

## **Guía de observación para Presentación en PowerPoint**

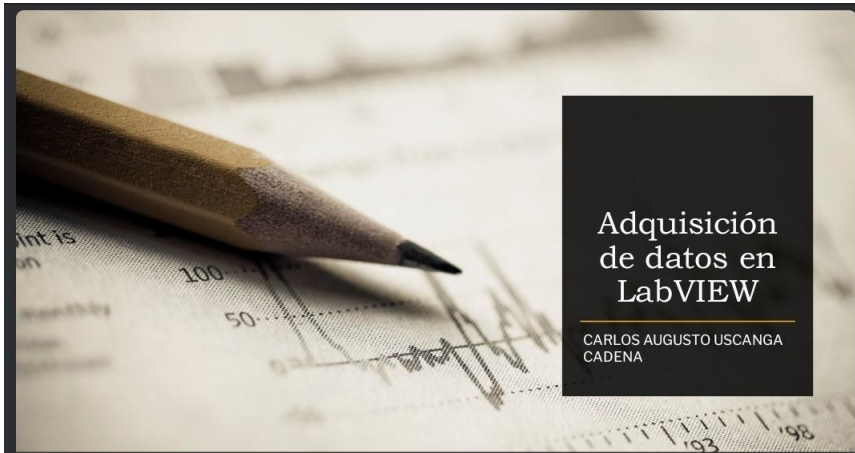
Nombre asignatura: Visualización y Control de Procesos

Tema: Adquisición de datos en Labview

Nombre del alumno o integrantes del equipo: \_\_\_\_\_

Nombre del docente: Ing. Yosafat Mortera Elías

<b>Criterios</b>	<b>Indicador máximo por criterio</b>	<b>Indicador de alcance total (30%)</b>
<b>1. Capacidad crítica y autocrítica del trabajo</b>	0-5	5
<b>2. Habilidad en el uso de TIC</b>	0-7	7
<b>3. Dominio del tema</b>	0-7	5
<b>4. Utilización de ejemplos acorde al tema explicado.</b>	0-7	6
<b>5. Manejo e inclusión de referencias bibliográficas</b>	0-4	4
<b>Total Indicador</b>	30	<b>27</b>



## Adquisición de datos en LabVIEW

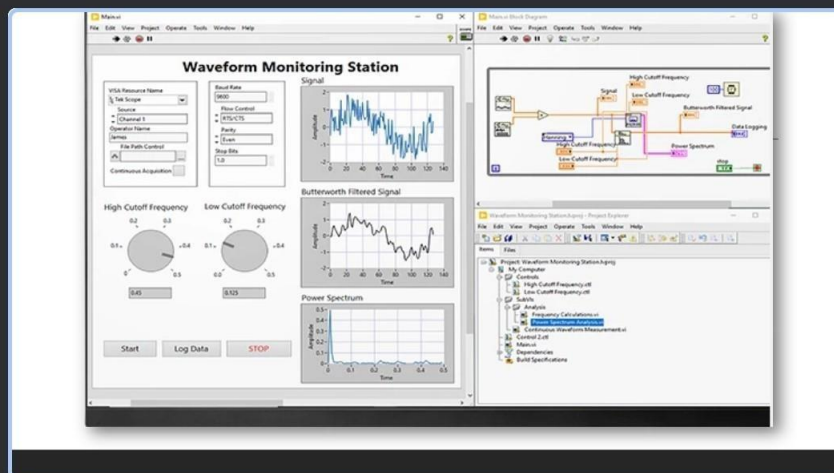
CARLOS AUGUSTO USCANGA CADENA

### ¿QUÉ ES LABVIEW?

LabVIEW es un entorno de programación gráfica que los ingenieros utilizan para desarrollar sistemas pruebas automatizadas de investigación, validación y producción.

¿Qué puedo hacer con LabVIEW?

Durante los últimos 35 años, LabVIEW ha sido la herramienta elegida por los ingenieros para desarrollar sistemas de pruebas automatizadas. Desde realizar una simple medida de voltaje hasta misiones espaciales, descubra cómo LabVIEW puede impulsar su próximo proyecto:



### ¿Qué es DAQ?

La adquisición de datos (DAQ) es el proceso de medir un fenómeno eléctrico o físico como voltaje, corriente, temperatura, presión o sonido. Un sistema DAQ consiste de sensores, hardware de medidas DAQ y una PC con software programable como LabVIEW.

