

INGENIERÍA MECATRÓNICA



DI_CERO

DIEGO CERVANTES RODRÍGUEZ

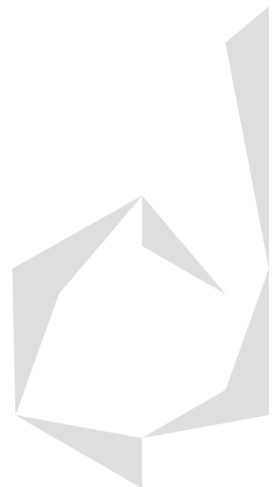
INSTRUMENTACIÓN VIRTUAL

NI LABVIEW 2020 (32-BIT)

Visión Artificial:
Cubo RGB en LabVIEW

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
Show Context Help: Descripción de Bloques y sus Terminales.....	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: Firma de una Imagen	9
Introducción Teórica – Firma de una Imagen	9
Programa: Escala de Grises y Binarizado – Identificación de Bordes	9
Programa: Escala de Grises y Binarizado – Identificación de Bordes	9
Programa: Medición de Distancias del Centroide al Contorno	10
Programa: Identificación de Figura - Correlación.....	10
Programa: Firma de una Imagen – Reconocimiento de Figuras.....	11



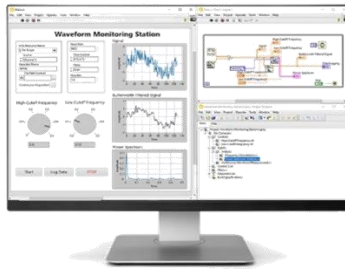
Introducción Teórica de LabVIEW:

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.

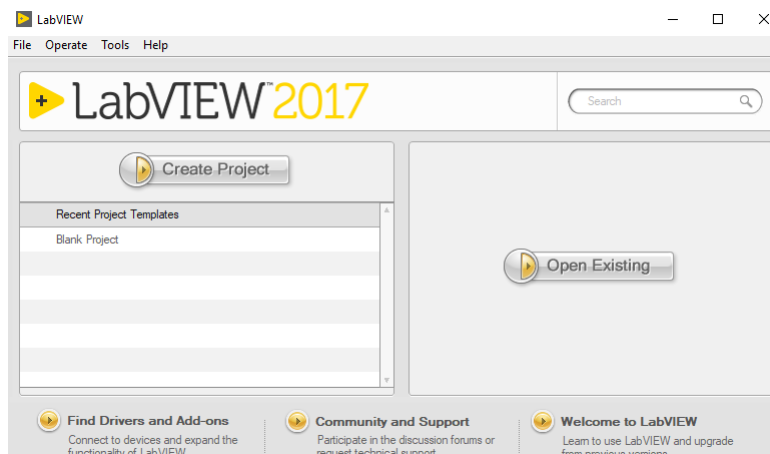
- **Instrumentación Tradicional:** El hardware es más usado, como por ejemplo con los circuitos integrados de un osciloscopio.
- **Instrumentación Virtual:** El software es el más utilizado y sus funciones son modulares, como lo es en una tarjeta de desarrollo de National Instruments.

