

# **CAPÍTULO 6**

# **INSTRUMENTO**

# **VIRTUAL**

Como se ha ido señalando a lo largo de esta memoria, el objetivo del proyecto no es otro que el de diseñar una implementación adecuada para el estimador de frecuencia diseñado. Finalmente los requisitos previstos nos llevan a implementar un instrumento virtual con LabView.

## **6.1. Introducción a LabView**

LabView (Laboratory Virtual Instrument Engineering Workbench), de National Instruments, es un sistema de programación gráfico diseñado para el desarrollo de distintas aplicaciones como el análisis de datos, la adquisición de datos y el control de instrumentos. La versión de LabView utilizada para este proyecto es la 8.6.

El desarrollo de este programa tiene como fin la obtención de un instrumento virtual.

Un instrumento virtual es un módulo software que simula el panel frontal de instrumentos de medida que, apoyándose en elementos hardware accesibles por computadora (tarjetas de adquisición, tarjetas DSP, instrumentos accesibles vía GPIB, VXI, RS-232, USB Ethernet) realiza una serie de medidas como si se tratase de un instrumento real. De este modo, cuando se ejecuta un programa que funciona como instrumento virtual o VI (Virtual Instrument), en la pantalla de la computadora aparece un panel cuya función es idéntica a la de un instrumento físico, facilitando la visualización y el control del aparato. A partir de los datos reflejados en el panel frontal, el VI debe actuar recogiendo o generando señales, como lo haría su homónimo.

Cuando se crea un VI en LabView se trabaja con dos ventanas: una en la que se implementará el Panel de Ejecución (panel de mando y visualización) y otra que soportará el nivel de programación, el que será llamado Panel de Programación.

Para la creación del Panel de Ejecución se dispone de una librería de controles e indicadores de todo tipo y la posibilidad de crear más, diseñado por el propio usuario.

Cuando se añade un control a este panel desde la librería se crea una variable cuyos valores vendrán determinados por lo que el usuario ajuste desde el panel; inmediatamente, aparece un terminal en la ventana de programación representándolo. El nivel de programación del VI consistirá en conectar estos terminales a bloques funcionales, hasta obtener el resultado que se desee visualizar. Los bloques funcionales son iconos de entrada y salidas que se conectan entre sí mediante cables ficticios por donde fluyen los datos, constituyendo el nivel de programación del VI.

## 6.2. Implementación en LabView

La implementación del sistema en LabView se idea principalmente de la forma siguiente:

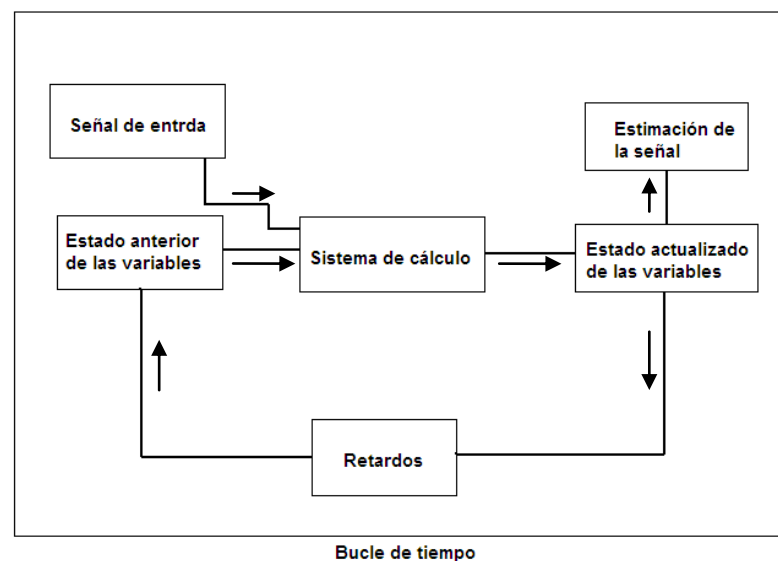


Figura6. 1 diagrama conceptual de la implementación en LabView

donde el retardo se refiere al la diferencia del tiempo entre el estado anterior y el actualizado de las variables, en la primer bucle de la ejecución éste presenta como condición inicial de las variables de entradas.

### 6.2.1. Problemas de la implementación

Al pasar el sistema probado en Matlab al LabView se encuentran varios problemas que dificultan la implementación del sistema, entre ellos los más raros son:

1. La ganancia  $\rho$  ya no tiene concurrencia con las pruebas anteriores, se ha de incrementar muchísimo más para que funcione el sistema.

2. La derivada de la señal de entrada presenta ruidos significativos que interviene la estabilidad del sistema.

Una posible causa que se encuentra para dar explicación a estos problemas es que las variables dinámicas se declaran en LabView en formato float 64. La precisión de este formato es mucho menor que la precisión que se ha utilizado en Matlab (1e-13). Además, como el sistema está basado totalmente en cálculos numéricos, la reducción de precisión puede empeorar mucho los cálculos especialmente a la derivada de la señal de entrada que puede provocar la inestabilidad de sistema.

### 6.2.2. Soluciones y mejoras

Para solucionar los problema encontrados se plantea primero un tercer filtro que es de tipo paso bajo y se pone justamente delante de la ecuación de derivada para filtrar los posibles ruidos de alta frecuencia, la señal que sale de este filtro no debe llevar ni ruidos de alta frecuencia ni componente de directa que es filtrado anteriormente con el filtro de componente de directa. Este filtro paso bajo se presenta en el sistema con la formula siguiente:

```
/*filtramos los posibles ruidos de alta frecuencia con un filtro paso bajo*/  
/*sf2 es la señal filtrada final, es una señal pura sin componente directo*/  
/*ni ruido de alta frecuencia, la utilizamos como la señal de entrada de nuestro sistema*/  
sf2=h*(-0.25*sf2a+0.25*sf)+sf2a;
```

Al poner este filtro ha sido necesario de retocar y encontrar experimentalmente las  $\alpha$  de todos los filtros para encontrar el nuevo equilibrio del sistema. Una vez que funcionan bien los filtros ya se puede buscar la ganancia  $\rho$  para el rango de frecuencia que se interesa.

Además en lugar de obtener la frecuencia de la señal de entrada al cuadrado en unidades de rad/s, se ha añadido la siguiente formula al final del programa con el objetivo de dar una estimación exacta en Hz.

```
/*hacemos la raíz de FB divide entre 2*pi, para sacar una medición exacta*/  
/*en Hz de la señal de entrada*/  
sal=sqrt(FB)/(2*3.1415926);
```

Hasta este momento el sistema sigue teniendo problema que una vez que se estabiliza el sistema se presenta una pequeña oscilación en la estimación, la cual cosa introduce un error relativo elevado de la lectura. Esto no obliga a buscar un método para eliminar dicha oscilación.

### 6.2.3. Introducción a la modificación de $\sigma$

Los sistemas dinámicos que tiene una expresión matemática de la siguiente manera (ver [8])

$$\theta = -ge_1y$$

donde  $g$  es una constante,  $e_1$  es la diferencia entre dos variables dinámicas y la  $y$  es una variable dinámica, presenta una baja robustez en la estabilización.

Loannou & Kokotovic [1983] sugirió modificando la ley de parámetro actualizado para contrarrestar la tendencia de los valores de parámetro en las regiones de inestabilidad en la ausencia de excitación persistentes. La formula original es la siguiente:

$$\dot{\theta} = -ge_1y - \sigma\theta \quad (6.1)$$

donde  $\sigma$  es elegido pequeño positivo para salvar  $\theta$  del crecimiento sin limites.

Hay dos formulas interesantes que están basas en la modificación 6.1 son:

$$\dot{\theta} = -ge_1y - \sigma(\theta - \theta_0) \quad (6.2)$$

donde  $\theta_0$  es una estimación previa de  $\theta$ .

$$\dot{\theta} = -ge_1y - \sigma|e_1|\theta \quad (6.3)$$

Tanto (6.2) como (6.3) intenta capturar el espíritu de (6.1) con su devolución de cursar  $\theta \rightarrow 0$  si  $e_1$  es pequeño. La ecuación (6.2) trata de sesgar la dirección de la derivada hacia la  $\theta$  en lugar de 0 y la (6.3) trata de apagar la derivada hacia 0 cuando  $e_1$  es pequeño. La principal ventaja de (6.3) es que conserva las características del algoritmo sin pérdidas (tales como la convergencia de los parámetros a sus valores reales cuando la excitación es persistente). Además, el algoritmo de (6.3) puede ser menos susceptible a la rotura de (6.1), aunque esta afirmación no ha sido totalmente justificada.

