

Filtro FIR pasa bajo con selección de banda

Trabajo Práctico Final - Circuitos Lógicos Programables

Especialización en Sistemas Embebidos

Autor: Gerardo Alexis Vilcamiza Espinoza

Resumen

Se presenta el diseño, implementación y simulación de un filtro FIR pasa bajo con frecuencia de muestreo de 48 kHz, orden 63 y cuantización Q1.15. El filtro incorpora un selector de coeficientes de 2 bits para conmutar entre cuatro frecuencias de corte: 0.5 kHz, 1 kHz, 5 kHz y 10 kHz. La implementación se realizó en VHDL y la verificación en ModelSim y Vivado, utilizando una señal continua con diferentes frecuencias para evaluar la respuesta. Se documentan criterios de diseño, arquitectura, consideraciones de aritmética fija, latencia, y resultados esperados.

1. Introducción

Los filtros de respuesta al impulso finita (FIR) son ampliamente utilizados en sistemas embebidos por su estabilidad incondicional y la posibilidad de implementar respuestas con fase lineal. Un filtro FIR realiza una convolución discreta entre la señal de entrada y un conjunto finito de coeficientes. La ecuación de salida es

$$y[n] = \sum_{k=0}^M h[k] x[n-k], \quad (1)$$

donde M es el orden y $h[k]$ los coeficientes. En este trabajo se diseña un FIR pasa bajo con cuatro bandas seleccionables mediante hardware, lo que permite evaluar el desempeño del mismo circuito bajo distintas frecuencias de corte sin re-sintetizar.

2. Especificaciones y criterios de diseño

2.1. Requerimientos funcionales

- Frecuencia de muestreo: 48 kHz.
- Tipo de filtro: pasa bajo FIR, fase aproximadamente lineal.
- Orden: 63 taps efectivos (64 coeficientes).
- Cuantización de coeficientes e I/O: Q1.15 en 16 bits con signo.
- Selector de banda: 2 bits para conmutar 4 bancadas de coeficientes.
- Frecuencias de corte nominales: 0.5 kHz, 1 kHz, 5 kHz, 10 kHz.

2.2. Criterios de diseño de los coeficientes

- Método de ventana Hamming para un buen compromiso entre ancho de transición y rizado.
- Normalización de ganancia en DC para mantener amplitud unidad en banda pasante.
- Cuantización simétrica a Q1.15 con recorte al rango $[-32768, 32767]$.

2.3. Métricas esperadas

- Atenuación en banda de detención acorde a ventana Hamming para 64 taps.
- Rizado en banda pasante típico menor a 0.05 dB tras normalización y cuantización.
- Retardo de grupo aproximadamente constante en banda pasante.

2.4. Tabla de especificaciones

Parámetro	Valor	Comentario
F_s	48 kHz	fija
Orden	63	64 coeficientes
Formato	Q1.15	16 bits con signo
Selector	2 bits	4 bancadas
F_c opciones	0.5, 1, 5, 10 kHz	nominales
Ventana	Hamming	diseño FIR

Cuadro 1: Especificaciones del filtro FIR con banco de coeficientes.

3. Gráficas de respuesta en frecuencia

Este apartado recoge las respuestas en frecuencia de las cuatro bancadas de coeficientes. Las figuras se obtienen en Python con la librería scipy, a partir de los coeficientes cuantizados Q1.15.

