

## Laboratorio 2: Implementación de filtros

Gonzalo G. Fernández

24 de septiembre de 2022

### Efectos de cuantización

Ejecutar los script de python con el objetivo de comprender los efectos de cuantización de los coeficientes.

- fir\_filter\_direct\_form.ipynb
- iir\_filter\_direct\_form.ipynb
- IIR.Filter.Design.ipynb

Instanciar y ejecutar el testbench de cada uno de los filtros.

- filtro\_fir.v, tb\_filtro\_fir.v
- iir.v, filter\_tb.v
- iir\_top.v, iir\_filter.v, coeffSec1.v, coeffSec2.v, coeffSec3.v, tb\_iir\_filter.v

Los resultados de las simulaciones de las Fig. 1, 2 y 3.

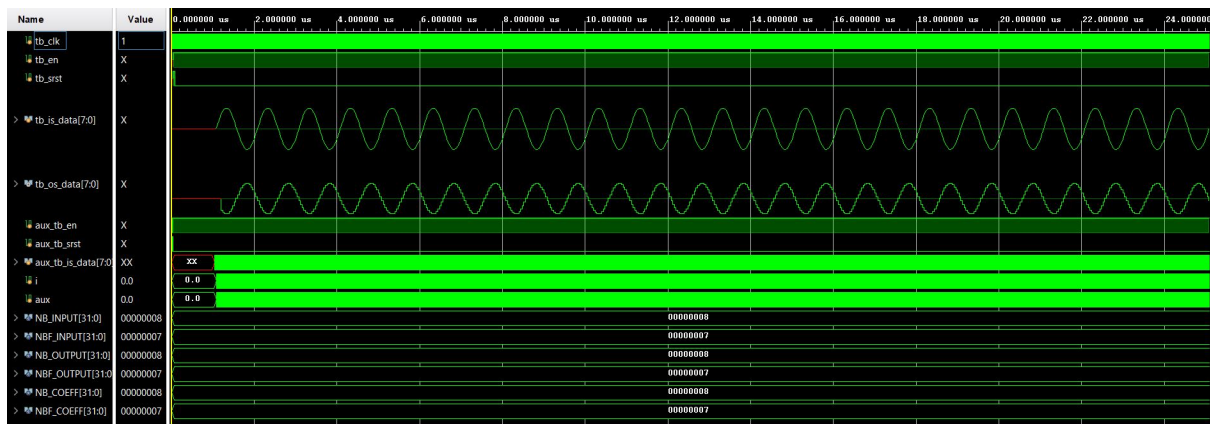


Figura 1: Forma de onda obtenida del testbench `tb_filtro_fir.v`

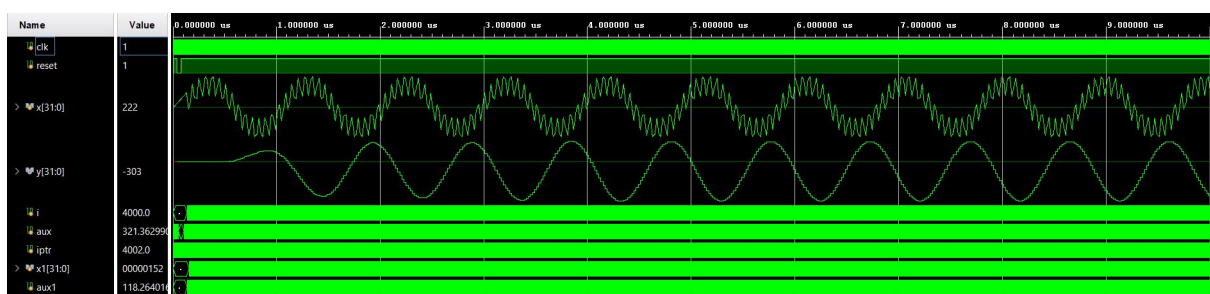


Figura 2: Forma de onda obtenida del testbench `filter_tb.v`

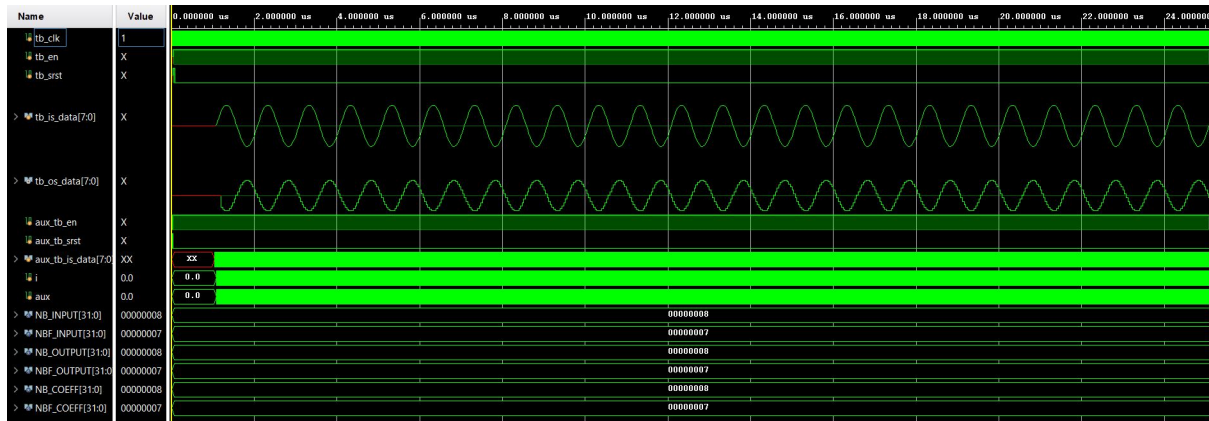


Figura 3: Forma de onda obtenida del testbench tb\_iir\_filter.v

## Primer modelo

Implementar en FPGA el filtro FIR según los siguientes archivos:

- top.design.v
- signal\_generator.v
- filtro\_fir.v
- SatTruncFP.v

Agregar los IPs VIO e ILA para controlar en forma remota el diseño.

## Laboratorio

Considerar un sistema de transmisión compuesto por una señal senoidal y un filtro pasa bajo con las siguientes características:

- Señal senoidal compuesta por dos frecuencias  $f_1 = 17kHz$  ( $A = 0,5$ ) y  $f_2 = 1,5kHz$  ( $A = 1,0$ )
- Frecuencia de muestreo  $f_s = 48kHz$
- Filtro pasa bajo con frecuencia de corte  $f_{cut} = 8kHz$

Desarrollo del modelo:

1. Utilizando el script de python coeff.ipynb, determinar los coeficientes del filtro para una frecuencia de corte de  $f_{cut} = 8kHz$ . El filtro debe tener una longitud de 15 coeficientes.
2. Realizar el diagrama de bloques del filtro (Fig. 4).
3. Generar un proyecto con los archivos entregados por la cátedra con la herramienta Vivado.
4. Configurar el archivo mem.hex con las señales senoidales especificadas previamente utilizando el script genmem.py.
5. Generar los coeficientes del filtro en Verilog con los valores de los coeficientes cuantizados (sintetizar cada filtro por separado).
6. Implementar en FPGA y graficar las señales senoidales pre y pos filtradas (Fig. 5, 6 y 7).

