

INGENIERÍA MECATRÓNICA



DI_CERO

DIEGO CERVANTES RODRÍGUEZ

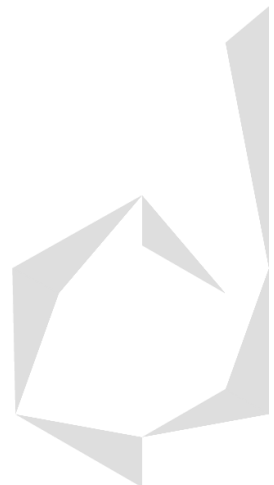
INSTRUMENTACIÓN VIRTUAL

NI LABVIEW 2020 (32-BIT)

Bucle while y Shift Register

Contenido

| | |
|---|----|
| Introducción Teórica de LabVIEW:..... | 2 |
| Introducción al Entorno de LabVIEW:..... | 2 |
| Front Panel: Ventana Gris con la Interfaz del Programa | 4 |
| Block Diagram: Ventana Blanca con la Lógica del Programa (Bloques) | 4 |
| Front Panel o Block Diagram - Show Context Help: Descripción de Bloques | 5 |
| Front Panel y Block Diagram: Navegar de una Ventana a Otra | 6 |
| Block Diagram - Cambiar Nombre a los Bloques: Nombre de los elementos en el Front Panel | 7 |
| Block Diagram - Highlight Execution: Correr Más Lento el Programa..... | 8 |
| Coertion dot: Conversión Automática de Datos por Parte de LabVIEW | 8 |
| Block Diagram - Clean Up Diagram: Organizar Automáticamente los Bloques del VI | 8 |
| Programa: Bucle while y Shift Register..... | 9 |
| Desarrollo del Programa: Gráfica SubVI y su Promedio de 5 Shift Register | 9 |
| Block Diagram - Bucle While: Uso de Memorias (Registros) Shift Register | 9 |
| Block Diagram - Bucle While - Shift Register: Registros de Memoria dentro de un Ciclo..... | 9 |
| Block Diagram - User Libraries: Ejecutar un SubVI Creado por Nosotros Mismos..... | 11 |
| Block Diagram - Compound Arithmetic: Operaciones Matemáticas Booleanas, Aritméticas, etc. | 12 |
| Block Diagram - Divide: Dividir dos Variables Numéricas Entre Sí | 13 |
| Block Diagram - Bundle: Juntar Varios Tipos de Datos Para Mandarlos a un Cluster | 14 |
| Block Diagram - Wait Until Next ms Multiple: Temporizador en milisegundos..... | 15 |
| Front Panel - Waveform Chart: Ventana que Muestra una Señal (Dynamic Data) | 16 |
| Ejecución del Programa: Señal SubVI y su Promedio de 5 Shift Register | 17 |



Introducción Teórica de LabVIEW:

LabView sirve para poder usar la computadora como instrumento de medición, monitoreo, control y análisis de procesos y operaciones, esto se hace a través de una frecuencia de muestreo que se relaciona con mediciones de los dispositivos digitales y tiene que ver con la señal de reloj de la tarjeta de desarrollo, indicando cada cuánto tiempo se hará un muestreo de cualquier señal del mundo real.

La diferencia entre los instrumentos virtuales de medición y los reales es más que nada el precio, ya que un osciloscopio cuesta alrededor de \$10,000 y se puede hacer la misma función con LabView y un Arduino, que cuesta alrededor de \$170, además de que es modular, esto implica que se pueden agregar o quitar funcionalidades. La mejor tarjeta de desarrollo para hacer esto es la de NI Instruments, que es la creadora de LabVIEW.

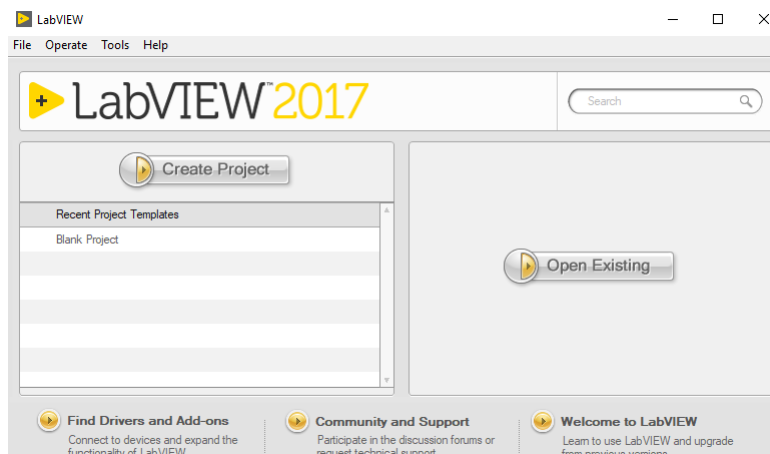
- **Instrumentación Tradicional:** El hardware es más usado, como por ejemplo con los circuitos integrados de un osciloscopio.
- **Instrumentación Virtual:** El software es el más utilizado y sus funciones son modulares, como lo es en una tarjeta de desarrollo de National Instruments.

La instrumentación virtual es empleada para la gestión de sistemas industriales y muy utilizado en compañías como: Ford, SpaceX, Accenture, Bosch, etc.

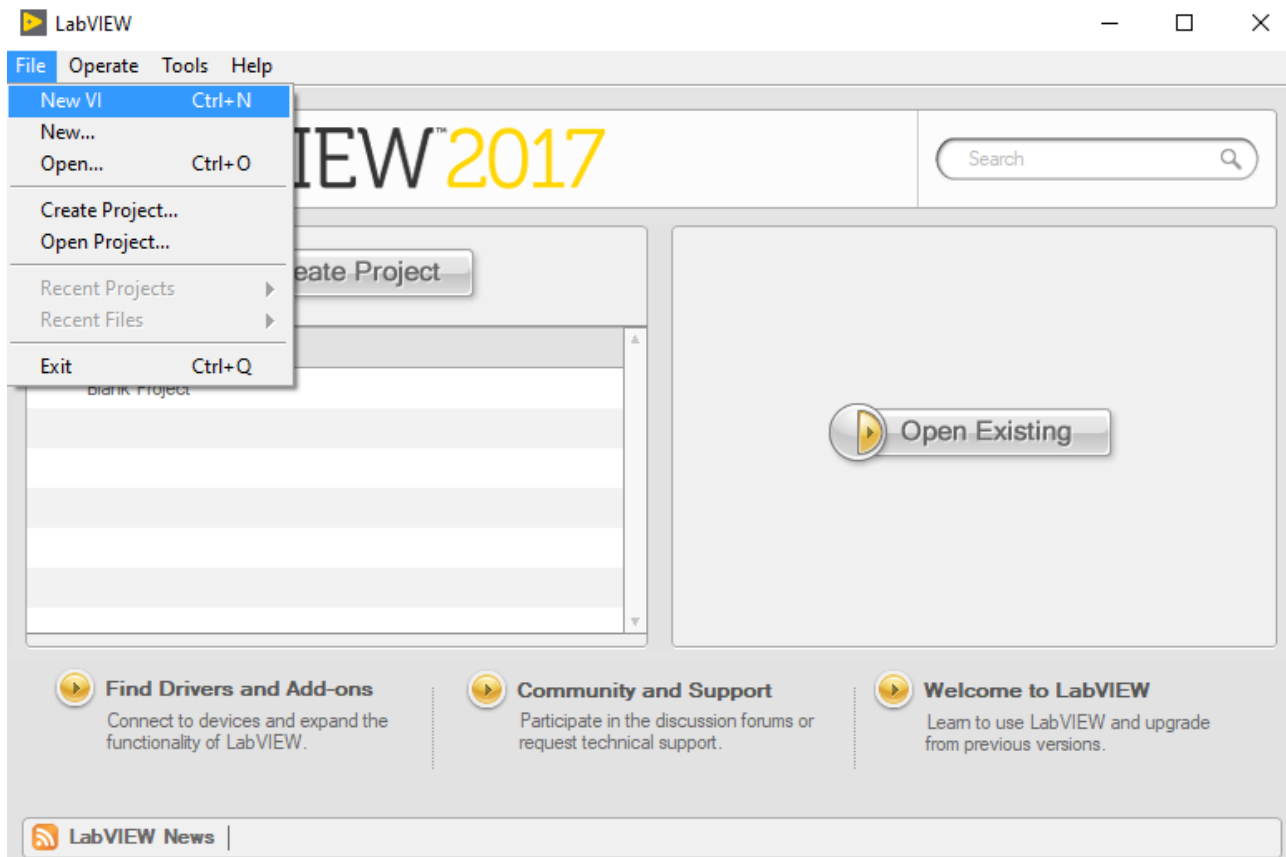


Introducción al Entorno de LabVIEW:

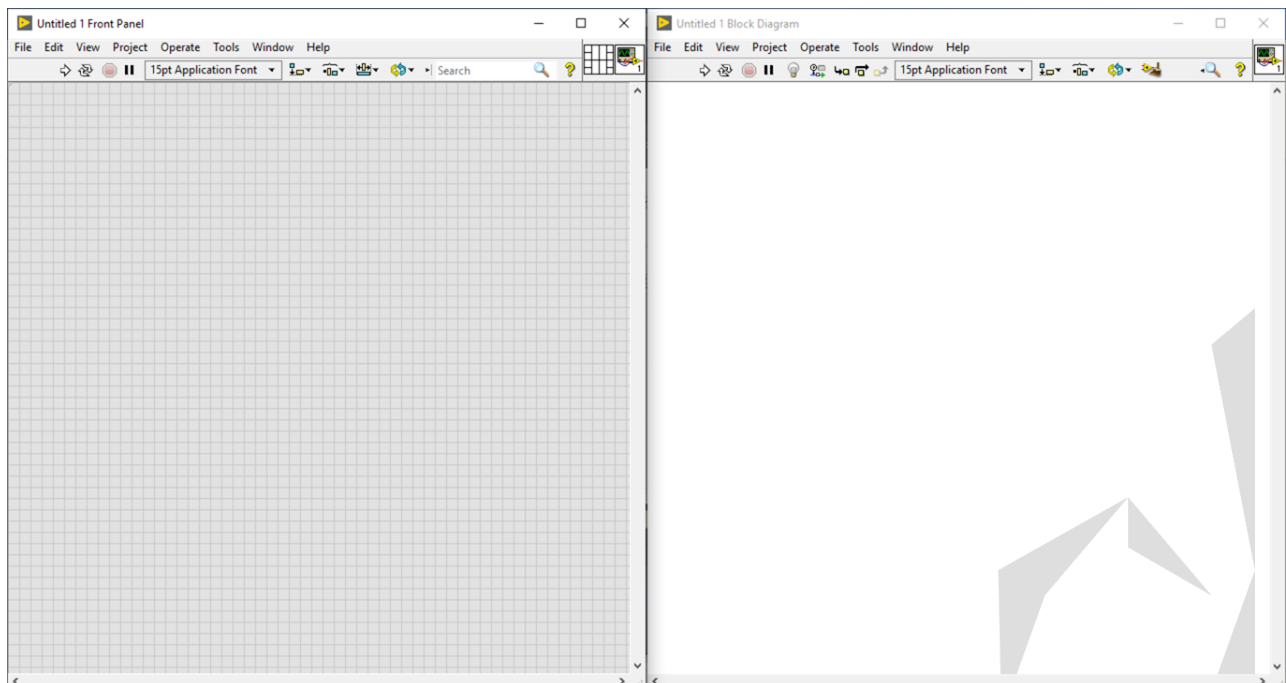
Un nuevo proyecto de LabView se abre por medio del botón de Create project que aparece inmediatamente cuando abra el programa.



VI se refiere a Virtual Instrument.

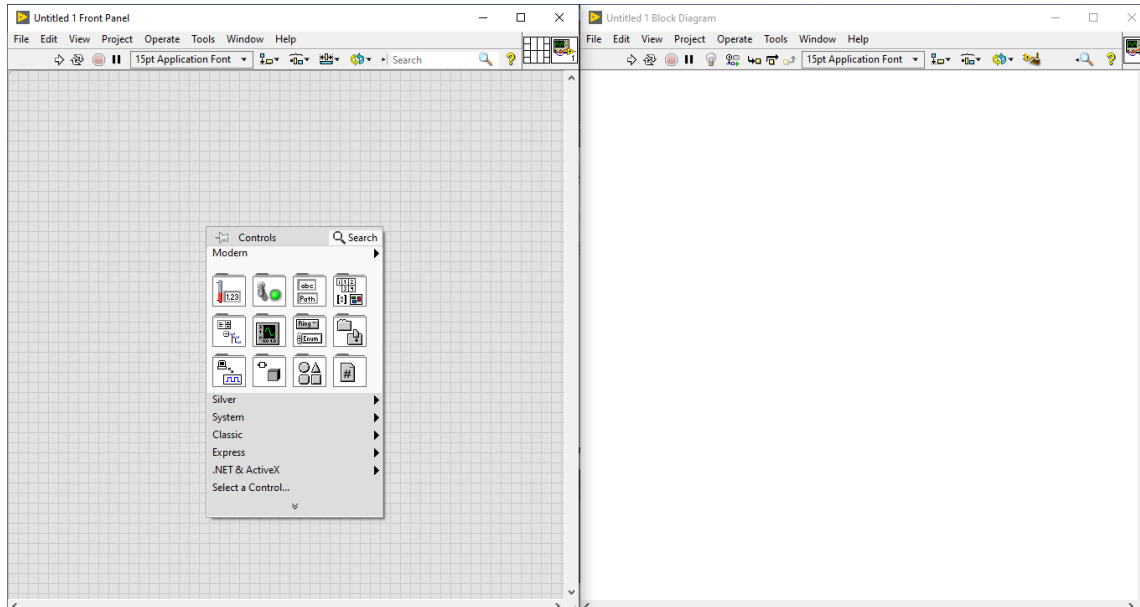


Al hacerlo me abrirá estas dos ventanas, en una de ellas se creará el programa con bloques (Ventana Block Diagram) y en la otra se verá la interfaz (Ventana Front Panel).



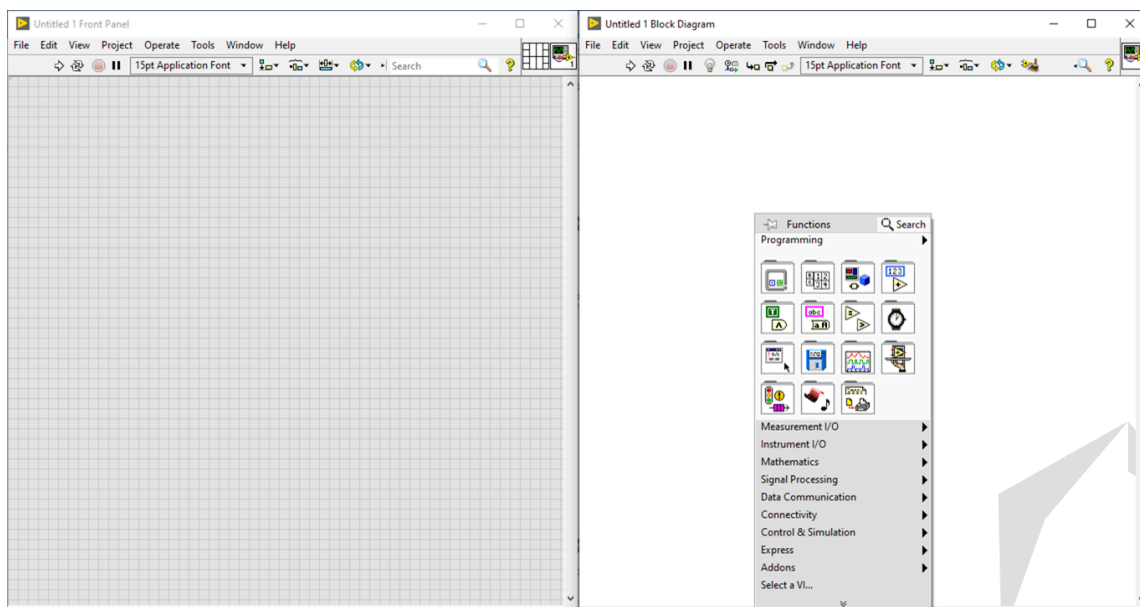
Front Panel: Ventana Gris con la Interfaz del Programa

En la ventana gris llamada **Front Panel**, es donde se observa la interfaz del Programa y se cuenta con el **control palette** que sirve para poder añadir elementos gráficos a la interfaz y aparece dando clic derecho en la pantalla gris. Si no aparece la otra ventana (blanca) por default, se debe seleccionar la opción **Window → Show Block Diagram** y con ello aparecerá.



