

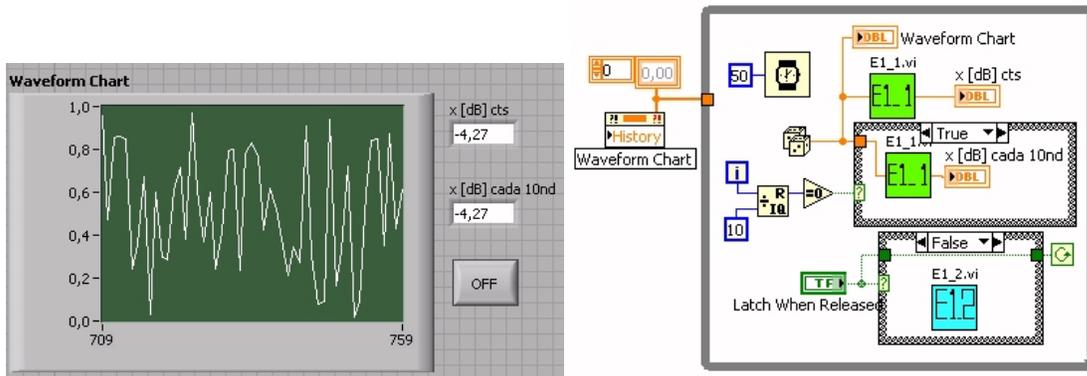
La edición de este material docente se ha realizado para ser utilizado como ejercicios suplementarios para las prácticas y actividades, que mediante el autoaprendizaje amplían y complementan el campo operativo de un curso generalista sobre instrumentación virtual . Sin una componente práctica significativa el ciclo de aprendizaje de cualquier asignatura no queda completo. Cada ejemplo está cuidadosamente diseñado para que el estudiante pueda emplearlo como punto de partida para su propia aplicación. El estudiante, en su proceso de aprendizaje debe asimilar los ejercicios básicos y posteriormente editar los suyos propios. Los 23 ejercicios se encuentran en el fichero ACTIVIDADES.llb el cual despliega su contenido cuando se abre en el entorno LabVIEW.

Ejercicio 1

1.- VI a emplear E1.vi

2.- Objetivo: Implementar un VI que represente número aleatorios, empleando dos subVIs: E1_1.vi y E1_2.vi

En las figuras adjuntas se muestra el panel frontal y el diagrama de bloques.

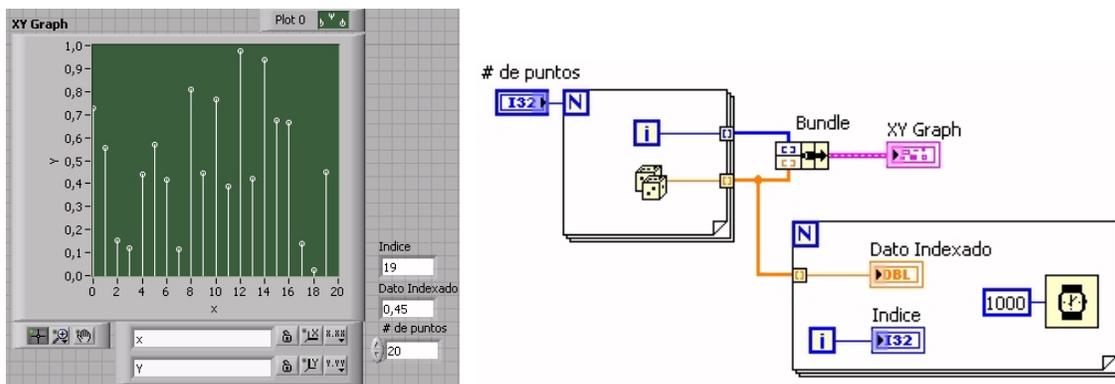


Ejercicio 2

1.- VI a emplear E2.vi

2.- Objetivo: Analizar la funcionalidad del autoindexamiento

En las figuras adjuntas se muestra el panel frontal y el diagrama de bloques.

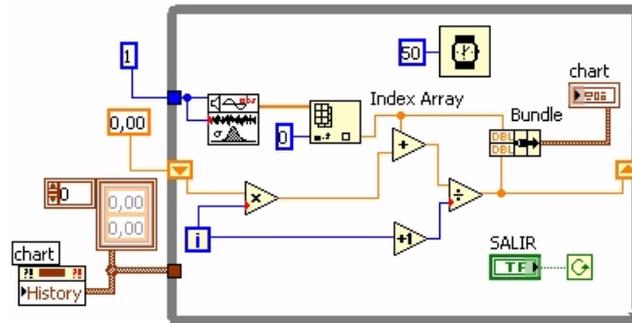
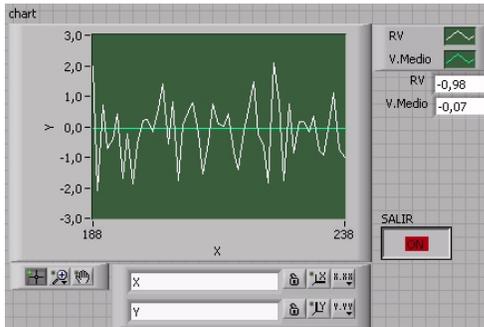


Ejercicio 3

1.- VI a emplear E3.vi

2.- Objetivo: Estudiar los tres tópicos siguientes: I) Registro de desplazamiento, II) Indexado de arrays, III) representación multicanal mediante la función **bundle**.

En las figuras adjuntas se muestra el panel frontal y el diagrama de bloques.

