

IMPLEMENTACIÓN DE UN FILTRO DIGITAL EN VHDL

Eduardo Lahuerta Aguilar, Carlos Medrano Sánchez, Pedro Ramos Lorente.

Departamento de Ingeniería Electrónica y Comunicaciones.
Universidad de Zaragoza.

443597@ingtt.unizar.es ctmedra@posta.unizar.es pramos@posta.unizar.es

ABSTRACT

This paper presents the implementation of a digital FIR filter on VHDL as an alternative to the use of DSP based boards. We have used ALTERA MAX+PLUSII software. Afterwards the filter is implemented in the FLEX 10K20RC240-C device from the same manufacturer. In our case, this device is part of an evaluation board by ALTERA, the UP1 board. We show thereby the possibilities of these programmable devices for digital signal processing. This is a field in which manufacturers offer a wide range of registered products [1].

1. FILTROS DIGITALES

El filtro digital implementado es un filtro FIR (respuesta impulsional finita). Los filtros FIR tienen dos ventajas fundamentales con respecto a los filtros IIR (de respuesta infinita) [2], [3]. La primera es que se pueden diseñar con fase lineal, por lo que no presentan distorsión de fase. Además, son incondicionalmente estables, ya que su función de transferencia tiene sólo ceros. La presencia de polos en la función de transferencia de un filtro IIR compromete su estabilidad. Como desventaja frente a los IIR, hay que señalar que para unas mismas características, un filtro IIR requiere un menor número de coeficientes que un filtro FIR. Los buenos resultados obtenidos con filtros FIR en esquemas de filtrado adaptativo basados en DSP y el deseo de comparar los resultados con este prototipo, sugieren el empleo de un esquema de respuesta impulsional finita.

2. DISEÑO DEL FILTRO.

Para el cálculo de los coeficientes del filtro hacemos uso de MATLAB, y más concretamente de la herramienta SPTOOL [4]. En esta herramienta se permite seleccionar de un modo

sencillo, basado en un interfaz gráfico de usuario de fácil manejo, el tipo y el orden del filtro, así como las frecuencias que definen la banda de paso y la banda de rechazo. Empleando el método de igualdad de rizado en la banda de paso y en la atenuada obtenemos los 16 coeficientes de nuestro filtro de orden 15:

$h(0)=h(15)=217$; $h(1)=h(14)=128$; $h(2)=h(13)=158$;
 $h(3)=h(12)=186$; $h(4)=h(11)=212$; $h(5)=h(10)=233$;
 $h(6)=h(9)=247$; $h(7)=h(8)=255$;

El módulo de la función de transferencia del filtro se muestra en la siguiente figura:

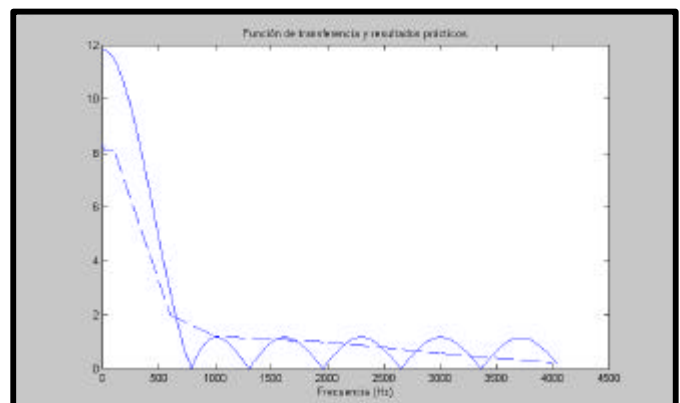


Figura 1. Función de transferencia del filtro FIR y resultados experimentales del filtrado

3. REALIZACIÓN DEL FILTRO EN VHDL.

Para implementar un filtro FIR en VHDL, y en un dispositivo FLEX, hacemos uso de la nota de aplicación de ALTERA número 73 [1], donde se muestra el esquema a seguir para llevar a cabo su desarrollo. La multiplicación y suma se pueden optimizar en los dispositivos FLEX utilizando el procedimiento de síntesis de funciones lógicas mediante multiplexores (LUT, Look-up-table). Las variables de selección del multiplexor dependen de la señal s y las entradas dependen de los coeficientes h . Para realizar el

filtro de 16 coeficientes, hacemos uso de dos filtros de 8 coeficientes cada uno. Los coeficientes del primer filtro son:

$$h(0)=h(7)=217; h(1)=h(6)=128; h(2)=h(5)=158; h(3)=h(4)=186$$

y su tabla será la siguiente:

S(4) _i	S(3) _i	S(2) _i	S(1) _i	Pi	resultado
0	0	0	0	0	0000000000
0	0	0	1	h(1)	0011011001
0	0	1	0	h(2)	0010000000
0	0	1	1	h(1)+h(2)	0101011001
0	1	0	0	h(3)	0010011110
0	1	0	1	h(3)+h(1)	0101110111
0	1	1	0	h(3)+h(2)	0100011110
0	1	1	1	h(3)+h(2)+h(1)	0111110111
1	0	0	0	h(4)	0010111010
1	0	0	1	h(4)+h(1)	0110010011
1	0	1	0	h(4)+h(2)	0100111010
1	0	1	1	h(4)+h(2)+h(1)	1000010011
1	1	0	0	h(4)+h(3)	0101011000
1	1	0	1	h(4)+h(3)+h(1)	1000110001
1	1	1	0	h(4)+h(3)+h(2)	0111011000
1	1	1	1	h(4)+h(3)+h(2)+h(1)	1010110001

Donde Pi son los productos parciales que deberán ser sumados con los pesos adecuados.

Los coeficientes y la tabla del segundo filtro son:

$$h(0)=h(7)=212; h(1)=h(6)=233; h(2)=h(5)=247; h(3)=h(4)=255;$$

S(4) _i	S(3) _i	S(2) _i	S(1) _i	Pi	resultado
0	0	0	0	0	0000000000
0	0	0	1	h(1)	0011010100
0	0	1	0	h(2)	0011101001
0	0	1	1	h(1)+h(2)	0110111101
0	1	0	0	h(3)	0011110111
0	1	0	1	h(3)+h(1)	0111001011
0	1	1	0	h(3)+h(2)	0111100000
0	1	1	1	h(3)+h(2)+h(1)	1010110100
1	0	0	0	h(4)	0011111111
1	0	0	1	h(4)+h(1)	0111010011
1	0	1	0	h(4)+h(2)	0111101000
1	0	1	1	h(4)+h(2)+h(1)	1010111100
1	1	0	0	h(4)+h(3)	0111110110
1	1	0	1	h(4)+h(3)+h(1)	1011001010
1	1	1	0	h(4)+h(3)+h(2)	1011011111
1	1	1	1	h(4)+h(3)+h(2)+h(1)	1110110011

Los dos filtros se colocan en serie, sumándose la salida de ambos, obteniéndose así la salida del filtro.

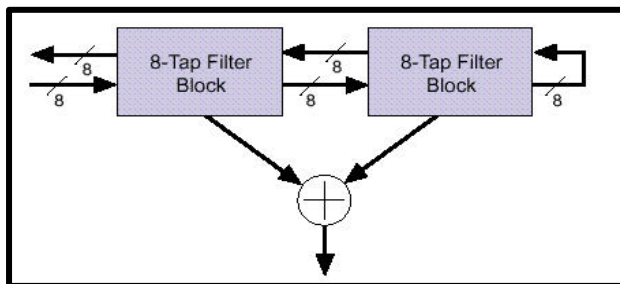


Figura 2: Diagrama de bloques del filtro.

Los elementos que integran el filtro: registros, sumadores, multiplicadores y tablas, se han desarrollado en VHDL.

4. DESARROLLO EXPERIMENTAL Y RESULTADOS.

Finalizado el proceso de desarrollo del filtro en VHDL, y después de realizar la compilación de éste y una simulación software, pasamos a implementarlo en el dispositivo FLEX10K20RC240-C

Las señales a filtrar deben estar en formato digital, por lo que la señal de entrada (los ficheros de audio digitalizados) se envían por el puerto paralelo del PC. Para ver la señal de salida ya filtrada, desarrollamos el hardware para realizar la conversión digital-analógica, basado en el conversor AD667. Las pruebas realizadas han sido de dos tipos: con una señal de voz mono de 8 bits, y con varias señales sinusoidales de distintas frecuencias. Con éstas últimas hemos realizado un barrido de frecuencia para verificar la bondad del filtrado. En cuanto al retardo introducido por el filtro, es de 7 ciclos de reloj, que en nuestro caso (dada una frecuencia de muestreo de 11025 Hz), suponen 0.6349 ms.

5. CONCLUSIONES.

Se ha implementado de forma satisfactoria (en términos del retardo introducido, coste, y selectividad del filtrado) un filtro FIR en un dispositivo programable de la familia FLEX10K de Altera. Este es un primer paso hacia un objetivo más ambicioso: estudiar la viabilidad del empleo de estos dispositivos como filtros adaptativos. En función de las prestaciones obtenidas, las posibilidades de optimización y el menor coste del esquema completo, la sustitución de sistemas basados en DSP por dispositivos lógicos programables se presenta como una opción interesante.

6. REFERENCIAS

- [1] "Implementing FIR Filters in FLEX Devices", Application note 73. Altera, www.altera.com
- [2] Proakis, J. G., Manolakis, D. G., "Tratamiento digital de señales". Prentice Hall. 1998.
- [3] Oppenheim, A.V., Schafer, R.W., "Discrete-Time Signal Processing". Prentice Hall. 1989.
- [4] "Signal Processing Toolbox User's Guide". The MathWorks Inc. 1999.