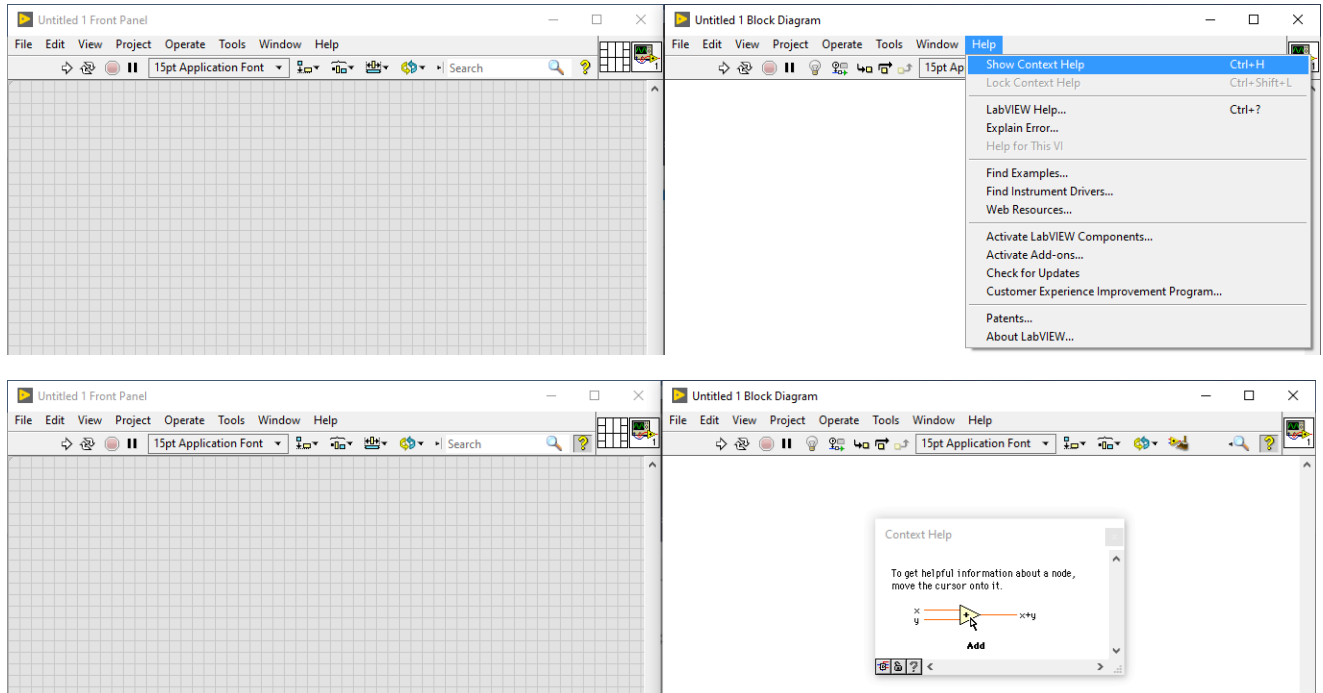
Block Diagram: Ventana Blanca con la Lógica del Programa (Bloques)

En la ventana blanca llamada **Block Diagram** aparece la **paleta de funciones** que sirve para introducir los elementos de programación en forma de bloques que se conectarán entre ellos y describirán la función del programa, aparece dando clic derecho en la pantalla gris. Si no aparece la ventana gris se debe seleccionar la opción **Windows → Show Front Panel** y con ello aparecerá.



Front Panel o Block Diagram - Show Context Help: Descripción de Bloques

Seleccionando la opción de Help → Show Context Help, aparecerá una ventana emergente que explicará las propiedades de los bloques que se puede seleccionar, mostrando una descripción de su función, imágenes explicativas y significado de sus pines de entrada y salida.



Las funciones o subrutinas son los elementos más básicos que pueden existir en LabView, dentro de ellas existe un código de bloque propio que describe sus funciones, pero además se cuenta con otros elementos:

VIs Express, VIs y Funciones

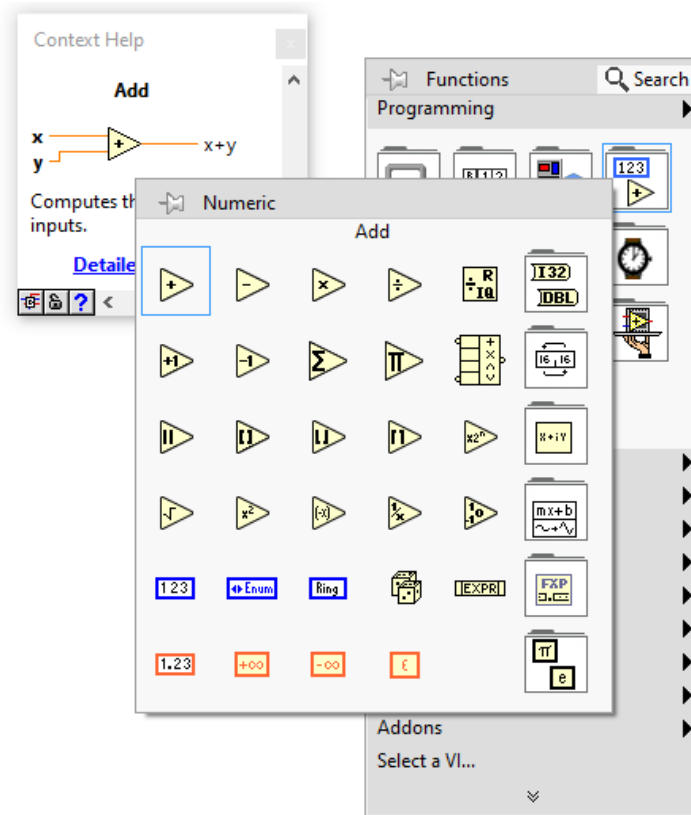


- **VIs Expreso:** VIs interactivos con pagina de dialogo configurable
- **VIs estándar:** VIs modulares y personalizables mediante cableado
- **Funciones:** Elementos fundamentales de operación de LabVIEW; no contiene panel frontal o diagrama de bloque



Función

En un bloque de código, las **terminales que aparezcan en negritas** son las que a fuerza deben estar **conectadas a algo**, las que no estén en negritas no deben estar conectadas a nada forzosamente.

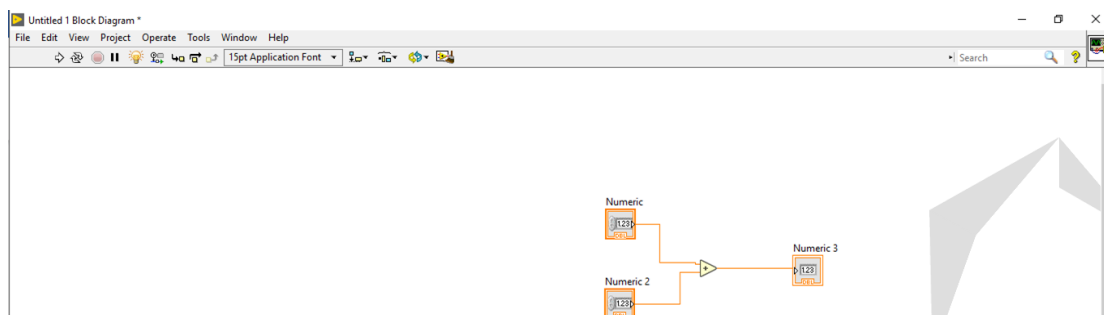


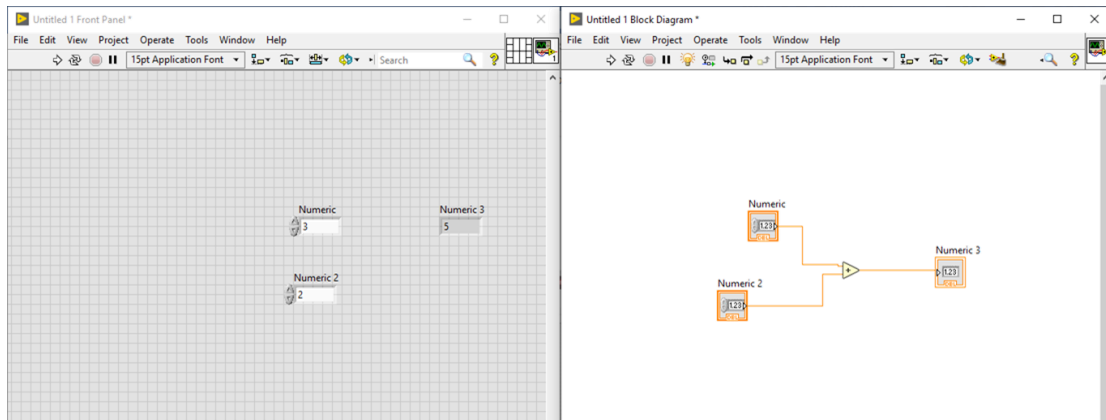
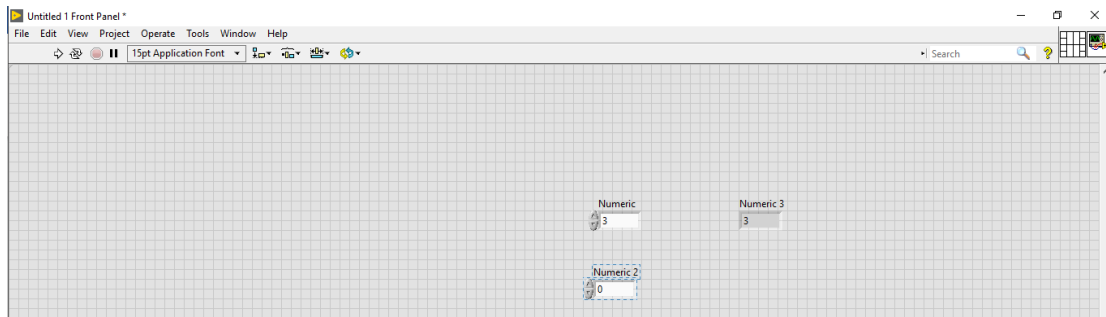
El programa es autocompilable, es decir que se corre por sí solo, por lo que si la flechita aparece rota es porque hay un error en el programa.