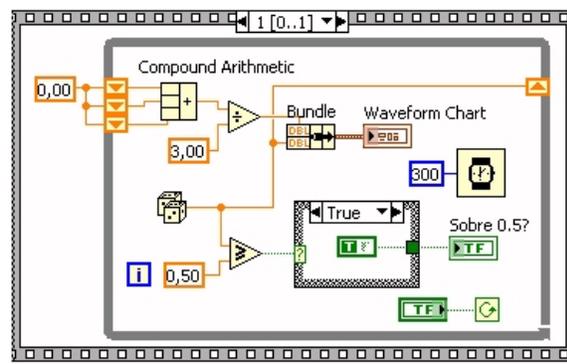
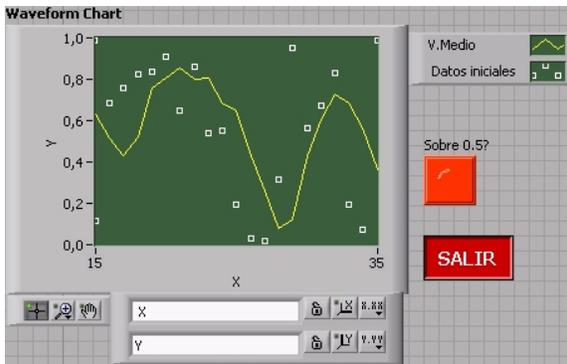


Ejercicio 4

1.- VI a emplear E4.vi

2.- Objetivo: Analizar la operatividad de una estructura **sequence** y el promediado mediante registros de desplazamiento case

En las figuras adjuntas se muestra el panel frontal y el diagrama de bloques.

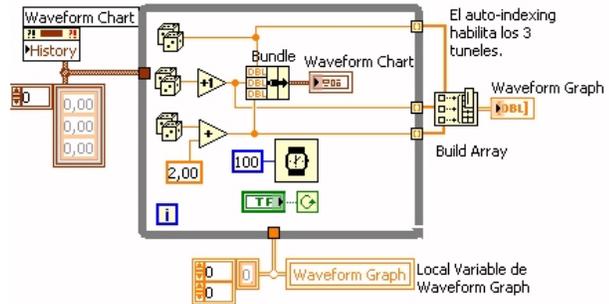
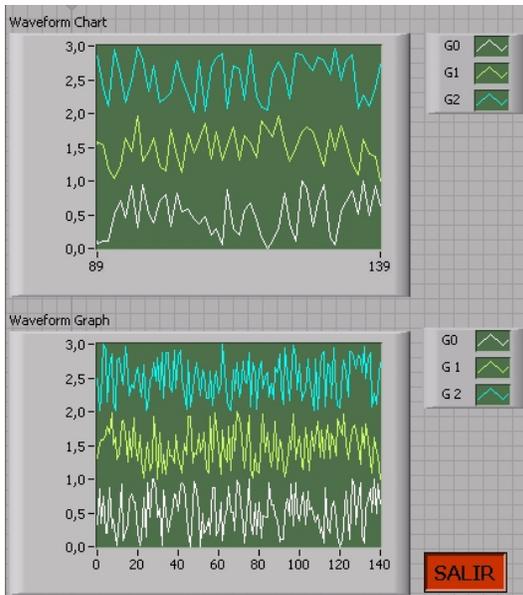


Ejercicio 5

1.- VI a emplear E5.vi

2.- Objetivo: Aprender la representación multicanal de datos mediante una **Waveform Chart** y **Waveform Graph**

En las figuras adjuntas se muestra el panel frontal de ambas gráficas y el diagrama de bloques.

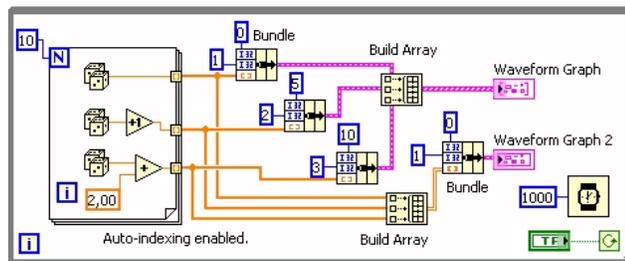


Ejercicio 6

1.- VI a emplear: E6.vi

2.- Objetivo: Aprender la representación multicanal de datos con distintos intervalos de representación

En las figuras adjuntas se muestra el panel frontal y diagrama de bloques.

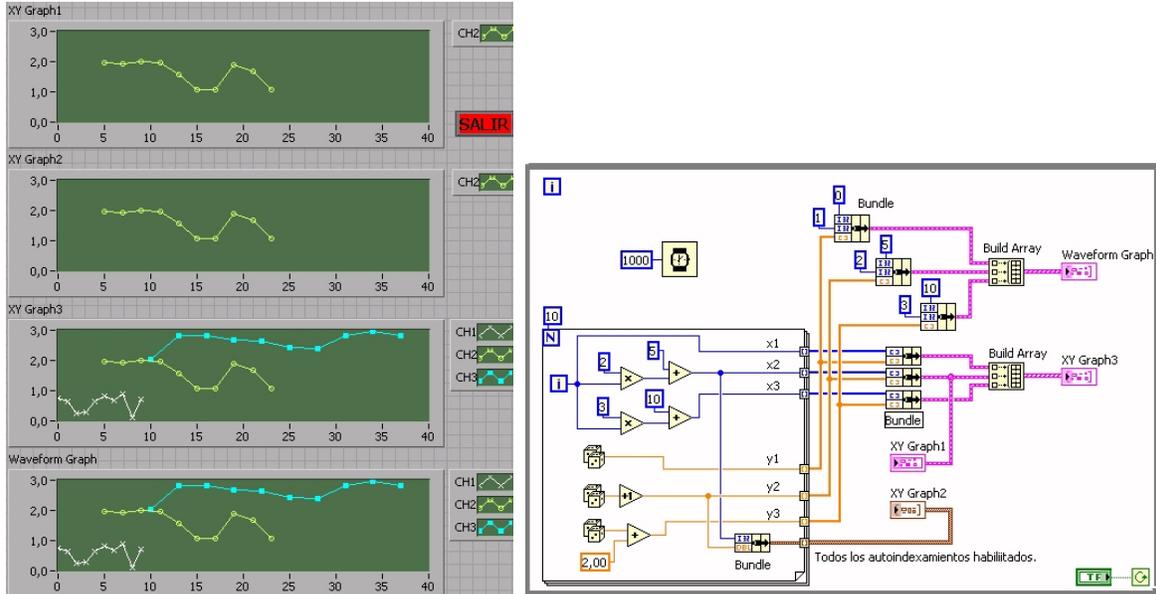


Ejercicio 7

1.- VI a emplear E7.vi

2.- Objetivo: Aprender la representación multicanal de datos mediante una **XY Graph**

En las figuras adjuntas se muestra el panel frontal y diagrama de bloques.