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

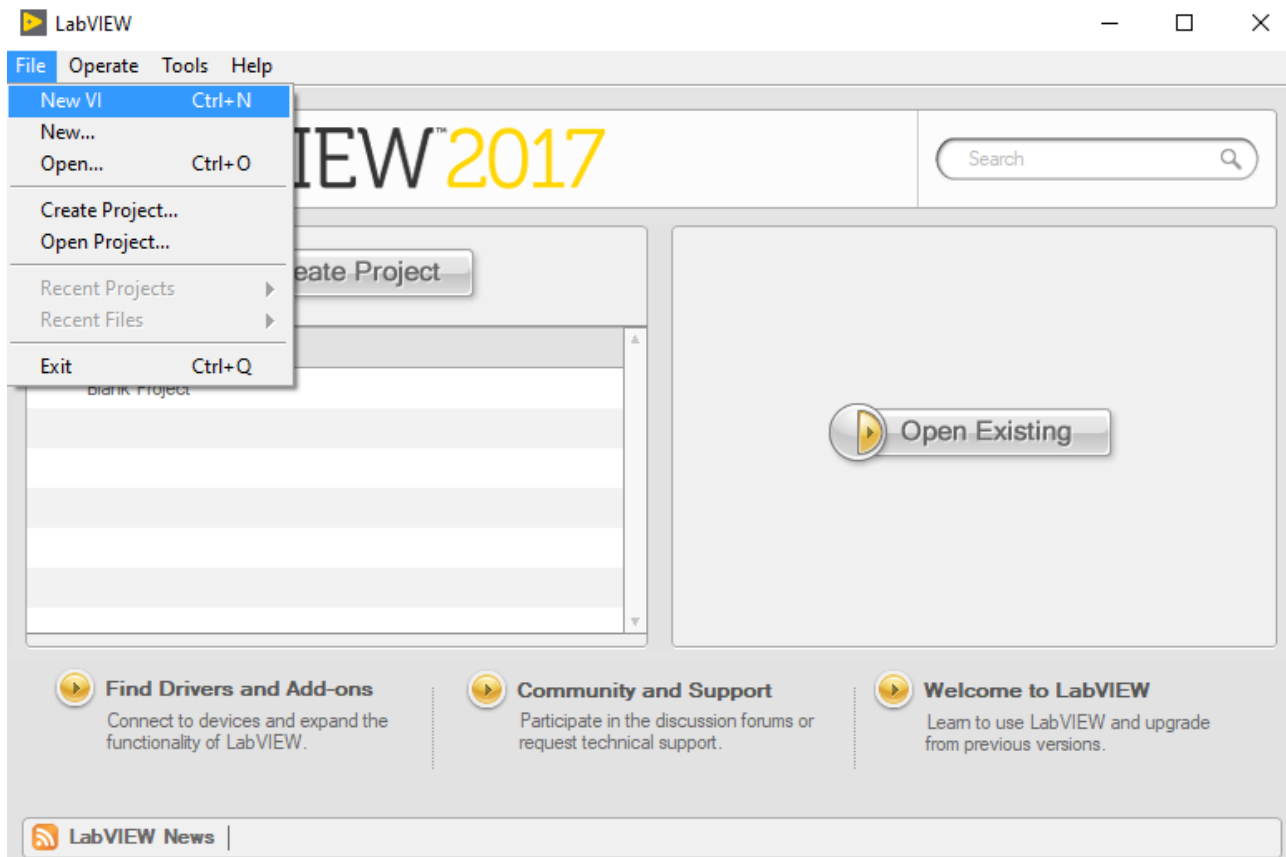


Introducción al Entorno de LabVIEW:

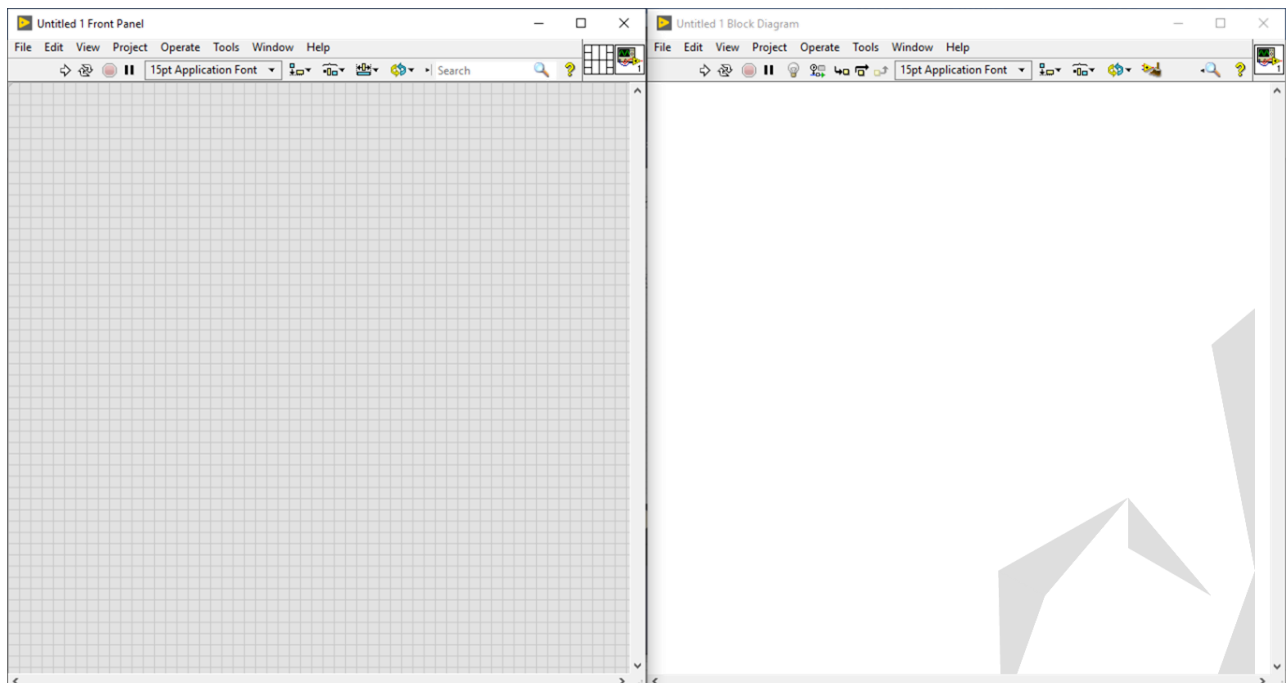
Un nuevo proyecto de LabView se abre por medio del botón de Create project que aparece inmediatamente cuando abra el programa.



VI se refiere a Virtual Instrument.