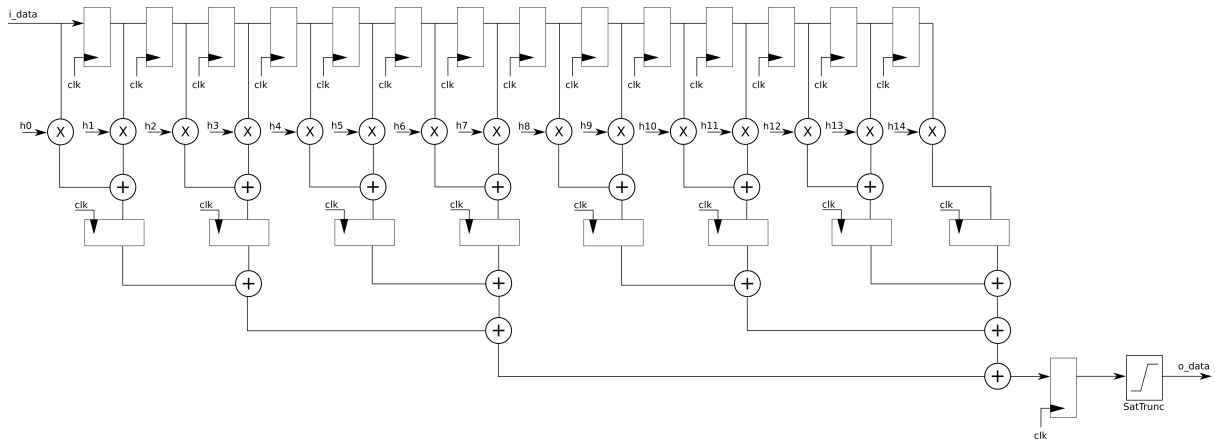


Figura 4: Diagrama de bloques del filtro

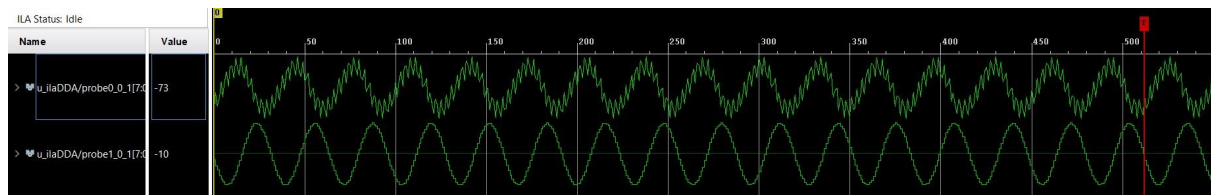


Figura 5: Forma de onda obtenida de ILA con implementación de filtro con frecuencia de corte  $0,5kHz$

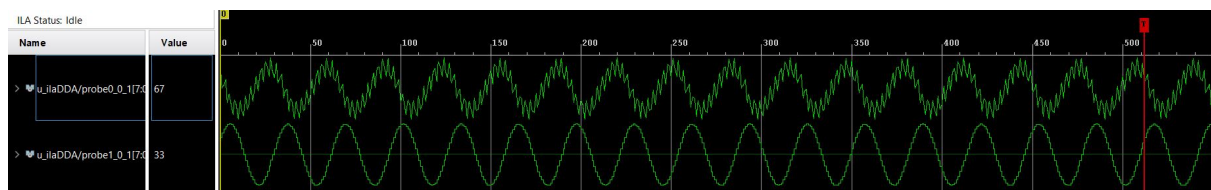


Figura 6: Forma de onda obtenida de ILA con implementación de filtro con frecuencia de corte  $8kHz$

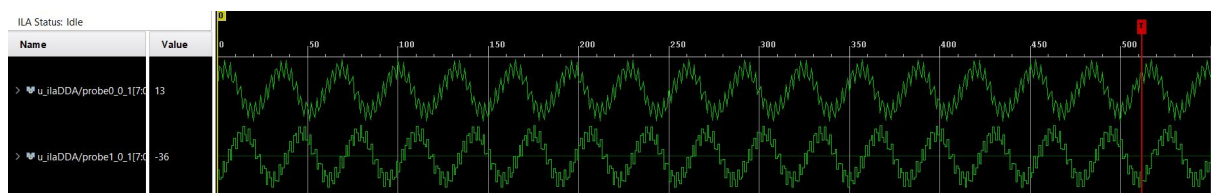


Figura 7: Forma de onda obtenida de ILA con implementación de filtro con frecuencia de corte  $18kHz$