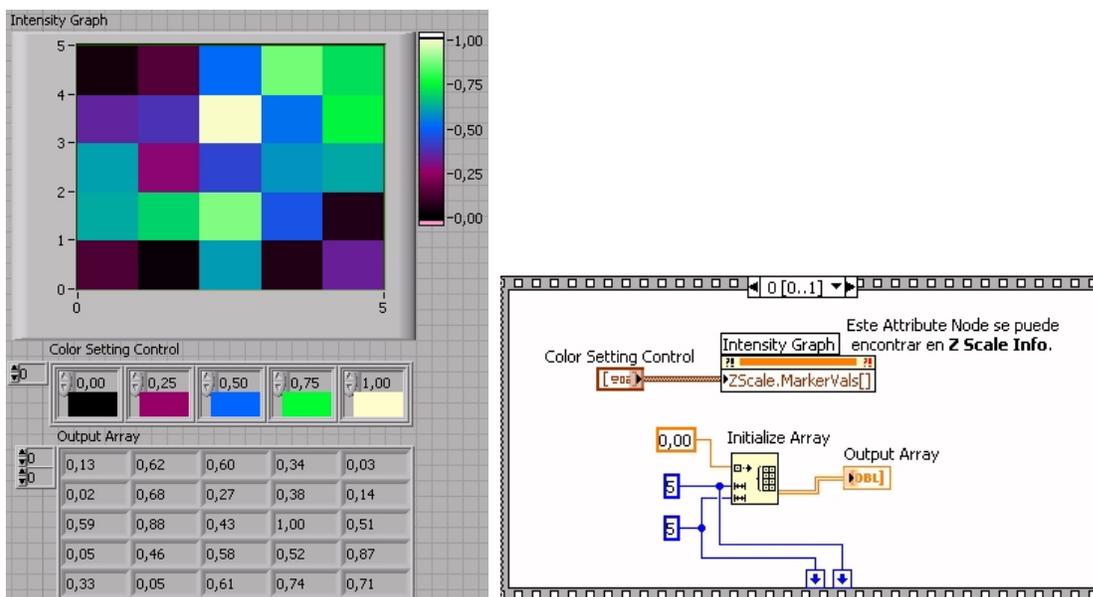


Ejercicio 8

1.- VI a emplear E8.vi

2.- Objetivo: Aprender a representar un array 2D mediante un gráfico de intensidad. Se emplea una estructura secuencia.

En las figuras adjuntas se muestra el panel frontal y diagrama de bloques.

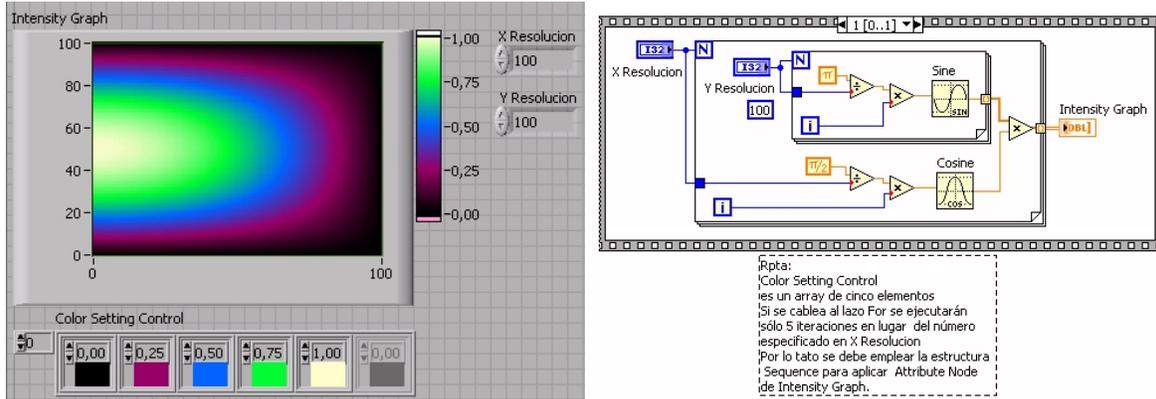


Ejercicio 9

1.- VI a emplear E9.vi

2.- Objetivo: Profundizar en el aprendizaje de la representación de un array 2D mediante un gráfico de intensidad. Se emplea una estructura secuencia.

En las figuras adjuntas se muestra el panel frontal y diagrama de bloques.

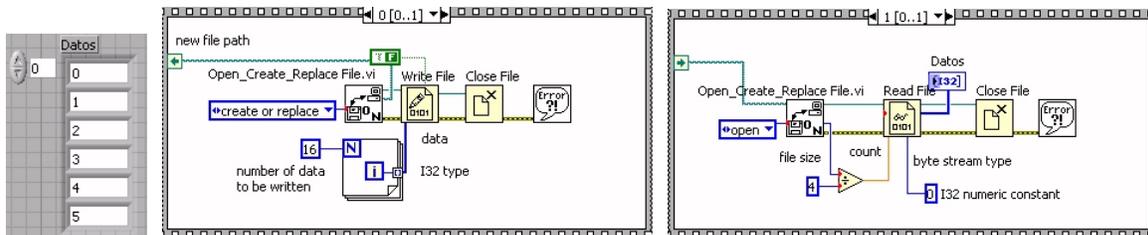


Ejercicio 10

1.- VI a emplear: E10.vi

2.- Objetivo: Aprender a almacenar y leer de un fichero un array de datos 1D.

En las figuras adjuntas se muestra el panel frontal y diagrama de bloques.