#### 6.2.4. Implementación final

Se aplica (6.3) de la teoría de modificación de  $\sigma$  a nuestro sistema con las formulas siguientes,

```
f1=rho*ya*(ya-x2a);  
teta=h*(f1-abs(ya-x2a)*tetaa*1)+tetaa;
```

para mejorar estabilidad de la estimación, el valor de  $\sigma$  se ha de encontrar como resultado de muchos experimentos.

Se debe tener en cuenta de que al introducir esta teoría, el sistema pierde la robustez de estimar señales de amplitud mayor, es decir, el sistema se vuelve inestable cuando la señal de entrada cuando la amplitud de la señal de entrada es mayor que un nivel relacionado con la  $\sigma$ . Este problema se puede compensar añadiendo un circuito de control de ganancia automática en la entrada de la tarjeta de adquisición. Sin embargo, nuestro sistema presentó robustez cuando la amplitud de la señal se movía de 0.5 a 3 (un rango bastante amplio).

La implementación final del sistema es mostrada en la figura 6.2.

```

float64 x2,teta,y,rho,h,sal,sp,DF,sf,sf2,FB,f1;

rho=300000000; /*ganancia rho*/
h=0.001;
/*filtramos el posible componente directo*/
/* sf es la señal filtrada*/
DF=h*(1*(-DFa+s))+DFa;
sf=s-DFa;

/*filtramos los posibles ruidos de alta frecuencia con un filtro paso bajo*/
/*sf2 es la señal filtrada final,es una señal pura sin componente directo*/
/* ni ruido de alta frecuencia,la utilizamos como la señal de entrada de nuestro sistema*/
sf2=h*(-0.25*sf2a+0.25*sf)+sf2a;

/*sistema de calculo*/
/*utilizamos la primera metodo de Euler para sacar la derivada de sf2, que es sp*/
sp=(sf2-sf2a)/h;
/*las tres formulas principales que utilizamos para los calculos*/
y=h*(sp-(tetaa))*(ya-x2a))+ya;
x2=h*(sf2-ya)+x2a;
f1=rho*ya*(ya-x2a);
teta=h*(f1-abs(ya-x2a)*tetaa*1)+tetaa;
FB=h*(-1*FBa+1*tetaa)+FBa;
/*hacemos la raiz de FB divide entre 2*pi, para sacar una medición exacta*/
/*en Hz de la señal de entrada*/
sal=sqrt(FB)/(2*3.1415926);

```

Figura6. 2 Programación completa del instrumento virtual.

### 6.3. Aspecto general del instrumento virtual

Después de realizar numerosas pruebas finalmente la implementación se queda como se muestra en la figura 6.2-6.3

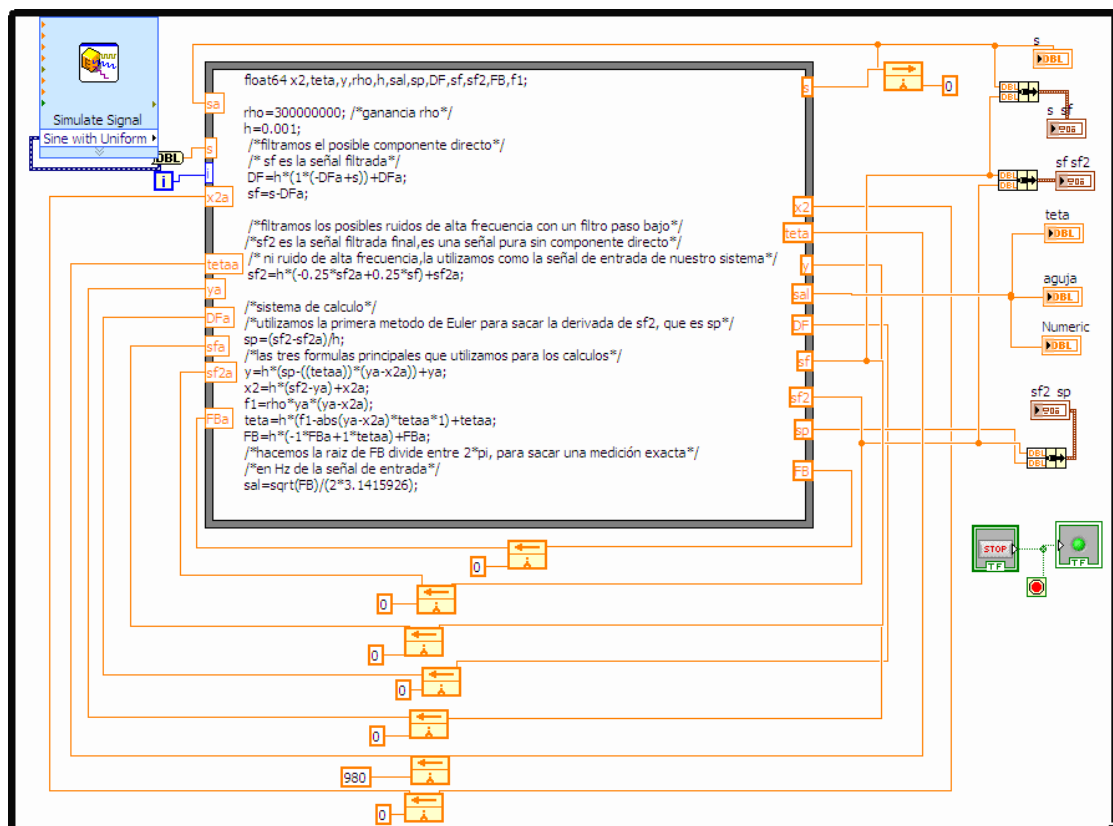


Figura6. 3 Panel de programación de instrumento virtual.

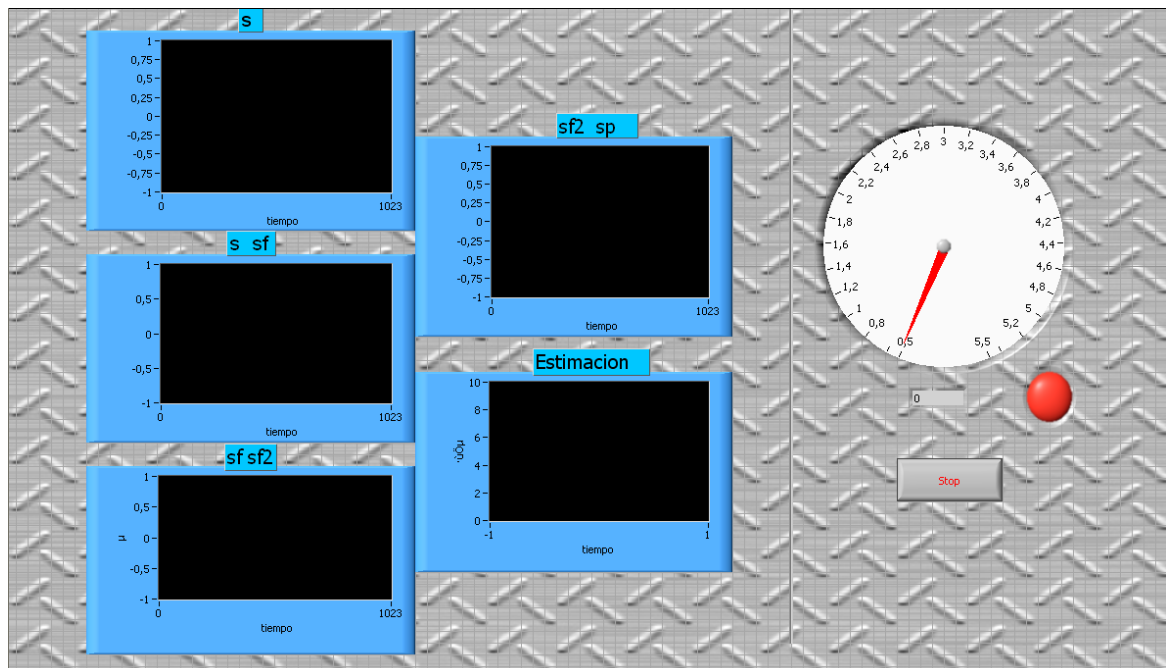


Figura6. 4 Panel de control del instrumento virtual.