Convertir datos del mundo real en información

**ACTIVIDAD: Problemario**

**MATERIA: Visualización y Control de Procesos**

**ALUMNO:**

**DOCENTE: Ing. Yosafat Mortera Elías**

**1.- REALIZA LO QUE SE TE INDICA.**

Se propone realizar un programa que genere una tensión analógica por el canal 0 de salida, a través del correspondiente convertidor D-A (DAC0, clema nº 20. Dicha tensión será seleccionada por el usuario del programa mediante un mando en la pantalla del PC, entre  $\pm 5V$ . Además, esta funcionalidad se debe mantener hasta que el usuario decida salir del

programa mediante un pulsador. Por ello, la ejecución será cíclica, con un periodo de repetición (periodo de muestreo) seleccionable entre 0 y 500 ms.

Para realizar esta tarea se pueden elegir dos opciones diferentes: la estructura de datos *while* con control de tiempo de ejecución o la estructura *event* con el evento *timeout*. Esta última tiene que estar dentro de una estructura de *buucle* para ejecutar su código cada vez que se genere el evento.

R= Estructura event con el evento timeout




## LISTA DE COTEJO: PROBLEMARIO

**Nombre asignatura:** Visualización y Control de Procesos

**Tema:** Adquisición de datos en Labview

**Nombre del alumno o integrantes del equipo:** Carlos Augusto Uscanga Cadena

**Nombre del docente:** Ing. Yosafat Mortera Elías

1.- LA CANTIDAD DE EJERCICIOS PROPUESTOS DEBE SER RESUELTA EN SU TOTALIDAD. (CANTIDAD DE PROBLEMAS)			
<p><u>EXCELENTE: 10 pts.</u> </p> <p>Resuelve todos los problemas presentados.</p>	<p><u>NOTABLE: 8 pts.</u></p> <p>Resuelve el 80% de los problemas presentados</p>	<p><u>SUFICIENTE: 6.5 pts.</u></p> <p>Resuelve el 65% de los problemas presentados</p>	<p><u>INSUFICIENTE: 5 pts.</u></p> <p>Resuelve el 50% de los problemas presentados</p>
2.-FORMA DE PRESENTAR Y ORDENAS EL PROCEDIMIENTO PARA LA RESOLUCIÓN DE LOS PROBLEMAS O EJERCICIOS			
<p><u>EXCELENTE: 10 pts.</u></p> <p>Resuelve todos los ejercicios usando los términos correctos y un procedimiento ordenado.</p>	<p><u>NOTABLE: 8 pts.</u></p> <p>Resuelve la mayoría de los ejercicios usando términos adecuados y un procedimiento correcto.</p>	<p><u>SUFICIENTE: 6.5 pts.</u> </p> <p>Resuelve los ejercicios llegando al resultado, pero no presenta un procedimiento ordenado.</p>	<p><u>INSUFICIENTE: 5 pts.</u></p> <p>No termina de resolver los ejercicios, y no presenta un procedimiento adecuado.</p>
3.-ENTREGA EN TIEMPO Y FORMA LA ACTIVIDAD			
<p><u>EXCELENTE: 10 pts.</u></p> <p>Entrega el trabajo en la fecha y hora acordada.</p>	<p><u>NOTABLE: 8 pts.</u> </p> <p>Entrega el trabajo con un día de retraso.</p>	<p><u>SUFICIENTE: 6.5 pts.</u></p> <p>Entrega el trabajo con dos días de retraso.</p>	<p><u>INSUFICIENTE: 5 pts.</u></p> <p>Entrega el trabajo con tres días de retraso.</p>

