

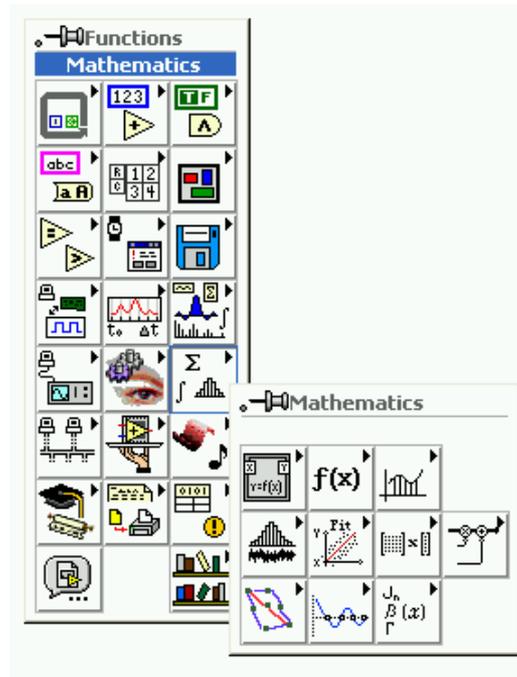


Figura 36. **Mathematics**



Communication – Use esta subpaleta para intercambiar datos entre aplicaciones.



Figura 37. **Communication**



Application Control – Use esta subpaleta para Programáticamente controlar VIs y aplicaciones de LabVIEW en su computador local o a través de una red.

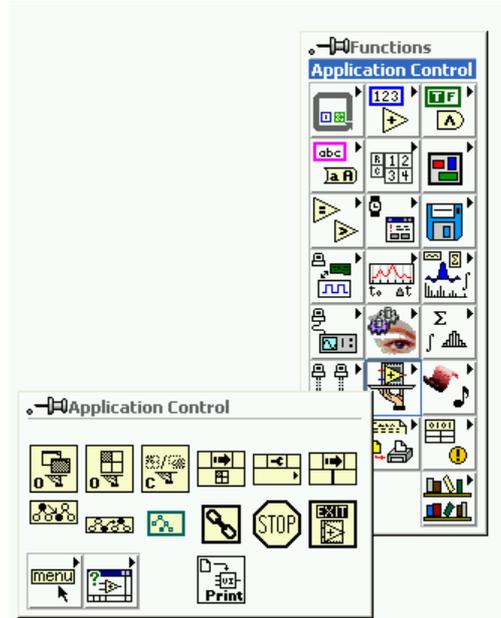


Figura 38. *Application Control*



Graphics & Sound – Use esta subpaleta para crear displays personalizados, importar y exportar datos desde archivos gráficos, y manipular sonidos.

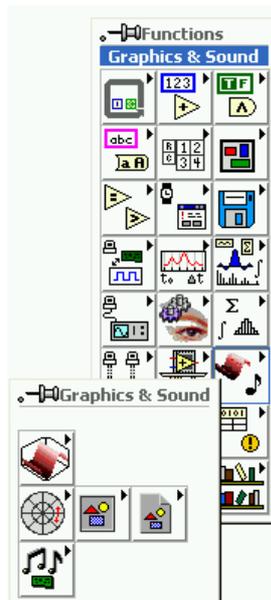


Figura 39. *Graphics & Sound*



Tutorial – Contiene VIs usados en las actividades en el tutorial de LabVIEW.

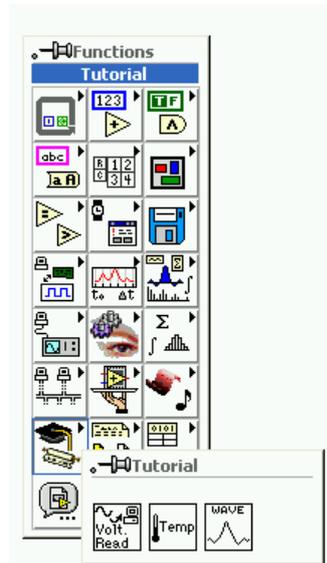


Figura 40. **Tutorial**



Report Generation – Úsela para crear y manipular reportes de las aplicaciones de LabVIEW.

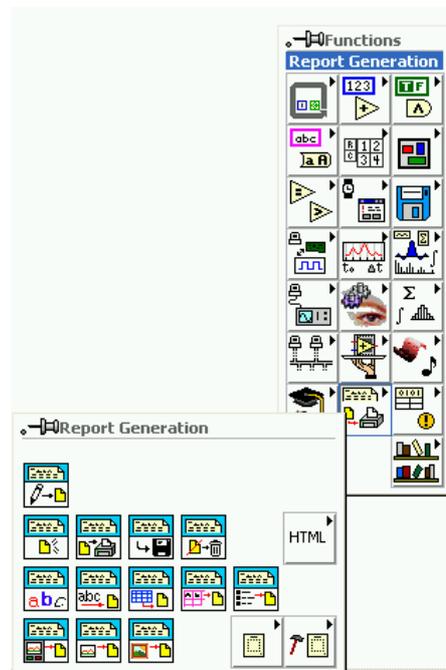


Figura 41. **Report Generation**



Advanced – Úsela para llamar el código desde las librerías, tales como, lazos dinámicos (DLLs), para manipular datos de LabVIEW para usar en otras aplicaciones.

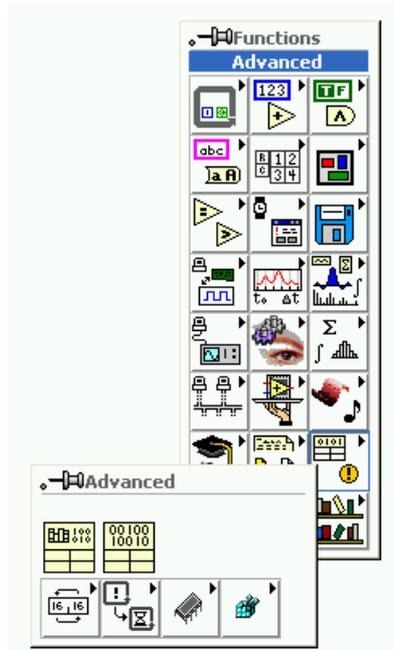


Figura 42. Advanced



Select a VI – Úselo para abrir un VI de la librería de VI.



User Libraries – Use esta subpaleta para ver las aplicaciones de los VI que usted ha construido y grabado. Usted puede adicionar VIs y controles a esta subpaleta.

4. PANEL FRONTAL

Es la interfaz del usuario con el instrumento virtual. En esta ventana se encuentran los elementos de entrada y salida de datos que comunican al instrumento con el usuario.

La siguiente imagen es un ejemplo de un panel frontal, en el cual se encuentran elementos indicadores (termómetro y display digital) y controladores (*Switch* selector de escala de temperatura).

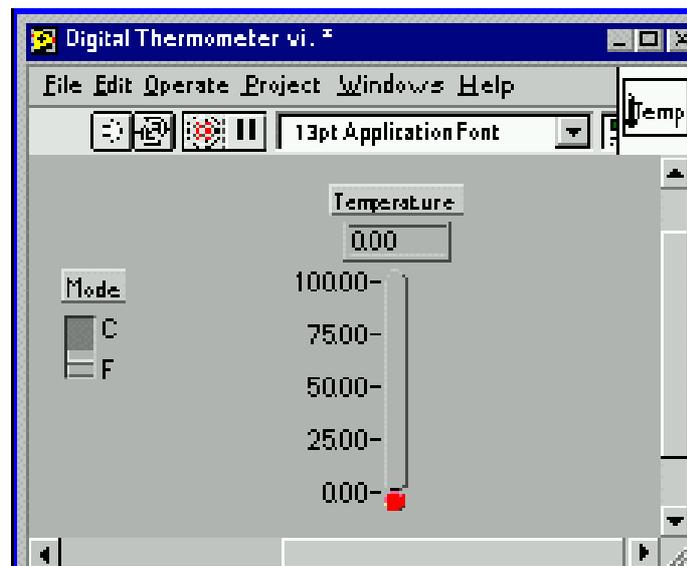


Figura 43. Ejemplo de Panel Frontal

CONSTRUYENDO EL PANEL FRONTAL

Usted construye el PANEL FRONTAL colocando controladores e indicadores, los cuales son los terminales interactivos de entradas y salidas de los VI respectivamente.

Los **controles** son, botones, pulsadores, diales, y otros dispositivos de entrada. Los cuales simulan dispositivos de entrada del instrumento y suministran datos al diagrama de bloques del VI.

Los **indicadores** son gráficos, *LEDs*, y otros despliegues. Éstos simulan dispositivos de salida del instrumento y despliegan datos que el diagrama de bloques adquiere o genera.

Cada control o indicador tiene un menú abreviado, usted puede usarlo para cambiar varios atributos o seleccionar los diferentes ítems del menú.

Cuando usted coloca un control o indicador en el panel frontal, un terminal correspondiente aparece en el diagrama de bloques, como se muestra en la figura.

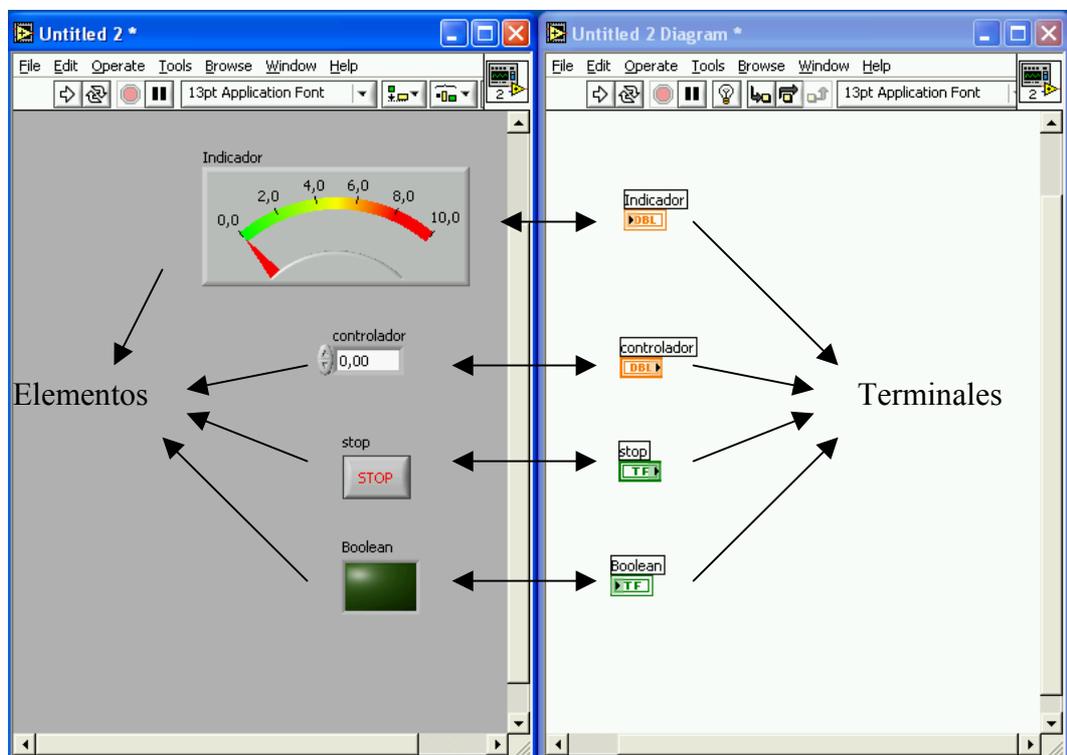
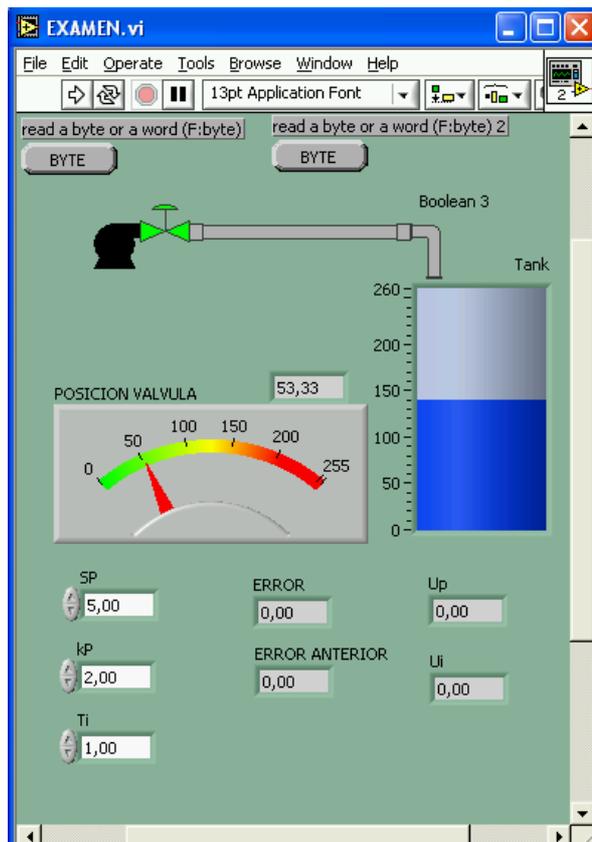


Figura 44. Elementos del Panel Frontal y sus equivalentes en el Diagrama de Bloques



El panel frontal se diseña de acuerdo con las características de la aplicación y puede contener todos los elementos que sean necesarios para la misma.

Para que su aplicación tenga una buena presentación usted puede hacer uso de elementos de la librería como lo son tubos, válvulas, bombas y otros elementos los cuales le ayudan a simular el proceso y le dan animación a dicha aplicación.