El panel de control está diseñado considerando las necesidades de los diseñadores del sistema de control sísmico a que se va a incorporar este instrumento virtual. Se divide principalmente en dos partes:

1. Analizador del sistema

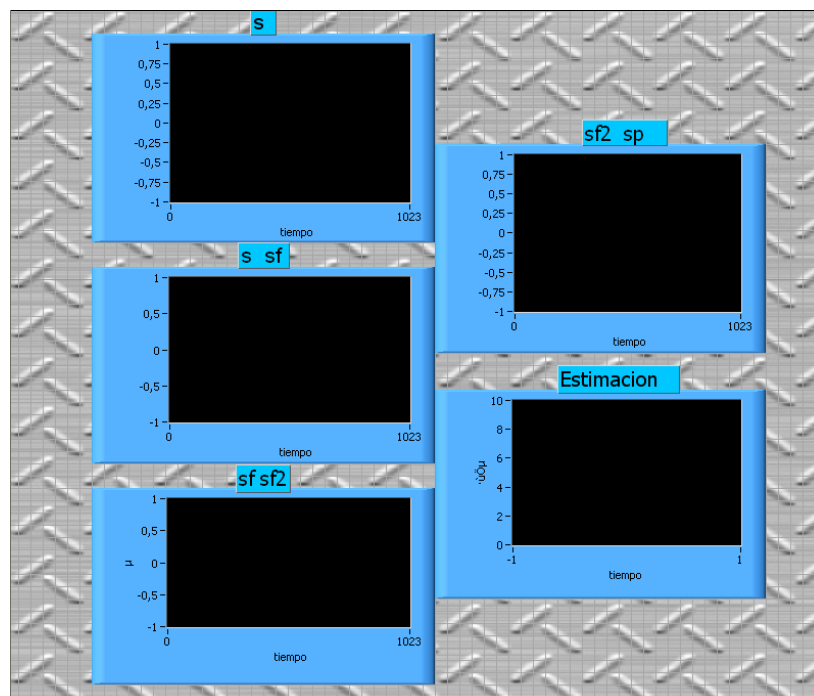
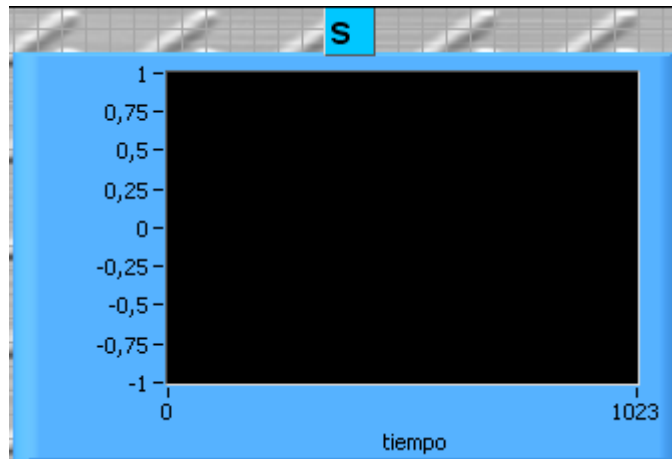


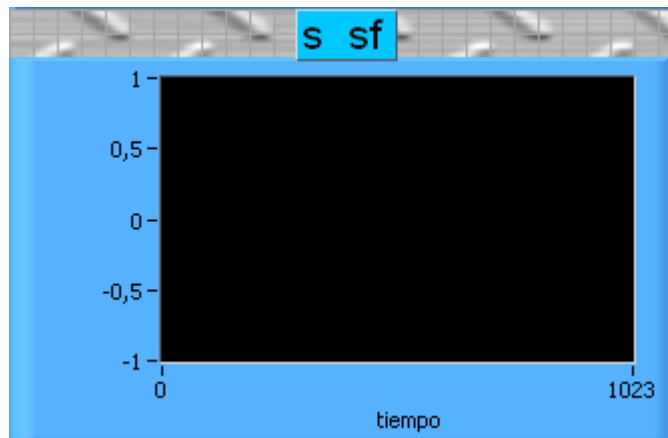
Figura6. 5 Analizador de sistema.

Esta parte como se muestra en la figura 6.4 está formada por cinco ventanas de visualización en que se muestran las variables dinámicas más interesantes del sistema.



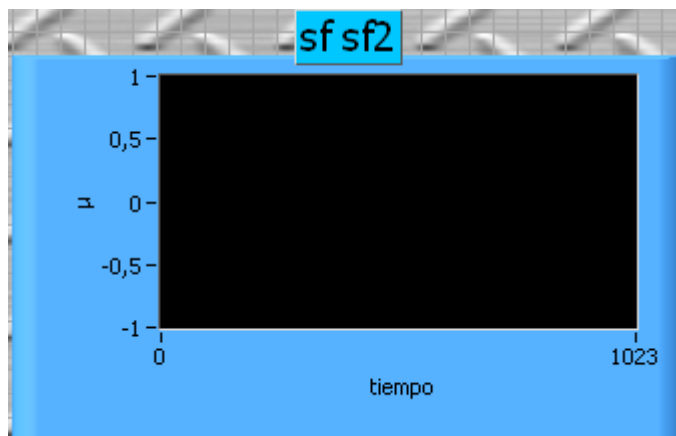
*Figura6. 6 Visualizador de la señal de entrada.*

En la ventana de la figura 6.5 se muestra la señal de entrada original, permite saber el aspecto general de fuente de señal.



*Figura6. 7 Visualizador de variable s y sf.*

En esta ventana de la figura 6.6 se muestran simultáneamente la señal de entrada y la señal filtrada de posibles componentes de directa, se permite saber visualmente si el filtro de componente de directa está funcionando correctamente.



*Figura6. 8 Visualizador de variable sf y sf2.*

En esta ventana de la figura 6.7 se visualizan la señal filtrada de componente de directa y la señal filtrada de todos los posibles ruidos que puede llevar la señal de entrada, se permite saber visualmente si el filtro de paso bajo de la entrada está bajo su correcto funcionamiento y los aspectos de la señal de entrada del sistema de estimación.

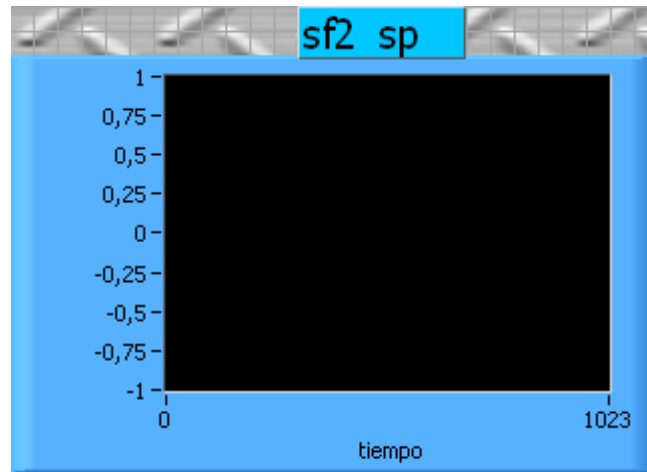


Figura6. 9 Visualizador de variable sf2 y sp.

Se muestran en esta ventana de figura 6.8 la señal filtrada de los posibles ruidos, que es la señal de entrada del sistema de estimación, y su derivada. Se permite saber visualmente si la derivada está bien hecha y si hay algunos saltos numéricos indeseables que podrían intervenir resultado de estimación y la estabilidad del sistema.

#### 1. Visualizador de estimación

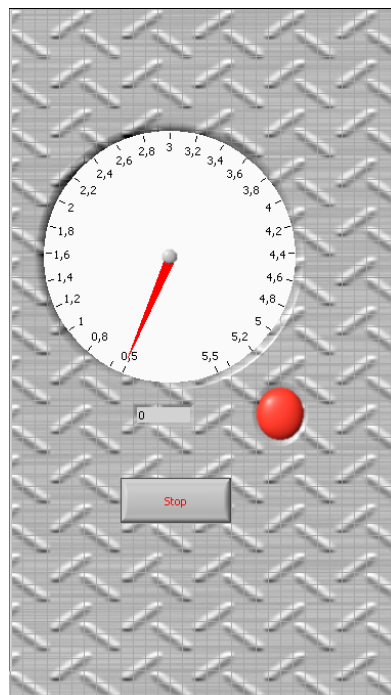


Figura6. 10 Visualizador de estimación.



Esta parte está formada por un botón de paro, un indicador de aguja que permite interpretar aproximadamente la estimación de forma intuitiva y rápida, un display donde indica el valor exacto de la estimación, y un indicador de luz que indica el estado de funcionamiento del instrumento (ON/OFF)

## 6.4. Sincronización del instrumento

La sincronización del instrumento virtual se refiere a ajustar los parámetros  $\alpha, \rho, \sigma$  del sistema para que funcione con rapidez y exactitud aceptables para un rango de frecuencias.