**TOTAL: 24.5 PTS.**

**ACTIVIDAD: Problemario**

**MATERIA: Visualización y Control de Procesos**

**ALUMNO: Carlos Augusto Uscoanga Cordera 1**

**DOCENTE: Ing. Yosafat Mortera Elías**

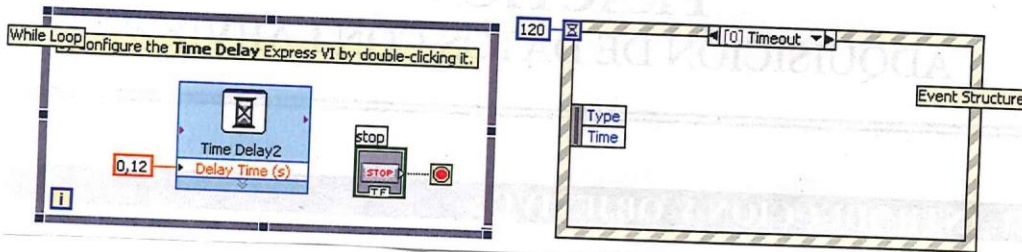
**1.- REALIZA LO QUE SE TE INDICA.**

Se propone realizar un programa que genere una tensión analógica por el canal 0 de salida, a través del correspondiente convertidor D-A (DAC0, clema nº 20. Dicha tensión será seleccionada por el usuario del programa mediante un mando en la pantalla del PC, entre  $\pm 5V$ . Además, esta funcionalidad se debe mantener hasta que el usuario decida salir del

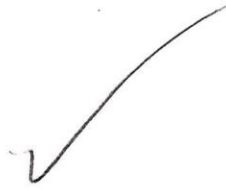
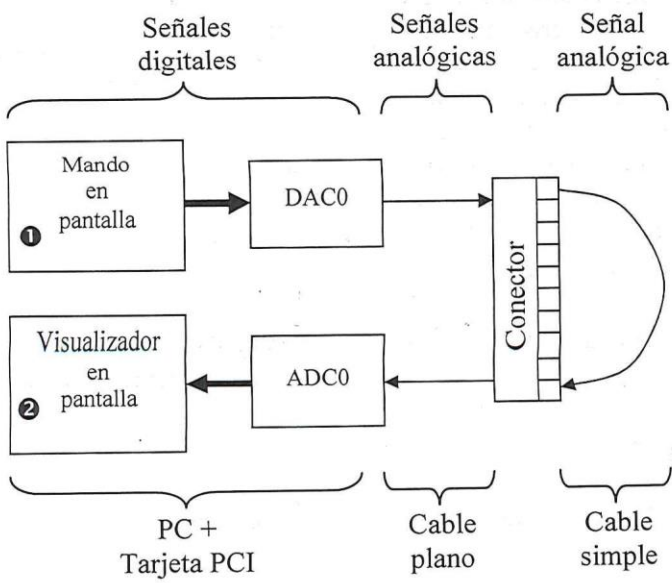
programa mediante un pulsador. Por ello, la ejecución será cíclica, con un periodo de repetición (periodo de muestreo) seleccionable entre 0 y 500 ms.

Para realizar esta tarea se pueden elegir dos opciones diferentes: la estructura de datos *while* con control de tiempo de ejecución o la estructura *event* con el evento *timeout*. Esta última tiene que estar dentro de una estructura de *blucle* para ejecutar su código cada vez que se genere el evento.

R= Estructura event con el evento timeout



Para comprobar que la tensión se genera correctamente se podría medir con la misma con un voltímetro o un osciloscopio conectado al pin de salida correspondiente al DAC0. Sin embargo, para evitar esto, se ha conectado eléctricamente el canal 0 de salida al canal 0 de entrada. Así, mediante el convertidor A-D C ADC0, clema nº 2 del conector de la figura 11.1) se podrá ver la medida en la pantalla del PC. La figura 11.3 muestra un diagrama del Esquema propuesto.



Por cada ciclo de ejecución no solo había que hacer una conversión DA sino una AD, mostrándose la evolución temporal de las tensiones medidas en un gráfico de tipo *chart* (2). Adicionalmente, se deben presentar en pantalla en todo momento los valores máximo y mínimo alcanzados hasta el momento por la señal adquirida.