Front Panel y Block Diagram: Navegar de una Ventana a Otra

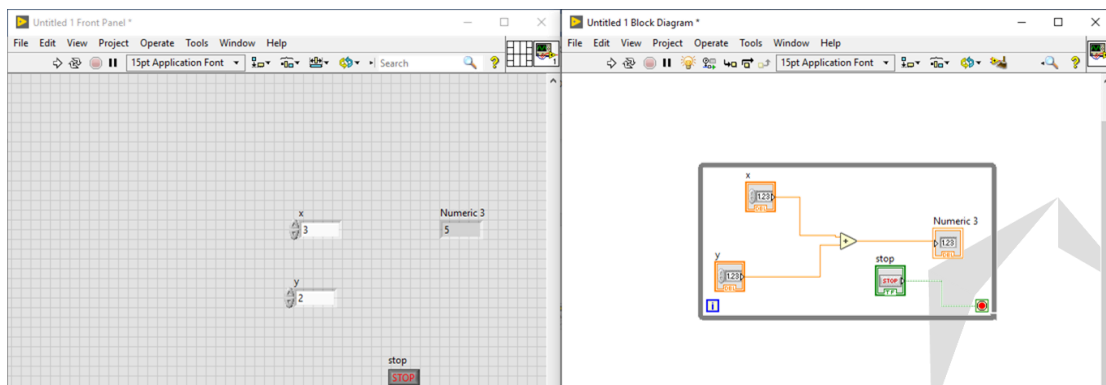
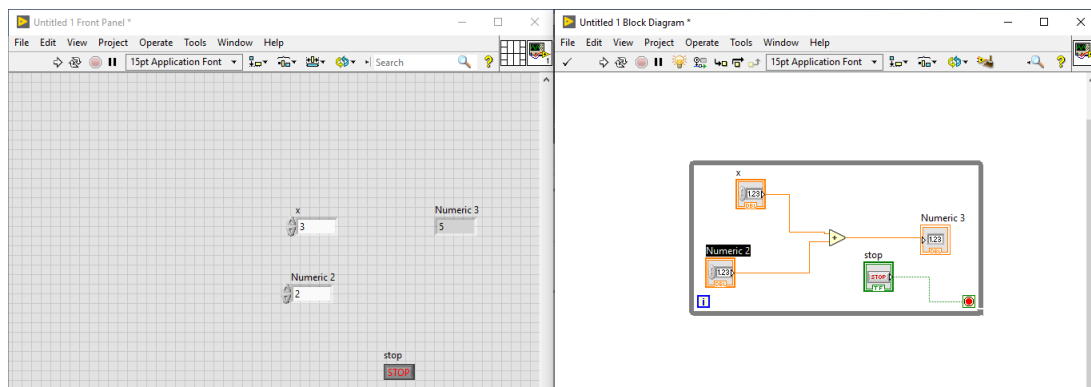
Al dar doble clic en el bloque de la pantalla blanca, me llevará al punto donde se encuentra el mismo bloque, pero en la pantalla gris.

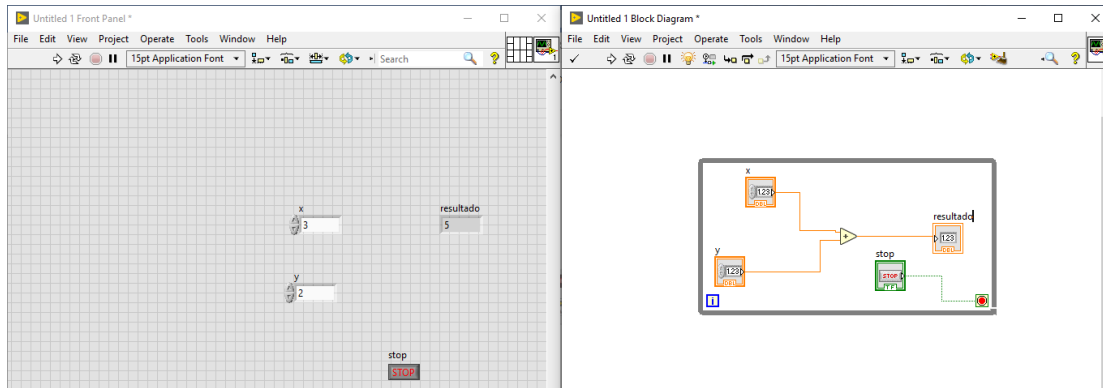




Block Diagram - Cambiar Nombre a los Bloques: Nombre de los elementos en el Front Panel

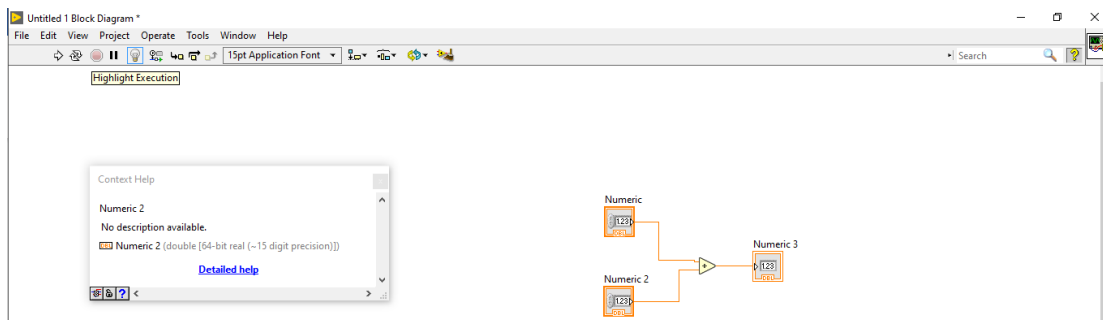
El nombre de los elementos de las interfaces se puede cambiar desde el Block Diagram, cambiándole literal el nombre a los bloques.





Block Diagram - Highlight Execution: Correr Más Lento el Programa

Podemos presionar el foquito del menú superior para ver el funcionamiento de programa de manera más lenta.

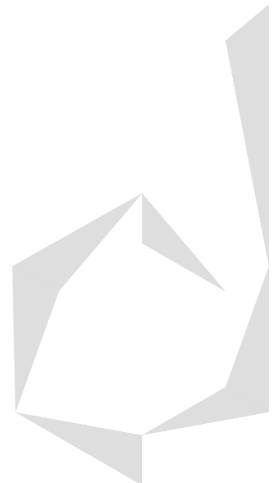


Coertion dot: Conversión Automática de Datos por Parte de LabVIEW

Aparece un punto rojo en la terminal del bloque llamado coercion dot, este lo que me dice es que los tipos de datos en la conexión son distintos, por lo que LabVIEW está forzando una conversión de un tipo de dato a otro, el problema es que en este tipo de conversión yo no sé si se están perdiendo datos, por eso debemos evitar el uso de coercion dots porque usa direcciones de memoria o recursos de la computadora sin que yo tenga control de ellos.

Block Diagram - Clean Up Diagram: Organizar Automáticamente los Bloques del VI

Con el botón de Clean Up Diagram que se encuentra en la parte superior derecha del Block Diagram se organizan mejor y de forma automática mis elementos.



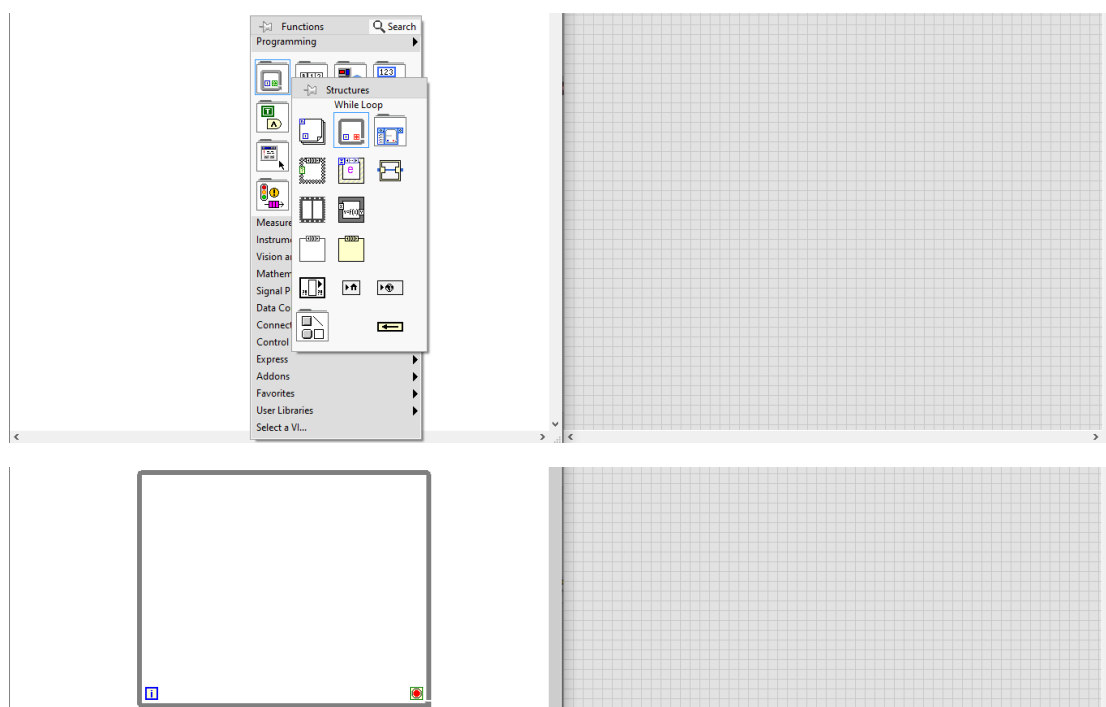
Programa: Bucle while y Shift Register

Movimiento de datos en una memoria de registros Shift Register, proveniente de un bucle for en donde además por medio de un Waveform Chart se grafica un Array obtenido del mismo Shift Register.