En el caso de interesar un rango de frecuencia de 0.1Hz-10Hz, es imprescindible dividir este en ciertos rangos de frecuencia más pequeños para conseguir un buen funcionamiento del sistema. Para cada rango de frecuencia dividido se debe disponer un instrumento virtual cuyos parámetros  $\alpha, \rho, \sigma$  haya sido ajustado de forma experimental para conseguir la rapidez y exactitud de estimación. Esto también lo hacen otros sistemas de medición. Por ejemplo, el selector de rango de lectura al medir resistencias en un multímetro, entre otros más.

La pregunta es "¿En cuántos rango se debe dividir el rango de frecuencia de mayor interés?" Esto ya es el criterio que debe tomar el usuario, mientras en más rangos de frecuencias se divide, mayor exactitud y rapidez se consigue.

En este apartado se presenta el procedimiento de la sincronización para un instrumento virtual trabaje en un rango de frecuencia de 1Hz-4Hz.

1. Definir los dos extremos de sincronización, si se quiere el rango de frecuencia final queda en 1Hz-4Hz, la sincronización se debe empezar por 0.5Hz-5Hz, siempre se debe poner un rango más amplio que lo deseado.
2. Utilizar el bloque de generador de señal para simular la señal de mayor frecuencia del rango de frecuencia definido (5Hz) con ruido blanco gaussiano, se toma esta señal como señal de entrada.
3. Utilizar el analizador de sistema del panel de control para saber visualmente si funciona bien los filtros.
4. Ajustar las  $\alpha$  hasta que se consiga una señal de entrada del sistema (sf2) buena (sin componente de directa ni ruidos de alta frecuencia).
5. Una vez que se encuentra e las  $\alpha$ , fijar  $\sigma$  a 1 y poner un  $\rho$  muy elevado para que el sistema se vuelva a inestabilizarse.
6. Reducir el  $\rho$  para que se estabilice la estimación (si se reduce más de tres décadas el  $\rho$ , y el sistema no estabiliza se debe aumentar una década el  $\sigma$ .)
7. Repetir el paso 6 observando el comportamiento de la estimación, si no se converge en 5Hz se debe aumentar el  $\rho$ , si se converge en 5Hz pero con muchas oscilaciones, se debe aumentar el  $\sigma$ , ambos valores pueden provocar la inestabilidad del sistema, el objetivo final es que la estimación se converge en 5Hz sin oscilaciones considerables.
8. Una vez que se consiga paso la estimación correcta de 5Hz, aplicar los  $\rho, \sigma$  conseguidos para estimar la frecuencia menor del rango de frecuencia definido (0.5Hz)
9. Repetir los paso 6 y 7 hasta que la estimación se converge en 0.5Hz

10. Aplicar los  $\rho$ ,  $\sigma$  conseguidos para estimar la frecuencia 4Hz, ir bajando o subiendo los  $\rho$ ,  $\sigma$  según los criterios del paso 6 y 7 dependiendo del comportamiento de la estimación.

11. Repetir el paso 10 para estimar la frecuencia 1Hz.

12. Repetir el paso 10 y 11 paso el resto de frecuencias interesadas del rango de frecuencia deseada (1.5Hz, 2Hz, 2.5Hz, 3Hz, 3.5Hz )

El instrumento sincronizado es resultado de muchos experimentos siguiendo estos pasos.

## 6.5. Demostración del funcionamiento

En este apartado se demuestra el funcionamiento robusto del instrumento virtual con diferentes señales de entrada.

### 6.5.1. Señal sinusoidal no perturbada

El funcionamiento básico del sistema es probado con una señal sinusoidal de 2Hz/s como se muestra en las figuras 6.11-6.12, se ve que la estimación de la señal está realizada correctamente.

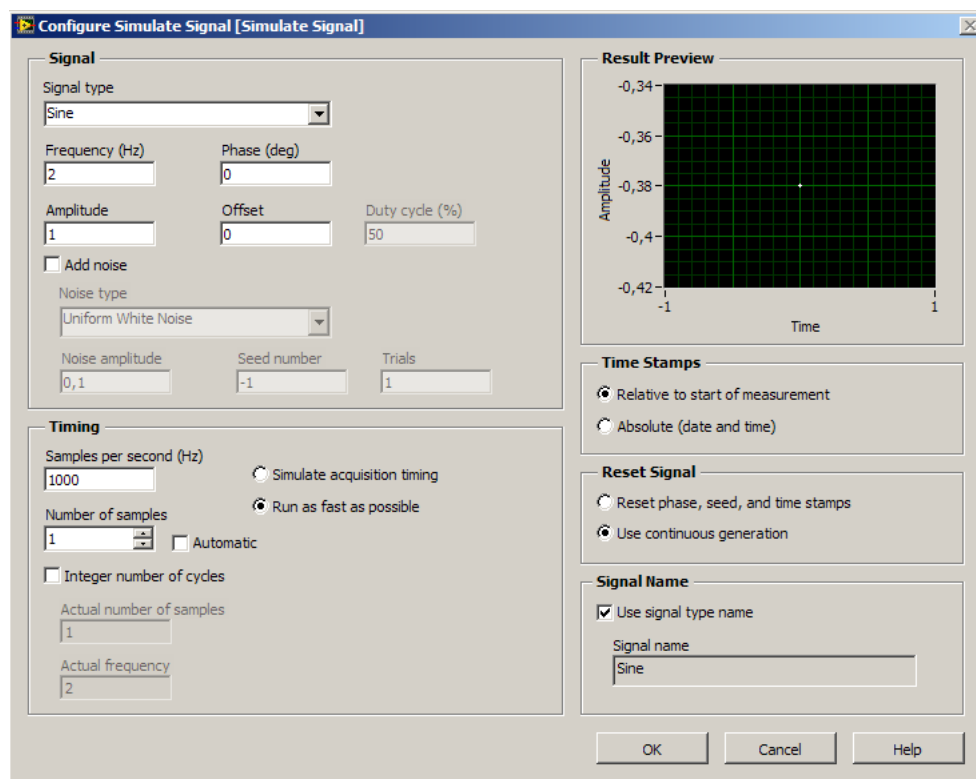
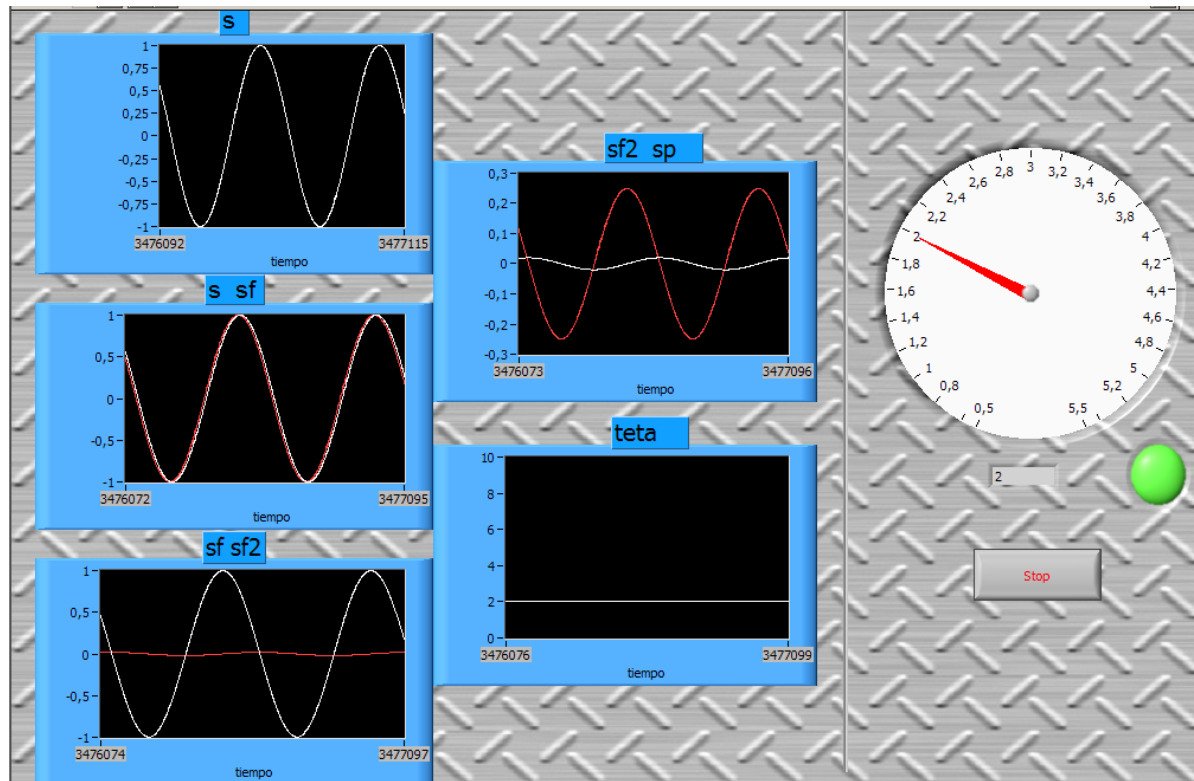


Figura6. 11 Configuración de la señal de entrada para caso sinusoidal no perturbado.