Figura 45. Ejemplo de Diseño de un Panel Frontal

5. DIAGRAMA DE BLOQUES

El diagrama de bloques contiene el código gráfico para su VI. La siguiente imagen es un ejemplo de una ventana de un diagrama de bloques. En ésta se puede observar elementos terminales (controlador *deg C*, indicador *deg F*), funciones matemáticas (\times , $+$) y bloques de constantes. También se puede observar los cables que unen los distintos elementos.

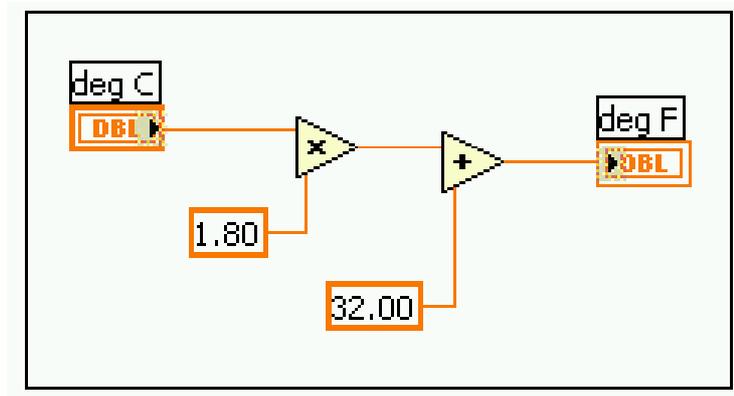


Figura 46. Ejemplo de un Diagrama de Bloques

CONSTRUYENDO EL DIAGRAMA DE BLOQUES

Los valores que se ingresan a través del panel frontal usando los controladores, llegan al diagrama de bloques por medio del bloque terminal correspondiente y pueden ser procesados a través de las distintas **funciones**. La respuesta del proceso o señal de salida la se puede visualizar nuevamente en el panel frontal, ya que ésta sale desde el terminal hacia el indicador correspondiente.

Los datos que ingresan del medio exterior a través de tarjetas, puertos y otros, también son procesados en el diagrama de bloques con la ayuda de las funciones y el resultado de dicho proceso se puede visualizar en el panel frontal dándoles salidas a través de un terminal de un indicador.

Las **funciones** proporcionan las operaciones matemáticas, lógicas, secuenciales entre otras, necesarias para la programación gráfica en LabVIEW.

Los **terminales** son la interfase entre el panel frontal y el diagrama de bloques. Usted no puede borrar un terminal desde el diagrama de bloques. Los terminales desaparecen únicamente después de que usted borra su correspondiente objeto en el panel frontal.

Las **estructuras**. En la paleta de funciones la primera opción es la de las estructuras. Éstas controlan el flujo del programa, bien sea mediante la secuenciación de acciones, ejecución de bucles, etc.

Los **cables** unen las funciones y terminales entre sí y definen la dirección del flujo de datos. Cuando se realiza una conexión indebida estos aparecen en forma punteada.

Usted puede reubicar cualquier objeto del diagrama de bloques arrastrándolo con el mouse con el fin de organizarlo sin afectar las conexiones del mismo.

6. ICONO Y CRISTAL CONECTOR

Existe un tercera parte importante de un VI, este es el icono y cristal conector.

Esta es la interfase entre un VI cuando es utilizado como un subVI.

Un **icono** es una representación gráfica de un VI, éste puede contener texto, imágenes o una combinación de ambas. Si usted usa un VI como un subVI, el icono identifica el subVI cuando éste es llamado desde el diagrama de bloques de otro VI.

Éste está localizado en la esquina superior derecha de las ventanas del panel frontal y del diagrama de bloque. El icono es lo que usted ve en el diagrama de bloque cuando usa un VI como subVI.

Use el **cristal conector** para colocar los terminales de entrada y salida del VI de tal manera que usted pueda usarlo como un subVI en otro VI.

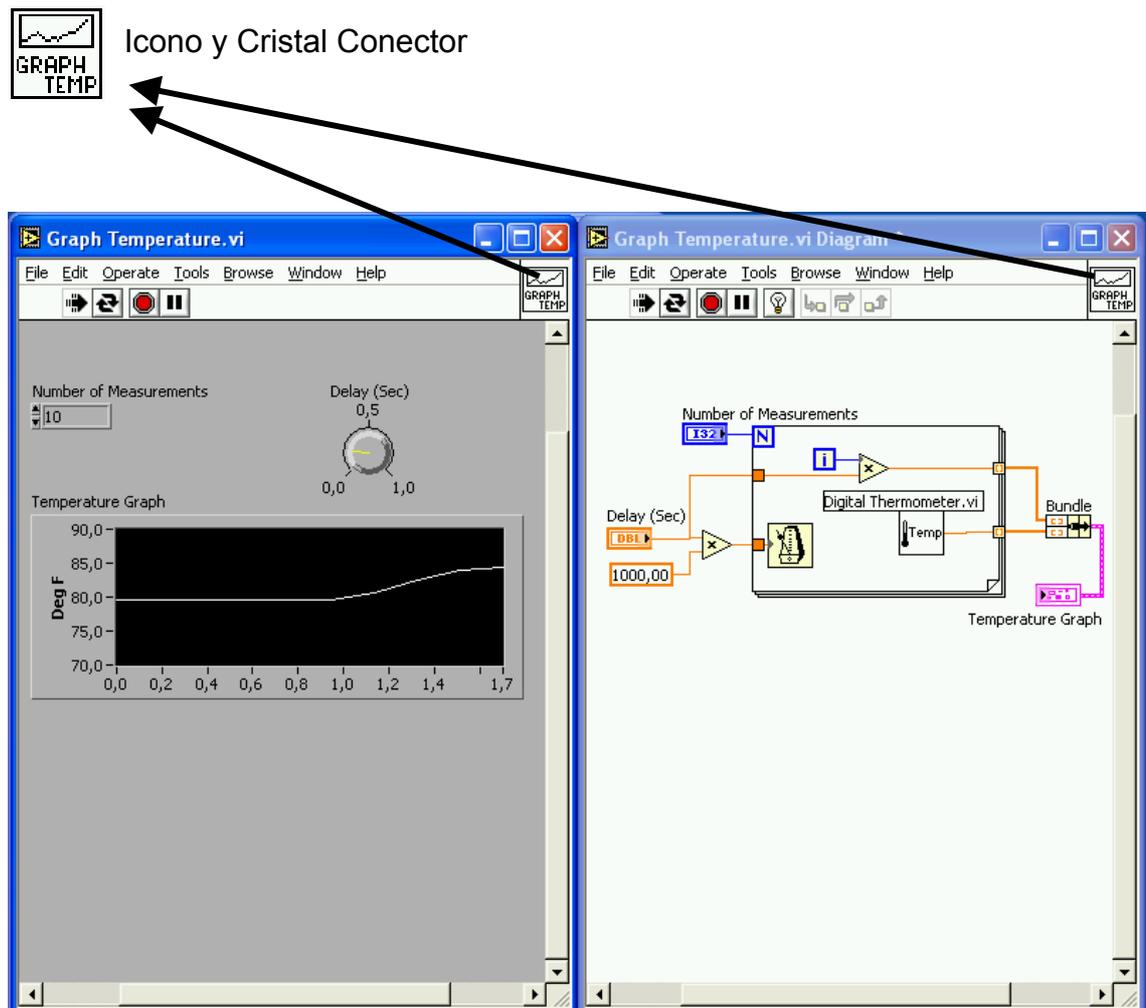


Figura 47. Icono y Cristal Conector

CREANDO EL ICONO Y CRISTAL CONECTOR

Después de haber construido el panel frontal y el diagrama de bloques de un VI, se debe construir el **icono y cristal conector** para poder utilizar el VI como un subVI.

Siga los siguientes pasos para **crear o editar el icono** de un VI.

- ✓ Haga clic derecho sobre el icono en la parte superior derecha de la ventana del panel frontal o del diagrama de bloques y seleccione **Edit Icon** del menú abreviado para mostrar la caja de dialogo del editor de icono.

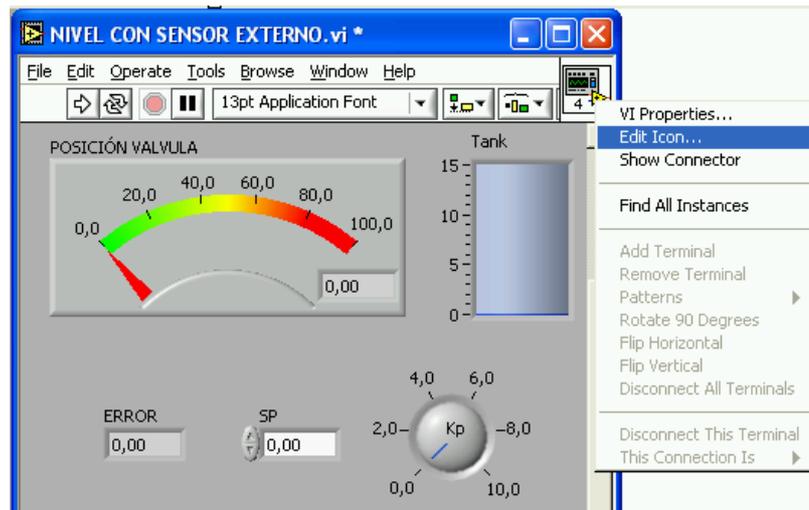


Figura 48. Menú Desplegable del Icono

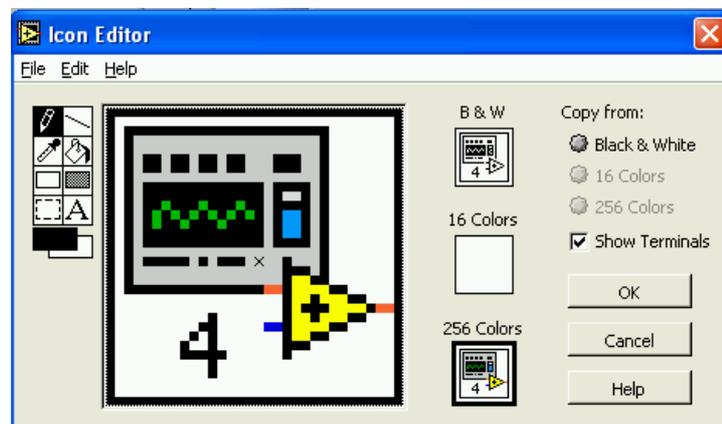


Figura 49. Editor del Icono

- ✓ Haga clic en la caja de 16 ó 256 colores para seleccionar cual icono quiere usted crear
- ✓ Marque la casilla de verificación de **Show Terminals** para mostrar un contorno del cristal conector en el área de edición. El cristal conector es solo una guía y no aparece en su icono final.

- ✓ Use las herramientas en el lado izquierdo de la caja de diálogo del editor de icono para crear el diseño del icono en el área de edición, éste puede contener texto, imágenes o una combinación de ambas.

- ✓ Use la opción **Copy from** en el lado derecho de la caja de diálogo de edición de icono de un icono de color a un icono blanco y negro y viceversa. Después de esto de click en el botón **OK** para completar el cambio.

Una vez creado el icono es necesario **construir el cristal conector** y para esto deben seguirse los siguientes pasos:

- ✓ Haga clic derecho en el icono de la parte superior derecha de la ventana del panel frontal y seleccione **Show Connector** del menú abreviado para mostrar el cristal conector. Este reemplaza al icono.

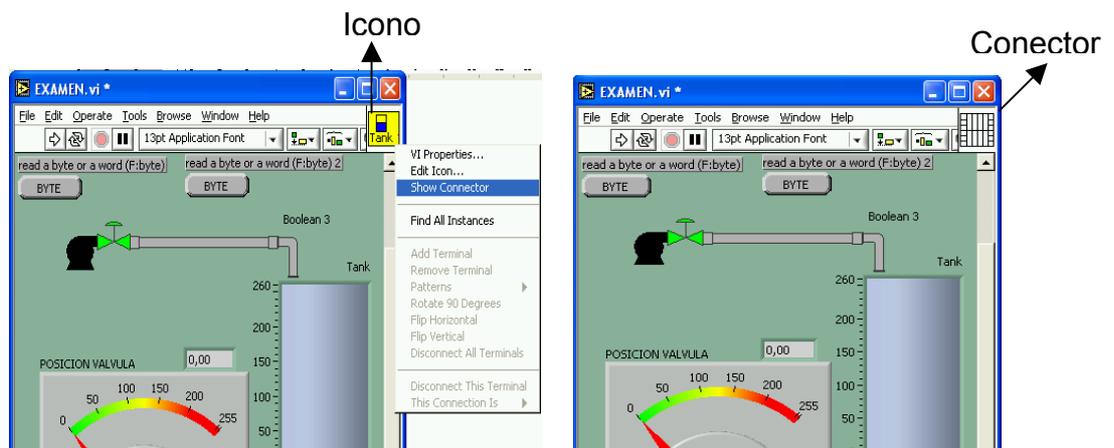


Figura 50. Ubicación del Icono y Cristal Conector

- ✓ Haga clic sobre un terminal del cristal conector. La herramienta automáticamente cambia a la herramienta de cableado y el terminal se torna negro. Aunque usted use la herramienta de cableado para asignar terminales sobre el cristal conector a los controles e indicadores del panel frontal, ningún cable es dibujado entre el cristal conector y estos controles e indicadores.
- ✓ Haga clic en el controlador o indicador del panel frontal el cual usted quiere asignar al terminal. Un cuadro destellante resaltará el objeto.
- ✓ Haga clic en un espacio abierto del panel frontal. El cuadro desaparece y el terminal cambia al color del tipo de dato del controlador o indicador al cual usted conectó el terminal.
- ✓ Repita los pasos anteriores para cada control o indicador al cual usted desea asignar un terminal.
- ✓ Guarde el VI para conservar todos los cambios realizados en éste.