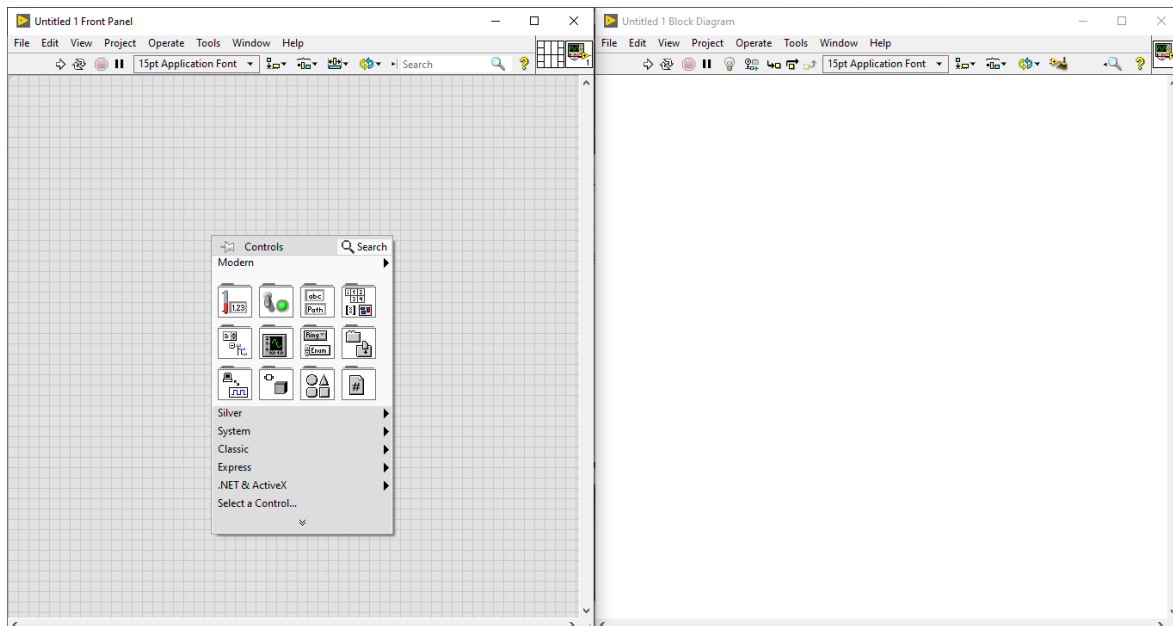


Al hacerlo me abrirá estas dos ventanas, en una de ellas se creará el programa con bloques (Ventana Block Diagram) y en la otra se verá la interfaz (Ventana Front Panel).



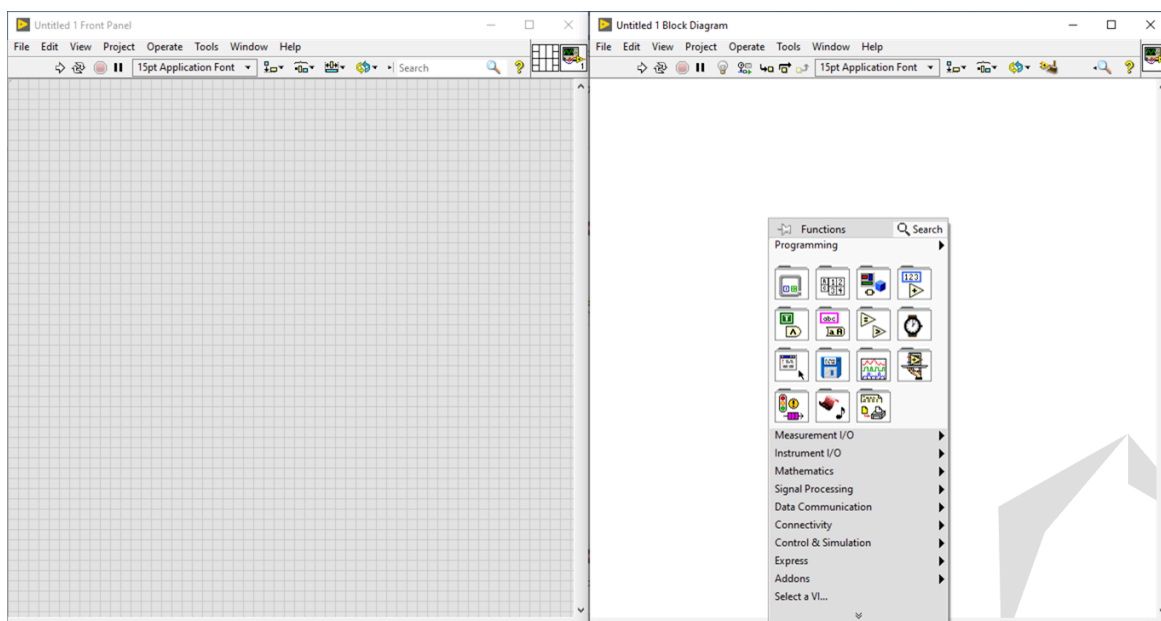
Front Panel: Ventana Gris con la Interfaz del Programa

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.