*Figura6. 12 Resultado de estimación para caso sinusoidal no perturbado.*

### *6.5.2. Señal sinusoidal perturbada*

Para demostrar la robustez del sistema, se pone a la entrada una señal sinusoidal de 2.5Hz/s con ruido blanco gaussiano como se muestra en las figuras 6.13-6.14, se observa no se ha dado ningún problema para estimar la frecuencia correctamente.

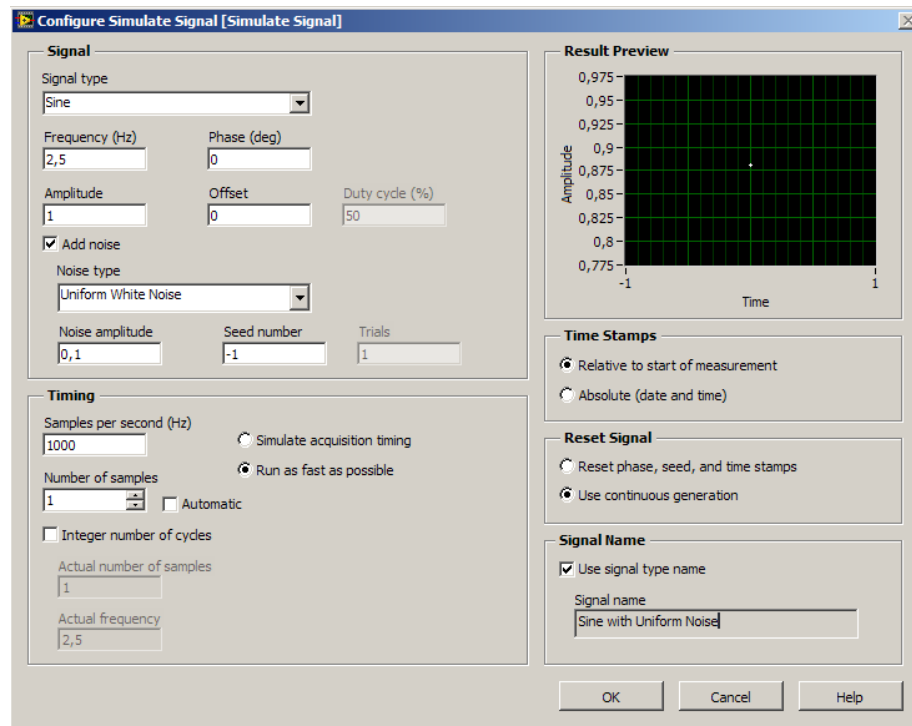


Figura6. 13 Configuración de la señal de entrada para caso sinusoidal perturbado

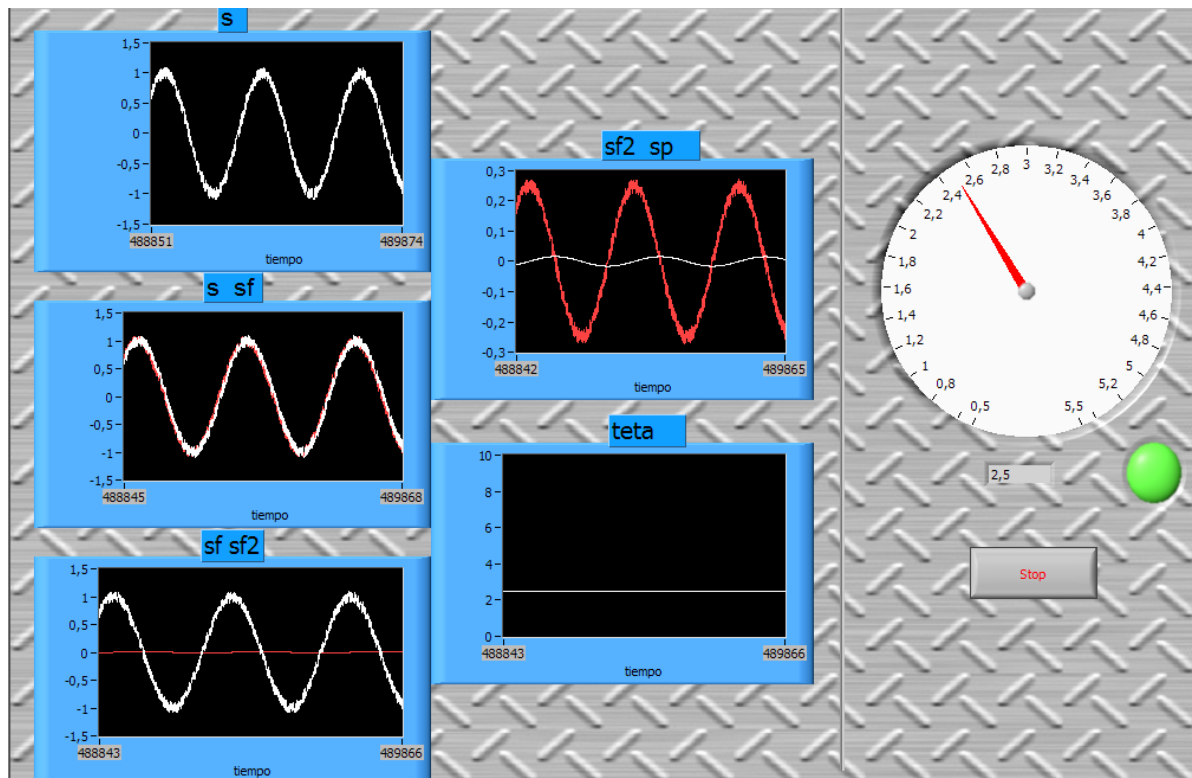
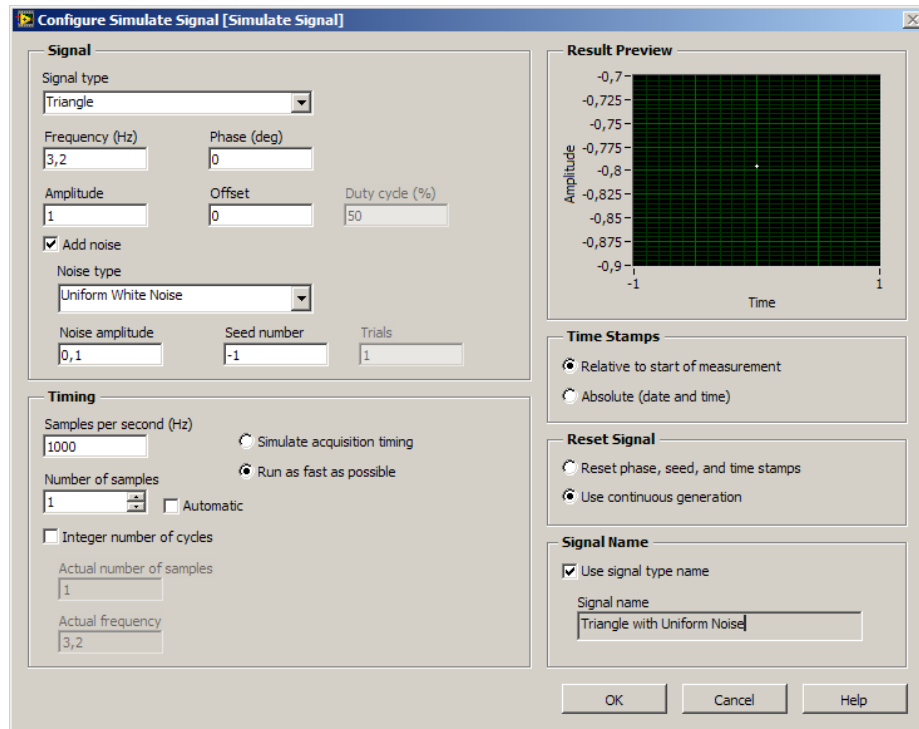