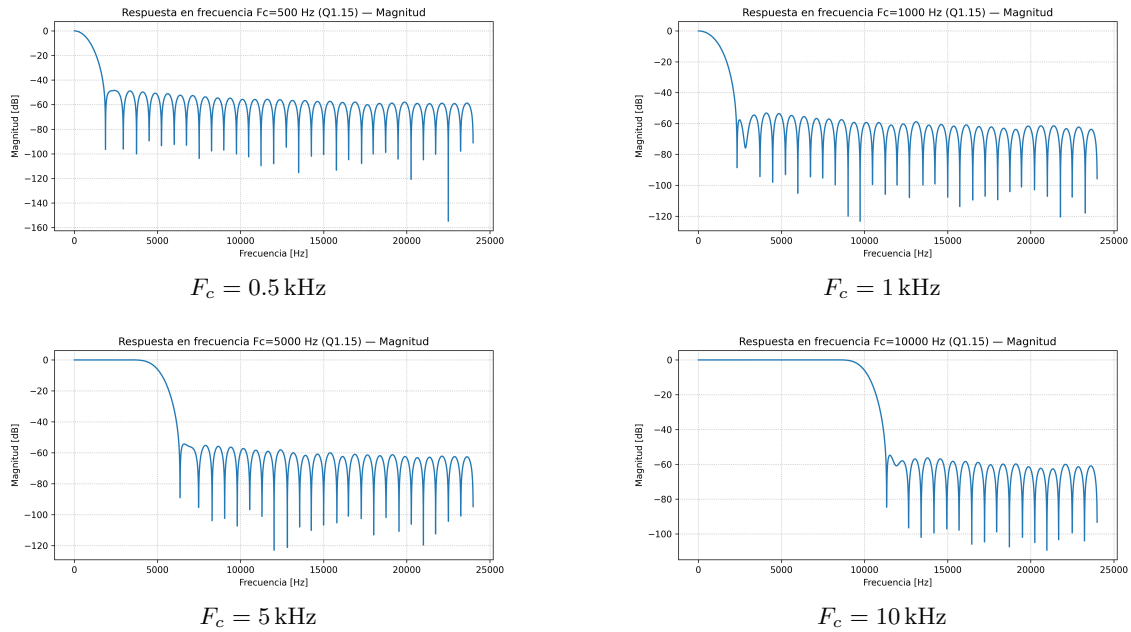


Figura 1: Respuestas en frecuencia del filtro para distintas frecuencias de corte.

4. Arquitectura e implementación

4.1. Visión general

El sistema se compone de:

1. Un paquete de coeficientes con cuatro bancadas $\text{COEFFS}[0..3]$, cada una con 64 coeficientes Q1.15.
2. Un bloque FIR *fir_lp_var* con:
 - Registro de desplazamiento de 64 muestras.
 - Multiplicación 16x16 por tap y acumulación ensanchada.
 - Redondeo y saturación hacia 16 bits.
 - Selector de 2 bits que elige la bancada de coeficientes en tiempo de ejecución.

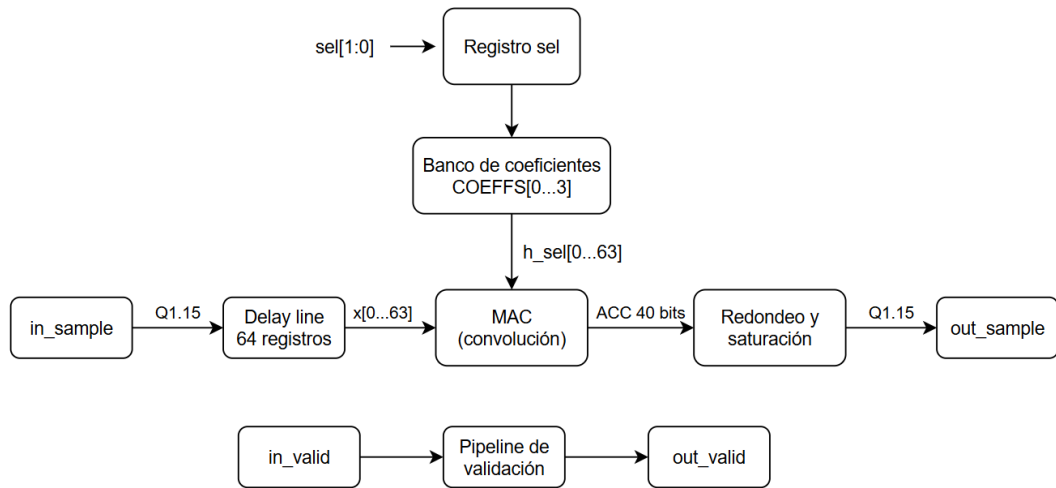


Figura 2: Diagrama en bloques del sistema.