Desarrollo del Programa: Gráfica SubVI y su Promedio de 5 Shift Register

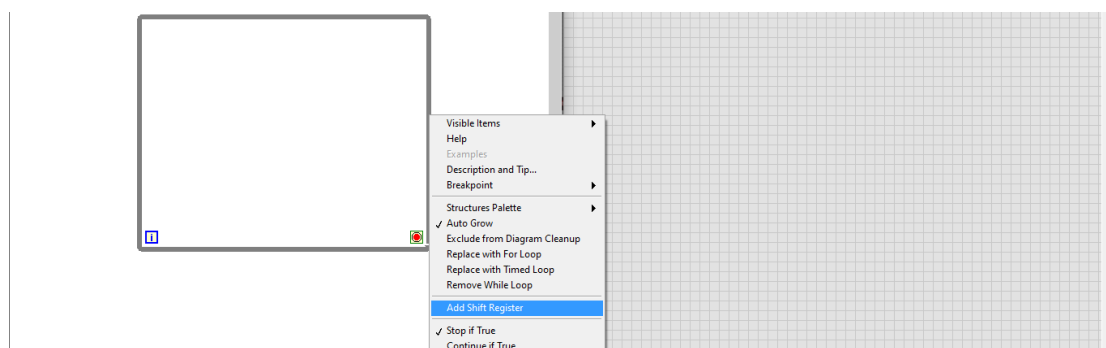
Block Diagram - Bucle While: Uso de Memorias (Registros) Shift Register

El ciclo while hace que el programa se ejecute un número infinito de veces, detenido solo por un control de botón de STOP, además cuenta con una variable i que cuenta las iteraciones que va realizando el bucle.

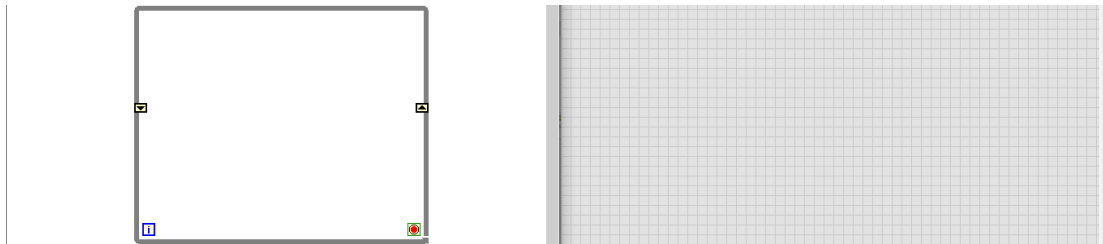


Block Diagram - Bucle While - Shift Register: Registros de Memoria dentro de un Ciclo

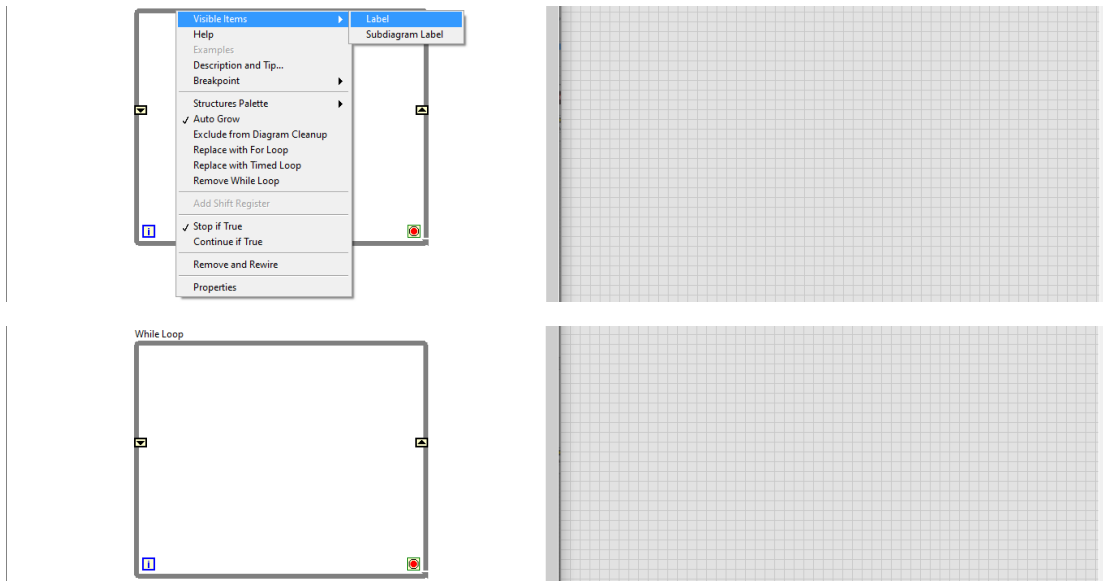
Ahora vamos a dar clic derecho al bucle y seleccionar la opción de Add Shift Register, esto es para que almacenemos en una dirección de memoria temporal algún dato generado en un ciclo for o while.



Cuando el Shift Register se muestra de color negro es porque no se le ha asignado un dato.

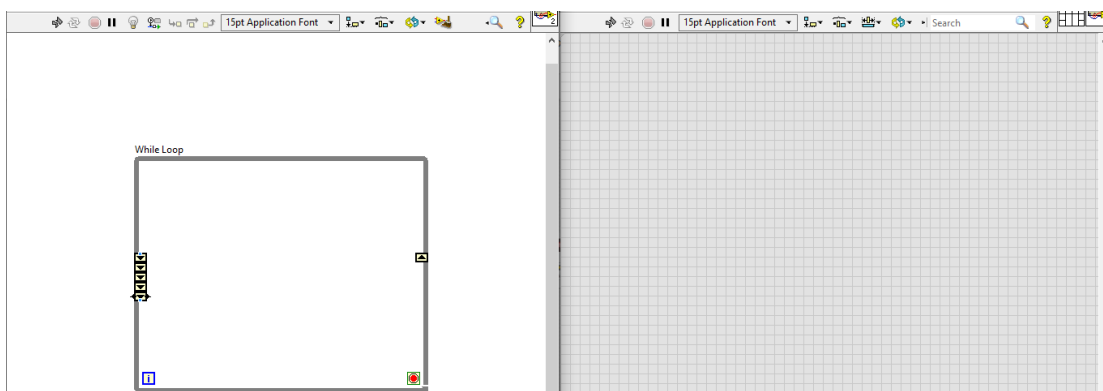


Mostrar nombre del bloque: Clic derecho → Visible Items → Label.