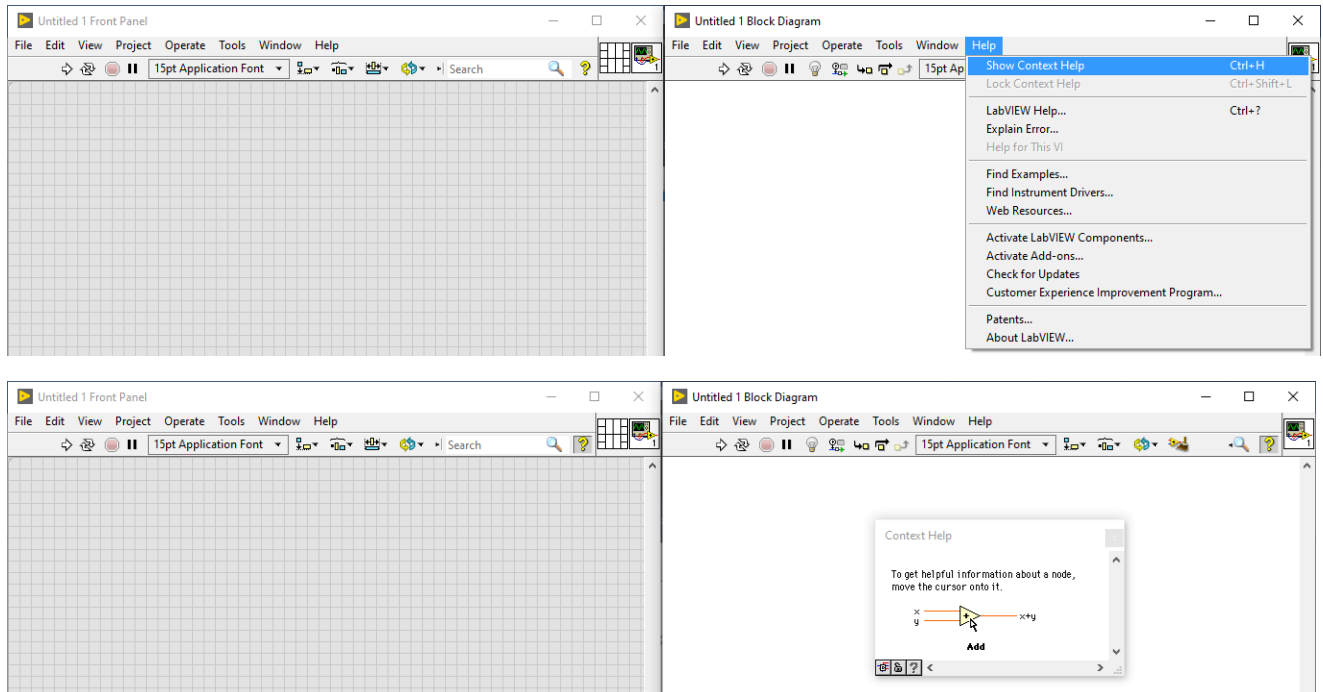
Block Diagram: Ventana Blanca con la Lógica del Programa (Bloques)

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.



Show Context Help: Descripción de Bloques y sus Terminales

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.



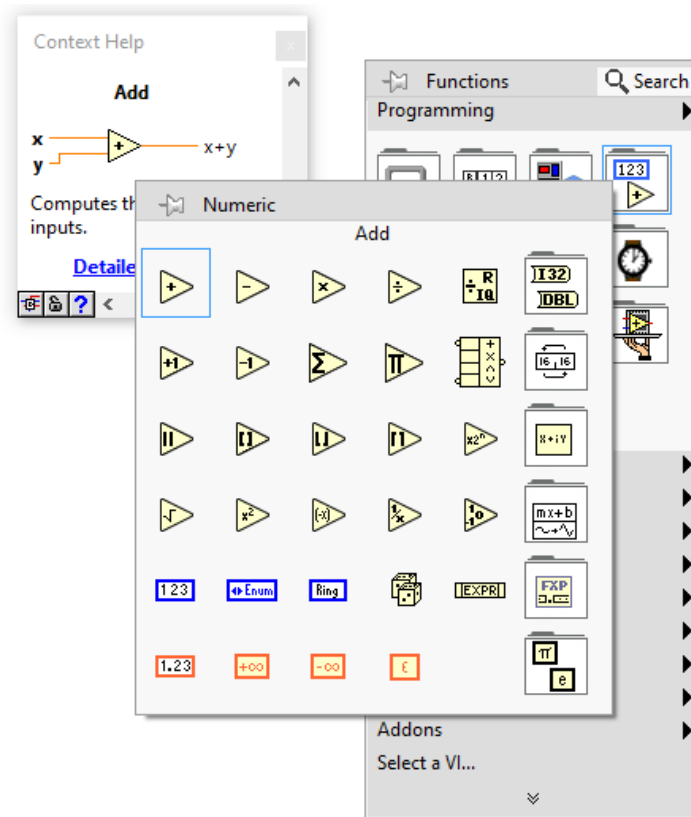
Las funciones o subrutinas son los elementos más básicos que pueden existir en LabView, dentro de ellas existe un código de bloque propio que describe sus funciones, pero además se cuenta con otros elementos:

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



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.

