

Practicario de Visualización y Control de Procesos

Nombre: _____ Grupo: _____

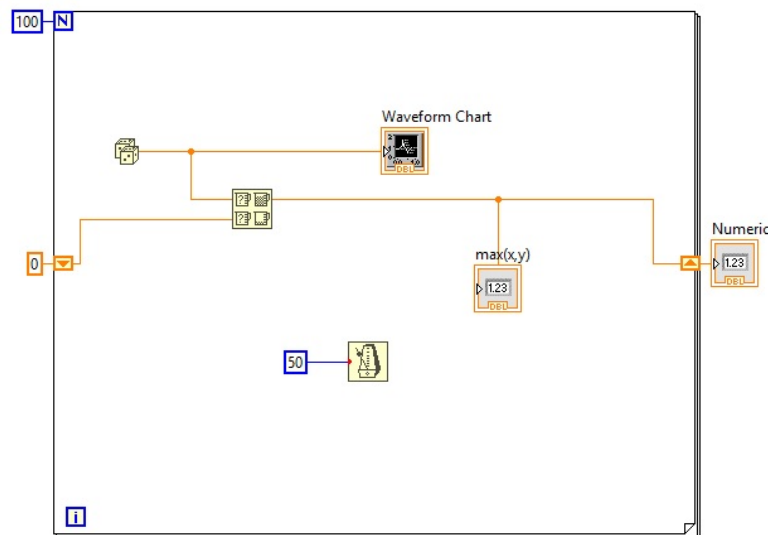
Prof.Dr.Enrique García Trinidad
Tecnológico Nacional de México
Tecnológico de Estudios Superiores de Huixquilucan
`enrique.g.t@huixquilucan.tecnm.mx`

Actividad 6

Realice un programa que permita encontrar el máximo de 100 números generados aleatoriamente. Represente gráficamente los números generados.

Solución:

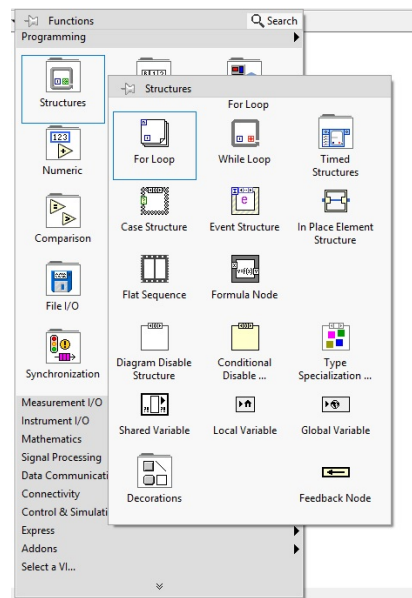
El ejercicio una vez terminado debe conducir a una representación en Lenguaje G como la que se muestra a continuación en la ventana de diagrama o código del LabVIEW:



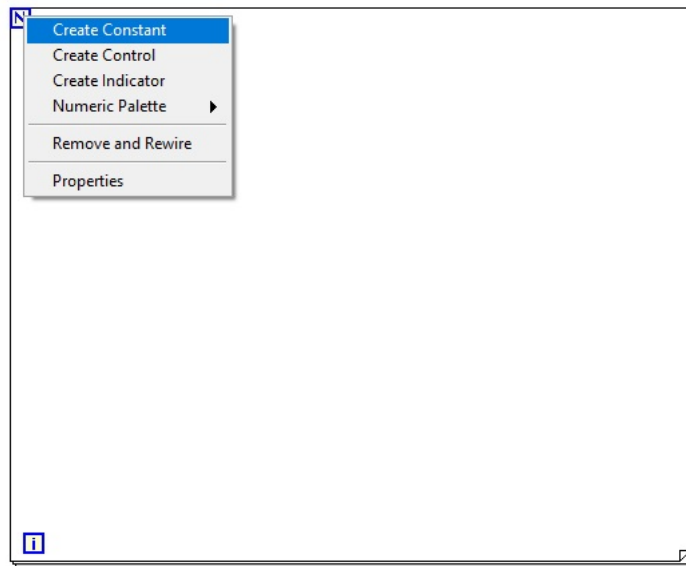
Veamos como desarrollar paso a paso dicho ejercicio. Primero, una explicación de lo que se nos pide. Se necesitan generar 100 números aleatorios, lo que implica utilizar un generador de números aleatorios y un lazo de control de programa o repetición que permita que ese generador de números aleatorios nos entregue 100 de dichos

números. Cada vez que se nos entregue un número, o sea, con cada iteración del bucle o lazo, se debe comparar el número generado con el número máximo hasta ahora generado. Eso nos indica que debemos realizar una comparación que busque el máximo y almacenarlo para comparar en cada iteración con el nuevo número. Se nos pide que representemos gráficamente dichos números por lo que debemos agregar un gráfico a nuestro diseño encargado de plotear en cada iteración el número aleatorio a procesar. Para realizar este programa debemos hacer las siguientes operaciones:

1. Primero debemos crear el lazo de control o bucle. Para ello seleccionamos de la paleta de funciones del LabVIEW las funciones de estructuras y dentro de ella la del ciclo **For**. El ciclo **For** es el escogido porque en este caso necesitamos repetir la operación de comparación en la búsqueda de un máximo 100 veces, invariablemente. Si estuviéramos sujetos a alguna condición entonces deberíamos de haber seleccionado un lazo condicional, o sea, un **While**. La siguiente figura ilustra cómo hacer la selección:



2. Una vez que se ha puesto el ciclo **For** debemos definir el alcance del mismo. Se recuerda que un ciclo **For** en LabVIEW siempre va desde 0 hasta $N - 1$, siendo N el valor a definir. Aquí queremos realizar las operaciones 100 veces por lo que debemos definir para N un valor de 100. Esto se realiza creando una constante y conectándola con N . Es muy sencillo, la siguiente figura ilustra como hacerlo. El menú que ahí aparece se obtiene dando clic derecho encima de N :

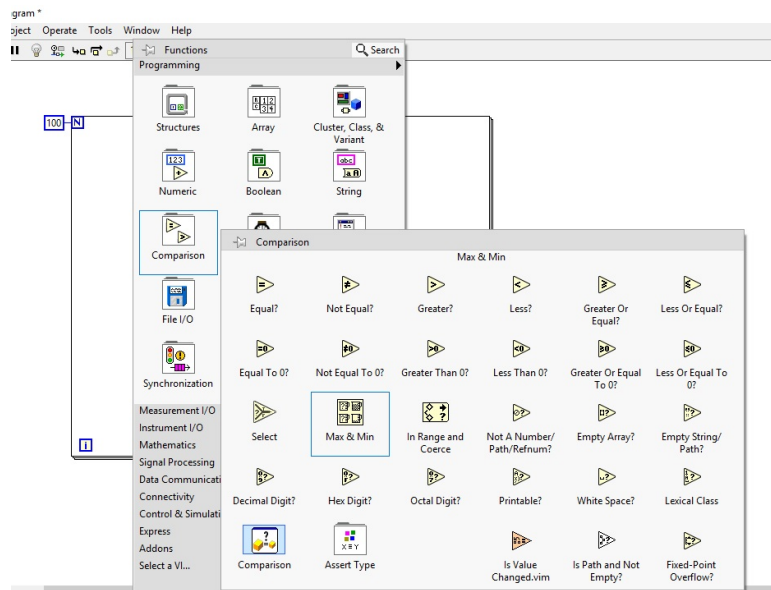


3. Cuando la constante se crea, aparece con valor 0. Debemos cambiar dicho valor y poner 100, como se muestra en la siguiente figura:

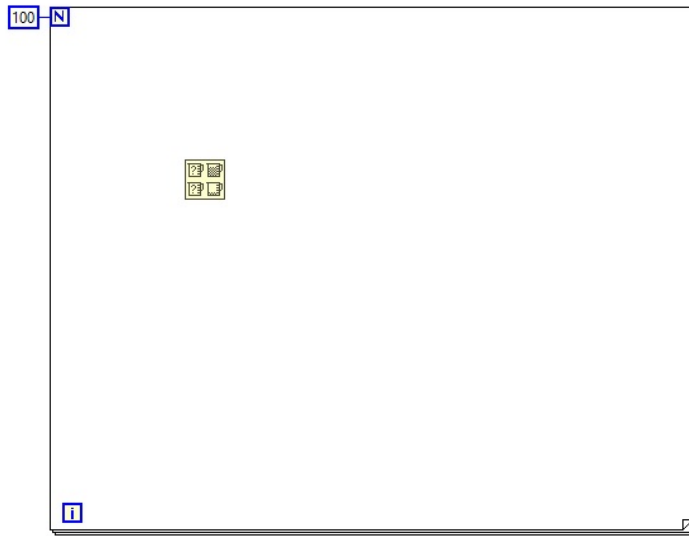


4. Necesitamos ahora generar un número aleatorio y compararlo con algún máximo. Por supuesto, el primer número aleatorio no tiene referencia de comparación, pero como debemos comparar dentro del ciclo for, basta con compararlo con el menor