7. LAZOS Y ESTRUCTURAS

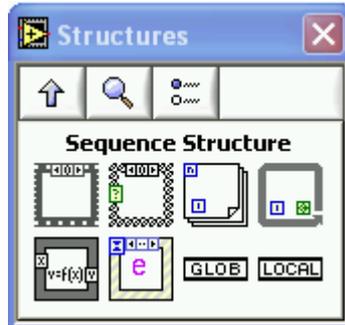


Figura 51. *Structure*

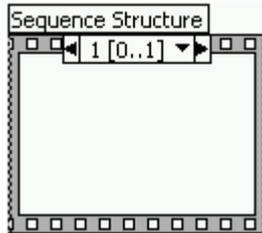
Las estructuras son representaciones gráficas de los lazos y declaraciones de caso de los lenguajes de programación basados en texto.

Use las estructuras sobre el diagrama de bloques para repetir bloques de código y para ejecutar códigos condicionalmente o en un orden específico.

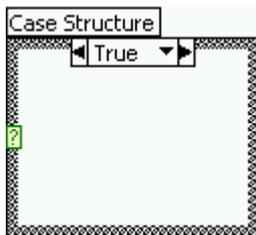
Al igual que otros nodos, las estructuras tienen terminales que las conectan con otros nodos del bloque de diagramas; se ejecutan automáticamente cuando el dato de entrada está disponible, y suministran datos a los cables de salida cuando la ejecución se ha completado.

Cada estructura tiene un borde distintivo y de tamaño ajustable para encapsular la sección del bloque de diagramas que se ejecuta según las reglas de la estructura.

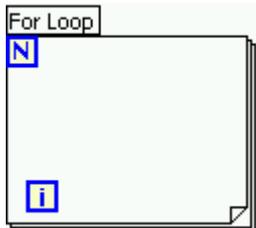
LabVIEW ofrece seis diferentes tipos de estructuras que son:



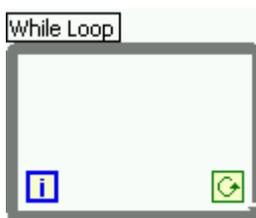
Sequense Structure: Consiste de uno o mas subdiagramas o cuadros que se ejecutan secuencialmente. De clic derecho en el borde de la estructura para añadir y borrar cuadros, o crear secuencias locales para pasar datos entre cuadros.



Case Structure: Ejecuta un subdiagrama dependiendo del valor de entrada pasado a la estructura a través del terminal de selección.

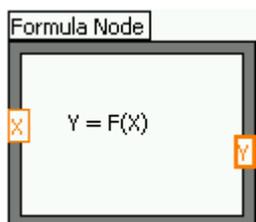


For Loop: Ejecuta su subdiagrama n veces, donde n es el valor cableado al terminal contador (N). El terminal de iteraciones (i) proporciona la cuenta de la correspondiente iteración del lazo, la cual va desde 0 a n-1.

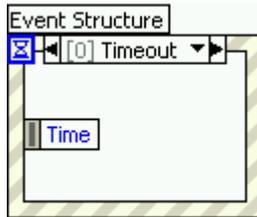


While Loop: Repite el subdiagrama dentro de este hasta que el terminal condicional, y terminal de entrada reciban un valor booleano en particular.

De clic derecho sobre el terminal condicional y seleccione **Stop if True** o **Continue if True** del menú abreviado para configurar el lazo.



Formula Node: Evalúa expresiones y formulas matemáticas en el diagrama de bloques.



Event Structure: Tiene uno o mas subdiagramas, o casos de eventos, exactamente uno de los cuales se ejecuta cuando la estructura se ejecuta.

La estructura de eventos espera hasta que un evento pasa en el panel frontal, entonces ejecuta el caso apropiado para manejar ese evento.

De clic sobre el borde de la estructura para añadir nuevos casos de eventos y configurar cuales eventos manejar.

8. COMO CONSTRUIR UN VI

En este apartado se construirá un VI sencillo como ejemplo para familiarizarse con la programación gráfica de LabVIEW.

A continuación se construirá un VI el cual realizará lo siguiente: Al ingresarle dos números a través de controladores digitales el VI ejecutará las cuatro operaciones matemáticas básicas (suma, resta, multiplicación y división), además se encenderá un *LED* en el caso de que el resultado de algunas de las operaciones sea mayor a un número de referencia (SP), este último debe ser ingresado a través de otro controlador digital. La respuesta de cada una de las operaciones debe ser visualizada a través de un display digital.

Siga los siguientes pasos para construir el VI:

1. Ingrese al programa dando doble click sobre el icono de LabVIEW en el escritorio. Una vez hecho esto aparecerá la siguiente ventana:



Figura 52. Ventana de Inicio de LabVIEW

2. Haga clic en **New VI** para empezar a construir el instrumento.

Aparecerá la siguiente ventana como se muestra en la figura:

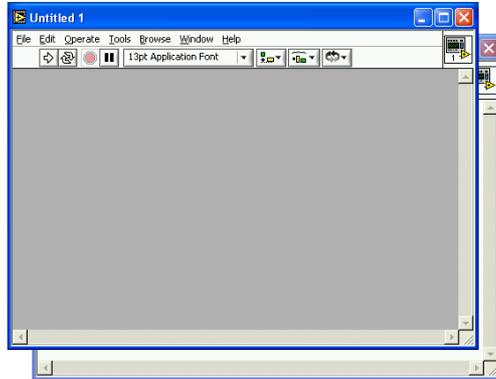


Figura 53. Ventanas de LabVIEW

3. Haga clic en Window y se abrirá el menú desplegable en el cual se escogerá la opción **Tile Left and Right** como se ve en la figura. Esta opción distribuye las ventanas de la siguiente manera, el panel frontal a la izquierda y el diagrama de bloques a la derecha.

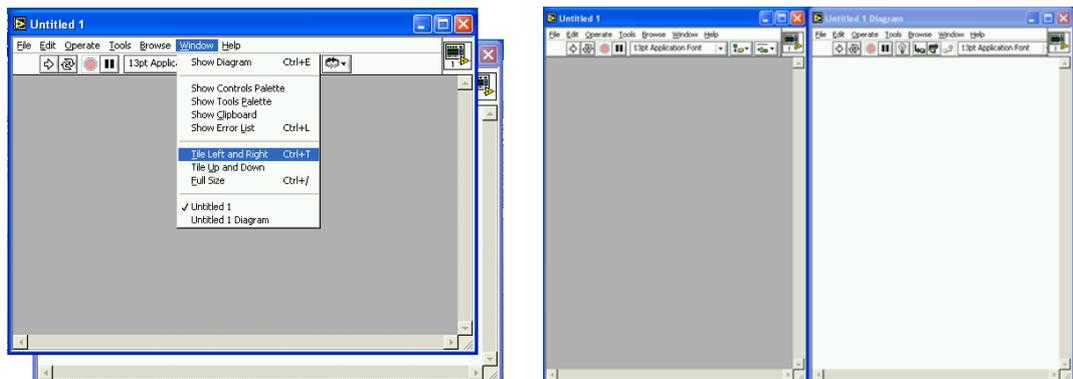


Figura 54. Despliegue de las Ventanas a Izquierda y Derecha

4. Proceda a colocar las paletas de funciones, control y herramientas. Para mayor comodidad coloque la paleta de funciones al lado inferior izquierdo de la pantalla, la paleta de controles al lado inferior derecho y la paleta de herramientas en la parte inferior central.

La paleta de controles se hace visible mientras se trabaje en el panel frontal y la paleta de funciones aparece cuando se trabaje en el diagrama de bloques, mientras que la paleta de herramientas permanece disponible para utilizarla ya sea en el panel frontal o en el diagrama de bloques.

Estas **paletas flotantes** se obtienen de la opción **Window** en la barra de menú principal.

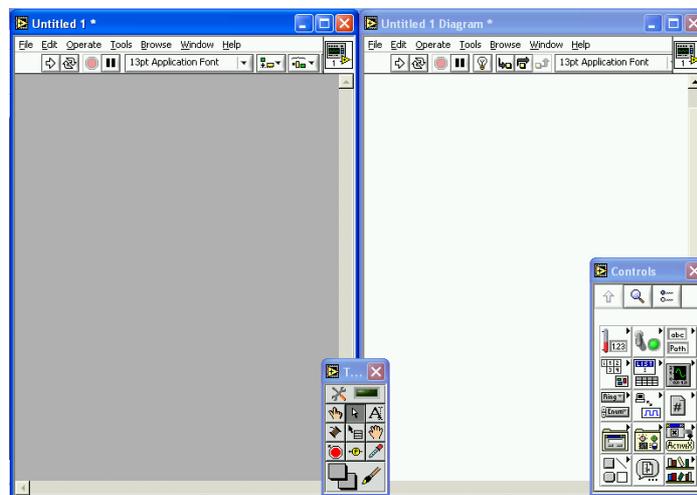


Figura 55. Ubicación de las Paletas

5. Diríjase a la paleta control y de clic en el botón **numeric**, automáticamente se desplegará la subpaleta con los distintos tipos de controladores e indicadores.
6. Seleccione tres controladores y cuatro indicadores digitales, para esto de click sobre el elemento a seleccionar y arrástrelo hasta el panel frontal, de click nuevamente para soltarlo, Haga esto elemento por elemento.

- Regrese a la paleta principal mediante el botón **up** y seleccione el botón **Boolean**. De la subpaleta tome un LED indicador y colóquelo en el panel frontal.

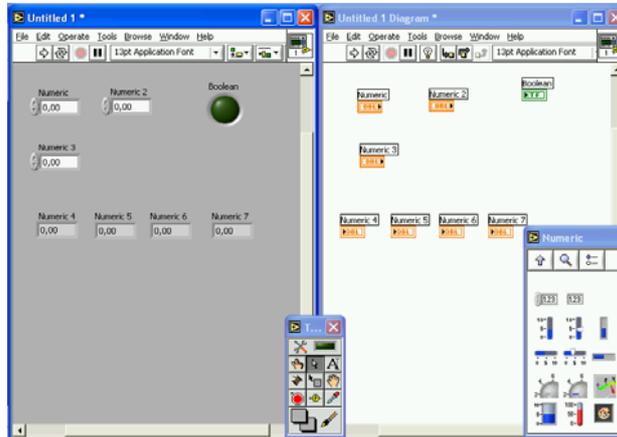


Figura 56. Organización de Elementos

- Utilice la herramienta de texto en la paleta de herramientas y cambie el nombre de la etiqueta de los elementos seleccionados en el panel frontal. Note que al hacer esto simultáneamente se cambian los nombres de las etiquetas de los correspondientes terminales en el diagrama de bloques.
- De click en el diagrama de bloques para que se visualice la paleta de funciones y seleccione de ésta el botón **numeric**.
Aparecerá la subpaleta con los distintos operadores matemáticos.
- Seleccione los elementos de suma, resta, multiplicación y división y colóquelos sobre el diagrama de bloques. Recuerde que debe hacerlo uno por uno.

11. Regrese a la paleta principal mediante el botón **up**, de clic en el botón **comparison** y seleccione el comparador **Greater? (>)**, colóquelo en el diagrama de bloques.
12. Organice los elementos del diagrama de bloques para poder conectarlos entre sí.
13. Con la herramienta de cableado de la paleta de herramientas interconecte todos los elementos como se muestra en la figura.

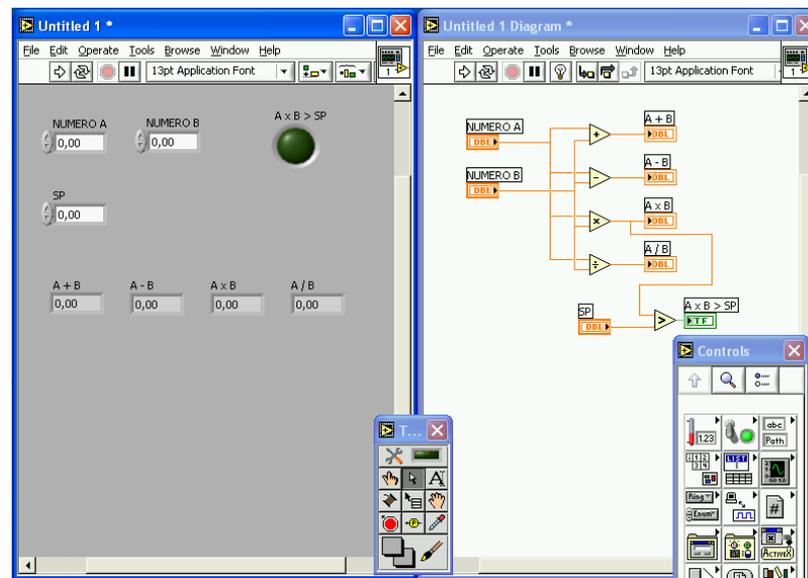


Figura 57. Cableado

14. Oprima el botón **Run Continuously** para ejecutar el VI.
15. Cambie los valores de los controladores **A**, **B** y **SP**. Observe los valores de salida en los indicadores al igual que el comportamiento del led, el cual debe encender cuando el valor de la multiplicación sea superior al valor de **SP**.
16. Salve el VI.

9. ACTIVIDADES

A continuación usted encontrará una serie de actividades que le ayudaran a entender mejor el uso de LabVIEW y le darán destreza en la programación.