La figura 11.7 y figura 11.8. muestran respectivamente el panel frontal y el diagrama de bloques del ejercicio. Desde el entorno habrew tan solo hay que utilizar los VLS' adecuados para acceder al exterior del PC, por ejemplo en la pestaña de funciones seleccionar: All functions => NI measurements => Data acquisition => para seleccionar los VLS' correspondientes a la interacción con las entradas y salidas analógicas. En la figura 11.4 se muestra el proceso de elección de ventanas hasta llegar los VLS.

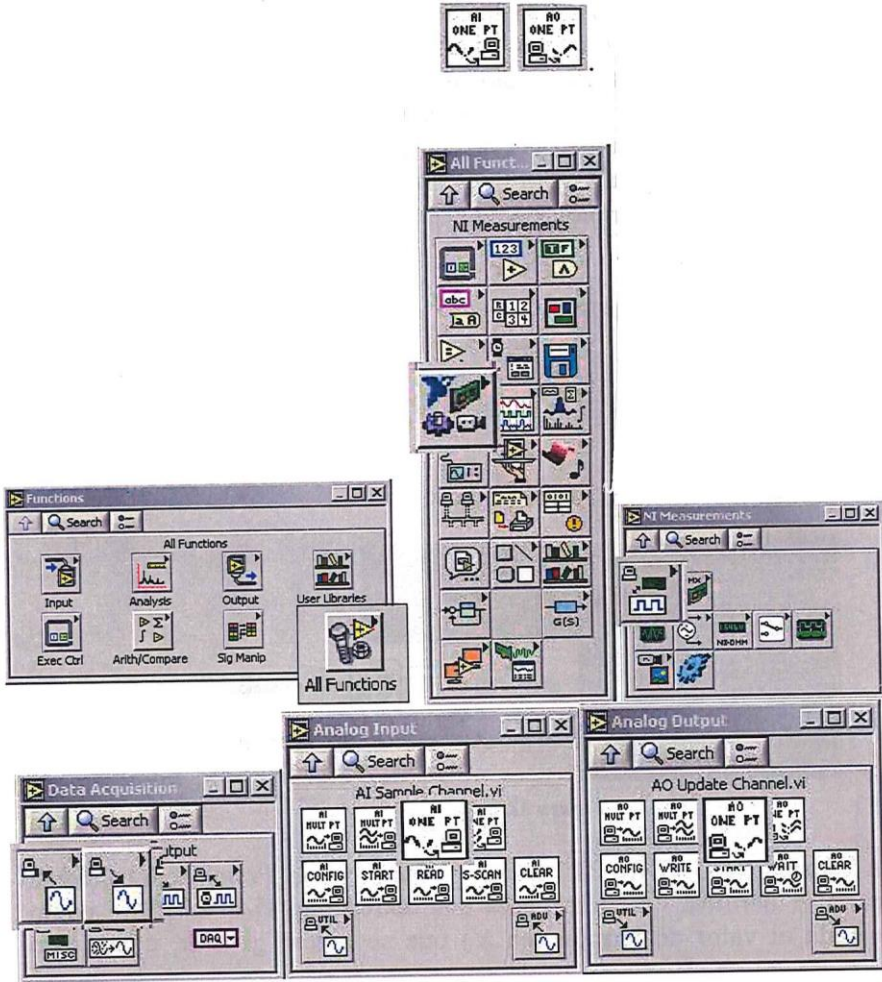


Figura II. 4

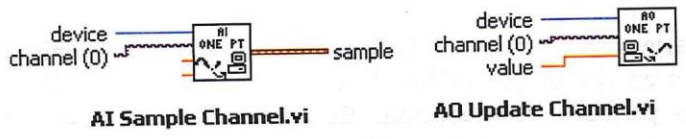


Figura II. 5

Como se puede ver en la figura II.5 existiendo dos tipos de parametros de entrada comunes a ambos VIs al dispositivo (1 en este laboratorio) y el canal con el que se desea interactuar (0 en esta practica)

El VI One PT (Analog Input One Point o AI sample Channel) figura II.5 izquierda que permite hacer una conversión AD. Se debe tener especial cuidado puesto que como se puede apreciar en la figura II.5 la linea o parametro de salida muestra por defecto un tipo de datos mas complejo que un escalar. Esto es así porque incluyen datos adicionales de la adquisición, además del propio valor de tensión captado. Para hacer que la salida de VI sea simplemente un escalar contenido el valor de la tensión analógica adquirida en (V)