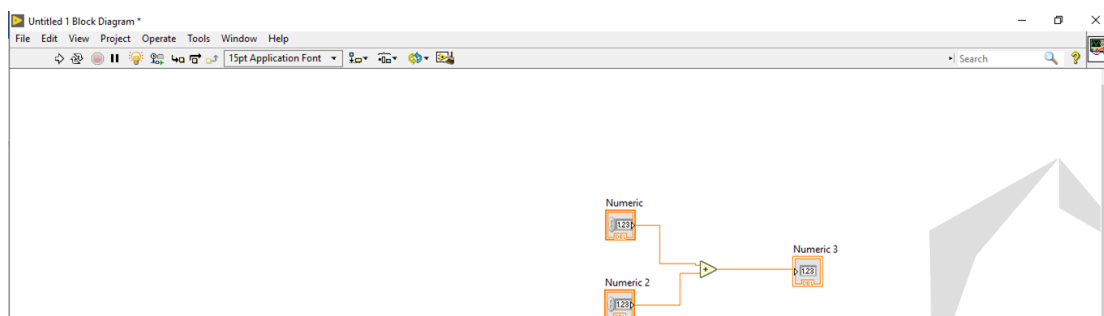


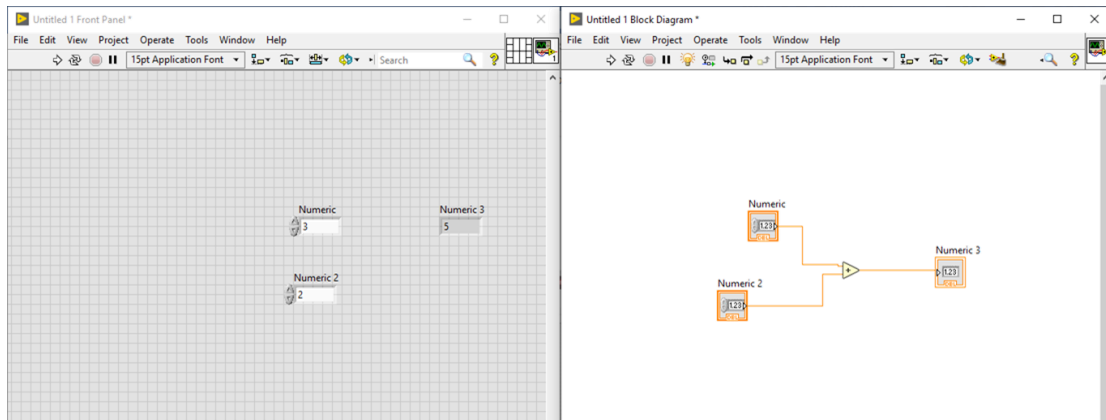
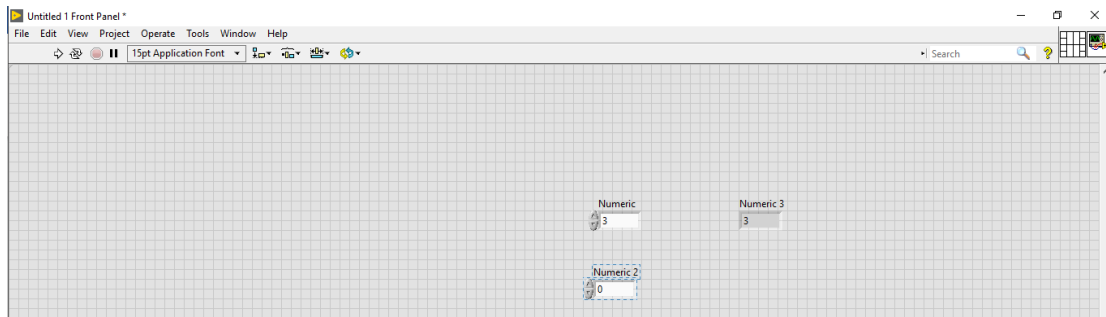
El programa es autocompilable, es decir que se corre por sí solo, por lo que si la flechita aparece rota es porque hay un error en el programa.



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

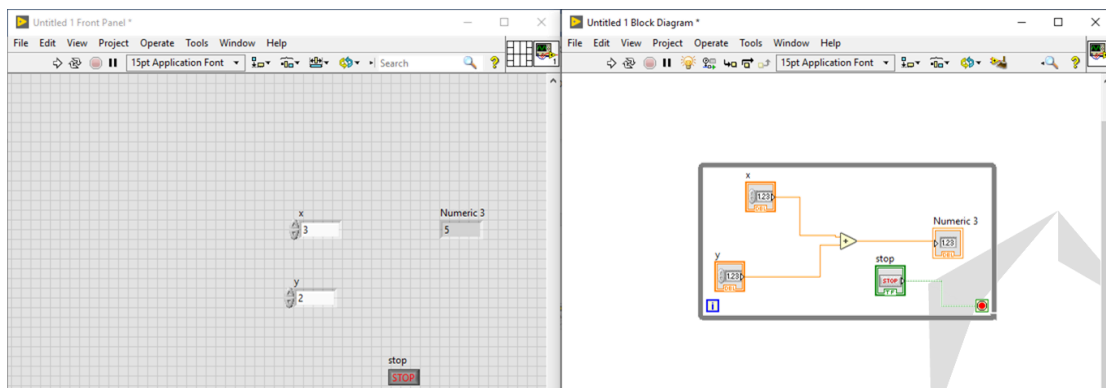
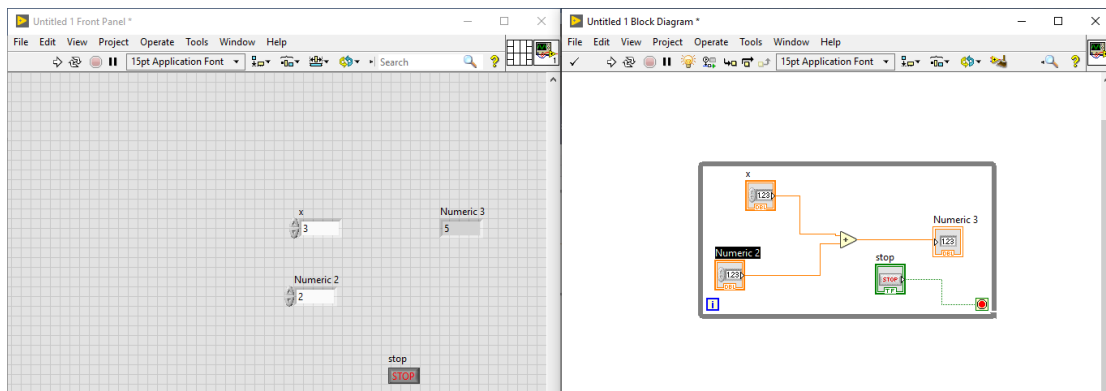
Al dar doble clic en el bloque de la pantalla blanca, me llevará al punto donde se encuentra el mismo bloque, pero en la pantalla gris.