Figura6. 14 Resultado de estimación para caso sinusoidal perturbado

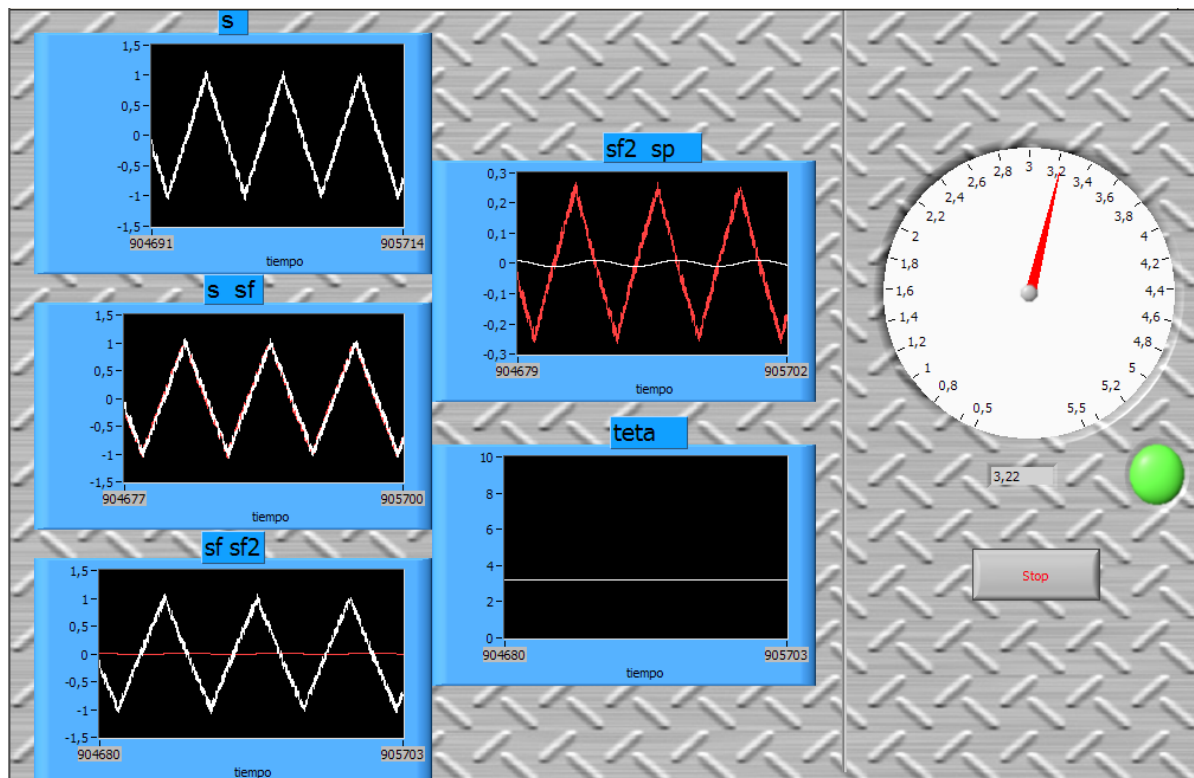
### 6.5.3. Señal triangular perturbada

Este instrumento virtual no solo puede estimar señales sinusoidales, también se sirve para estimar otros tipos de señales. En este apartado la señal de entrada es triangular de 3.2Hz/s con ruido blanco gaussiano como se muestra en las

figuras 6.15-6.16, la estimación ha sido realizada correctamente con un error relativo aceptable (0.6%)



*Figura6. 15 Configuración de la señal de entrada para caso triangular perturbado.*



*Figura6. 16 Resultado de estimación para caso triangular perturbado.*

#### 6.5.4. Señal cuadrada perturbada

En este apartado la señal de entrada es cuadrada de 1.8Hz/s con ruido blanco gaussiano como se muestra en las figuras 6.17-6.18, la estimación ha sido realizada con un error relativo de 8.3 %, si no se exige mucho la exactitud de la estimación, se puede interpretar que es un resultado deseado.

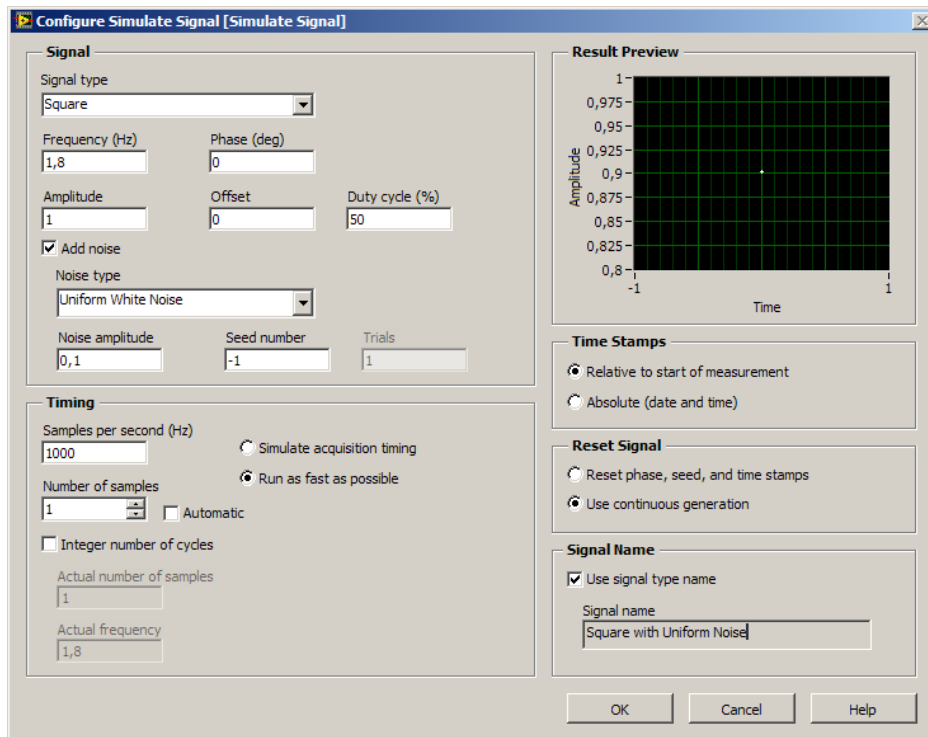


Figura6. 17 Configuración de la señal de entrada para caso triangular perturbado.

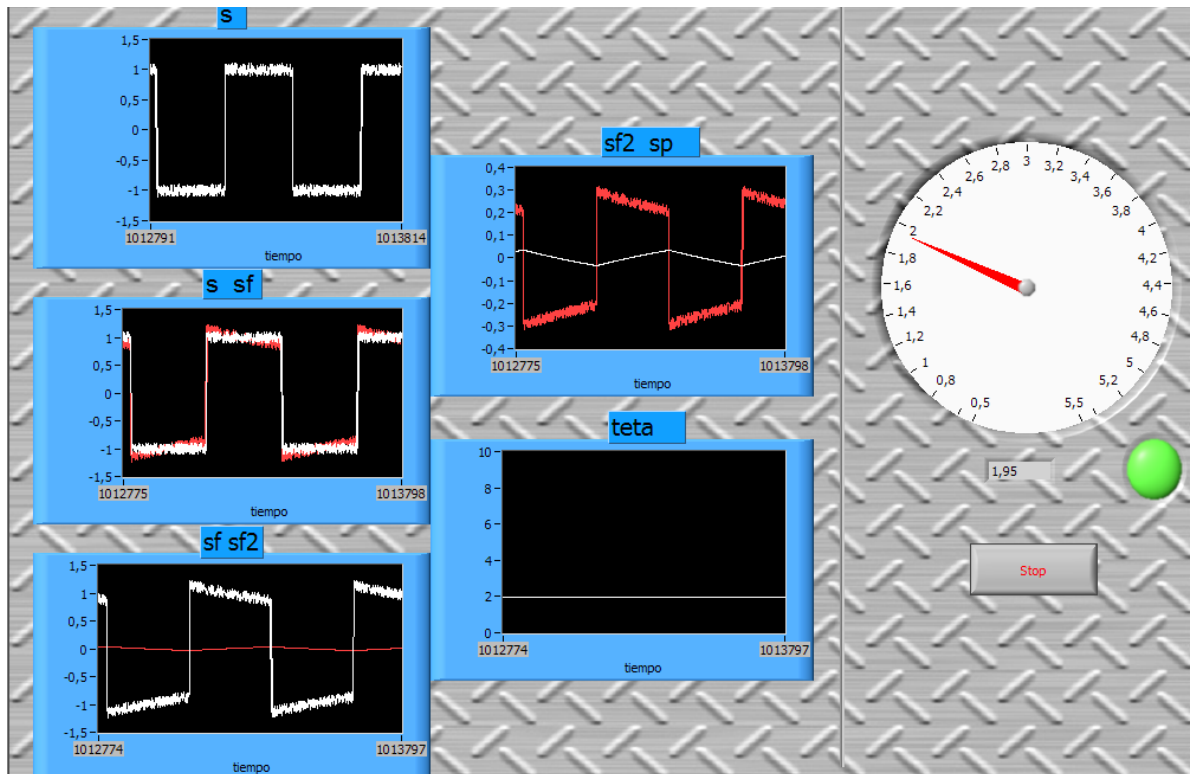


Figura6. 18 Resultado de estimación para caso cuadrado perturbado.