Cabe anotar que la habilidad para programar en LabVIEW, aunque se adquiere con la practica, también requiere del ingenio personal del programador, ya que no existen reglas para hacer los programas sino que cada programador desarrolla la programación de la aplicación a su manera y entre mas conocimientos e ingenio tenga, mas eficientes serán los programas que éste realice.

OBJETIVO DE LAS ACTIVIDADES

Estas actividades tienen como objetivo introducirlo en la programación para que usted aprenda el comportamiento del software por medio de un VI sencillo y luego se irá incrementando la complejidad de las mismas para que usted vaya conociendo el uso de las distintas herramientas y elementos necesarios para la programación.

Por medio de estas actividades usted aprenderá a

1. Realizar simulación de procesos.
2. Utilizar bloques de PID.
3. Utilizar el puerto paralelo del PC.
4. Utilizar el puerto serial.

5. Utilizar una tarjeta de adquisición de datos

9.1. ACTIVIDAD 1

En esta actividad, se simulará el llenado de un tanque de manera incremental sin ningún tipo de control; el valor del incremento podrá ser variado por el usuario según desee a través de un controlador digital.

Para realizar esta actividad siga los siguientes pasos.

1. Cree un nuevo VI.
2. Guárdelo como Actividad 1
3. Arme el panel frontal con los siguientes elementos y nómbralos como se indica
 - Un controlador digital (Incremento)
 - Un indicador digital (Nivel Anterior)
 - Un indicador tipo tanque (Tanque)
4. Cambie la escala del tanque utilizando la herramienta de texto, escriba el valor máximo deseado (100) en la parte superior de la escala, automáticamente la escala cambia de cero al valor deseado (0-100).

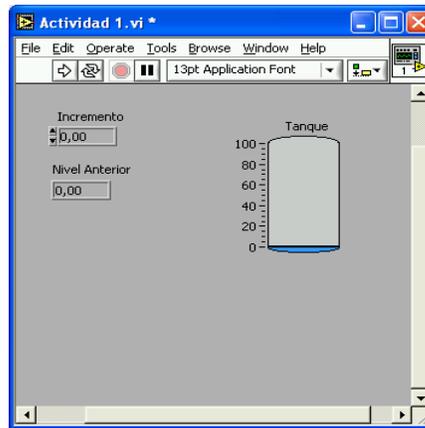


Figura 58. Actividad 1 Panel Frontal

5. En el diagrama de bloques coloque una estructura de secuencia (Sequence Structure), un sumador y un retardador de tiempo (Wait Until Next ms Múltiple)

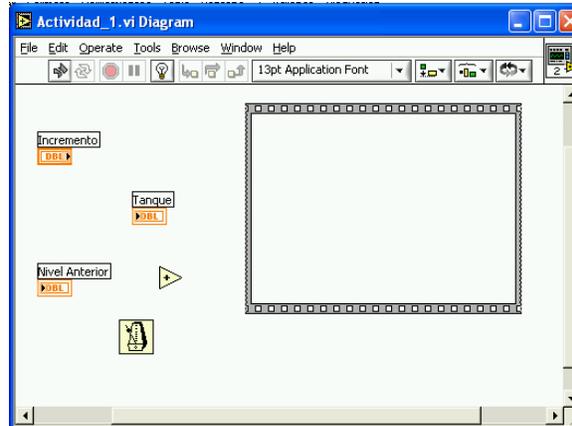


Figura 59. Actividad 1 Elementos del diagrama de bloques

6. Organice el diagrama de bloques como se muestra en la figura.

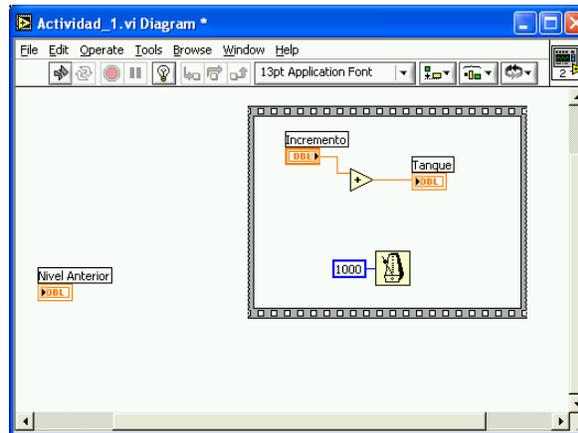


Figura 60. Organización del Diagrama A.1

Para colocar el tiempo del retardador, de clic derecho sobre éste y seleccione del menú abreviado la opción **Create/Constant**.

Al colocar el valor de 1000 este retardador hará esperar 1000 ms (1seg.) entre la ejecución de una y otra secuencia. Esto es necesario para poder visualizar la simulación; de lo contrario el proceso se hará tan rápido que usted no alcanzaría a notar los cambios.

7. De clic derecho sobre el borde de la estructura de secuencia y seleccione del menú desplegable la opción **Add Frame After**. Esta opción crea un segundo cuadro de secuencia en el cual se colocara el indicador de Nivel Anterior.

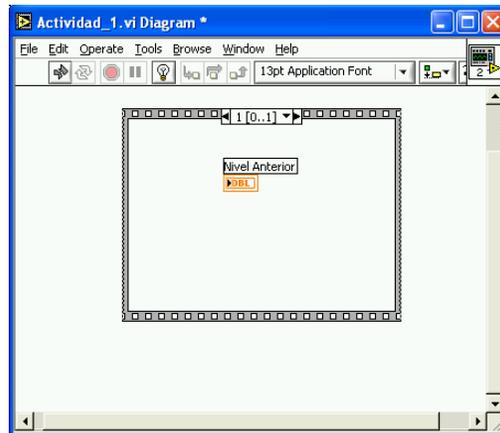


Figura 61. Cuadro de Secuencia

8. de clic en el selector de cuadro de secuencia y regrese al cuadro inicial (0)
9. De clic derecho nuevamente sobre el borde del cuadro y seleccione la opción **Add Séquense Local**. Conéctela como se indica.

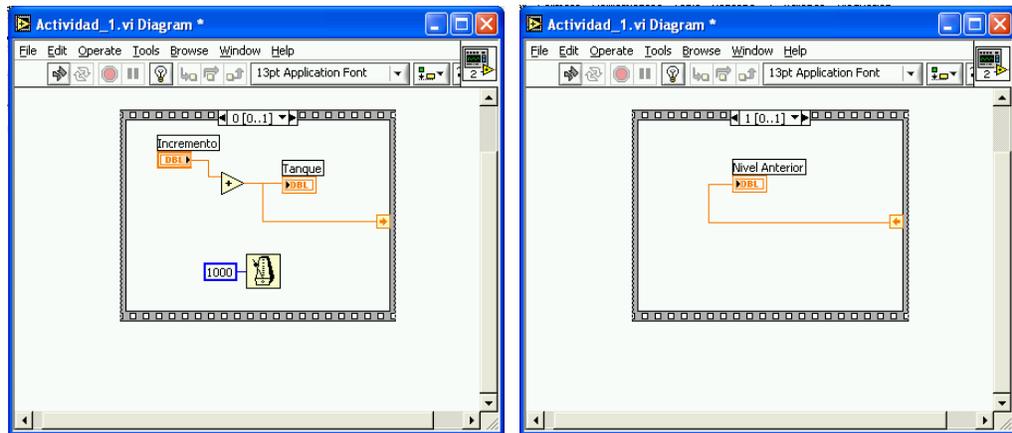


Figura 62. Conexión de Secuencia Local

Esta secuencia local sirve para pasar datos entre cuadros.

10. De click derecho sobre el terminal del indicador de Nivel Anterior y escoja del desplegable la opción **Create/Local Variable**. Saque este elemento del cuadro de secuencias (1).
11. De click derecho sobre esta variable local y escoja del desplegable la opción **Change To Read**
12. Regrese nuevamente al cuadro de secuencia cero (0) e ingrese esta variable y conéctela como se indica.

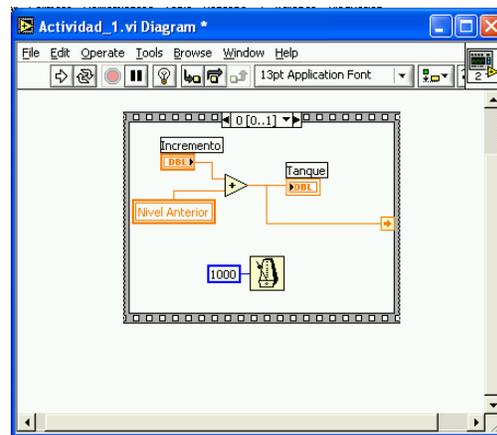


Figura 63. Creación de Variable Local

13. Guarde los cambios
14. Regrésese al panel frontal y corra el VI continuamente.
15. Cambie los valores del incremento y observe lo que sucede.
16. Regrese nuevamente al diagrama de bloques y haga click en la opción **Highlight Execution**. De la barra de herramientas y observe el flujo de datos del programa.

9.2. ACTIVIDAD 2

En esta actividad, se simulará el llenado de un tanque de manera incremental; pero a diferencia de la actividad anterior se ejerce control para detener el llenado en un valor deseado (SP), el cual puede ser elegido por el usuario al igual que el valor del incremento a través de controladores digitales.

Para realizar esta actividad siga los siguientes pasos:

1. Cree un nuevo VI.
2. Guárdelo como Actividad 2
3. Organice el panel frontal como se muestra en la figura.

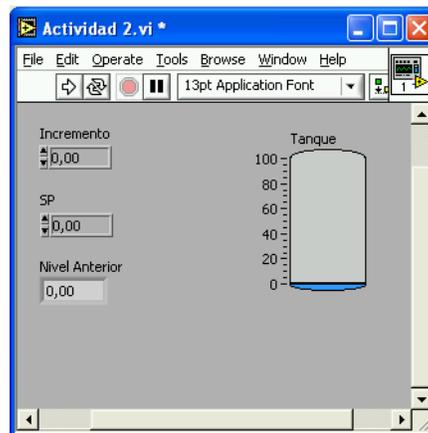


Figura 64. Actividad 2 Panel Frontal

4. Organice el diagrama de bloques como se muestra en la figura.

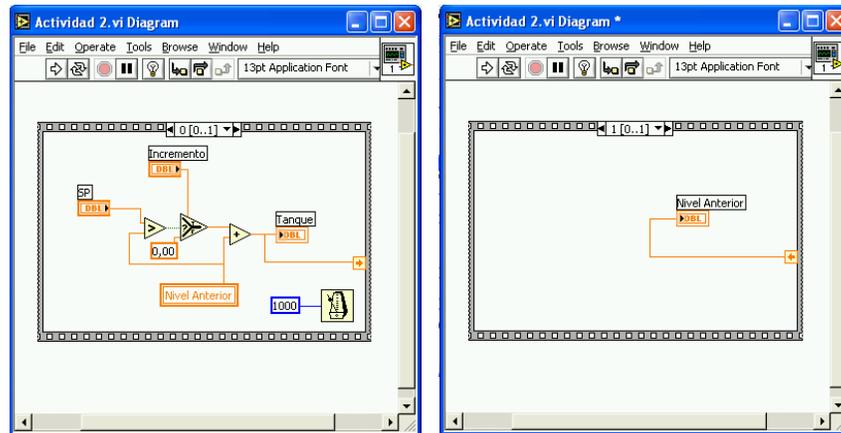


Figura 65. Organización y Cableado del Diagrama de Bloques A.2.

La función de comparación mayor que ($>$) compara el valor del nivel del tanque con el valor del nivel deseado.

La función de comparación **select** entrega el valor conectado al terminal de entrada **t** o **f** dependiendo del valor booleano conectado al terminal de la incógnita **?**.

Mientras el nivel del tanque sea inferior al nivel deseado, la función select permitirá el paso del incremento al sumador; de lo contrario accederá el paso del valor constante cero (**0,00**), lo cual hará que el nivel del tanque sea igual al ultimo nivel almacenado.

5. Guarde el VI
6. Corra el VI continuamente.
7. Ingrese el valor del nivel deseado en el controlador SP.
8. Ingrese el Valor del incremento.

9. Observe como el llenado del tanque se detiene al llegar al valor deseado (SP).

9.3. ACTIVIDAD 3

En esta actividad, se simulara el llenado de un tanque de manera proporcional y se ejercerá control para detener el llenado en un valor deseado (SP), el cual puede ser elegido por el usuario al igual que el valor de la variable de proporcionalidad (K_p). Además se instalara una señal de alarma que nos indique que el tanque alcanzó el nivel deseado.

Siga los siguientes pasos para llevar acabo esta actividad.

1. Cree un nuevo VI.
2. Guárdelo como Actividad 3
3. Organice el panel frontal como se muestra en la figura.

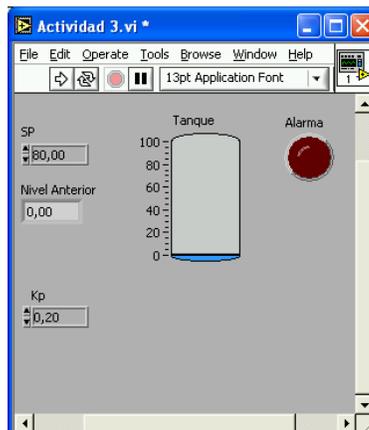


Figura 66. Actividad 3 Panel Frontal

- Arme el diagrama de bloques como se muestra en la figura. Recuerde los pasos de la actividad 1.