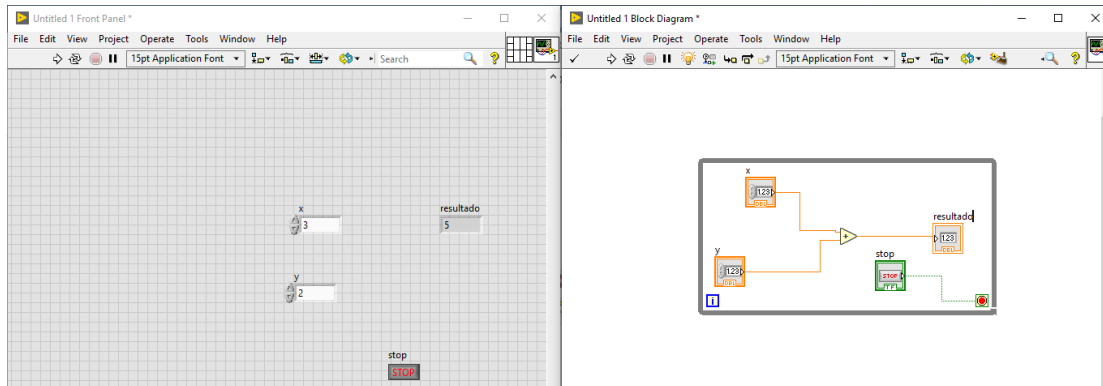




Block Diagram - Cambiar Nombre a los Bloques: Nombre de los elementos en el Front Panel

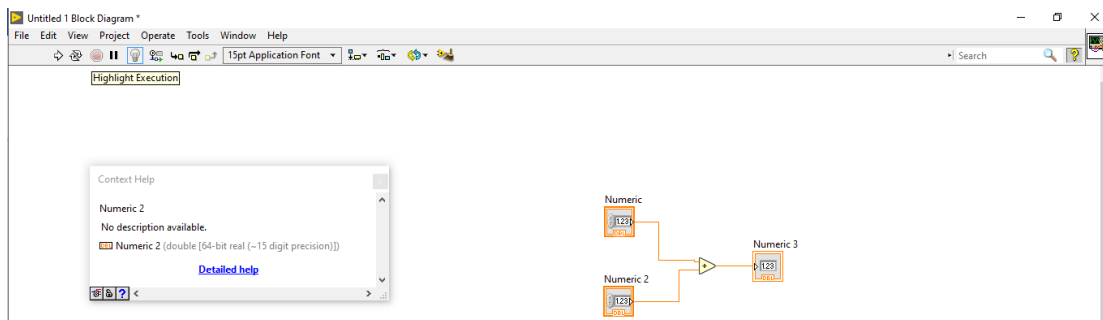
El nombre de los elementos de las interfaces se puede cambiar desde el Block Diagram, cambiándole literal el nombre a los bloques.





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.

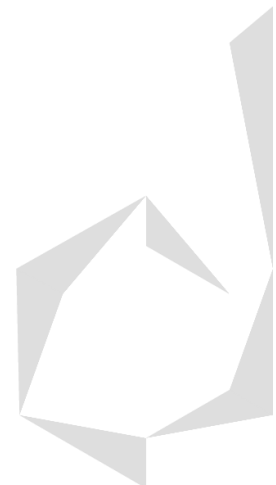


Coertion dot: Conversión Automática de Datos por Parte de LabVIEW

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.