Se ha de elegir la opción *Select Type* > *scaled value* en el menú emergente 2 que muestra la figura 11.6. En esta figura se representa el VI con salida de tipo de escalar (línea simple)

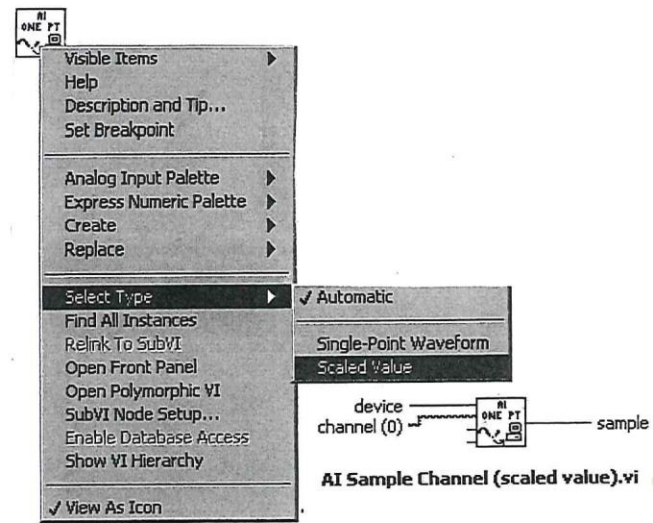


Figura II. 6

Por parte, la función "AO one PT" (Analog Output One Point o AO update channel) figura 11.5 derecha, permite hacer una conversión DA. Requiere como parámetro de entrada el valor de tensión (en V) que se requiere generar como dato escalar.

24.5

Nota:

El procedimiento que emplea no está ordenado en su totalidad.

Sin embargo, si llegas al resultado y cumple con lo que se le pide.



Nombre: Erlo Augusto Usanga Cadena

Carrera: Ingeniería Mecánica Semestre: 10<sup>o</sup>

Grupo: 811-C

### EXAMEN UNIDAD 3.

#### VISUALIZACION Y CONTROL DE PROCESOS

A) Eliminar la parte correspondiente a la detección de máximos y mínimos en la Figura II. 8 y representar en el Waveform Chart la señal proveniente del ADC0 y ésta misma retardada 3 y 6 periodos de ejecución. Es decir, deben representarse 3 gráficas simultáneamente, con un retardo de tres periodos entre cada una de ellas.

B) Posteriormente, se propone generar un subVI que calcule la diferencia entre dos datos de entrada y la divida entre un tercero. Este subVI se usará en el programa principal para calcular una derivada aproximada de la señal que se toma del ADC0. Para ello, habrá que restar a cada muestra la anterior y dividir la diferencia entre el período de tiempo que las separa. El Waveform Chart debe representar en este caso la señal del ADC0 y su derivada.