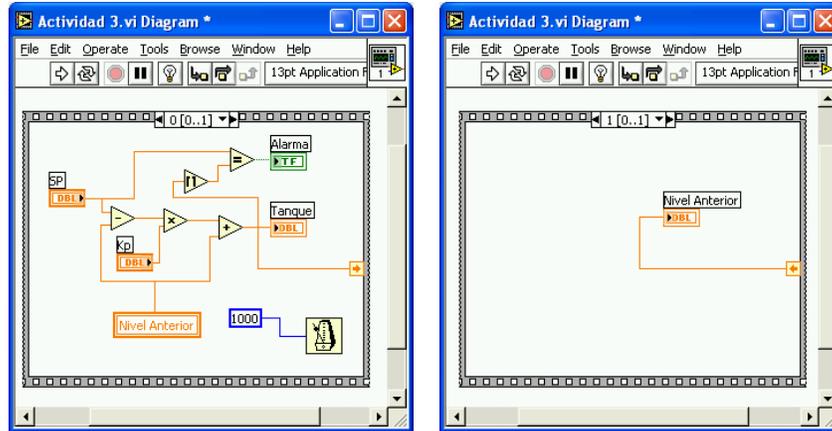


Figura 67. Organización y Cableado Diagrama de Bloques A.3.

- Salve el VI.
- Corra el VI continuamente.
- Ingrese el valor del nivel deseado en el controlador SP.
- Ingrese un Valor al controlador Kp entre 0 y 1.
- Observe el comportamiento del programa.

La función de resta (-) saca la diferencia entre el valor deseado y el nivel del tanque

La función por (x) multiplica esta diferencia por la constante de proporcionalidad Kp y su resultado será el nuevo incremento que se agregara al indicador del tanque.

La función de comparación igual a (=) nos da una señal booleana la cual encenderá una señal de alarma cuando el nivel del tanque sea igual al valor deseado.

La función Round To Infinity redondea la entrada al entero más próximo. Esto es necesario ya que la aproximación de manera proporcional toma muchas iteraciones y tiende al infinito haciendo el error cada vez más pequeño pero nunca igual a cero.

9.4. ACTIVIDAD 4

En esta actividad, se simulara el proceso de llenado de un tanque elevado el cual depende tanto del nivel de agua almacenado en el tanque inferior como del flujo de consumo de los usuarios.

Deben cumplirse las siguientes consideraciones

- La bomba no funcionara a menos que
 - a) El nivel del tanque inferior sea superior a 20
 - b) El nivel del tanque elevado sea menor de 90
 - c) La válvula de paso se encuentre abierta.

- El nivel del tanque inferior (Acueducto) variara de acuerdo con:
 - a) El flujo de salida o flujo de bombeo
 - b) El flujo de entrada o despacho

- El nivel del tanque elevado (Usuario) variara de acuerdo a:

- a) El flujo de entrada o flujo de bombeo
- b) El flujo de salida o consumo.

Siga los siguientes pasos para llevar a cabo esta actividad.

1. Cree un nuevo VI
2. Organice el panel frontal como se muestra en la figura.

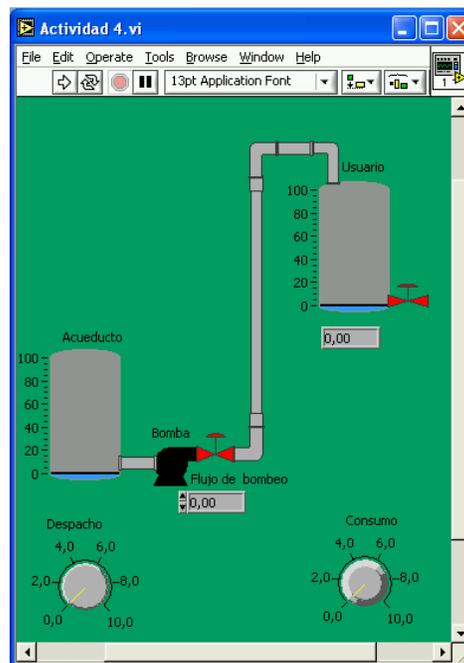


Figura 68. Actividad 4 Panel Frontal

3. Arme el diagrama de bloques según se indica

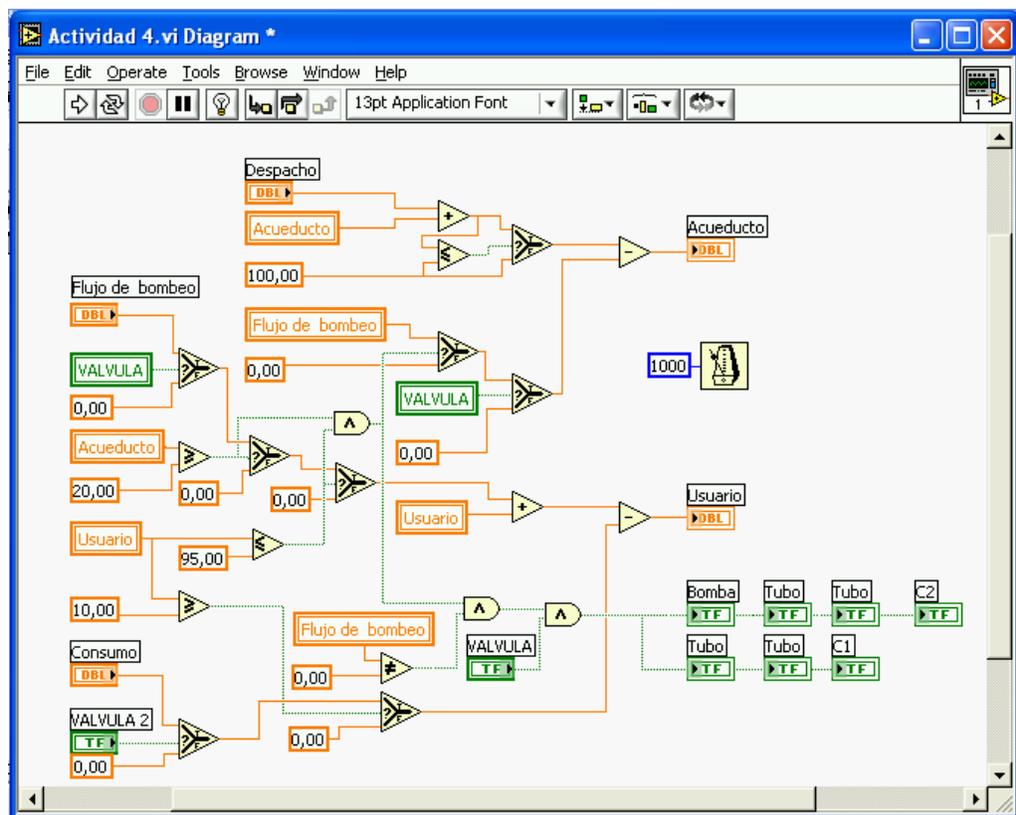


Figura 69. Organización y Cableado del Diagrama de Bloques A.4.

4. Guarde el VI como actividad 4
5. Corra el VI continuamente
6. Dele valores a los flujos de despacho, de bombeo y de consumo.
7. De click sobre la válvula de la bomba y observe.
8. De click sobre la válvula del tanque de los usuarios y observe.
9. Varíe los valores de los flujos y observe lo que sucede.

9.5. ACTIVIDAD 5

El objetivo primordial de esta actividad es el envío de datos a través del puerto paralelo del PC.

A continuación se elaborara un VI para que encienda uno a uno los LEDs correspondientes a cada una de las posiciones del numero binario de ocho bits. Recuerde que los valores de las posiciones, cuando existe un uno en ésta, en un numero binario de ocho bits son, de izquierda a derecha (del LSB al MSB), son 2^0 (1), 2^1 (2), 2^2 (4), 2^3 (8), 2^4 (16), 2^5 (32), 2^6 (64), 2^7 (128).

Al ejecutarse este VI verá encenderse los LEDs desde el que representa el valor menos significativo hasta el que representa el valor mas significativo, uno a uno.

Para mayor información sobre conexiones y visualización de datos por el puerto paralelo del PC, ver anexo A.

Siga los siguientes pasos para la elaboración de este VI.

1. coloque en el panel frontal los siguientes elementos.
 - Dos indicadores digitales
 - Un *array*
 - Un *Round LED*
2. Coloque el Round LED sobre el Array y luego arrastre la esquina inferior del array hasta desplegar ocho LEDs (lo equivalente al numero de Bits).
3. Nombre cada uno de los elementos como se indica en la figura.

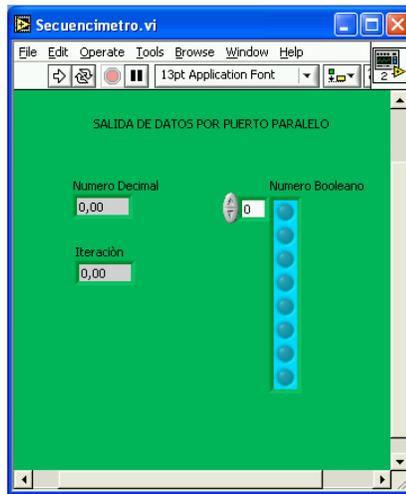


Figura 70. Actividad 5 Panel Frontal

4. Además de los terminales que ya se encuentran en el Diagrama de Bloques, coloque los siguientes elementos.
 - Un **For Loop**. (paleta de Funciones/*Structure/For Loop*)
 - Un **Out Port**.(paleta de Funciones/*Avanced/Port I/O /Out Port*)
 - Un **Scale By Power Of 2**. (Paleta de funciones/*Numeric/Scaled By Power Of 2*).
 - Un **Number To Boolean Array**. (paleta de *funciones/Bolean/Number To Bolean Array*).
 - Un **Wait Until Next ms Multiple**. (Paleta de funciones/*Time & Dialog/Wait Until Next ms Multiple*).
5. Organice el diagrama de bloques como se muestra en la figura

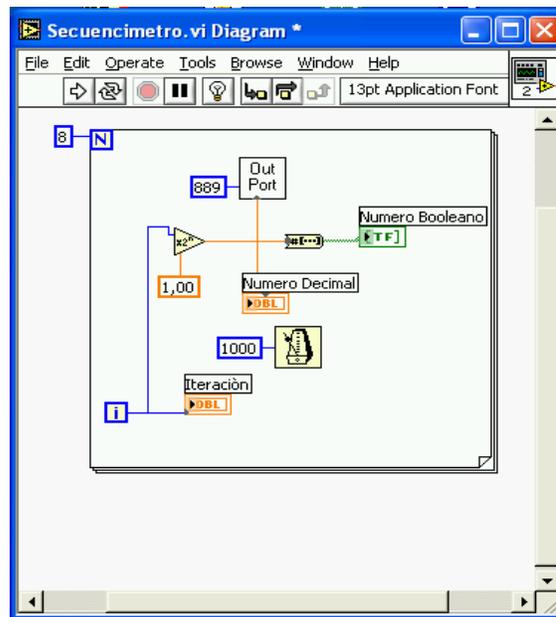


Figura 71. Organización y Cableado del Diagrama de Bloques A.5.

6. Salve el VI.
7. Corra continuamente el VI.
8. Observe los LEDs a la salida del puerto y compare con el indicador del panel frontal.

9.6. ACTIVIDAD 6

El objetivo primordial de esta actividad es la recepción de datos a través del puerto paralelo del PC.

En esta actividad asumiremos que existen dos sensores de presencia de personas que detectan la entrada de personas a dos áreas restringidas, enviando en este caso, un 1 (uno) lógico por el pin 15 del puerto paralelo por el sensor del área 1 y un 1 (uno) lógico por el pin 13 del puerto paralelo por el sensor del área 2.

Se construirá un VI que indique la presencia de personas ya sea en el área 1, en el área 2 o en ambas.

Para mayor información sobre conexiones y entrada de datos por el puerto paralelo del PC, ver anexo A.

Para construir este VI siga los siguientes pasos.

1. Cree un nuevo VI.
2. Coloque los siguientes elementos.
 - Tres **Square LED** (paleta de control/*Boolean/Square LED*).
 - Un **Display Digital** (paleta de controles/*Numeric/Digital Indicator*).
 - Un **Array** (paleta de controles/*Array & Cluster/Array*).
 - Un **Round LED** (paleta de controles/*Boolean/Round LED*).
3. Arrastre la esquina inferior del Array hasta obtener ocho LEDs, (uno por cada pin de entrada del puerto).
4. Organice el panel frontal como se indica en la figura.



Figura 72. Actividad 6 Panel Frontal

5. Además de los terminales que ya se encuentran en el diagrama de bloques, coloque los siguientes elementos.
- Una compuerta **And** (paleta de *funciones/Bolean/And*).
 - Un **Number To Boolean Array**. (paleta de *funciones/Bolean/Number To Boolean Array*).
 - Un **Wait Until Next ms Multiple**. (Paleta de *funciones/Time & Dialog/Wait Until Next ms Multiple*).
 - Un **In Port**. (paleta de *Funciones/Avanced/Port I/O /In Port*).
 - Tres comparadores de igualdad (paleta de *funciones/Comparición/Equal?*).
6. Organice el Diagrama de bloques como se muestra en la figura.

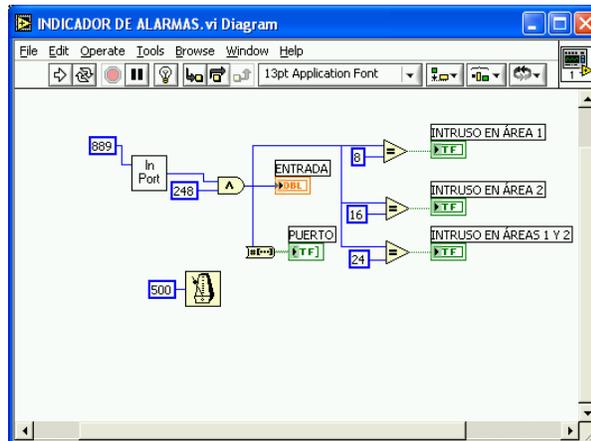


Figura 73. Organización y Cableado del Diagrama de Bloques A.6.