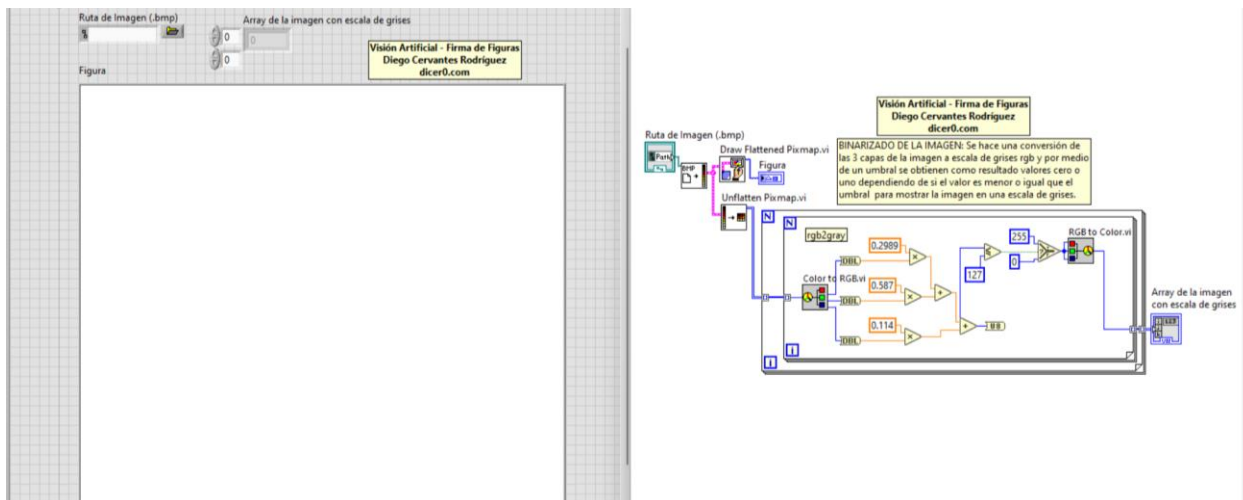
Programa: Firma de una Imagen

Introducción Teórica – Firma de una Imagen

La firma, también conocida como huella digital, es una representación única y característica de una imagen que se utiliza para identificarla o verificar su autenticidad. En realidad, la firma es un vector o gráfica que se crea al medir todas las distancias que se forman en una figura al medir desde su centroide hasta todos los puntos de su contorno, para ello es necesario identificar entonces el centroide y los bordos de la imagen, es más sencillo realizar este proceso con archivos de imagen con extensión .bmp porque su estructura es más fina.

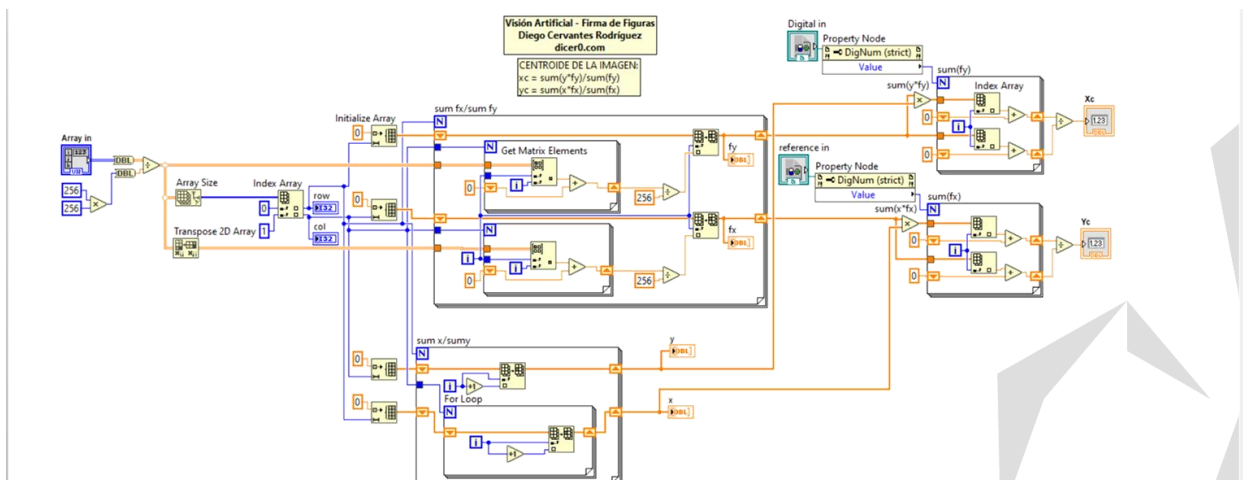
Programa: Escala de Grises y Binarizado – Identificación de Bordes

Primero que nada, se debe obtener la escala de grises de la imagen y binarizarla, para separar de mejor forma el contorno de la imagen.



Programa: Escala de Grises y Binarizado – Identificación de Bordes

Después se debe obtener el centroide de la figura ya binarizada.