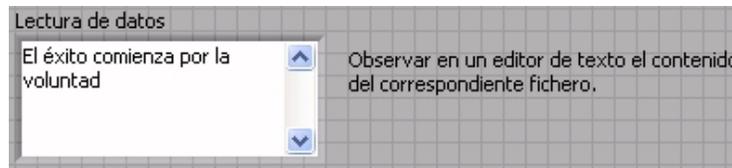


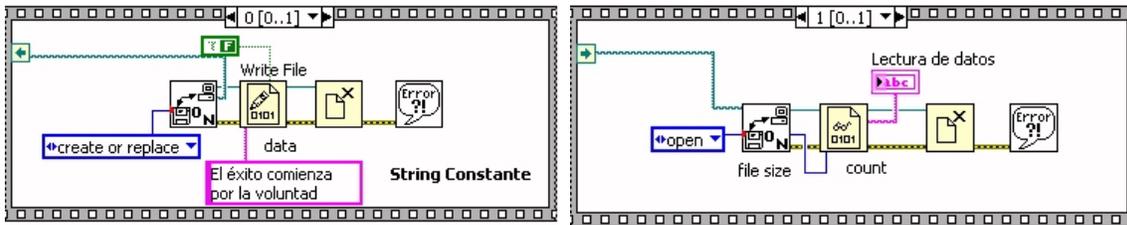
Ejercicio 11

1.- VI a emplear: E11.vi

2.- Objetivo: Aprender a almacenar y leer de un fichero de datos ASCII.

En las figuras adjuntas se muestra el panel frontal y diagrama de bloques.



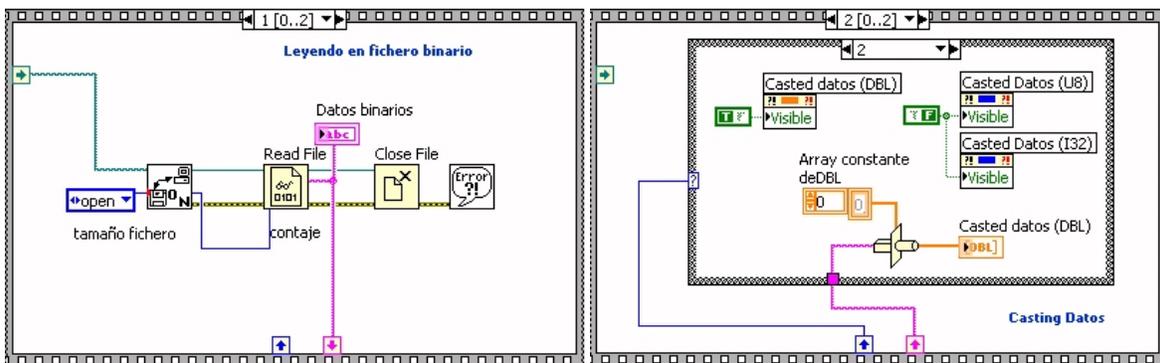
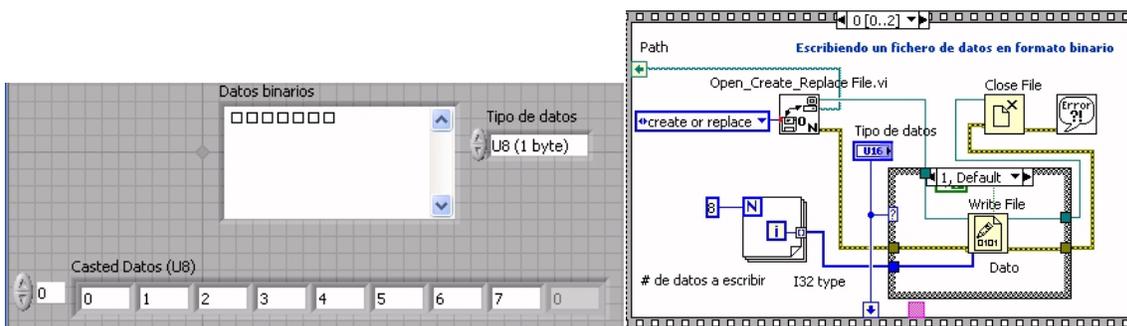


Ejercicio 12

1.- VI a emplear: E12.vi

2.- Objetivo: Aprender a convertir un string de datos en formato numéricos.

En las figuras adjuntas se muestra el panel frontal y diagrama de bloques.

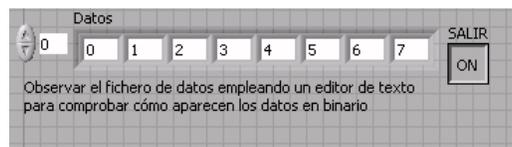


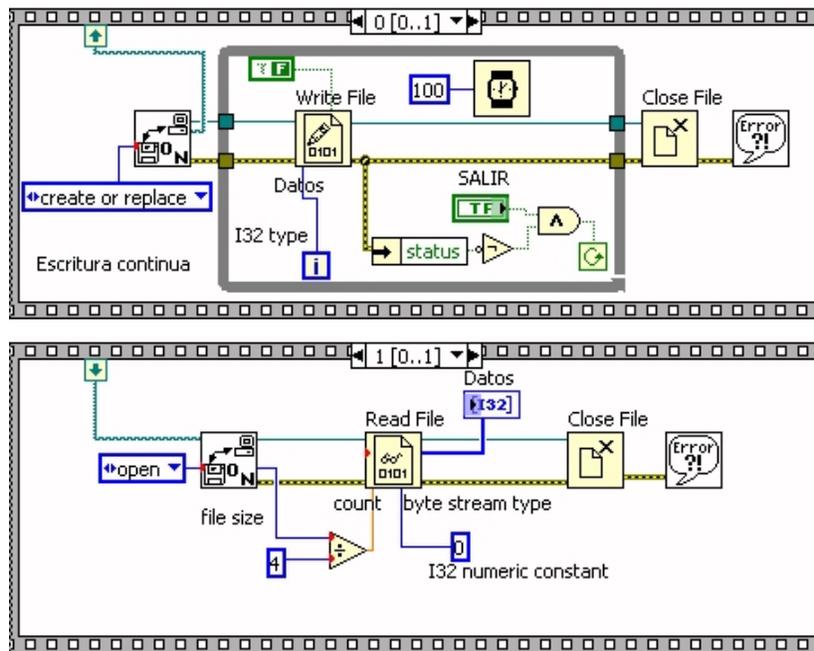
Ejercicio 13

1.- VI a emplear: E13.vi

2.- Objetivo: Aprender a escribir en un fichero binario de forma continua.