7. Guarde el VI.
8. Corra el VI continuamente.
9. Asegúrese de que todos los pines de entrada estén conectados a un cero lógico.
10. Ponga un uno lógico (5Vdc) en el pin 15 del puerto y observe lo que sucede en el diagrama de bloques.
11. Repita el paso anterior para el pin 13.
12. Ponga un uno lógico (5V dc) en ambos pines (13 y 15) y observe.

9.7. ACTIVIDAD 7

Esta actividad tiene como objetivo el envío y recepción de datos a través del puerto serial del PC.

En ésta actividad se necesitaran dos computadores de los cuales uno enviara los datos y el otro los recibirá.

Se construirá un VI (computador 1) que envíe dos datos (Nivel y Temperatura) a través del puerto serial y otro VI (computador 2) que reciba dichos datos y los represente en el panel frontal.

Para construir el VI de envío de datos (computador 1), realice lo siguiente.

1. Cree un nuevo VI.
2. Arme el panel frontal como se indica en la figura.

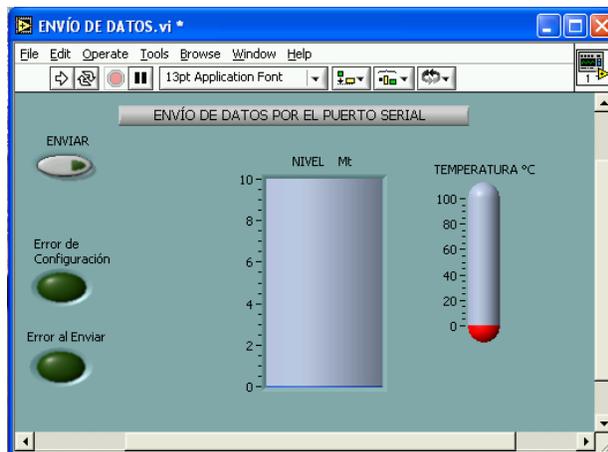
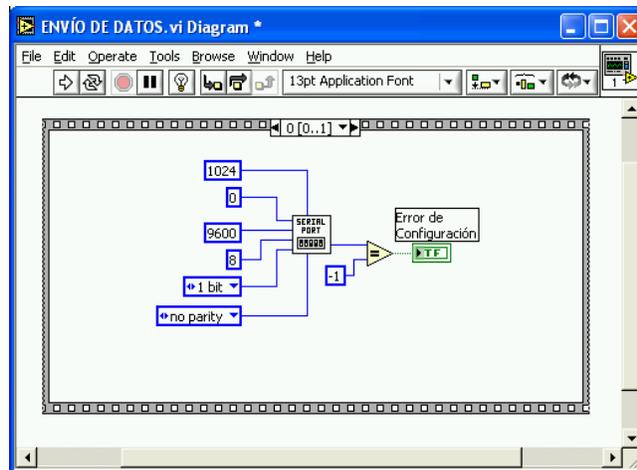


Figura 74. Panel Frontal para Envío

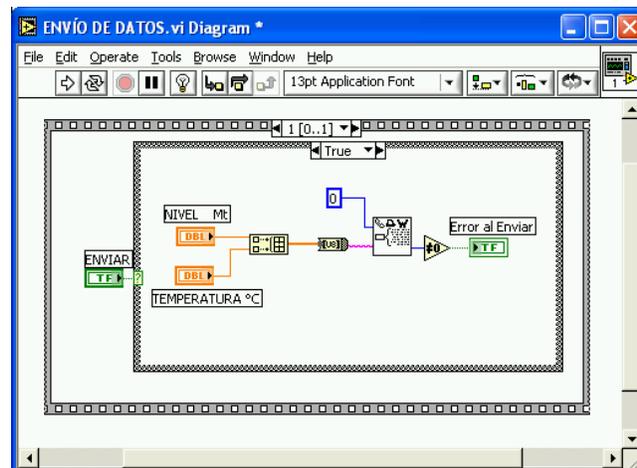
3. Coloque los siguientes elementos en el diagrama de bloques.
 - Un **SERIAL PORT Init .vi.** Ubicado en la paleta de funciones/Instrument I/O / I/O *Compatibility*/Serial *Compatibility*/SERIAL PORT Init.vi.
 - Un **Build Array** ubicado en la paleta de funciones/Array/Build Array
 - Un **Byte Array to String** se encuentra en la paleta de funciones/String/String/Array/Path Conversión/Byte Array to String

- Un **Serial Port Write** se encuentra en la paleta de funciones/*Instrument I/O / I/O Compatibility/Serial Compatibility/Serial Port Write. vi*

4. Arme el diagrama de bloques como se indica en la figura 75.



a. Configuración del Puerto Serial



b. Envío de Datos

Figura 75. Organización y Cableado del Diagrama de Bloques para Envío

5. Guarde el VI.

Para construir el VI de Recepción de datos (computador 2), realice lo siguiente.

1. Cree un nuevo VI.
2. Arme el panel frontal Como se muestra en la figura 76.

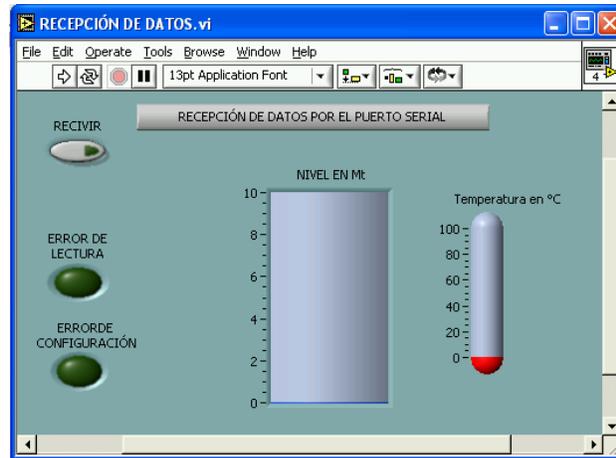
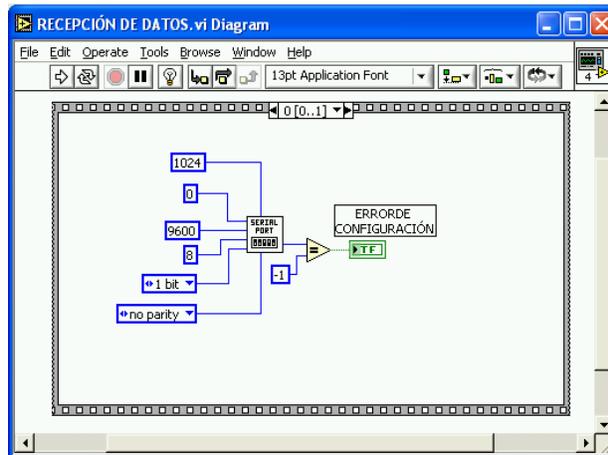


Figura 76. Panel Frontal para Recepción

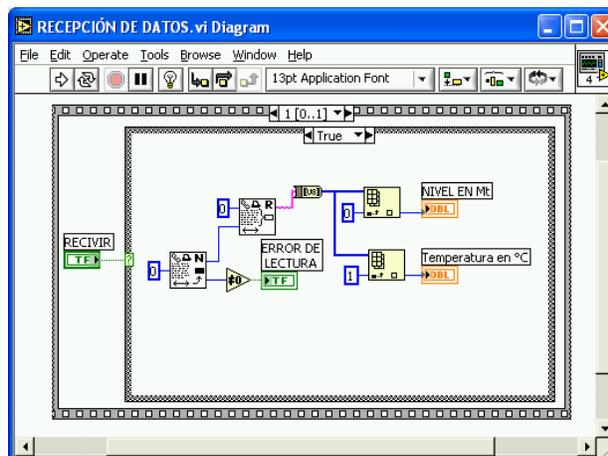
3. Coloque en el Diagrama de Bloques los siguientes elementos.
 - Un **SERIAL PORT Init.vi** Ubicado en la paleta de funciones/Instrument I/O / I/O *Compatibility*/Serial *Compatibility*/SERIAL PORT Init.vi.
 - Un **Bytes at Serial Port.vi** Ubicado en la paleta de funciones/Instrument I/O / I/O *Compatibility*/Serial *Compatibility*/Bytes at Serial Port.vi
 - Un **Serial Port Read.vi** Ubicado en la paleta de funciones/Instrument I/O / I/O *Compatibility*/Serial *Compatibility*/Serial Port Read.vi
 - Un **String to Byte Array** se encuentra en la paleta de funciones/*String*/*String*/*Array*/*Path Conversión*/ *String to Byte Array*

- dos **Index Array** Se encuentran en la paleta de funciones/Array/Index Array

4. Construya el diagrama de bloques según se muestra en la figura 75.



a. Configuración del puerto serial



b. Recepción de Datos

Figura 77. Organización y Cableado del Diagrama de Bloques para Recepción

5. Guarde el VI

Para ejecutar esta actividad siga los siguientes pasos.

1. Cree ambos VIs en sus computadores correspondientes

2. Conecte ambos computadores a través del puerto serial (ver Anexo 2)
3. Corra continuamente ambos VIs.
4. Cambie manualmente los valores de nivel y temperatura en el VI 1 (Envío de Datos)
5. Observe como se visualizan los datos en el VI 2 (Recepción de Datos).

NOTA: Para mayor información sobre el uso de los distintos elementos utilizados en esta actividad, de click derecho sobre el elemento de su interés y seleccione la opción **Help**

9.8. ACTIVIDAD 8

Esta actividad tiene como objetivo el envío y recepción de datos a través del puerto serial del PC y la realización de un control PID.

En esta actividad se simulara el proceso del llenado de un tanque a través del control **PID** de posición de una válvula.

Para esto se comunicaran entre si dos computadores de los cuales uno hará el papel de la planta (tanque y válvula), mientras que el otro será el controlador de posición de la válvula el cual recibe el valor del nivel del tanque proveniente de la “planta” y desarrolla el calculo de control para luego enviar la señal de apertura o cierre de dicha válvula a través del puerto serial del computador.

Para la realización de esta actividad se deben realizar dos VI, uno realizará la función de planta o actuador en el computador 1 y el otro hará la función de controlador PID en el computador 2.

Para construir el VI actuador (computador 1) siga los siguientes pasos:

1. Cree un nuevo VI.
2. Guárdelo con el nombre de **actuador**.
3. Construya el panel frontal como se muestra en la figura 78.

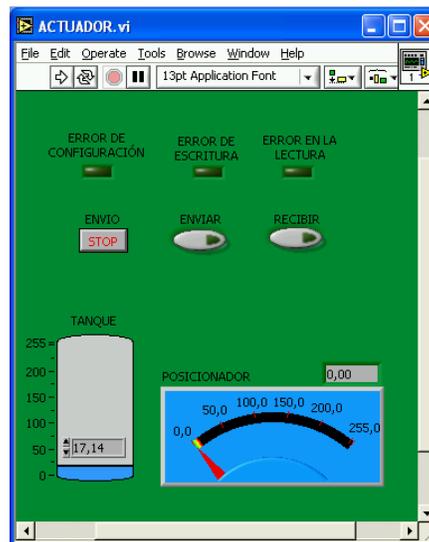
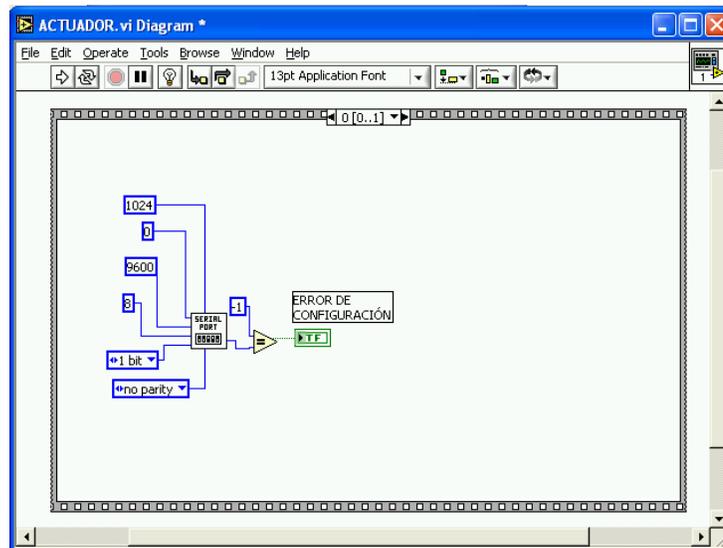
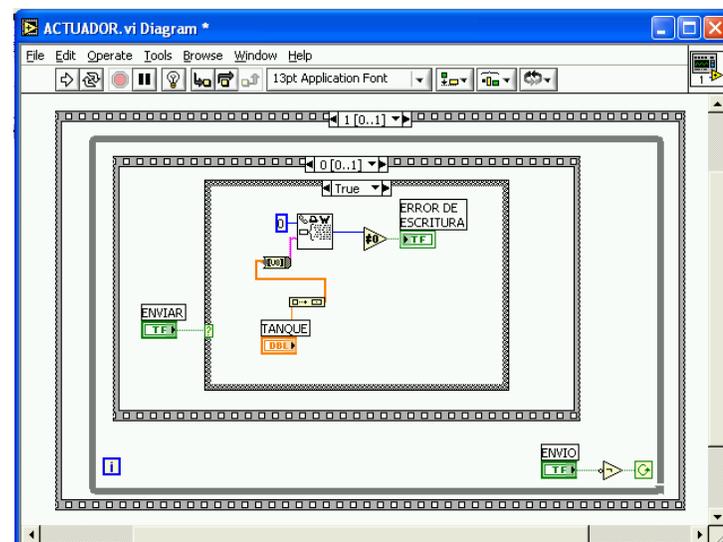