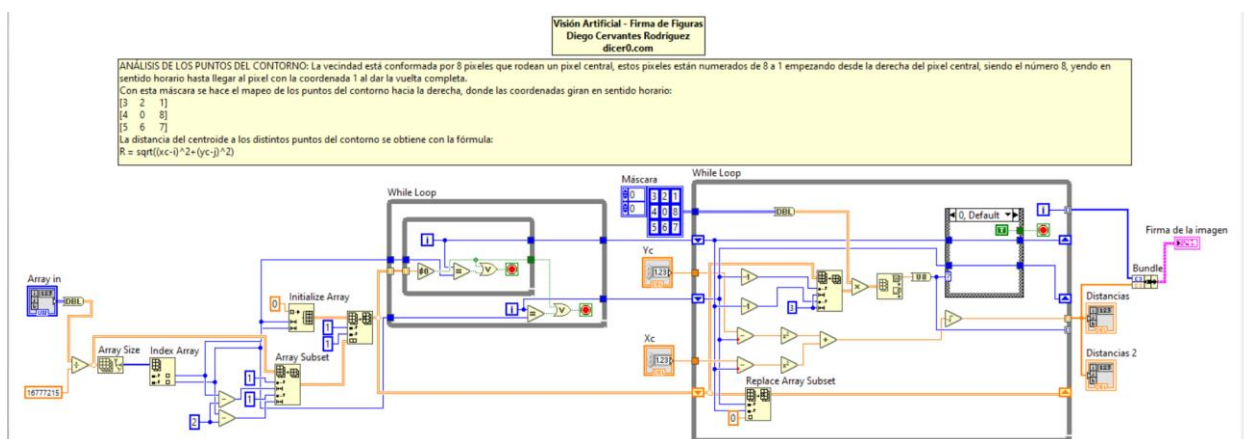
Programa: Medición de Distancias del Centroide al Contorno

Análisis de los Puntos del Contorno: La vecindad está conformada por 8 pixeles que rodean un píxel central, estos pixeles están numerados de 8 a 1 empezando desde la derecha del píxel central, siendo el número 8, yendo en sentido horario hasta llegar al píxel con la coordenada 1 al dar la vuelta completa. Con esta máscara se hace el mapeo de los puntos del contorno hacia la derecha, donde las coordenadas giran en sentido horario:

$$\begin{bmatrix} 3 & 2 & 1 \\ 4 & 0 & 8 \\ 5 & 6 & 7 \end{bmatrix}$$

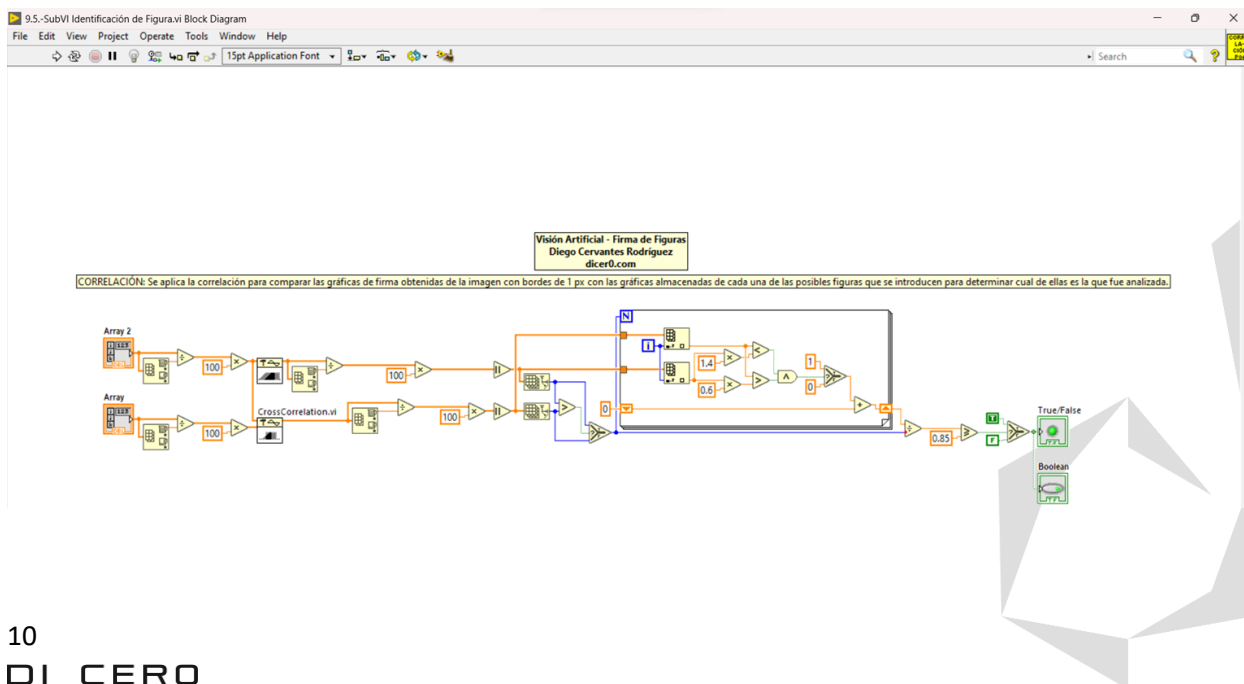
La distancia del centroide a los distintos puntos del contorno se obtiene con la fórmula:

$$R = \sqrt{(xc - i)^2 + (yc - j)^2}$$



Programa: Identificación de Figura - Correlación

Correlación: Se aplica la correlación para comparar las gráficas de firma obtenidas de la imagen con bordes de 1px con las gráficas almacenadas de cada una de las posibles figuras que se introducen para determinar cuál de ellas es la que fue analizada.



Programa: Firma de una Imagen – Reconocimiento de Figuras