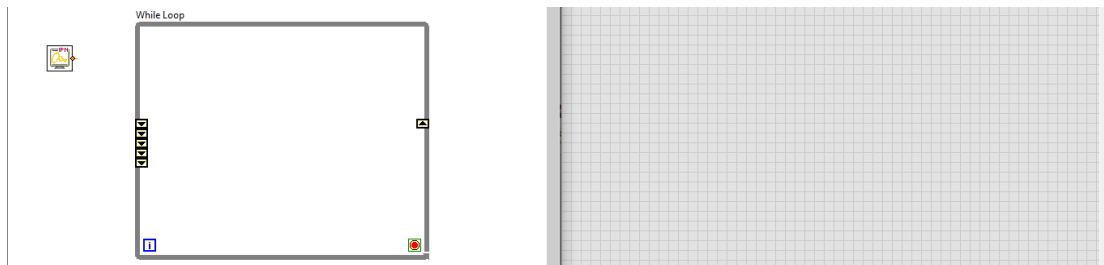
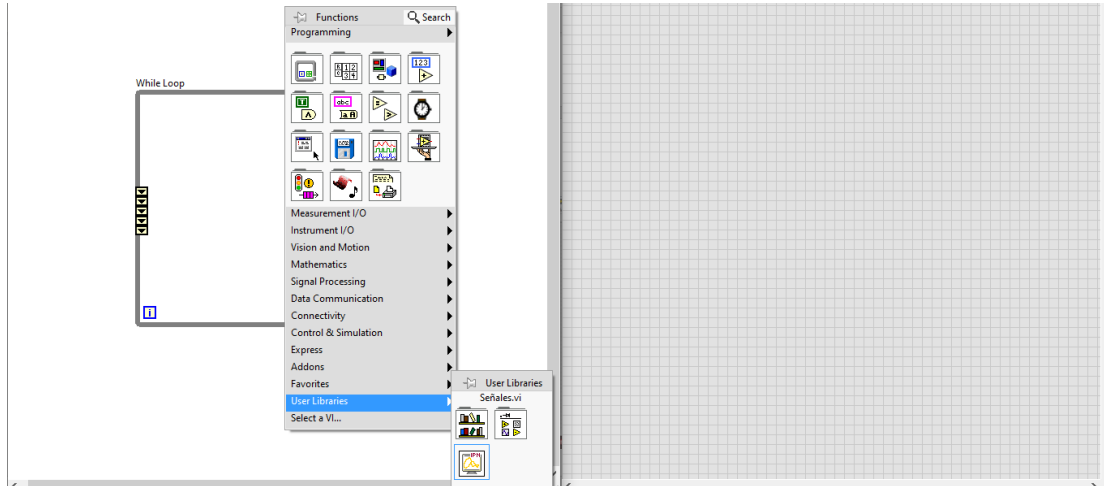
Si lo extiendo puedo ver las direcciones de memoria a donde se van a estar asignando los valores, cuando un dato entre al primer registro (posición de arriba hacia abajo) del Shift Register, si tenía un dato almacenado ahí, lo va a mover a la siguiente posición (hacia abajo) y así se estarán moviendo y moviendo los datos en los registros del Shift Register mientras vayan entrando más y más datos a la memoria, hasta que el último dato ya no se pueda mover a otra posición, entonces se borrará ese dato.

En este caso vamos a asignar 5 posiciones, llamadas registros de corrimiento (los de la izquierda) donde entrarán datos por medio del registro de la derecha.

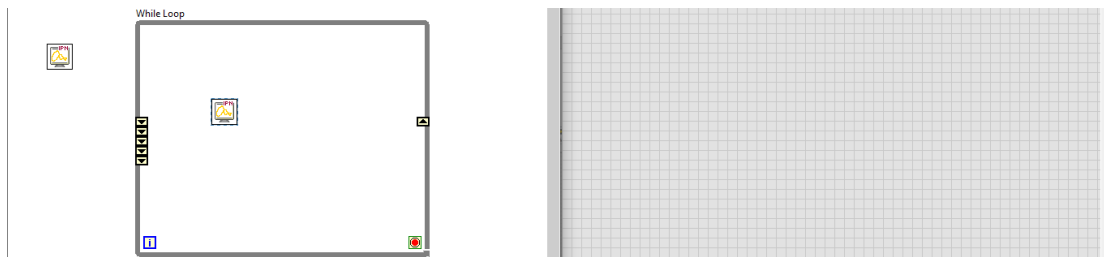


Block Diagram - User Libraries: Ejecutar un SubVI Creado por Nosotros Mismos

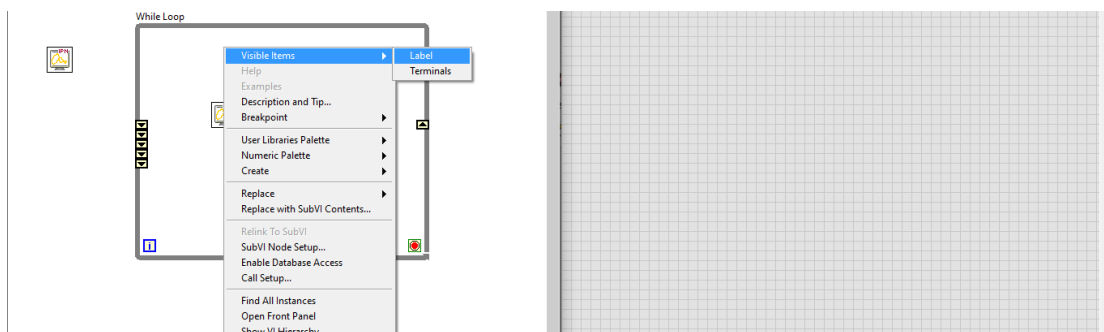
Las SubVI son diagramas de bloque creadas por nosotros mismos que podemos reutilizar en otros programas, las que podemos acceder desde las librerías de LabVIEW estarán guardadas en la siguiente ruta: **D:\Program Files (x86)\National Instruments\LabVIEW 2020\user.lib**

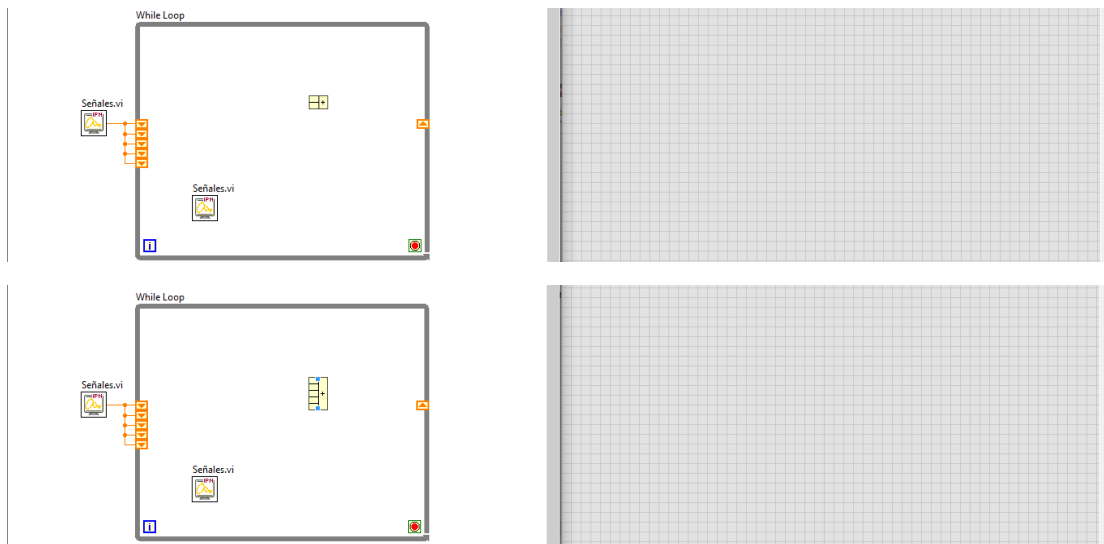


El subvi Señales crea una señal de números aleatorios que van de 0 a 100.

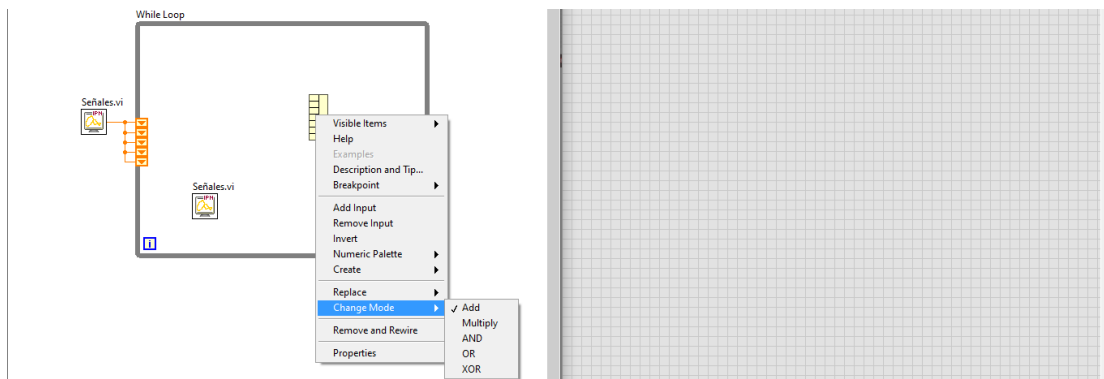


Mostrar nombre del bloque: Clic derecho → Visible Items → Label.