Figura 78. Actividad 8 Panel Frontal actuador

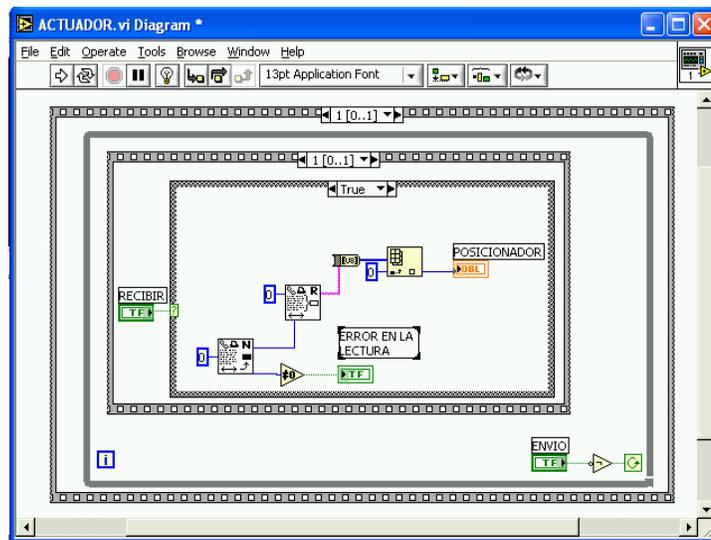
4. Construya el diagrama de bloque como se muestra a continuación



a. Configuración del Puerto Serial



b. Envío de Datos



c. Recepción de Datos

Figura 79. Organización y Cableado del Diagrama de Bloques Actuador

5. Salve los cambios.

Para construir el VI controlador (computador 2) realice lo siguiente:

1. Cree un nuevo VI.
2. Guárdelo con el nombre de **controlador**.
3. Construya el panel frontal como se muestra en la figura 80.

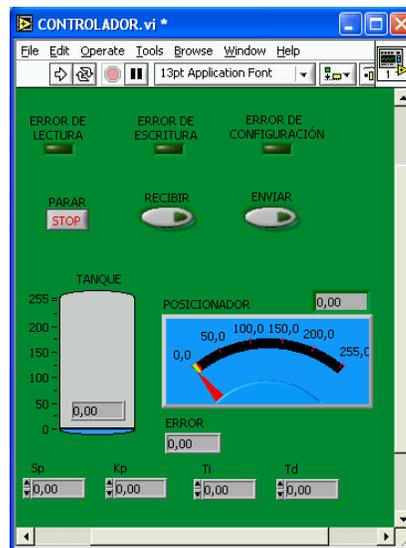
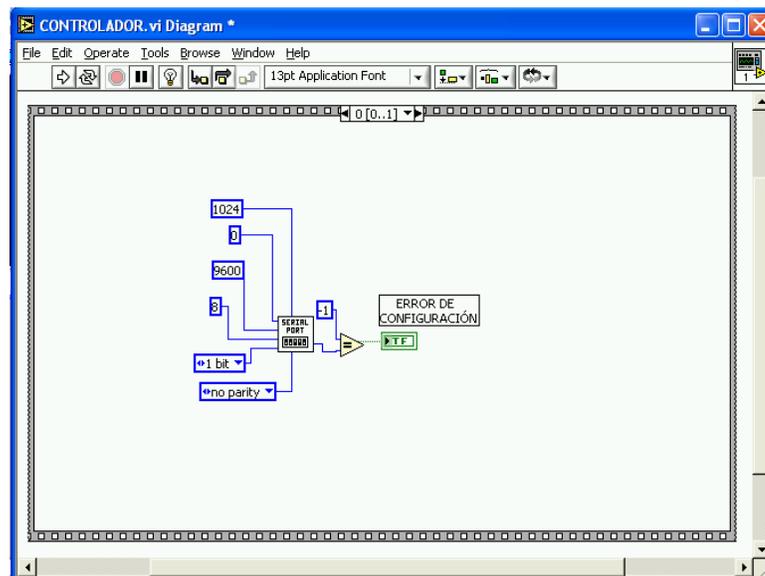
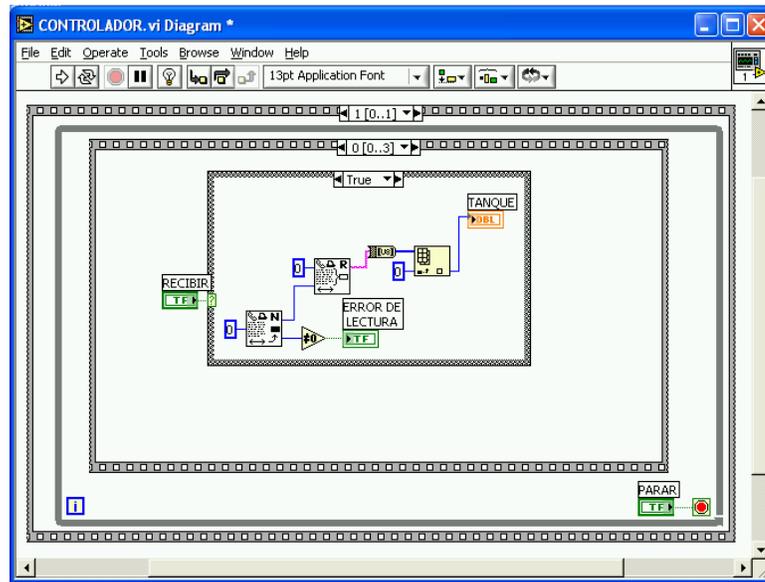


Figura 80. Actividad 8 Panel Frontal controlador

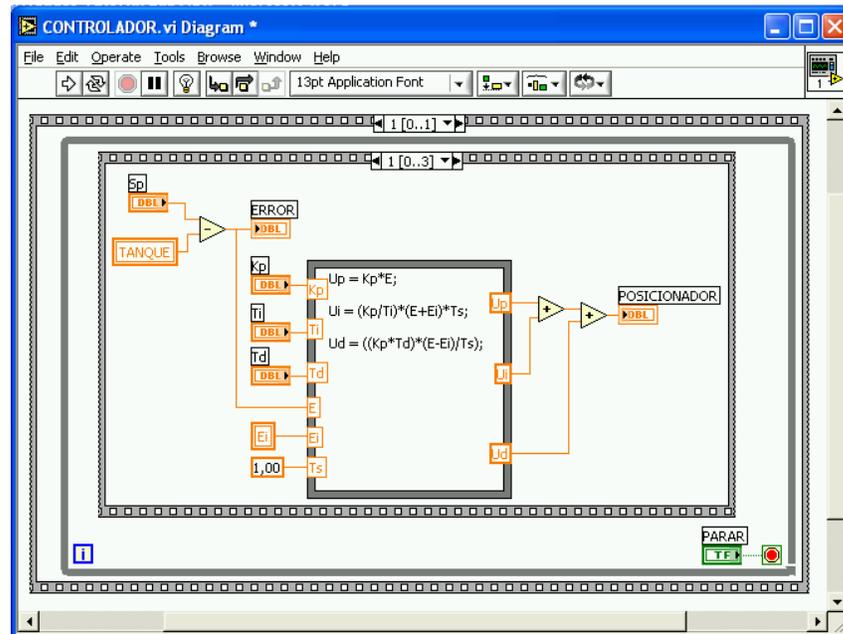
4. Construya el diagrama de bloque como se muestra a continuación



a. Configuración del Puerto Serial



b. Recepción de Datos



c. Ecuaciones de Control

Para llevar a cabo esta actividad conecte los dos computadores entre si, a través de los puertos seriales de los mismos. Ver anexo B.

Corra continuamente ambos VIs y cambie los valores del nivel del tanque en el VI actuador; observe el comportamiento del controlador y su respuesta sobre la posición de la válvula (indicador).

9.9. ACTIVIDAD 9

El objetivo de esta actividad es recibir un dato análogo a través de una tarjeta de adquisición de datos.

Este VI adquiere un simple valor de entrada análoga desde un canal determinado.

Para configurar la tarjeta siga los siguientes pasos.

1. Inserte la tarjeta en el puerto correspondiente. (ISA, PCI, etc.)

Recuerde que debe hacerlo con el computador apagado.

2. Encienda el computador. (Windows automáticamente reconocerá el nuevo hardware y LabVIEW proporcionara el software de instalación (*Driver*).
3. haga doble click sobre el icono de *Measurement & Automations Explorer* que se encuentra sobre el escritorio.



Figura 82. Icono para software de configuración de tarjeta.

4. una vez realizado el paso anterior le aparecerá la siguiente ventana, en la cual debe presionar el botón F5 de su teclado y luego presione OK sobre el botón de dicha ventana.

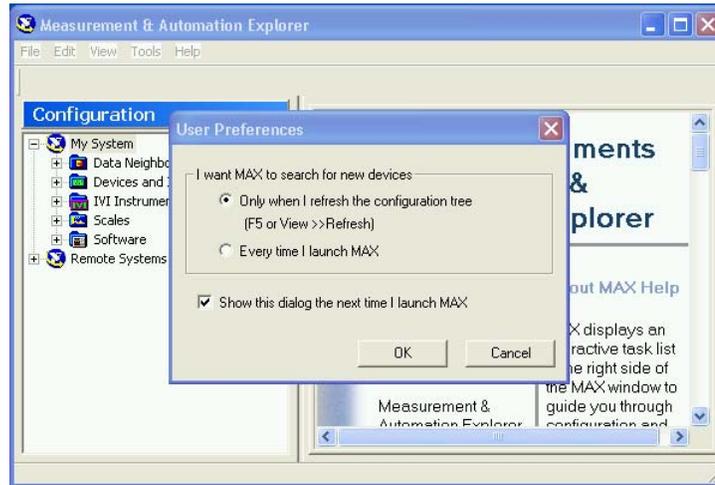


Figura 83. Caja de Dialogo para Identificación de la Tarjeta

5. De click derecho sobre la opción **Data Neighborhood** de la ventana de configuración y seleccione **Create New** para crear un nuevo canal virtual.
6. En la ventana que le aparece seleccione la opción **Virtual Channel** y haga click en el botón **Finish**.

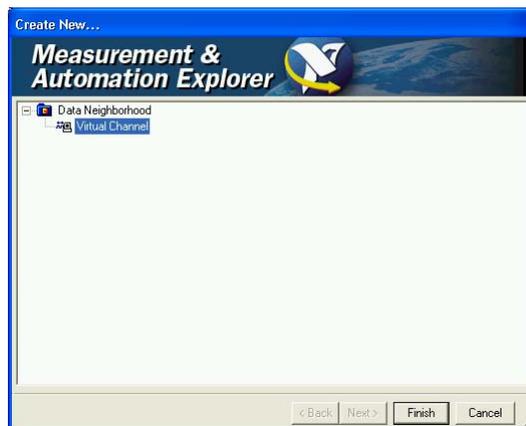


Figura 84. Ventana para Abrir el asistente de creación de un nuevo canal.

7. Siga los pasos del asistente de creación de un nuevo canal.



Figura 85. Asistente de creación de un nuevo canal.

8. Cierre la ventana de *Measurement & Automations Explorer*.

Una vez configurada la tarjeta y creado el canal de entrada análoga necesario para este VI. siga los siguientes pasos.

1. construya el panel Frontal Como se indica en la siguiente figura.

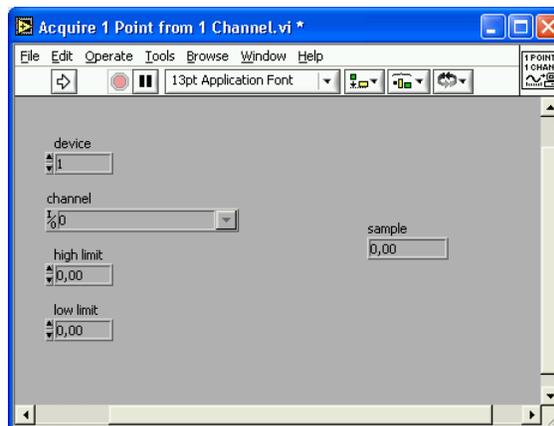


Figura 86. Actividad 9 Panel Frontal

2. arme el Diagrama de Bloques tal como se indica.

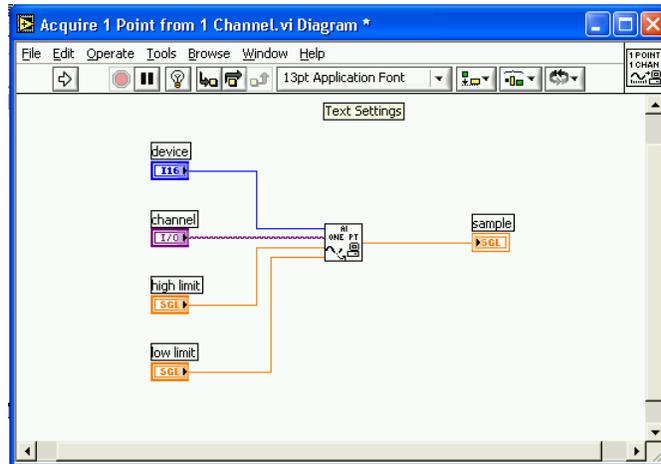


Figura 87. Organización y Cableado del Diagrama de Bloques A.9.