### 6.5.5. Señal de diente de sierra perturbada

El punto débil de este instrumento es medir señales de forma de diente de sierra, como se muestra en las figuras 6.19-6.20 cuando la señal de entrada es de tipo diente de sierra de 3.4Hz/s con ruido blanco gaussiano, la estimación se converge a un resultado con error relativo notable.

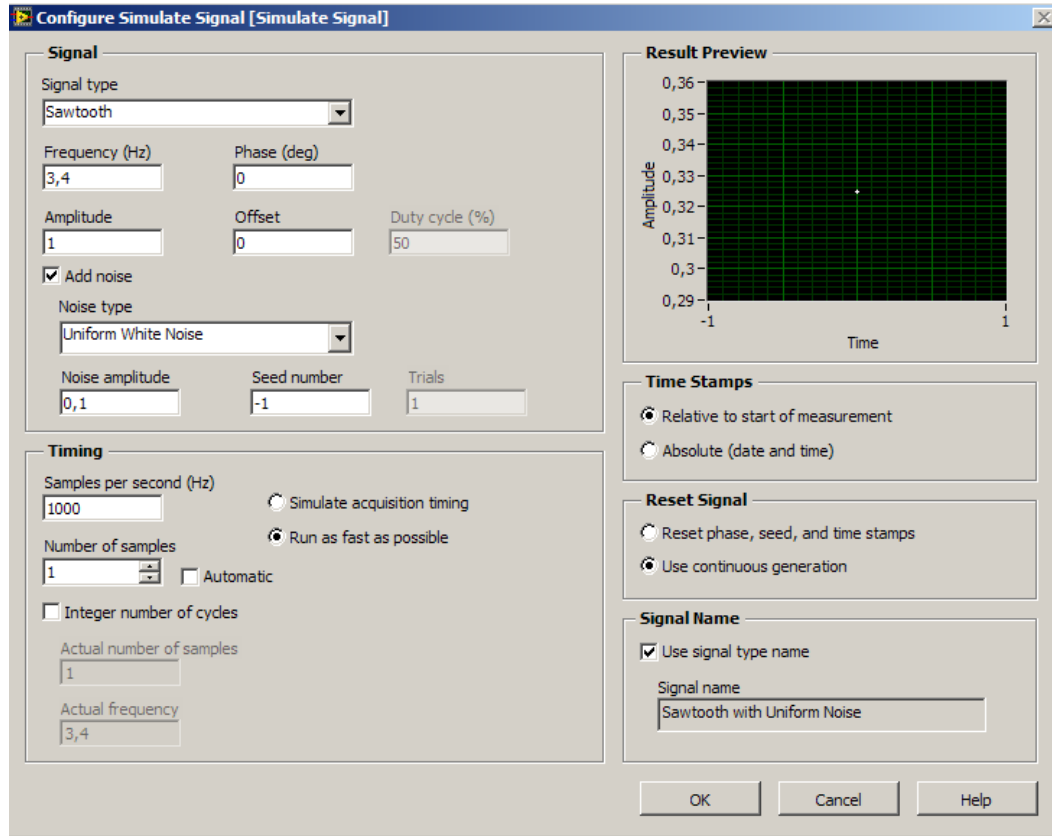


Figura6. 19 Configuración de la señal de entrada para caso diente de sierra perturbado.

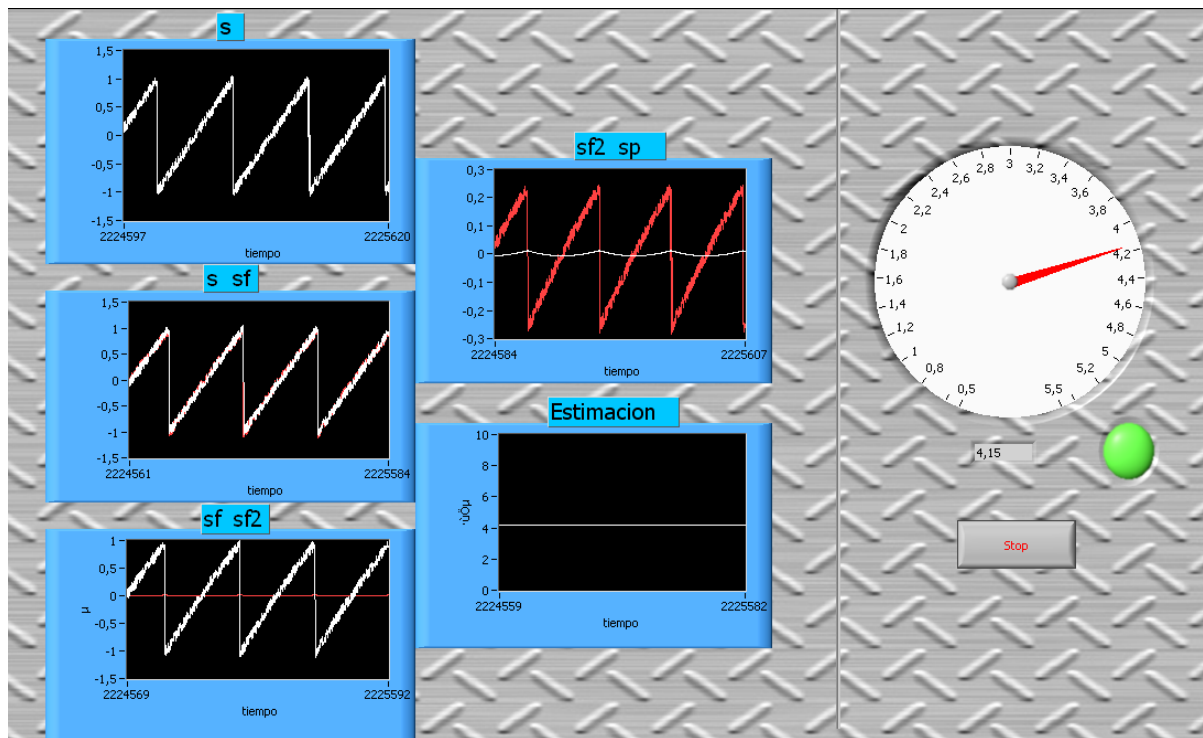


Figura6. 20 Resultado de estimación para caso diente de sierra perturbado.

## 6.6. Adquisición de la señal

Teóricamente el instrumento virtual puede funcionar con las tarjetas de adquisición de datos con pequeñas modificaciones como se muestra en la figura 6.21, se intentó utilizar el modelo NI9201 de National Instrument, pero hubo problemas con la configuración de la tarjeta y no se llegó a conseguir el su correcto funcionamiento.