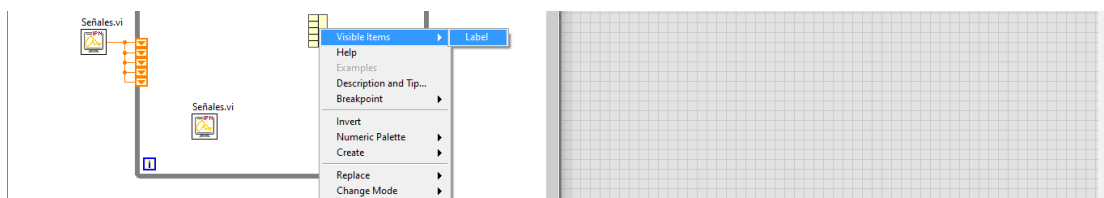




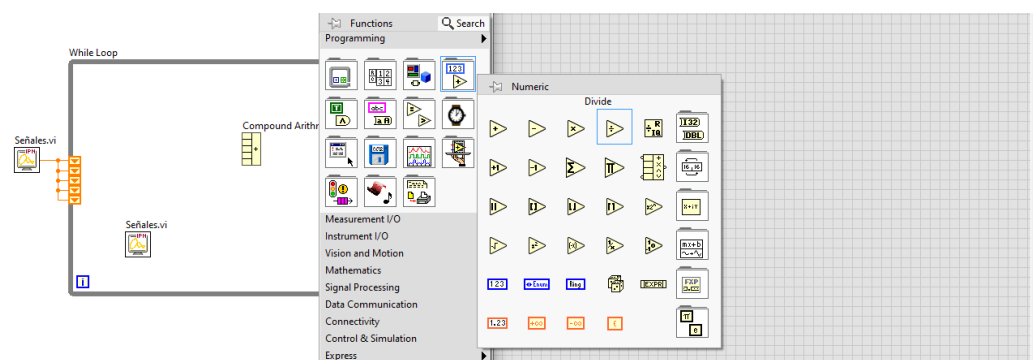
En este caso se va a elegir la operación Add, por lo que es una suma de 5 entradas.

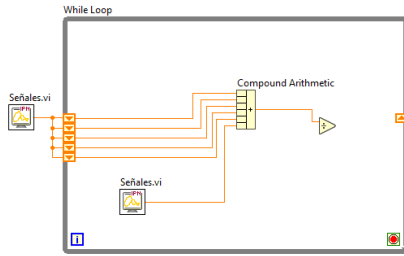


Mostrar nombre del bloque: Clic derecho → Visible Items → Label.

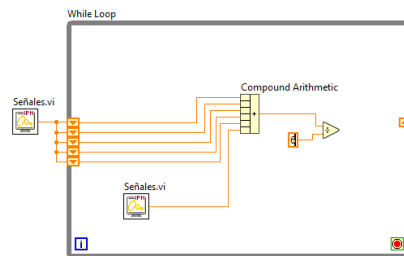
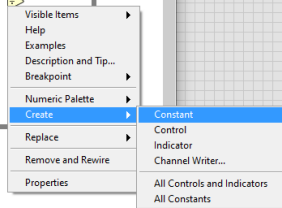
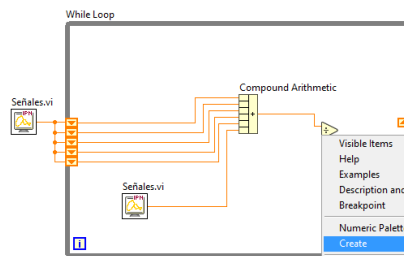


Block Diagram - Divide: Dividir dos Variables Numéricas Entre Sí



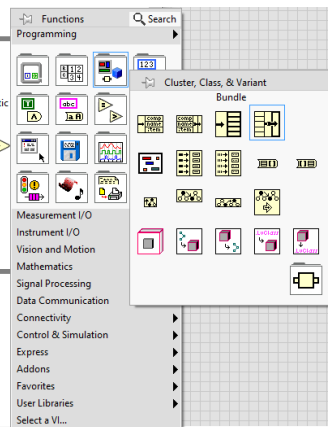
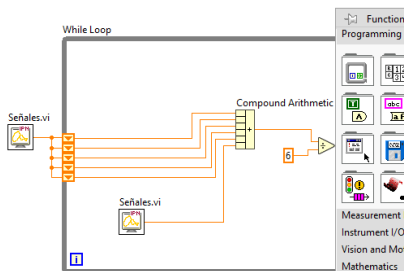


Crear una Constante para un Bloque: Clic derecho en la terminal del bloque de interés → Create → Constant. Constante para dividir toda la suma de los 5 registros de memoria del Shift Register entre un número, se divide entre 6 porque corresponde a 1 señal original + 5 registros de memoria.

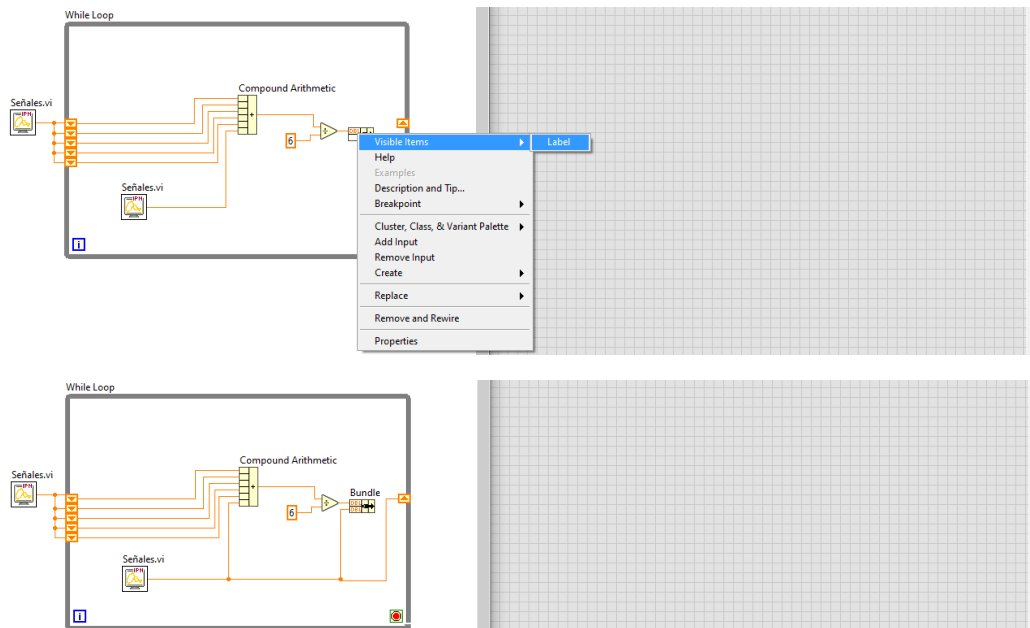


Block Diagram - Bundle: Juntar Varios Tipos de Datos Para Mandarlos a un Cluster

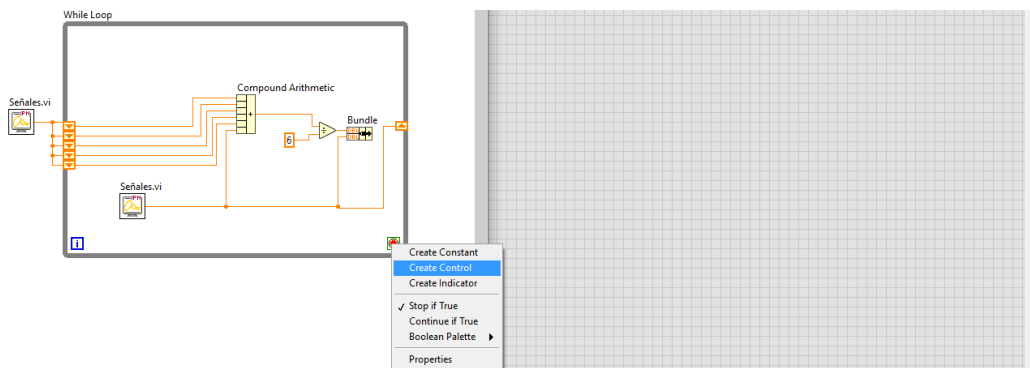
El tipo de dato Cluster es perteneciente únicamente al entorno de desarrollo de LabVIEW y representa un tipo de dato definido por el usuario que recibe y encapsula varios, pero para poder realizar esto se debe incluir un bloque intermedio llamado Bundle, que se encarga de juntar todos los tipos de datos distintos o iguales antes de enviarlos al bloque de Cluster.



Mostrar nombre del bloque: Clic derecho → Visible Items → Label.

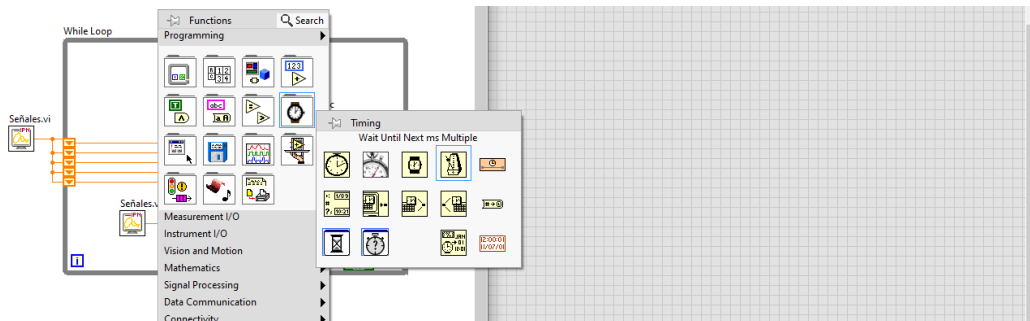


Crear un Control para un Bloque: Clic derecho en el bloque → Create Control.