4.2. Diagrama de bloques del filtro

4.3. Aritmética fija y ancho del acumulador

La entrada y los coeficientes se representan en Q1.15. La multiplicación 16x16 genera un producto de 32 bits con signo. Para la suma de 64 productos se emplea un acumulador de 40 bits, lo que provee margen frente a la suma de magnitudes y a errores de cuantización. Al finalizar se aplica redondeo hacia el bit 16, seguido de saturación al rango Q1.15.

4.4. Latencia y temporización

La arquitectura directa con registro de desplazamiento introduce una latencia de un ciclo entre la captura de la muestra y la entrega del resultado redondeado, además del retardo de fase del FIR por su naturaleza lineal. Si fuera necesario incrementar frecuencia de reloj se pueden introducir etapas de pipeline en el árbol de suma o en las multiplicaciones.

4.5. Selector de coeficientes

El selector interpreta $sel[1:0]$ como un índice de 0 a 3 y direcciona la bancada correspondiente sin reconfiguración adicional. Esto permite cambiar F_c en tiempo de simulación o durante operación sin costo de re-síntesis.

5. Metodología de verificación

5.1. Señal de estímulo por segmentos

Para evaluar el comportamiento en banda pasante y detención se inyecta una señal continua formada por tramos sinusoidales con frecuencias representativas: 300 Hz, 800 Hz, 2 kHz, 8 kHz y 12 kHz. Cada tramo dura 1 s a 48 kHz. La cuantización es Q1.15. Este estímulo permite observar la atenuación progresiva al conmutar F_c y comparar la salida con la entrada.

$$0.7 \sin(2\pi \cdot 300t) + 0.3 \sin(2\pi \cdot 800t) + 0.3 \sin(2\pi \cdot 2000t) + 0.2 \sin(2\pi \cdot 8000t) + 0.1 \sin(2\pi \cdot 12000t) \quad (2)$$

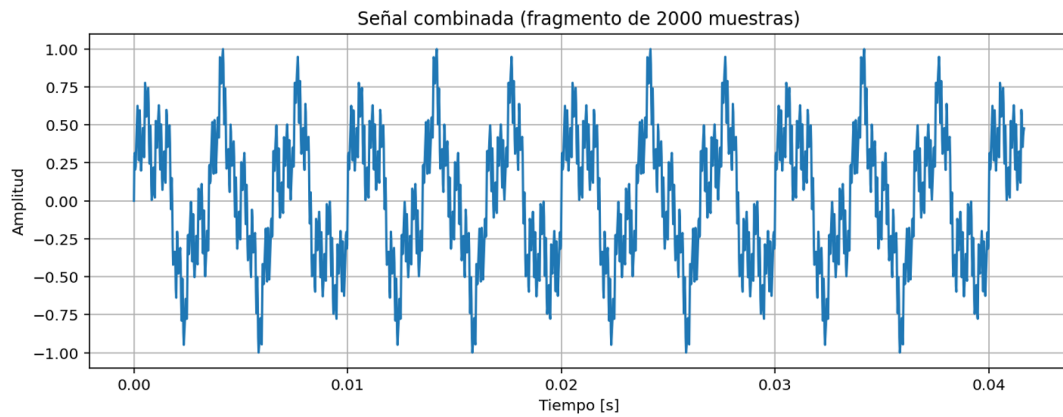


Figura 3: Señal de entrada para la simulación del modelo.

5.2. Banco de pruebas y observables

El banco de pruebas en Vivado:

- Lee las muestras desde `stimulus.txt` y aplica una por ciclo con `in_valid=1`.
- Conmuta `sel` en cuatro tramos del archivo para activar cada bancada.
- Registra en ondas: `clk`, `rst`, `in_valid`, `in_sample`, `sel`, `out_valid`, `out_sample`, y señales internas como `acc` y `xreg`.