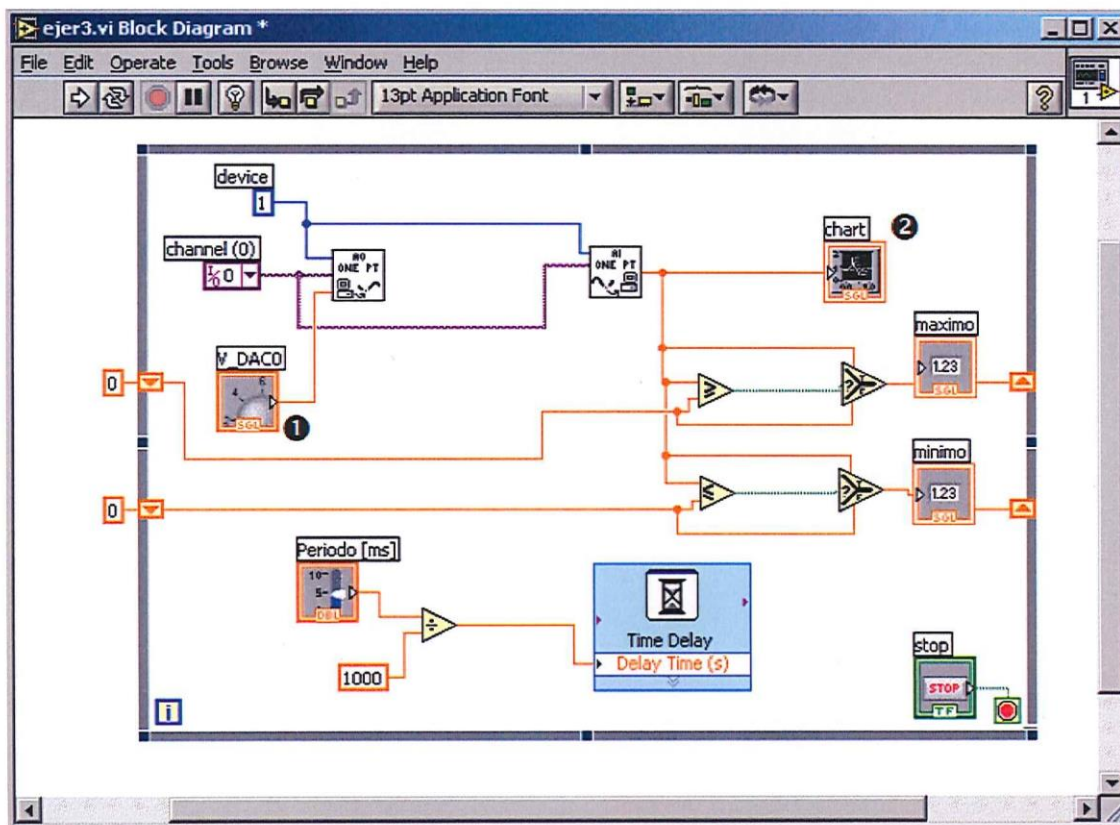


Figura II. 8

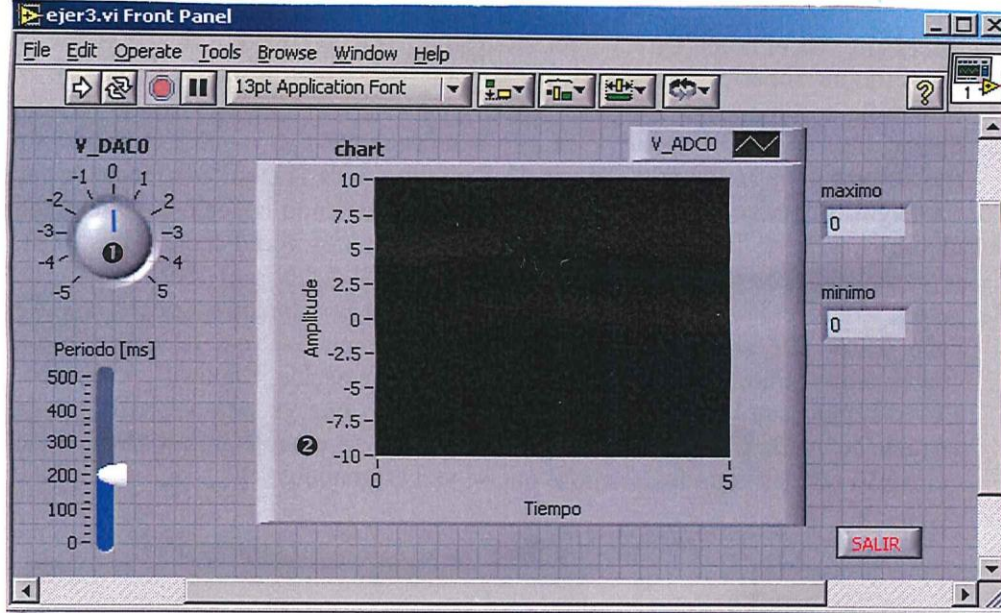
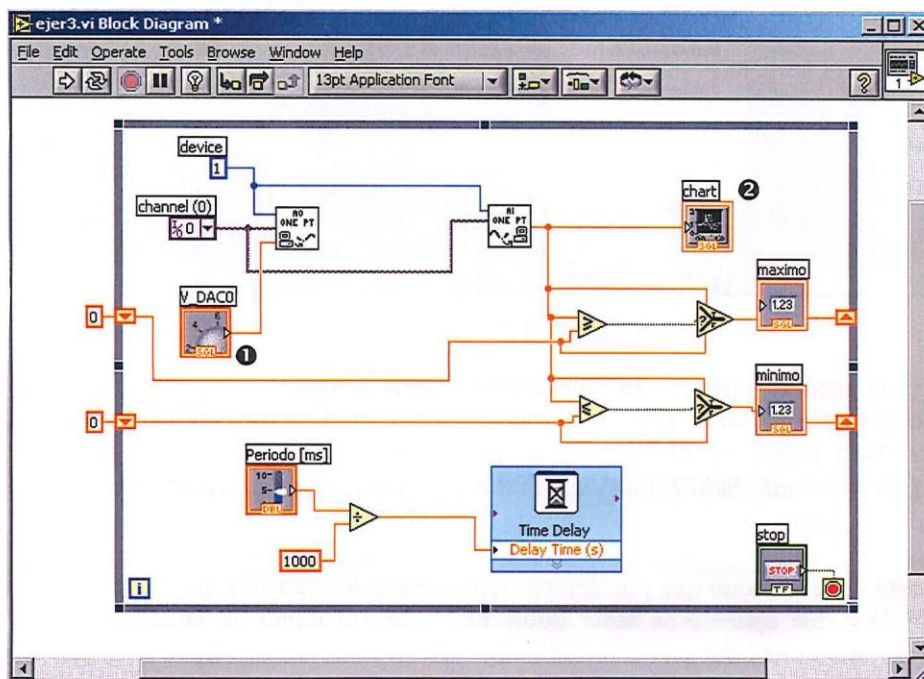