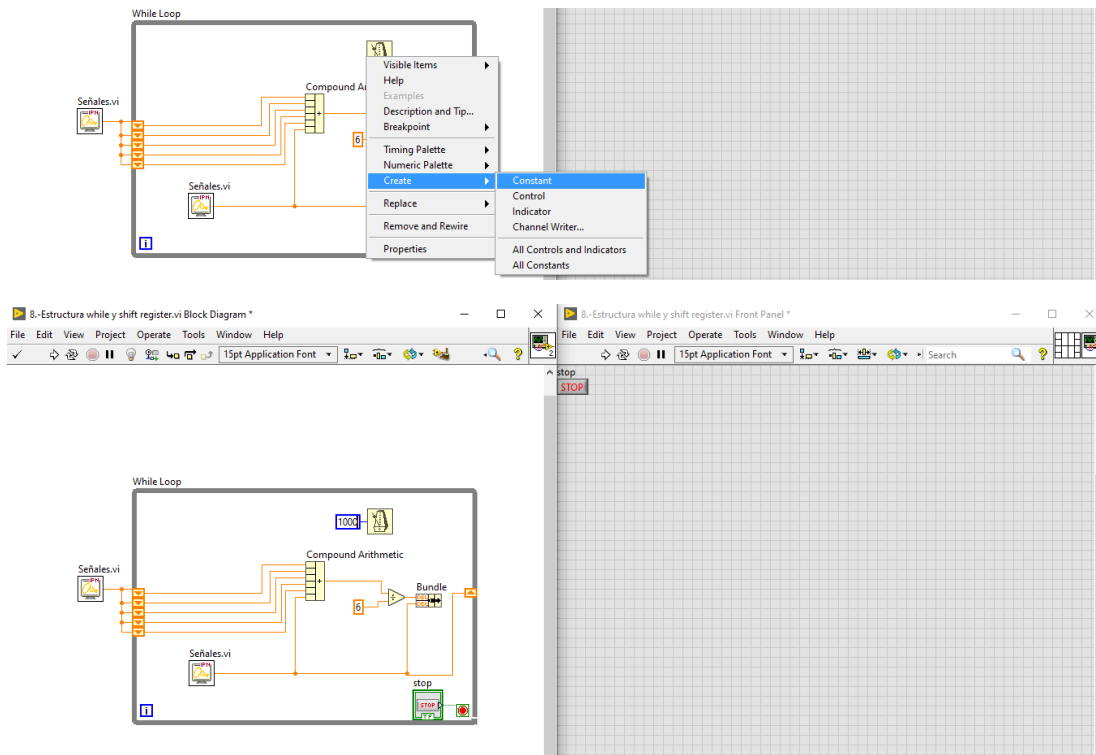


Block Diagram - Wait Until Next ms Multiple: Temporizador en milisegundos

El bloque de wait until se utiliza cuando se debe hacer un retraso de tiempo (delay) por ciertos segundos, para de esta manera parar la ejecución del programa por un cierto tiempo, en específico para que corra este bloque se debe crear una constante dando clic derecho sobre ella y declarando el tiempo en milisegundos.

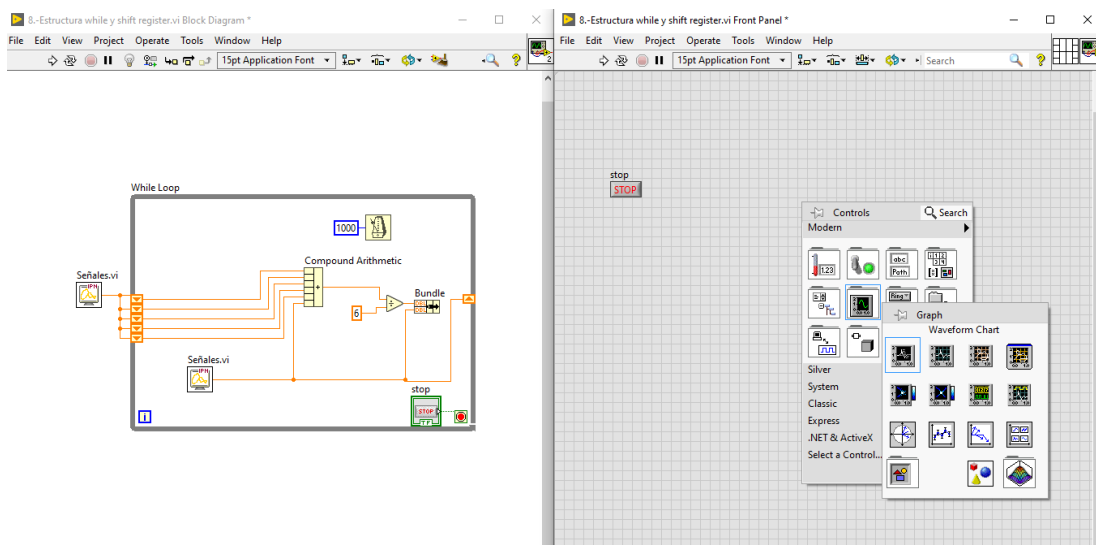


Crear una Constante para un Bloque: Clic derecho en la terminal del bloque de interés → Create → Constant. Tiempo de duración del temporizador en milisegundos.

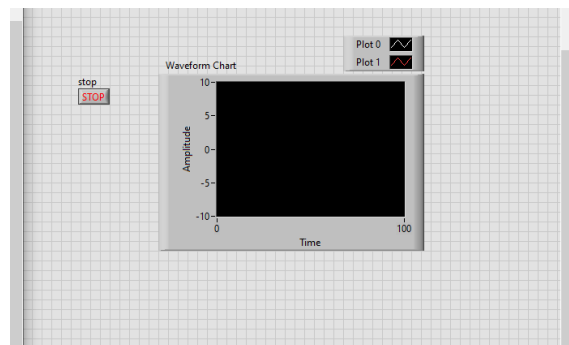
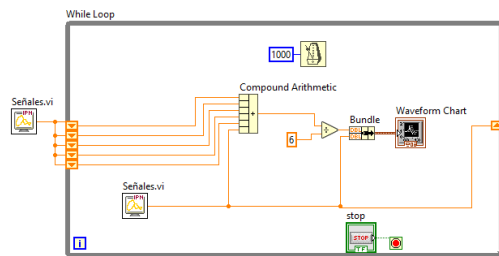
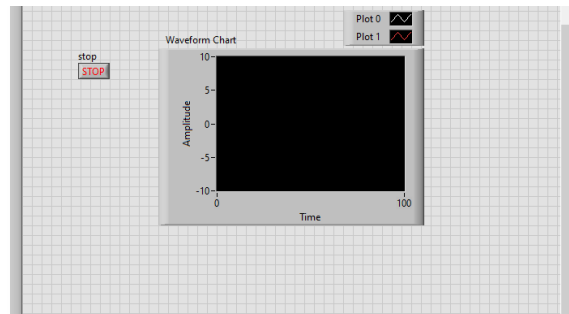
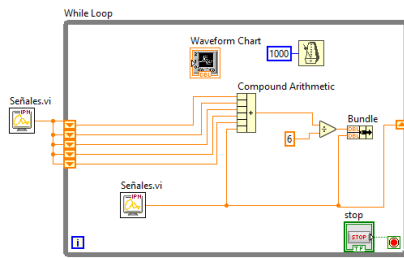
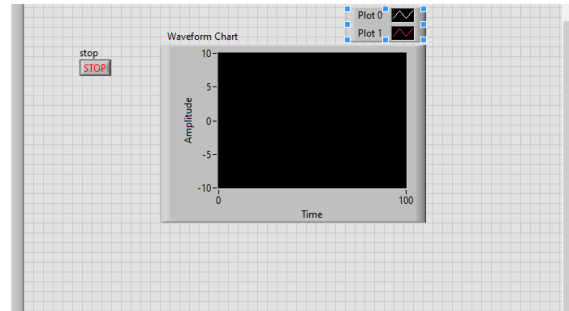
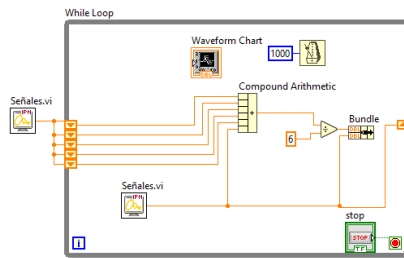


Front Panel - Waveform Chart: Ventana que Muestra una Señal (Dynamic Data)

El Waveform Chart muestra gráficas de tipo Dynamic Data, las cuales son señales virtuales o reales.



Al arrastrar el menú de la esquina superior derecha se accede a todas las gráficas del Waveform Chart (Dynamic Data), Plot 0 es el promedio de las 5 anteriores y la actual y el Plot 1 es la señal actual, que es la creada con el subVI Señales, por eso es que dividimos entre 6 la señal, para sacar el promedio de los Shift Register.



Ejecución del Programa: Señal SubVI y su Promedio de 5 Shift Register