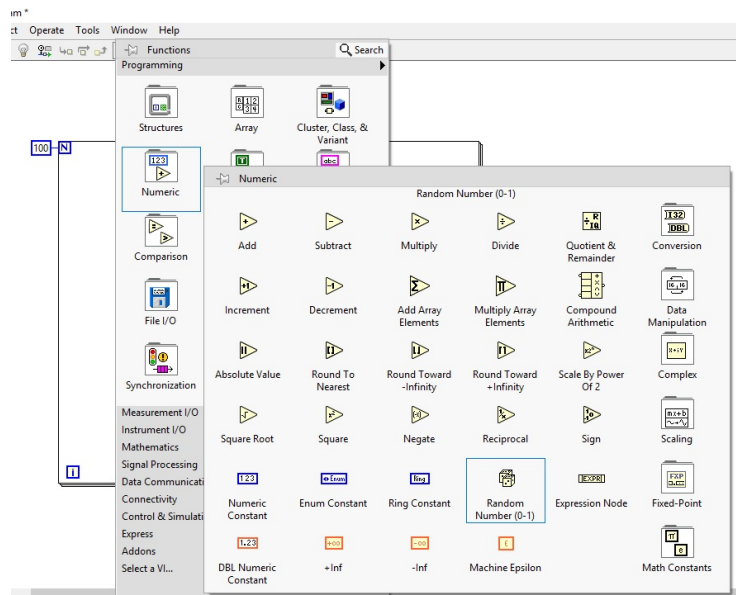
valor posible que se pudiera generar, en este caso 0, con lo cual garantizamos que el primer número aleatorio siempre será el primer máximo contra el cual comparar después los sucesivos que se generarán. LabView presenta un componente que realiza la comparación entre dos fuentes o números, y devuelve el mínimo valor de entre los dos y el máximo. La siguiente figura muestra como seleccionar dicho componente Functions/Comparison/Max & Min:



Se inserta en la ventana de diagrama dentro del ciclo `for`:



5. Necesitamos ahora encontrar alguna función que genere números aleatorios. LabVIEW tiene un componente en **Functions/Numeric/Random Number (0-1)**:



Si quisiéramos generar un número aleatorio en un rango diferente podemos multiplicar el número generado con esta función por el valor máximo que queremos generar. Puede modificar el ejercicio como estudio independiente ahora intentando buscar el

máximo de 100 números aleatorios en el rango de 0-50. Una vez introducidos en el diagrama el generador de números aleatorios y la función que devuelve el máximo entre dos números debemos entonces buscar la forma de almacenar el máximo que resulta de cada iteración para compararlo con el número generado en la siguiente iteración. Si se analiza bien esto lo que necesitamos es comparar el número generado en la iteración i (denotemos por $n(i)$ a dicho número) con el máximo que resultó de la iteración $i - 1$. Esta es una de las funciones que puede brindarnos el registro de desplazamiento o **Shift Register** de LabVIEW.

El **Shift Register** desde el punto de vista de la programación es una estructura tipo arreglo asociada con lazos de control o bucles de programa como el **for** y el **while** y que almacena valores anteriores producidos en dichos lazos. Así, por ejemplo, si obtenemos como resultado de una iteración i un valor $v(i)$ al final de dicha iteración, al comienzo de la próxima al i incrementarse en uno, dicho valor se almacena como $v(i - 1)$. Dos iteraciones hacia delante será $v(i - 2)$ y así sucesivamente. La siguiente tabla ilustra dicho concepto con ejemplos numéricos:

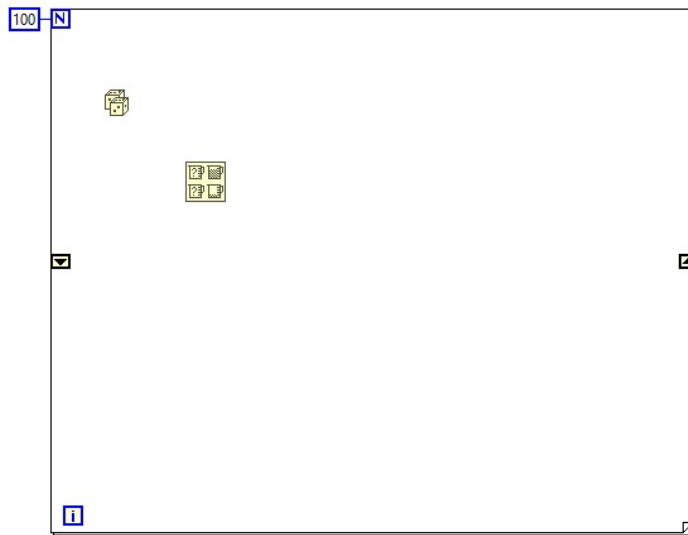
Iteracion	$V(i)$	$V(i - 1)$	$V(i - 2)$	$V(i - 3)$	$V(i - 4)$	$V(i - 5)$
0	5	-	-	-	-	-
1	3	5	-	-	-	-
2	0.7	3	5	-	-	-
3	1	0.7	3	5	-	-
4	-1	1	0.7	3	5	-
5	0	-1	1	0.7	3	5
6	2	0	-1	1	0.7	3

y así sucesivamente....

6. Para adicionar **Shift Registers** con LabVIEW se selecciona la estructura **for** y se da clic derecho sobre ella. El siguiente menú aparece:

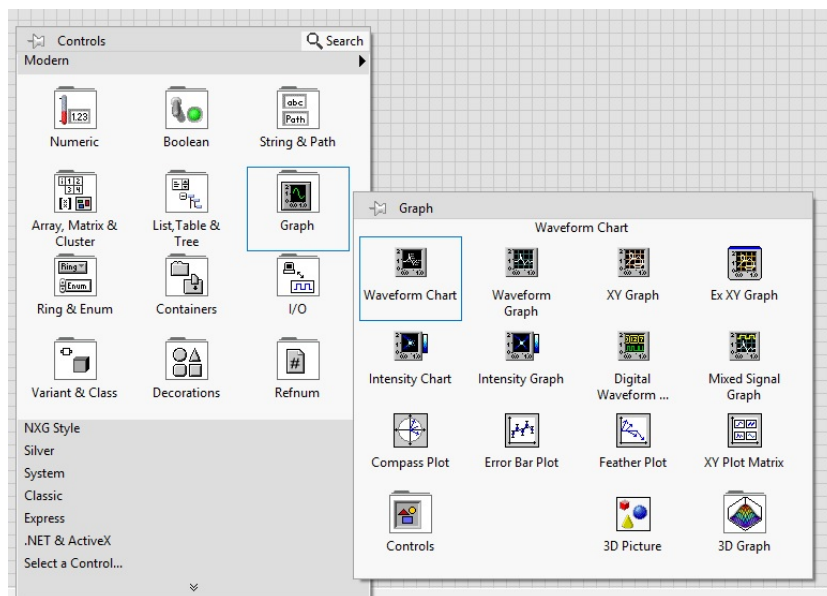


Al seleccionar de dicho menú la última opción, **Add Shift Register** (Adicionar registro de desplazamiento) se inserta en el diagrama como se muestra en la siguiente figura:

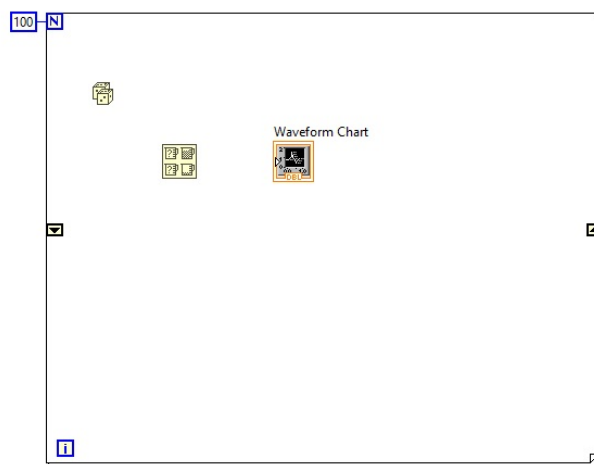
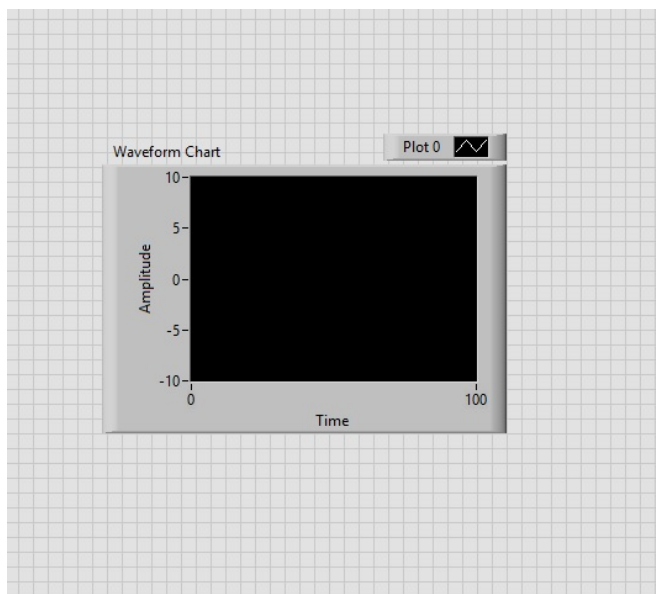


7. Antes de realizar las conexiones entre los elementos debemos además seleccionar algún componente que nos permita visualizar gráficamente los números aleatorios

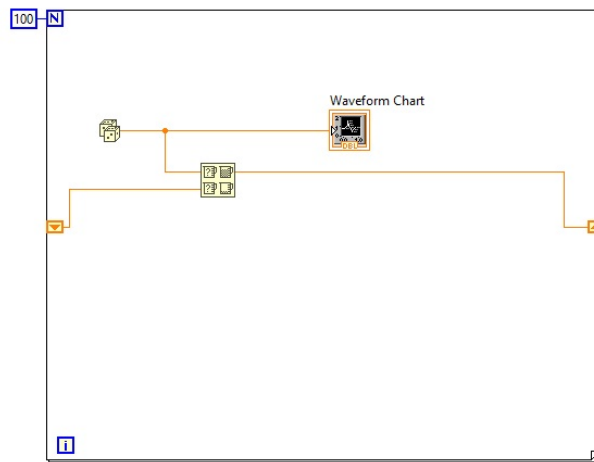
que se van generando. Para ello nos vamos al Panel Frontal y con clic derecho visualizamos el siguiente menú:



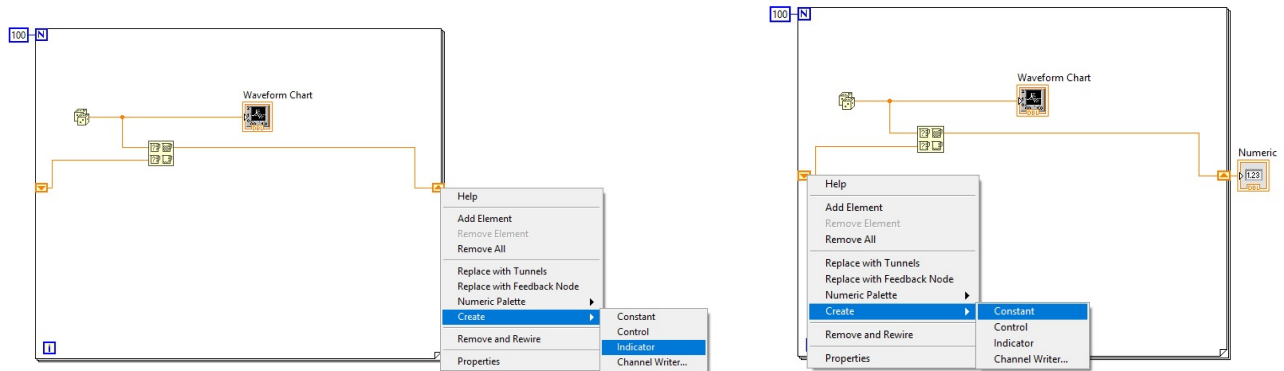
Al seleccionar la opción **Controls/Graph/Waveform Chart** obtenemos un gráfico como el que se representa en la pantalla frontal de la siguiente figura, donde también se representa el contenido de la ventana de diagrama hasta este momento:



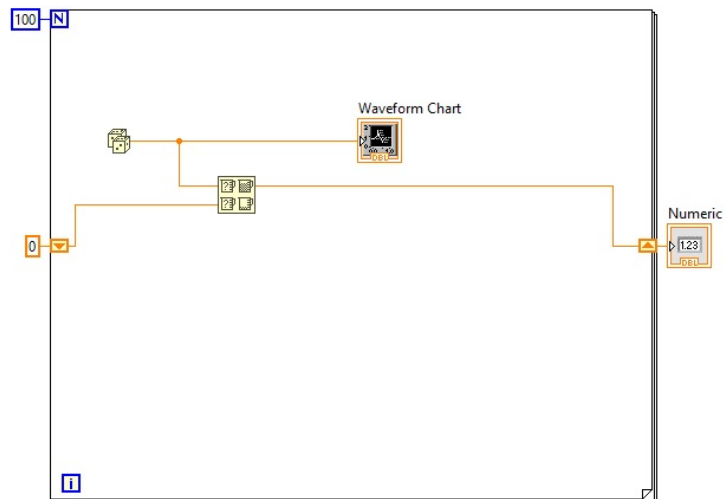
8. A continuación se realizan las conexiones entre los elementos. La salida del generador de números aleatorios se conecta al bloque de comparación de números, y a la entrada del control gráfico. El **Shift Register** de la derecha debe almacenar el último valor del máximo calculado y que se obtiene como salida del módulo de comparación. La siguiente figura muestra las conexiones a realizar:



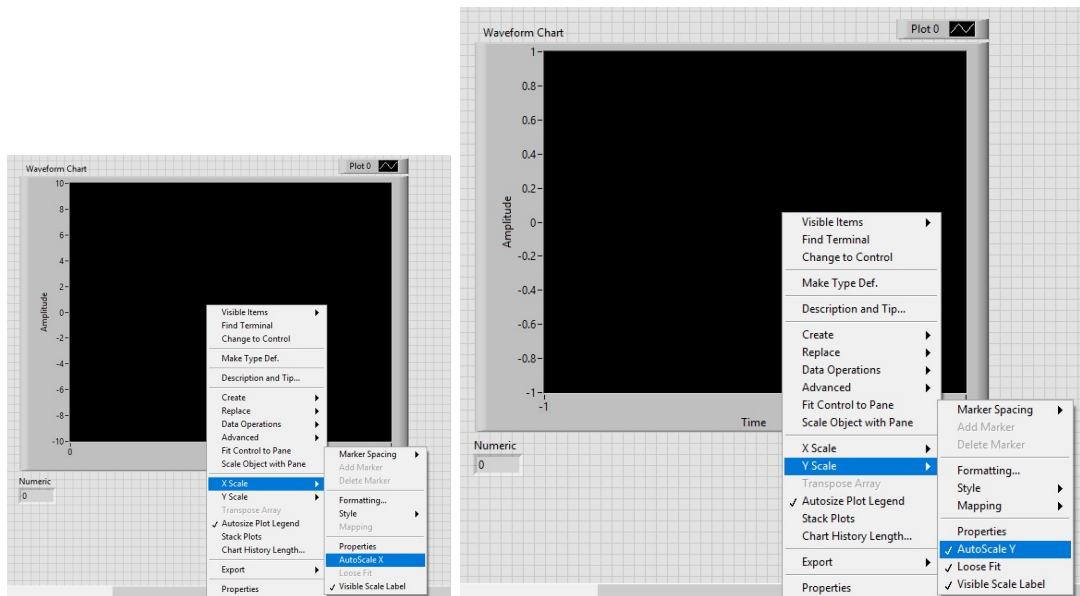
9. Una vez realizadas estas conexiones el ejercicio está prácticamente concluido. Note que la primera vez que se compara en el bloque de comparación se hace entre un primer número aleatorio y el valor del **Shift Register**. Ya se explicó como funciona el **Shift Register** y debemos notar que antes de concluir la primera iteración no tenemos un valor con el cual comenzar para el shift register que adquiere uno al final de cada iteración. Eso nos indica que tenemos que inicializarlo, darle un valor inicial con el cual empezar y evaluar la iteración 0 . Note además que cuando se llega a la última iteración el valor actual que contiene el shift register es el del último máximo calculado, y por tanto el valor que debemos reportar como resultado. Es por ello que entonces debemos crear un indicador para visualizar el último máximo y una constante para definir la condición inicial del **Shift Register**. Las figuras que se muestran a continuación ilustran como hacerlo. Los menús que allí aparecen se obtienen al dar clic derecho encima de los **Shift Registers**.