Figura II. 7



Un sub Vi es un Vi que, almacenado en un fichero con extensión Vi puede ser incluido en otros Vis de mayor nivel jerárquico. En general todo Vi puede convertirse en subvi, para lo que todas las líneas provenientes de controles en su diagrama de bloques, son susceptibles de constituir líneas entradas al sub Vi, y todas las líneas con destino a indicadores en su diagrama de bloques son susceptibles de constituir líneas de salida del sub Vi. La forma de asignar entradas y salidas en un Vi a conectores de entrada y salida del sub Vi que se requiere generar en el siguiente.

- 1º Hacer clic en el botón derecho del ratón sobre el icono que por defecto tiene el UI (esquina superior derecha de la ventana del panel Frontal)   
 Figura 11.9 izquierdo
- 2º Seleccionar show connector. Con ello aparecerá sobre el icono una matriz de dos columnas de rectángulos, figura 11.9 medio. La de izquierda tendría tantos rectángulos como control haya en el UI representando conexiones de entrada al subUI, la derecha tendría tantos rectángulos como indicadores haya representando conexiones de salida del subUI.
- 3º Ahora al pasar el ratón sobre esta matriz aparece la herramienta de cableado se debe hacer clic sobre el rectángulo (conector) que se requiere asignar a continuación sobre el control o indicador al que se asocia. Figura 11.9 derecha. El rectángulo cambia de color indicando con ello que ya está asignado. En la ventana de ayuda contextual Help si se tiene abierta va completándose el esquema del subUI en las entradas del subUI con las entradas y salidas que se van asignando.