En las figuras adjuntas se muestra el panel frontal y diagrama de bloques.



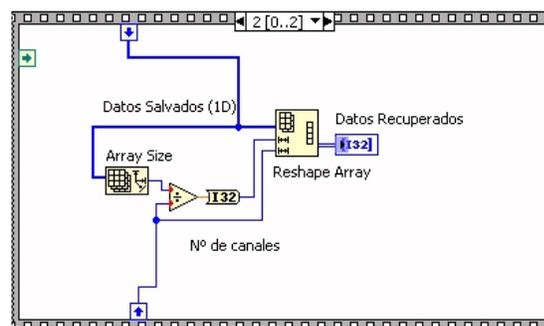
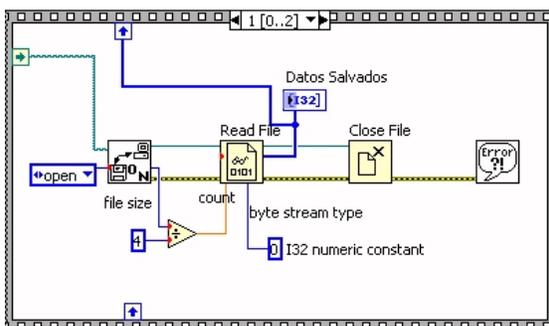
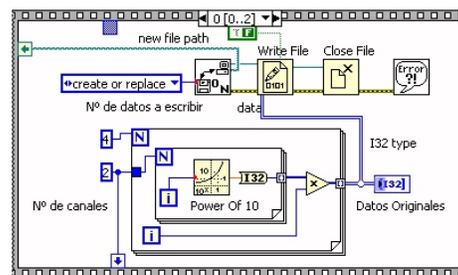
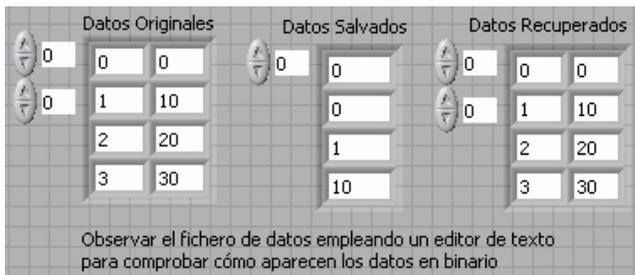


Ejercicio 14

1.- VI a emplear: E14.vi

2.- Objetivo: Aprender a escribir y leer de un fichero binario 2D

En las figuras adjuntas se muestra el panel frontal y diagrama de bloques.

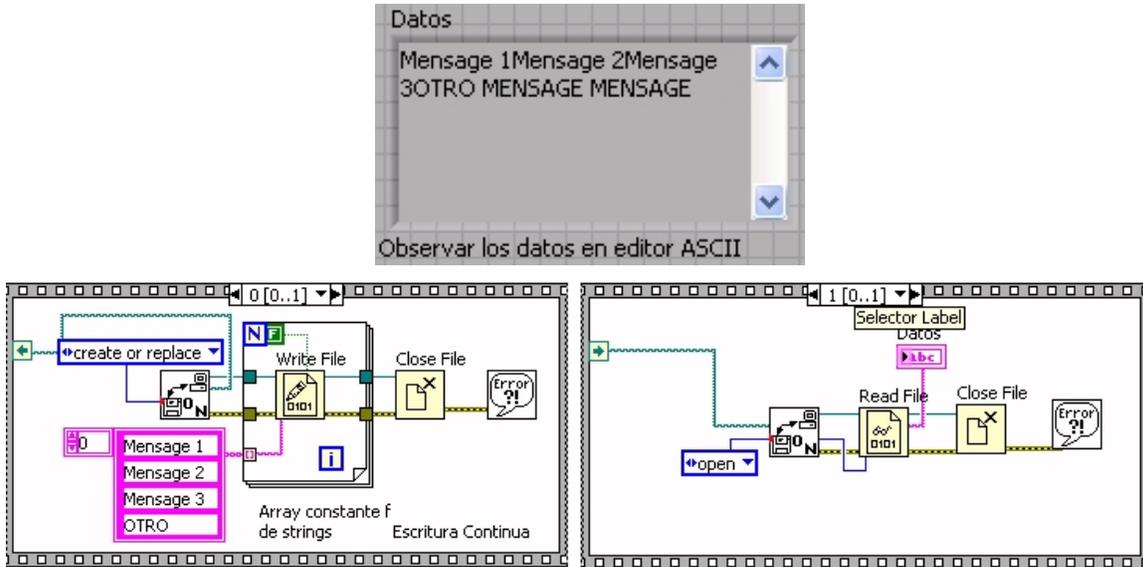


Ejercicio 15

1.- VI a emplear: E15.vi

2.- Objetivo: Aprender a escribir de forma continua datos en formato ASCII.

En las figuras adjuntas se muestra el panel frontal y diagrama de bloques.

