

Prueba 2: Procesado digital de la señal con FPGAs

Alumno:

10/5/2019

Se desea diseñar un filtro diezmador por 5 para reducir la tasa de muestreo a $f_{SL} = 50$ MHz de una señal capturada a $f_{SH} = 250$ MHz. El filtro diezmador tiene 45 coeficientes, ganancia 4, y su respuesta en frecuencia y respuesta al impulso se muestran en las figuras 1 y 2. Para la implementación del filtro se propone la arquitectura hardware que se muestra en la figura 3.

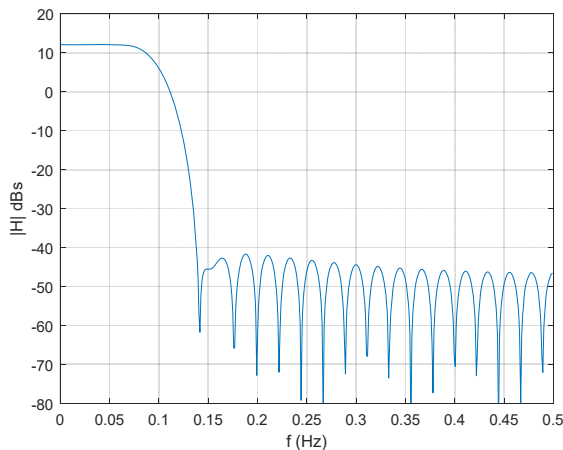


Figura 1. Respuesta en frecuencia del filtro

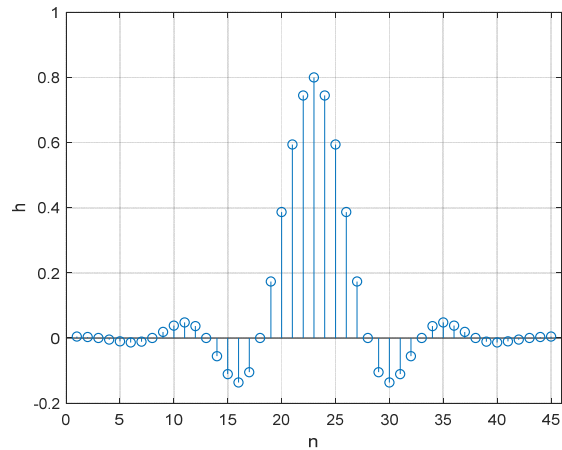


Figura 2. Respuesta al impulso

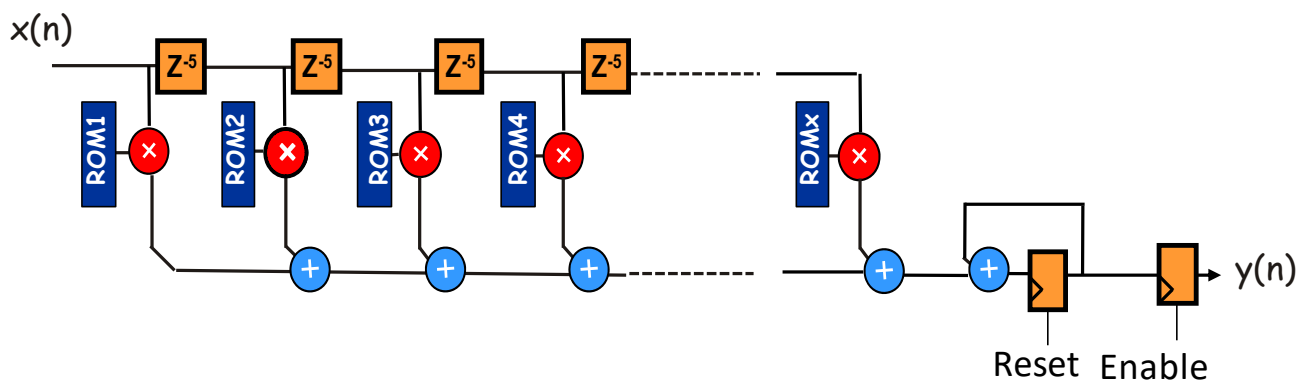


Figura 3. Arquitectura Polifásica del filtro diezmador

1) El filtro está formado por x etapas básicas formadas por un retardo de 5 ciclos, una ROM, un multiplicador y un sumador. ¿Cuántas etapas se requieren para la implementación del filtro? Justifique su respuesta. (1 punto)

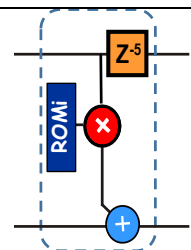


Figura 4. Celda básica

2) Indique a qué frecuencia debe funcionar cada uno de los componentes del filtro. Explique para qué sirven las señales de control Reset y Enable. ¿Qué secuencia deberá seguir cada una de ellas? (1.5 puntos)

3) Indique qué coeficientes se almacenan en cada memoria del filtro. (1 punto)

4) Dibuje (sobre el esquema de la figura 3) el camino crítico del circuito y halle la máxima frecuencia de funcionamiento sabiendo que $t_{\text{mult}}=3.5\text{ns}$ y $t_{\text{add}}=1.5\text{ns}$ (NOTA: considere nulos los tiempos de set-up y propagación de los registros para resolver este ejercicio). (1 punto)

