

Implementación de filtros FIR

Propósito

El propósito de este experimento es implementar filtros FIR en tiempo real.

Antecedentes

En este experimento, se implementará un filtro FIR en tiempo real. Hay varias cosas que hacen de este experimento un desafío. Primero, el programa para el filtrado debe ser implementado en C. El código C para implementar un filtro FIR es simple, pero puede causar problemas debido a que este programa ha de trabajar en tiempo real. En clases anteriores, las convoluciones se habían hecho a mano o en MATLAB. Esta es la primera vez que la convolución se hará en tiempo real.

En el filtro en tiempo real, la convolución es implementada utilizando la ecuación de diferencia directamente. Utilizando la señal $x[n]$ como la entrada y a $y[n]$ como la salida, la ecuación de diferencia para un filtro FIR es,

$$y[n] = h[0]x[n] + h[1]x[n-1] + \dots + h[N-1]x[n-N+1].$$

Este experimento es también un desafío porque éste puede ser el primer diseño implementado en hardware. El diseño debe cumplir las especificaciones en MATLAB pero puede fallar cuando se implemente en el DSK. El AIC utilizado para la entrada y la salida no es perfecto, y esto debe ser tenido en cuenta cuando se implemente el filtro. Por ejemplo, en uno de los diseños la salida debe estar entre 0 y -1 dB desde 1000 a 3000Hz. Si el AIC amplifica la señal, entonces ésta debe ser atenuada por el filtro digital con el fin de que permanezca por debajo de 0 dB.

Mostrar que el diseño cumple las especificaciones es el desafío final. Esto implica utilizar el generador de funciones junto con el osciloscopio y el multímetro para mostrar que el filtro es de fase lineal y que cumple las especificaciones de magnitud. El diseño y la implementación se hacen generalmente en pasos. Si el problema es muy simple puede ser posible diseñarlo e implementarlo todo al mismo tiempo, pero normalmente éste no es el caso. En muchos casos, el diseño y la implementación se dividen en partes, y cada parte es verificada que trabaja antes de pasar a la siguiente parte. En este experimento, el problema no es difícil, pero es difícil de verificar qué está pasando en el DSK si la salida no es lo que se espera. Yo he visto la misma cosa sucediendo muchas veces: un estudiante escribe el código completo para el DSK en C y lo ejecuta.

Éste no funciona, y el estudiante no tiene ni idea por qué. En vez de implementar la solución toda de una vez en el DSK, es mejor implementar un filtro muy simple en el PC. De esta forma, cada paso del problema será verificado fácilmente, y fácil de depurar. El programa escrito para este experimento ayudará también con otros experimentos y probablemente con su proyecto final.

Yo recomendaría el siguiente procedimiento. Comience con un programa en el PC que pueda leer y escribir números desde un archivo simple ASCII. Esto sería básicamente el programa *loopc.c* desde el DSK, pero para el PC. Una vez el programa esté funcionando, utilícelo para implementar un filtro FIR muy simple de, por ejemplo, tercer orden. Asegúrese de que los números son leídos y escritos desde los archivos simulando lo que está sucediendo en el DSK. En otras palabras, en vez de leer todos los valores al principio, lea un valor desde el archivo, compute la salida, escríbala en otro archivo, lea el siguiente valor, etc. Este programa en C en el PC es muy sencillo de depurar, porque usted puede escribir los valores en cada paso en la pantalla. Seleccione los coeficientes del filtro y la entrada de forma que la salida sea fácil de calcular a mano y verifique si el programa está haciendo lo que usted quiere. Una vez que este programa esté funcionando, vaya un paso adelante e implemente el filtro en el DSK.

Experimentos

Experimento 1.1

Diseñe un filtro FIR de fase lineal en Matlab para las especificaciones de magnitud dadas en la **Figura 1.1**. Implemente el filtro utilizando *firnc.asm*. El filtro debe cumplir las especificaciones en MATLAB, pero no necesita cumplir las especificaciones cuando se implemente en el DSK.