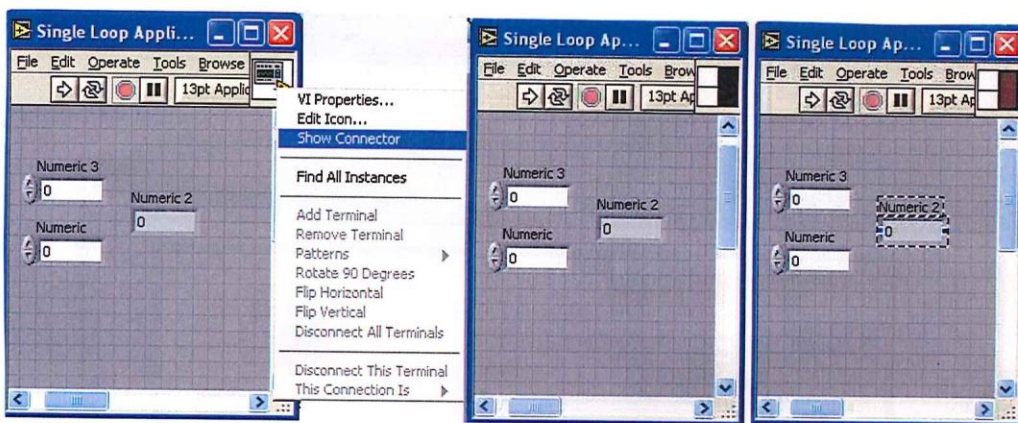
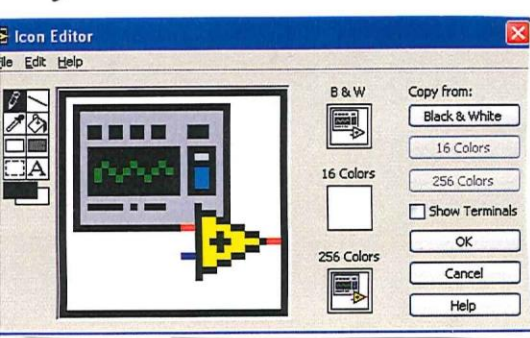


Figura II. 9

También es posible utilizar otro mecanismo particularmente interesante cuando se pretende generar un subVI a partir de un VI, consiste en seleccionar dicha parte (también ser el VI completo) sobre el diagrama de bloques y elegir la opción **Edit -> Create SubVI**. LabVIEW automáticamente asigna colores a entoda en indicando a salidas de selección queda lugar al subVI.

Puede cambiarse al icono que por defecto representa a los subVIs mediante un sencillo editor gráfico que aparece al hacer doble clic sobre el icono de la esquina superior derecha del panel frontal, según se muestra en la Figura 11.9 izquierdo apareciera la ventana de la Figura 11.10



Finalmente se debe salvar el subVI bien con la opción **File -> Save as** de la ventana del panel si se trabaja según el primer mecanismo explicado, bien con la opción **File -> Save as** de la ventana del panel que aparece al hacer el doble clic sobre el icono del subVI generado sobre el diagrama de bloques del VI jerárquicamente superior si se trabaja

según el segundo mecanismo explicado.

La Figura 11.11 indica los pasos a seguir para poder utilizar en cualquier diagrama de bloques del subVI salvando previamente el archivo.

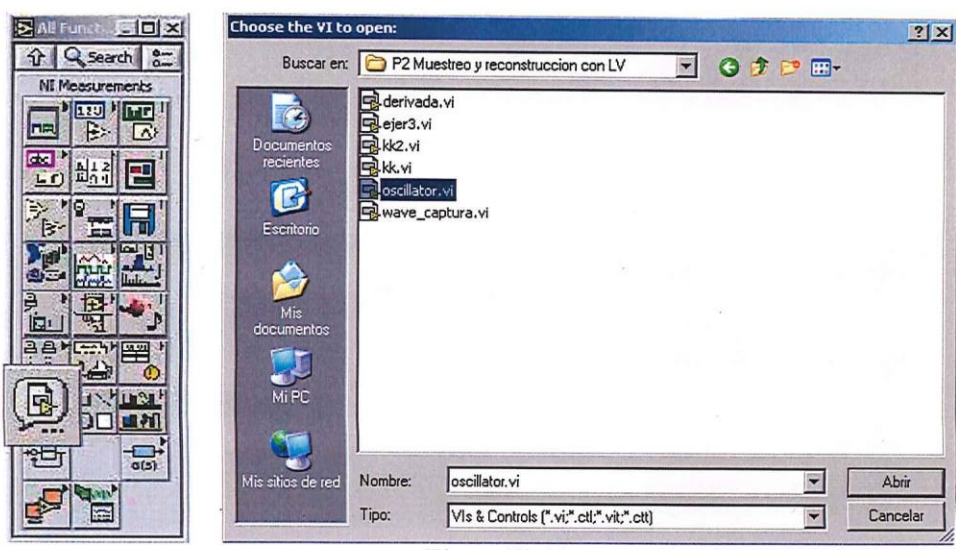


Figura II. 11