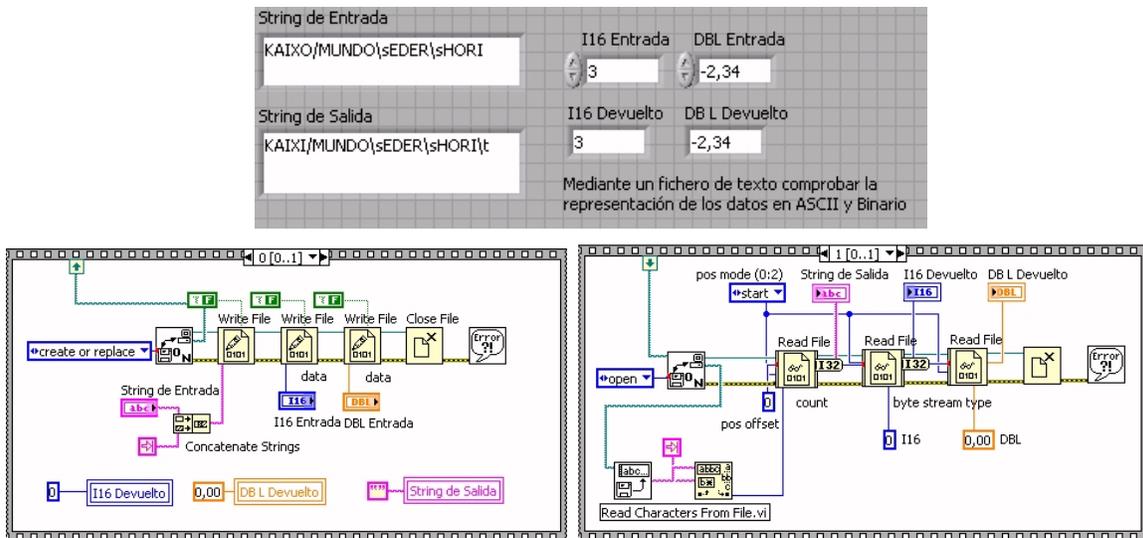


Ejercicio 16

1.- VI a emplear: E16.vi

2.- Objetivo: Aprender a almacenar en un fichero datos tanto en formato ASCII como en formato binario.

En las figuras adjuntas se muestra el panel frontal y diagrama de bloques.

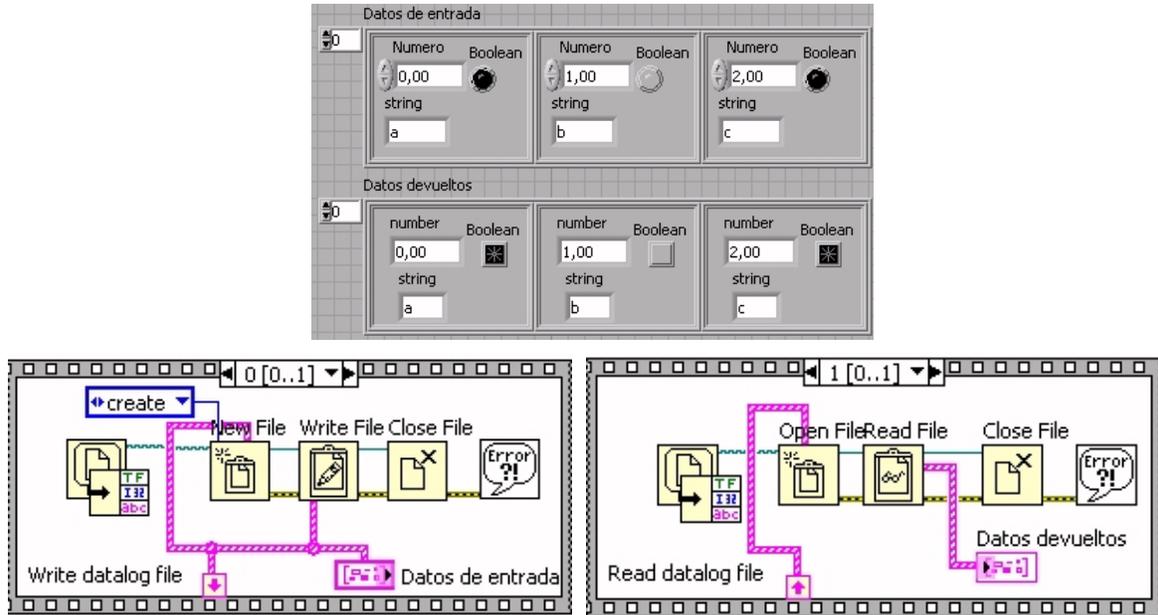


Ejercicio 17

1.- VI a emplear: E17.vi

2.- Objetivo: Aprender a almacenar y leer datos en clusters

En las figuras adjuntas se muestra el panel frontal y diagrama de bloques.

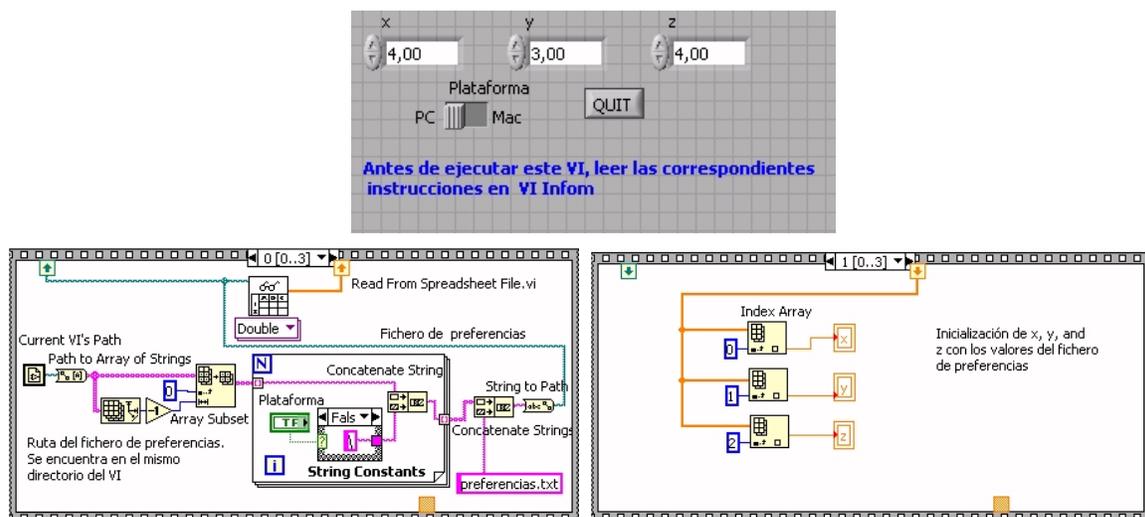


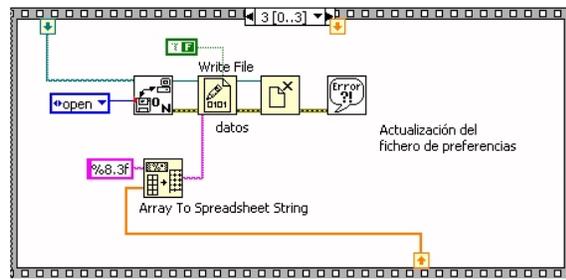
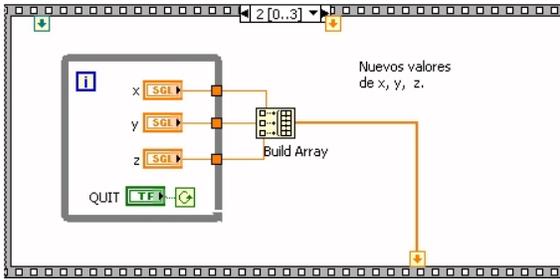
Ejercicio 18