6. Simulación

Como se observa en la Figura 4, se resalta en color cian la señal original de entrada y en naranja la señal de salida ya filtrada. En rojo se puede observar el seleccionador que cambia la frecuencia de corte del filtro.

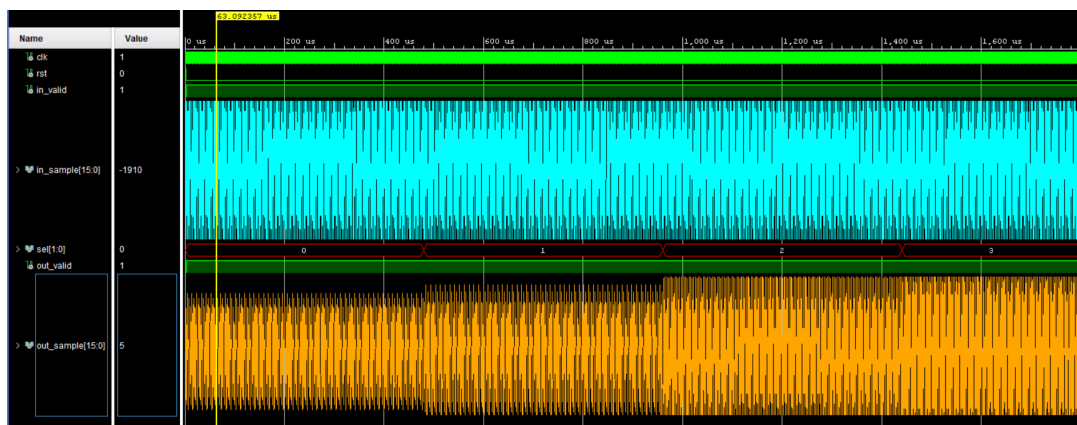


Figura 4: Resultados de simulación (vista completa).

En el testbench se definió que las frecuencias de corte vayan en orden creciente, es decir, de la que deja pasar solo frecuencias más bajas (filtrado más drástico) a la que deja pasar frecuencias más altas (filtrado más permisible). Un caso donde se nota bastante la diferencia y la repercusión del filtro es cuando se pasa de una frecuencia de corte de 1kHz a 5kHz, que es el que se muestra a continuación en la Figura 5.