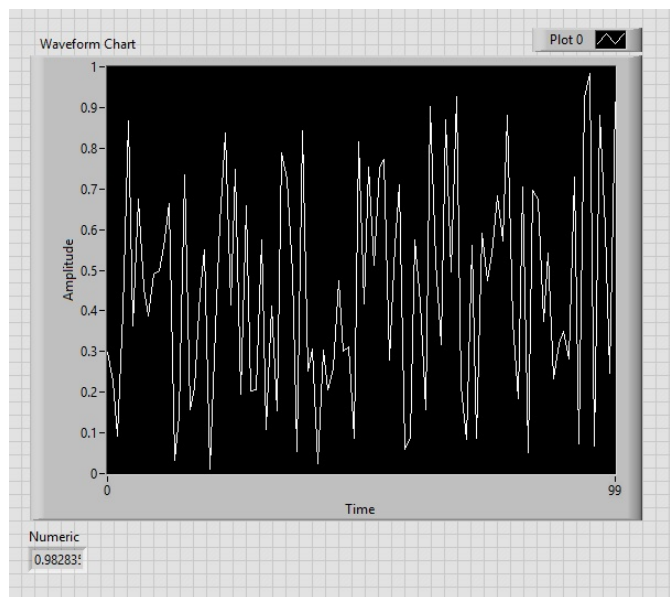
El resultado final se muestra en la siguiente figura. Note que el valor inicial del Shift Register es 0.00 pues es el mínimo que puede tomar el número aleatorio y no afectará la selección del primer máximo.



10. Antes de ejecutar el programa es conveniente modificar algunos parámetros del gráfico, en particular los relacionados con los ejes, que debemos modificar para permitir la auto escala. Las figuras a continuación muestran la manera de hacerlo (siempre clic derecho en el objeto para obtener el menú):

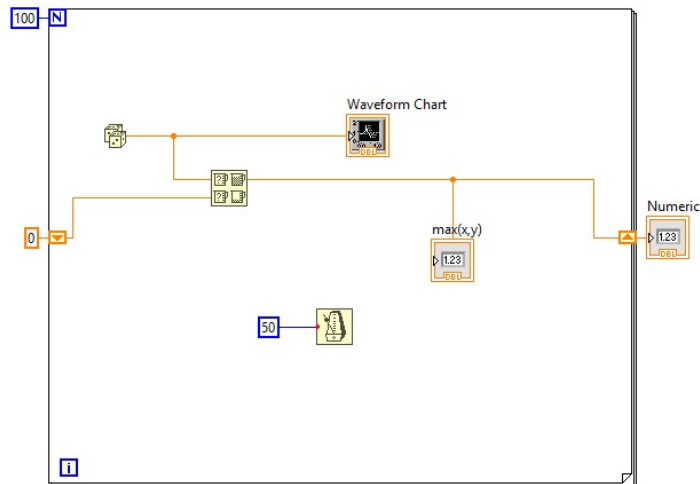


11. Finalmente salvamos el ejercicio y lo ejecutamos. Un ejemplo de ejecución se muestra en la siguiente figura:



Por último, señalar que si queremos que el gráfico se represente más lentamente podemos incluir una demora (Functions/Time & Dialog/ Wait Until Next ms Multiple)

o seleccionar el indicador de ejecución paso a paso. La siguiente figura muestra como quedaría el ejercicio para una demora de 50 milisegundos:



Entrega del reporte de actividades

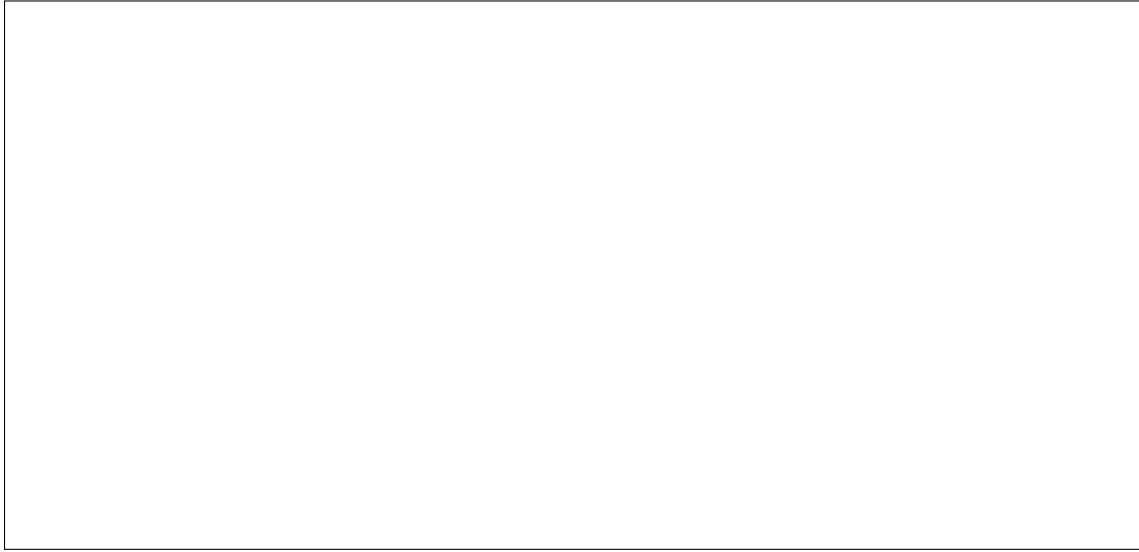
El reporte deberá ser entregado vía MS TEAMS en formato PDF, junto con un video del Instrumento Virtual funcionando, en la fecha establecida por el profesor. El reporte tiene un valor de 100pts.

Entrega del video

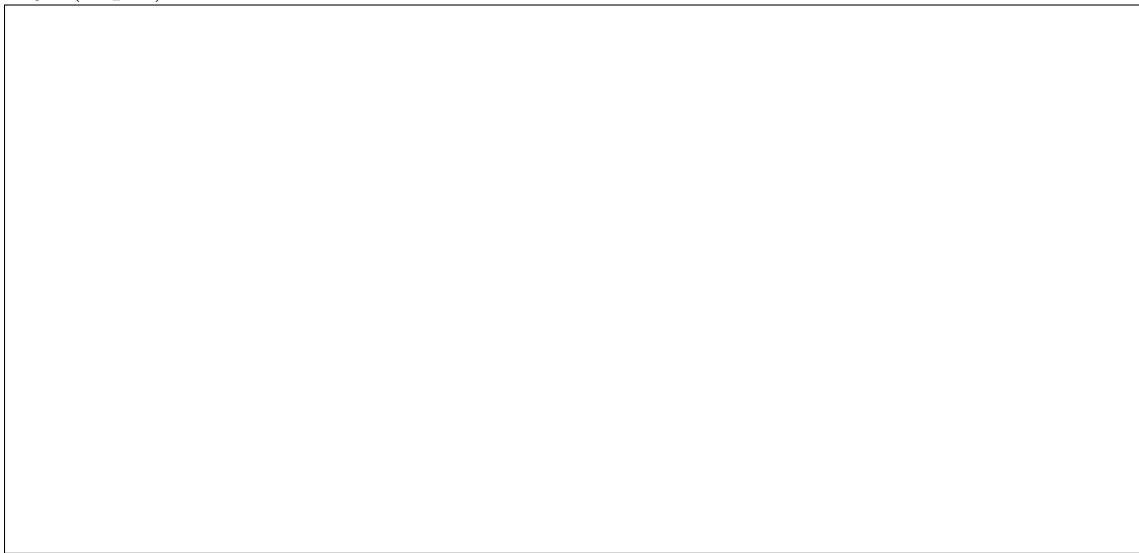
Subir a la plataforma MS TEAMS en la actividad correspondiente un video corto mostrando el funcionamiento del Instrumento Virtual desarrollado en este ejercicio.(20pts)

Panel Frontal y Diagrama de Bloques

Realice una captura de la pantalla del Panel Frontal y péguelo en la siguiente caja:(35pts)



Realice una captura de la pantalla del Diagrama de bloques y péguelo en la siguiente caja:(35pts)



Conclusiones

Redacte de manera breve los puntos más relevantes que le aportaron los conocimientos explorados a lo largo de la actividad.(10pts)

--

Evaluación del desempeño

Actividad:	Video	Panel Frontal	Diagrama de bloques	Conclusiones	Total
Puntos:	20	35	35	10	100
Calificación					