1.- VI a emplear: E18.vi

2.- Objetivo: Aprender a salvara parámetros de un VI después de su finalización y recuperarlos posteriormente cuando se vuelva a abrir.

En las figuras adjuntas se muestra el panel frontal y diagrama de bloques.



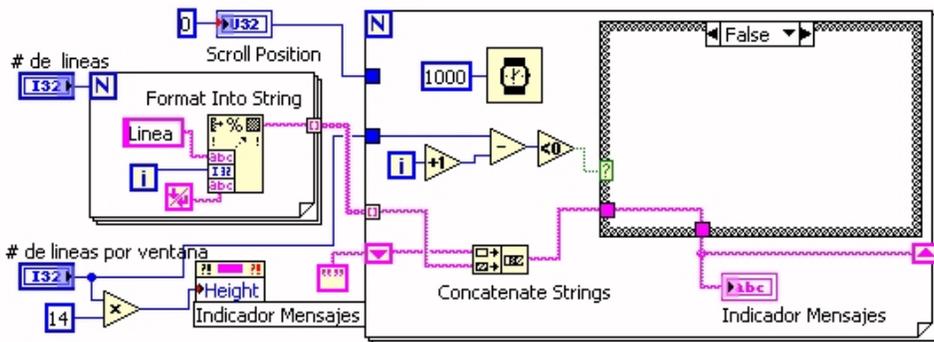
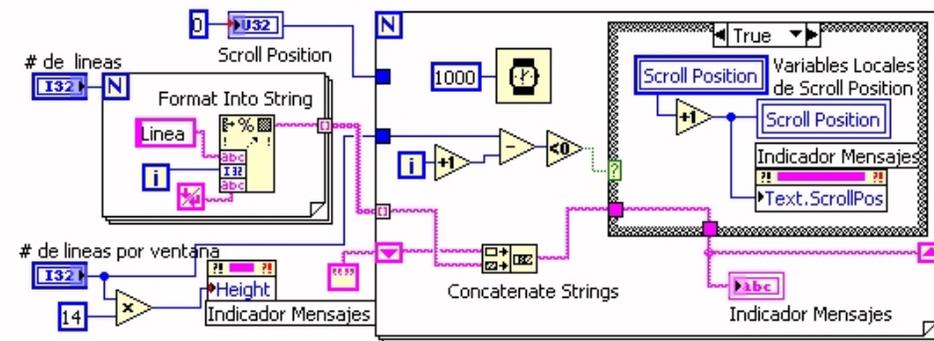
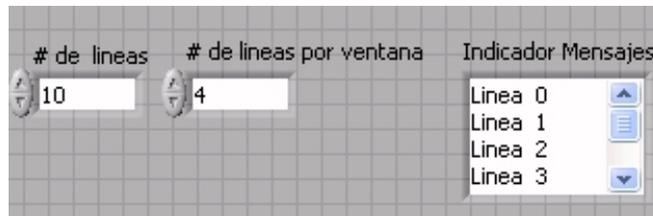


Ejercicio 19

1.- VI a emplear: E19.vi

2.- Objetivo: Scroll en un indicador de texto.

En las figuras adjuntas se muestra el panel frontal y diagrama de bloques.

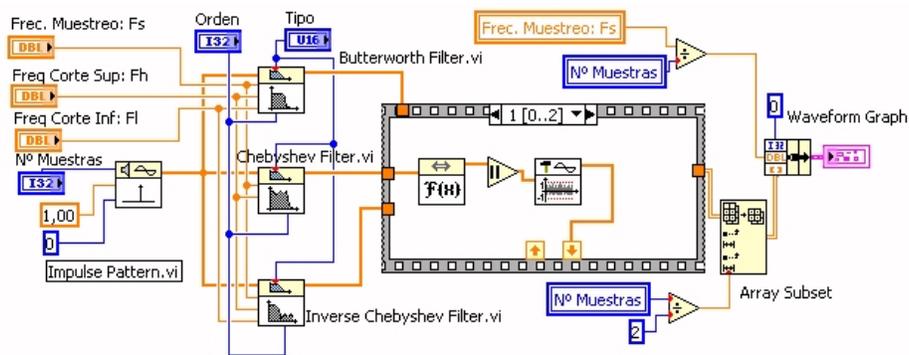
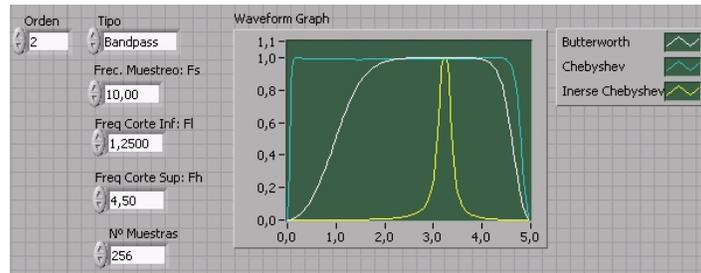


Ejercicio 20

1.- VI a emplear: E20.vi

2.- Objetivo: Seleccionar diferentes tipos de filtros digitales mediante su respuesta en frecuencia..

En las figuras adjuntas se muestra el panel frontal y diagrama de bloques.