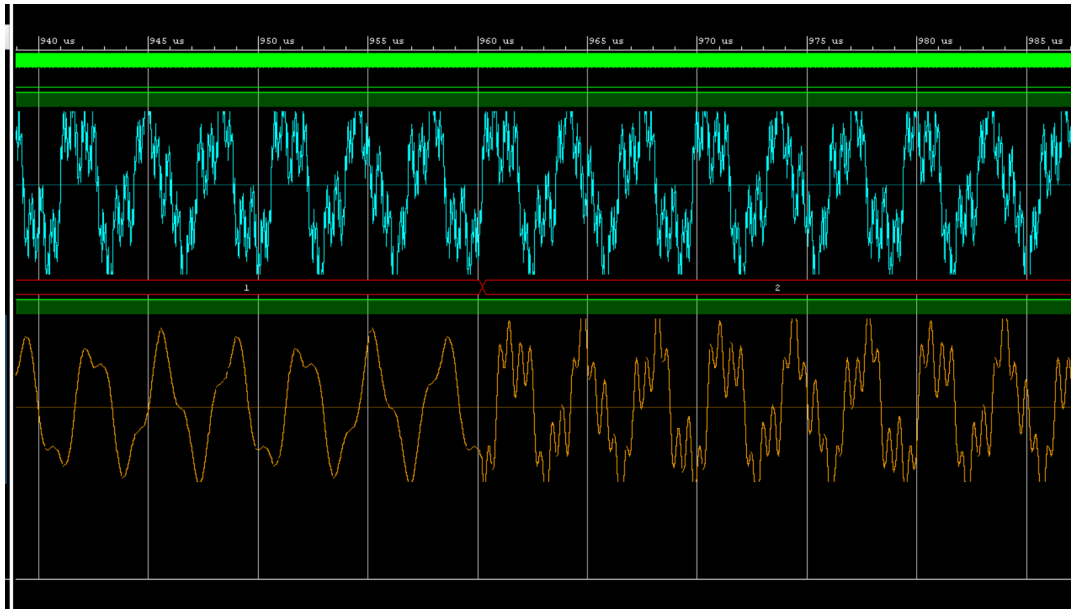


Figura 5: Resultados de simulación (vista pase de 1kHz a 5kHz).

En esa gráfica se observa claramente cómo la señal de salida pasa de una señal más limpia a un más ruidosa

De igual manera en la Figura 6, donde se pasa de una frecuencia de corte de 5kHz a 10KHz, donde se observa que al final la señal de salida es muy cercana a la señal original, ya que están pasando casi todas sus frecuencias, a excepción de su componente de 12kHz

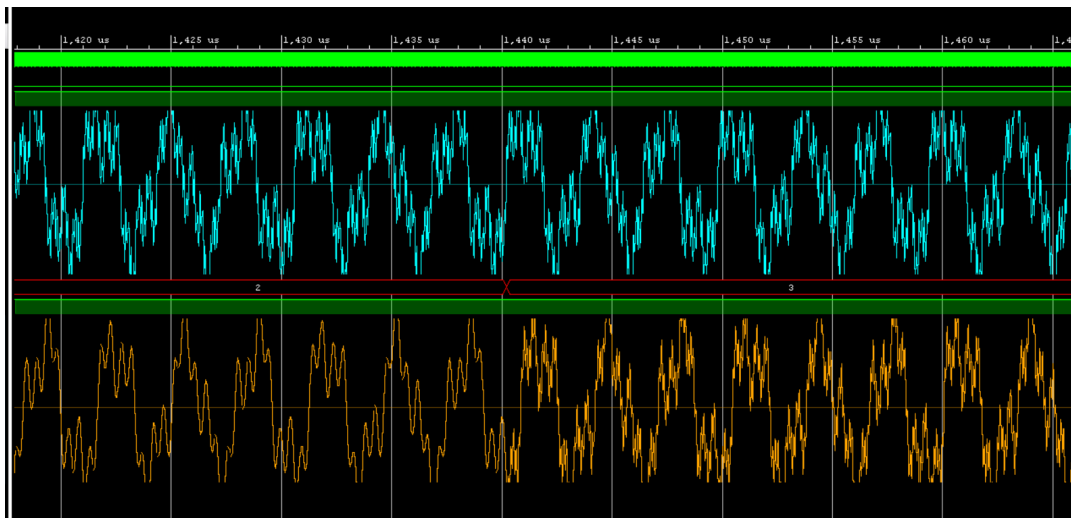


Figura 6: Resultados de simulación (vista pase de 5kHz a 10kHz).

7. Conclusiones

- Se implementó con éxito un filtro FIR pasa bajo con selección dinámica de coeficientes mediante hardware, permitiendo conmutar entre cuatro frecuencias de corte sin necesidad de re-síntesis. El diseño mostró un comportamiento estable, de fase aproximadamente lineal y con respuesta coherente con las especificaciones establecidas.
- El uso de aritmética fija en formato Q1.15, junto con un acumulador de 40 bits, proporcionó un equilibrio adecuado entre precisión numérica y eficiencia de recursos lógicos, evitando saturaciones internas durante las simulaciones.

- Los resultados confirman que la arquitectura es funcional, escalable y apta para ser integrada en sistemas embebidos de procesamiento digital de señales, donde se requiera ajuste de frecuencia de corte en tiempo real.