3. Salve el VI.
4. Ingrese una señal por el canal escogido en la configuración.
5. Corra el VI.

CONCLUSIONES

Con la finalización de esta monografía, se cuenta con una nueva herramienta de trabajo para que los estudiantes aprendan a interactuar con el Software de adquisición de datos LabVIEW 6.1 el cual a pesar de ser un software con lenguaje grafico necesita de un tutorial básico para introducir al estudiante en la programación y conocer sus herramientas.

Está monografía esta enfocada a incentivar a la comunidad estudiantil y en general a todo aquel que desee aprender el manejo del *Software* para realizar futuros proyectos encaminados a la automatización, control e instrumentación de problemas didácticos o problemas complejos de tipo industrial.

Finalmente, estamos satisfechos con el trabajo realizado, debido a la cantidad de conocimientos adquiridos durante su desarrollo y de entregar a la Institución una herramienta muy útil para la enseñanza.

BIBLIOGRAFIA

- Juan Contreras M. Modulo Sistemas Scadas. Cartagena de Indias, Abril de 2.003
- National Instruments. The Measurement and Automation Catalog. 2.002
- National Instruments. Tutorial Software LabVIEW 6.1.

ANEXO A

COMUNICACIÓN POR EL PUERTO PARALELO

El puerto paralelo, o puerto de la impresora, nos permite comunicar el PC con el mundo exterior enviando o recibiendo señales digitales.

CONECTOR DB25

El conector DB25 es el comúnmente empleado para comunicar el PC con la impresora. La siguiente figura muestra los terminales del conector DB25 “macho”, que se utilizará para conectar al conector DB25 “hembra” que trae la CPU del PC.

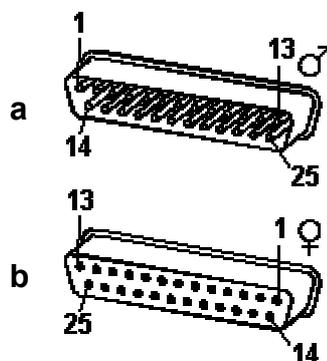


Figura 1. a. Conector DB25 macho b. Conector DB25 hembra

ENVIO DE DATOS POR EL PUERTO PARALELO DEL PC.

Para el envío de datos por el puerto paralelo del PC se emplean los pines 2 al 9 del terminal DB-25. para visualizar la información enviada se debe conectar el

circuito mostrado en la figura 2. El dato enviado corresponderá a un byte (numero binario o palabra de ocho bits), para lo cual se debe saber que por el pin 2 se envía el bit menos significativo (LSB) y por el pin 9 el bit más significativo (MSB).

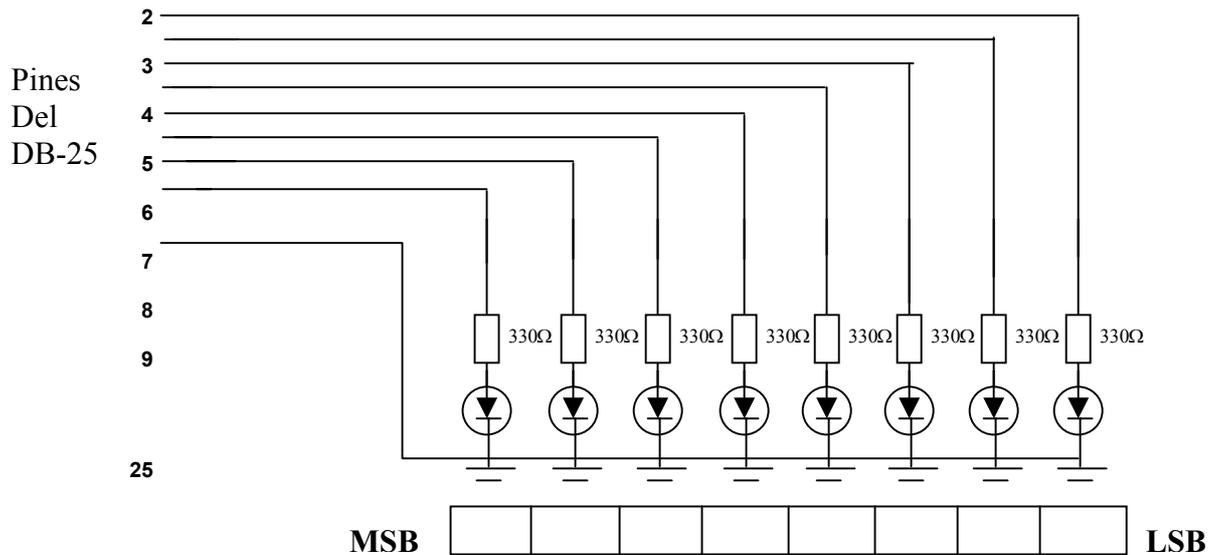


Figura 2. Circuito Para probar el envío de datos por el puerto Paralelo del PC

ENTRADA DE DATOS POR EL PUERTO PARALELO DEL PC.

El puerto paralelo del PC permite recibir información digital por los pines 11, 10, 12, 13, 15. del conector DB-25, la cual interpreta como un numero binario de ocho bits, donde el pin 11 representa la entrada del bit más significativo (MSB).

La figura 3 muestra una comparación entre la distribución de un numero binario de ocho bits y la asignación de los pines del puerto paralelo para entrada de datos

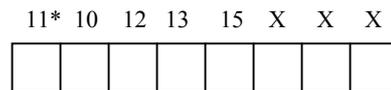


Figura 3. Pines del puerto paralelo del PC para entrada de datos

Se puede apreciar que en realidad el puerto paralelo del PC tiene disponible para recibir información externa, solo cinco (5) pines y no ocho (8). Los tres bits menos significativos no tienen pines asignados, por lo tanto, cada PC reconstruye el número binario asignando un valor de unos (1) o cero (0) a estos.

Es muy importante aclarar que el pin 11 tiene lógica negada. Esto quiere decir que si se conecta en este pin un uno lógico (5V dc) el PC lo toma como cero lógico o Viceversa.

Para un mayor entendimiento, construya el circuito de la figura 4 y conéctelo al PC. Se puede apreciar que al estar los pulsadores en la posición mostrada en la figura 4, se le están colocando a todos los pines un cero lógico ya que quedan conectados a referencia (cero voltios) por medio de la resistencia de 10K, a excepción del pin 11 que queda conectado a 5 V (1 lógico), pero como tiene lógica negada el PC lo tomara también como cero lógico.

Para evitar depender del PC en que se esta trabajando, haremos que estos tres bits menos significativos (X X X) tomen el valor de 000.

Recordemos que la compuerta lógica AND, tiene la siguiente tabla de verdad.

ENTRADAS	SALIDAS
0	0
0	1
1	0
1	1

Es decir, que si una de las entradas es cero la salida será siempre cero.

Si queremos que los bits representados por X X X sean siempre cero, hacemos la operación AND de la entrada binaria con el valor binario 1 1 1 1 1 0 0 0, que equivale al decimal 248.

De esta manera, sin importar cual sea el PC donde estemos trabajando, siempre que queramos recibir información por el puerto paralelo del PC hacemos una operación AND entre la entrada por el puerto y el valor decimal 248.

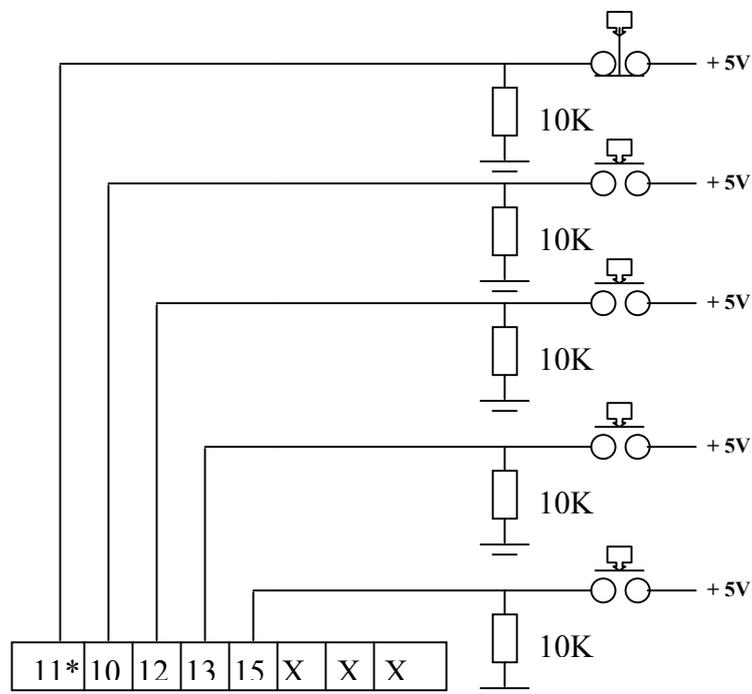


Figura 4. Circuito para adquisición de datos por el puerto paralelo

ANEXO B

COMUNICACIÓN A TRAVÉS DEL PUERTO SERIAL

Se transmite la información por un puerto que puede ser el COM1 o el COM2 de forma serial, osea a través de un solo cable, y cada bit pasa uno tras otro a alta velocidad. Para la comunicación entre computadores se establece un protocolo común para que la información sea entendida por ambos. Se debe definir el tamaño de los BUFFER para almacenar datos mientras se realiza la comunicación. También se debe definir si hay Handshaking, el cual consiste en que el que recibe cuando valla a tener lleno el buffer de información mande una instrucción (si es por software <Ctrl.-Q>, por hardware una línea). LabVIEW cuenta con funciones para iniciar, escribir y leer el puerto serial.

Como construir el cable *null modem* (serie). Con el cable de 7 hilos, suelde entre los dos conectores de pin a pin según se indica en figura 1.

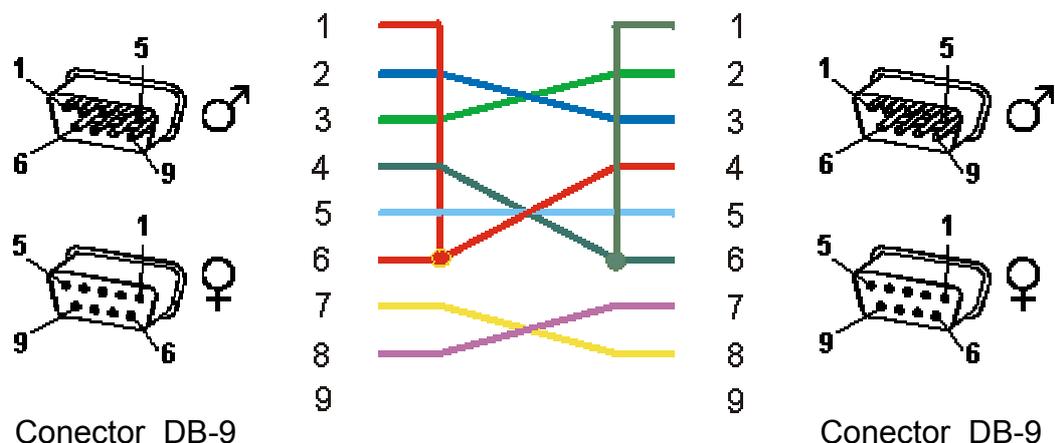


Figura 1. Interconexión de conectores para comunicación por puerto serial RS232

ANEXO C

COMUNICACIÓN CON TARJETAS DAQ

Las tarjetas DAQ son tarjetas insertables que permiten la entrada y salida de datos del computador a otros aparatos, donde se conectan sensores y actuadores, para interactuar con el mundo real. Los datos que entran y salen pueden ser señales digitales o análogas o simplemente conteos de ocurrencias digitales, tanto de entradas como de salida.

Las tarjetas se comportan como si fueran un puerto más en el computador, y poseen todo un protocolo y sistema de manejo, por lo que entender cada tarjeta. Como su funcionamiento, al igual que cualquier instrumento, requiere de tiempo y cuidado.

Existen tarjetas de alto desempeño, y de bajo. Las de alto son programables y facilitan altas tasas de manejo de información, pues son en cierta forma inteligentes y suficientes, tal como un sistema Stand Alone, y por tanto no comprometen mucho la velocidad y rendimiento del computador.

Las tarjetas de bajo desempeño requieren de un control directo del computador y se ven limitadas por la velocidad de este. El Windows en cierta forma es un sistema operativo que no trabaja en tiempo real, para operaciones donde la tasa de muestreo es muy alta, como en aplicaciones de audio, radar

vibraciones y videos, aunque para aplicaciones de lentitud considerable es bueno, como en controles de horno. En aplicaciones lentas Windows y tarjetas simples bastan por que los tiempos perdidos por el sistema de interrupciones de Windows (sea por mover el mouse o cualquier otra cosa) no afectan comparativamente. Para aplicaciones de alta velocidad y tiempo real, se requiere de hardware especial, osea tarjetas inteligentes que se programen y transfieran los datos a memoria, ya sea por rutinas de DMA (acceso directo a memoria) o por rutinas de interrupciones al procesador.

Las tarjetas como cualquier otro periférico, requiere de sus parámetros de programación y hasta protocolos de comunicación, por lo que se requiere de un software Driver que maneje lo bajo de programación, y deje en la superficie la posibilidad de programar aplicaciones con los beneficios de dichas tarjetas, de una forma sencillas.

LabVIEW ofrece acceso a los Drivers desde las rutinas de configuración. Los drivers disponibles son para las tarjetas de la NI National Instruments, pero en el mercado se consiguen drivers para otras marcas como PC-LAB.

La configuración se hace a través del programa anexo a LabVIEW, NI-DAQ.