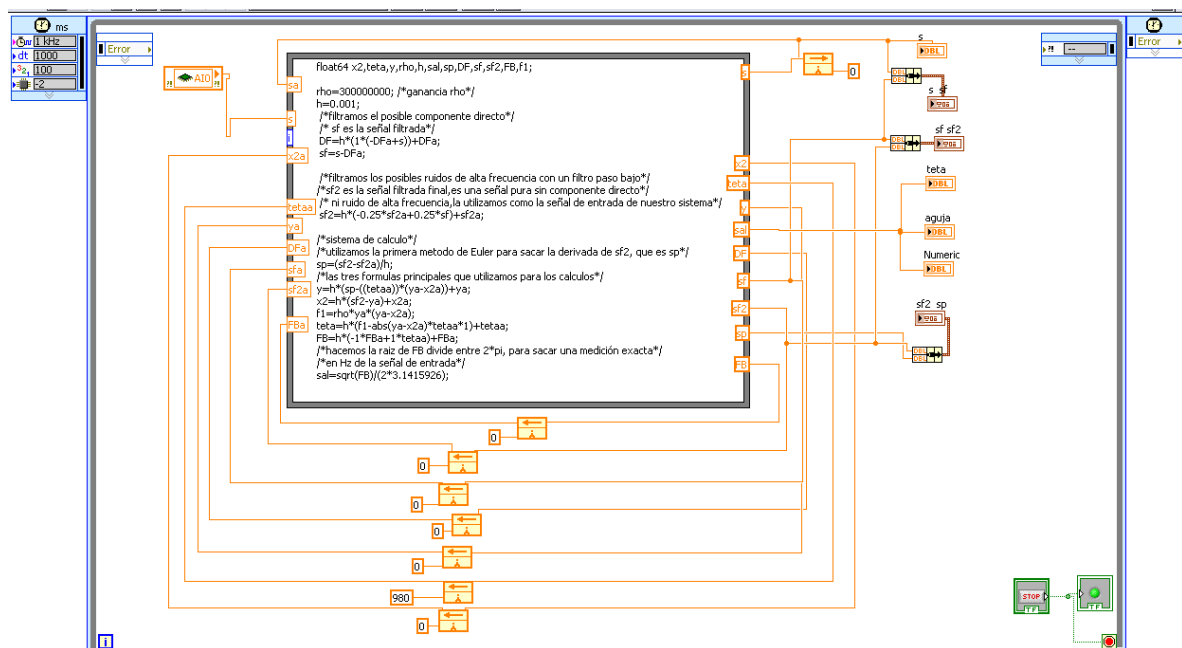


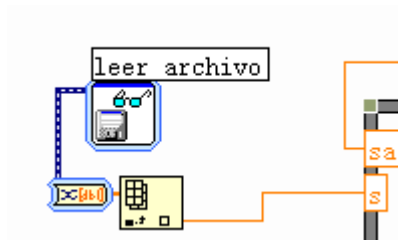
Figura6. 21 Configuración del sistema para trabajar con tarjeta de adquisición NI9201.



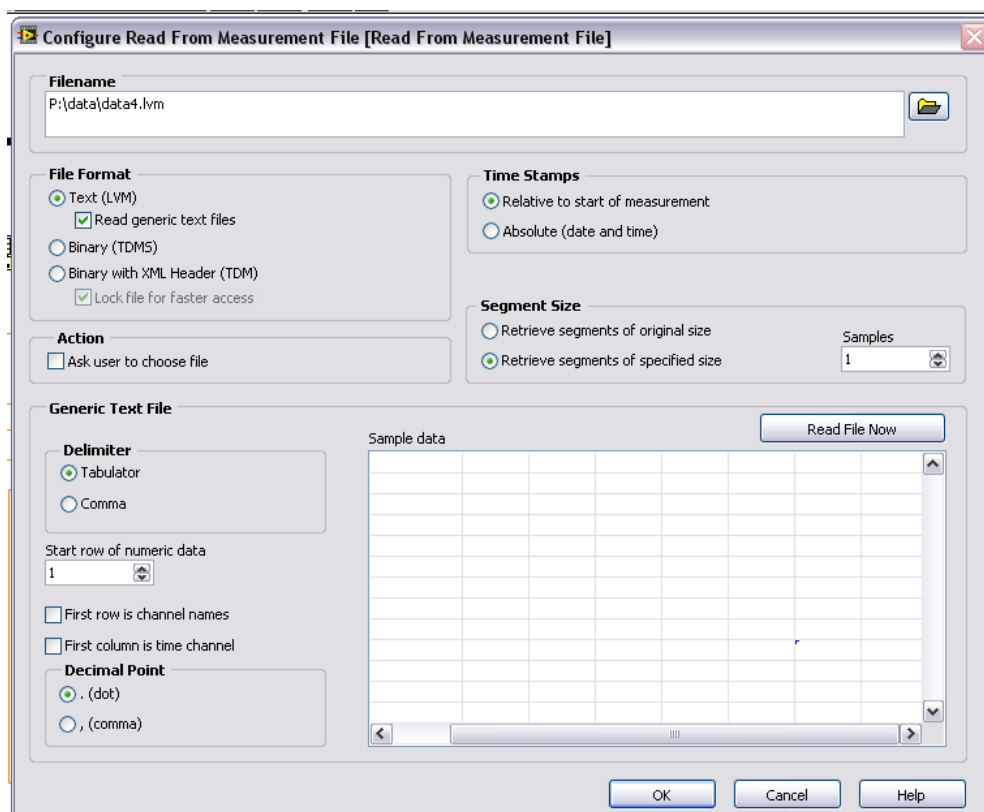
Para demostrar la validez de su funcionamiento de trabajar con las señales reales, se ha pedido ayuda a los expertos, los cuales capturaron una señal de 15Hz (con un tiempo de muestreo de 10KHz), generada por un generador de funciones montado con el chip ICL8038 [ver Anexo] y guardaron en un archivo de formato .lvm.

Con este archivo se ha hecho el experimento siguiente:

Se coge el bloque correspondiente de LabView para leer la señal capturada del archivo .lvm y se se toma este como la señal de entrada como se muestra en las figuras 6.22-6.23.



*Figura6. 22 Bloque de labview para leer archivo de formato .lvm*



*Figura6. 23 Configuración del bloque.*

Como se observa en la figura 6.24, la estimación se converge a 1.5, tal valor se se tiene que multiplicar por 10 debido a que el tiempo de muestreo que utiliza el sistema es de 1KHz que es 10 veces más pequeño que el tiempo de muestreo de la captura de la señal, eso implica una cambio de escala del tiempo real.

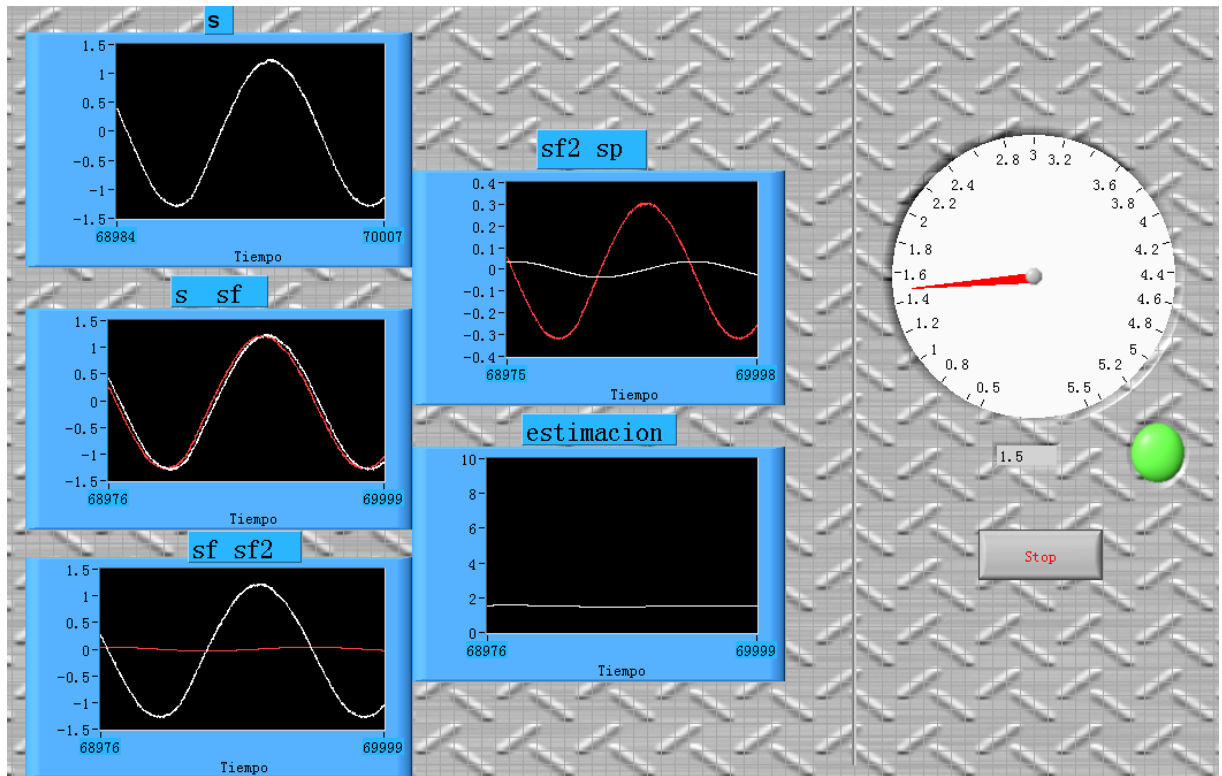


Figura6. 24 Resultado de estimación para caso de señal real.