- a. **Ajuste** la frecuencia de muestreo a 8kHz. ¿Cuál es la frecuencia de corte (-3dB) en Hz?
- b. **Cambie** la frecuencia de muestreo a 10kHz. ¿Cuál es la nueva frecuencia de corte? ¿Cuál debería ser teóricamente?
- c. **Reajuste** la frecuencia de muestreo a 8kHz y elimine el filtro "anti-aliasing". Conecte el osciloscopio de forma que pueda ver la entrada y la salida. Introduzca una frecuencia de 300Hz y determine el desfase entre la entrada y la salida. A continuación, verifique que el filtro es de fase lineal.
- d. **Conteste** la pregunta al final del enunciado. Con una frecuencia de muestreo de 8kHz y sin filtro "anti-aliasing", aumente la frecuencia de entrada desde 500Hz a 7.5kHz. ¿Qué ocurre con la señal? Explique por qué ocurre eso. ¿Es realmente un filtro de fase lineal?

Experimento 1.2

Diseñe un filtro FIR de fase lineal de mínimo orden para las especificaciones de magnitud dadas en la **Figura 1.2** en Matlab. Modifique **loopc.c** para implementar el filtro.

Muestre que el filtro que ha implementado en el DSK cumple las especificaciones.

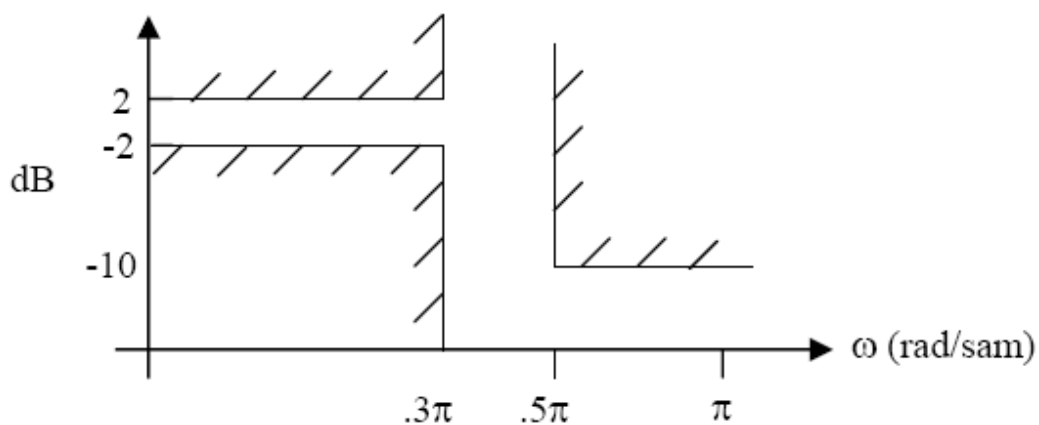


Figura 1.1. Especificaciones para el filtro pasobajo del experimento 1.1.

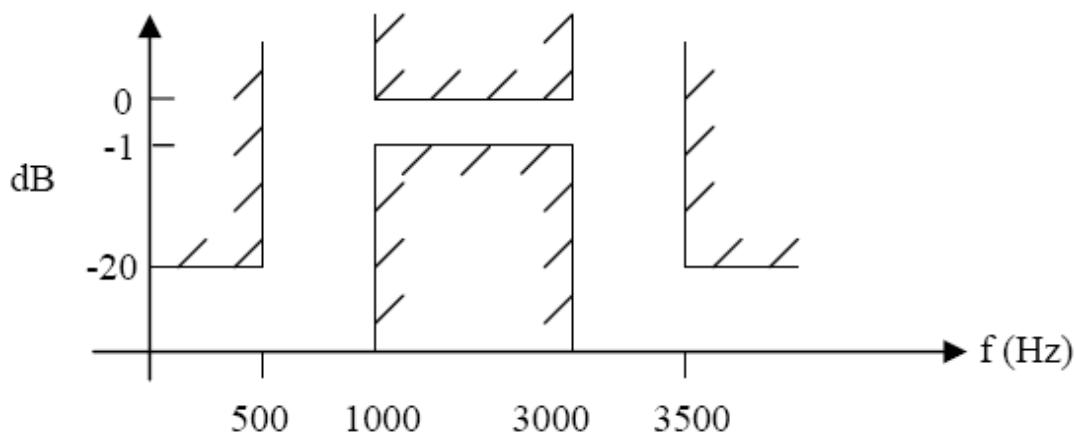


Figura 1.2. Especificaciones para el filtro pasobanda del experimento 1.2.