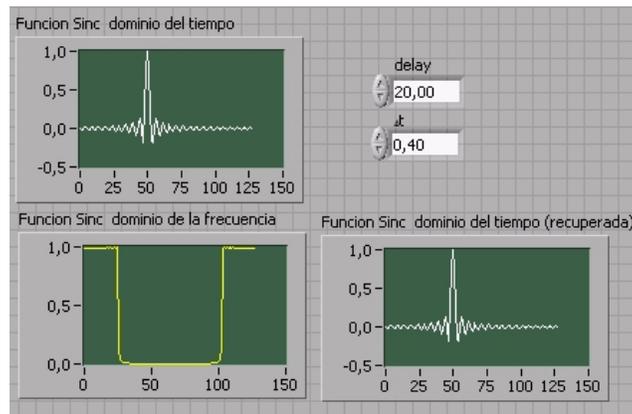


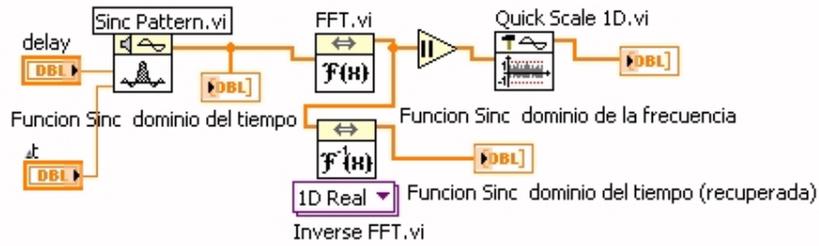
Ejercicio 21

1.- VI a emplear: E21.vi

2.- Objetivo: Aprender a emplear la FFT y su inversa

En las figuras adjuntas se muestra el panel frontal y diagrama de bloques.



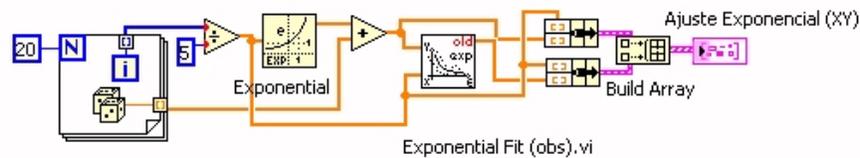
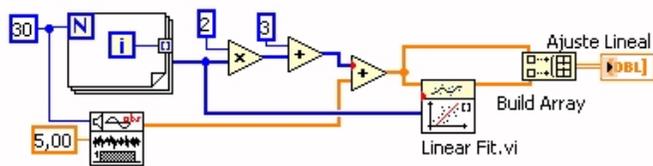
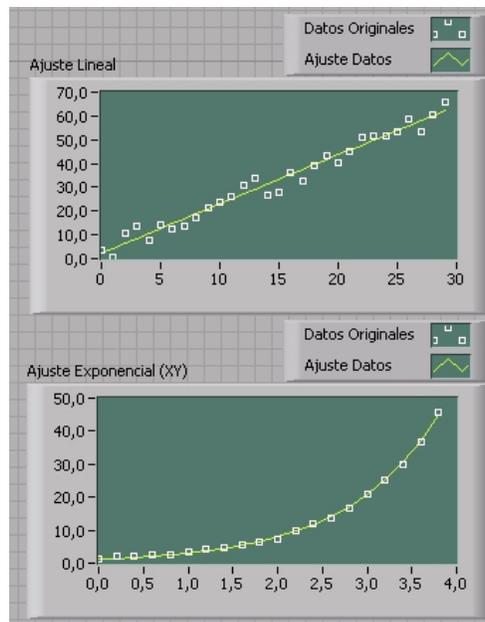


Ejercicio 22

1.- VI a emplear: E22.vi

2.- Objetivo: Aprender a realizar ajustes mediante funciones lineales y exponenciales

En las figuras adjuntas se muestra el panel frontal y diagrama de bloques.



Ejercicio 23

1.- VI a emplear: E23.vi

2.- Objetivo: Analizar la operatividad de una ocurrencia. Una ocurrencia es un mecanismo de sincronización mediante el cual estructuras que están operando en paralelo se notifican entre sí sobre un evento determinado. Operativamente funcionan de forma similar a un trigger por software. Una de las razones más importantes para el empleo de ocurrencias es evitar el mecanismo de polling, por ejemplo, mediante variables locales o globales. Para su implementación, en general se comienza llamando a la función Generate Occurrence para generar una referencia que debe ser enviada y compartida por las demás estructuras. Una vez realizada esta operación se puede esperar a que suceda una ocurrencia bien llamando a la función Wait on Occurrence, o mediante Set Occurrence para generar un evento. Esta operatividad se muestra en el diagrama de bloques de la figura adjunta

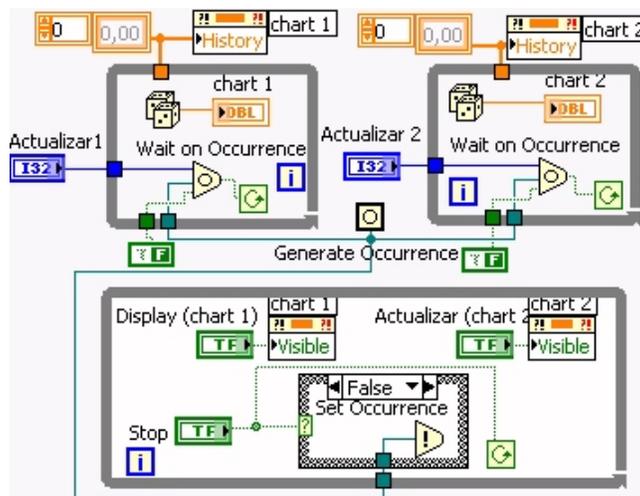


Fig. 10. Diagrama en bloques del VI sobre ocurrencias

En el ejemplo se desea controlar desde el botón Stop la finalización simultánea de ambos bucles while con distinta temporización. En el código mostrado se implementa una ocurrencia que será pasada a los dos bucles while. En el hilo inferior se activa el evento cuando en el panel frontal se presiona el botón Stop, activándose el hilo superior que generará la señal para finalizar ambos bucles while. En las figuras adjuntas se muestra el panel frontal