5) Proponga la segmentación adecuada para implementar el filtro diezmador en un dispositivo FPGA Virtex 7 y que pueda funcionar a la frecuencia de reloj de 250 MHz. Tenga en cuenta que el dispositivo Virtex 7 dispone de bloques DSP48, cuyo esquema se muestra a continuación. Indique cual es la latencia adicional del filtro una vez segmentado. (1.5 puntos)

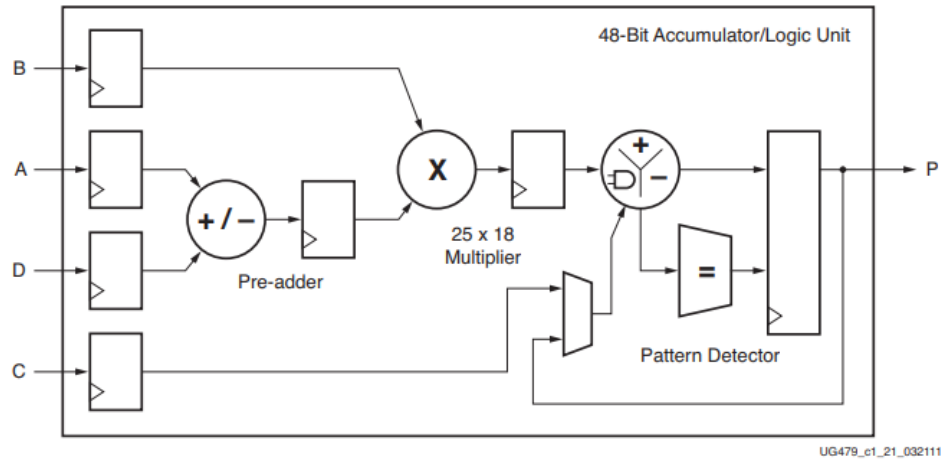


Figura 5. Estructura del DSP48E1 (Virtex-7)

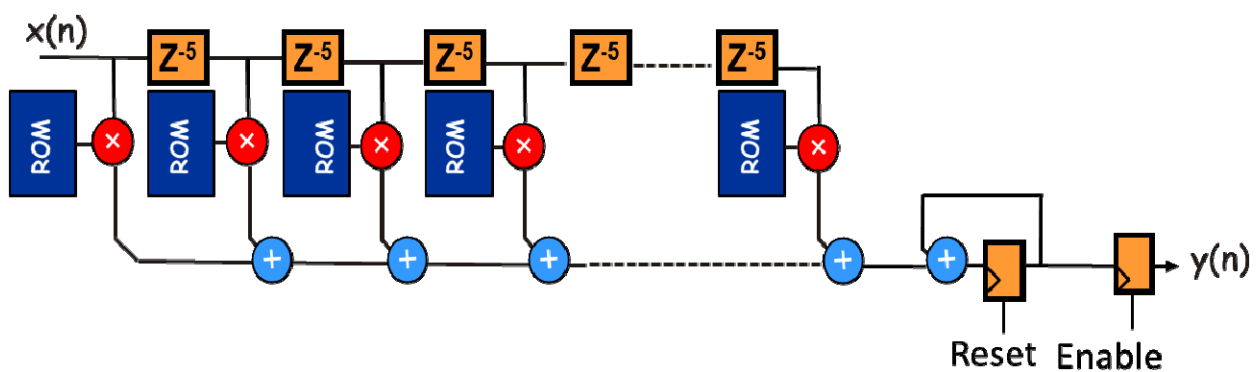


Figura 6. Estructura del filtro diezmador

6) Se desea cuantificar la entrada de datos y los coeficientes en complemento a dos con el formato [10,9] con signo. Se debe cuantificar la salida de forma que se evite el desbordamiento y se mantengan 9 bits fraccionales de precisión a la salida, mientras que internamente el filtro deberá operar con precisión completa. Cuantifique todos los operadores internos del filtro. Si en alguna conexión entre operadores se requiere cambiar el formato indíquelo con un bloque Q y escriba el nuevo formato a su salida. (1.5 puntos)

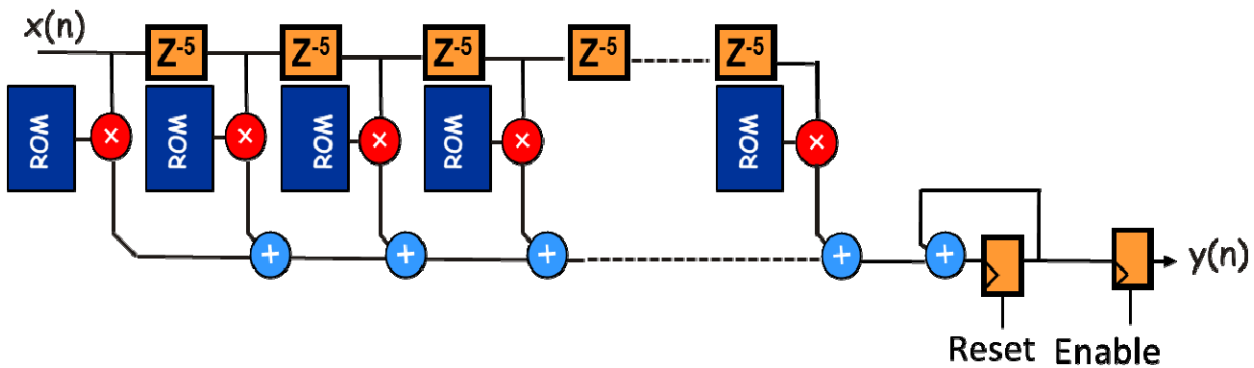
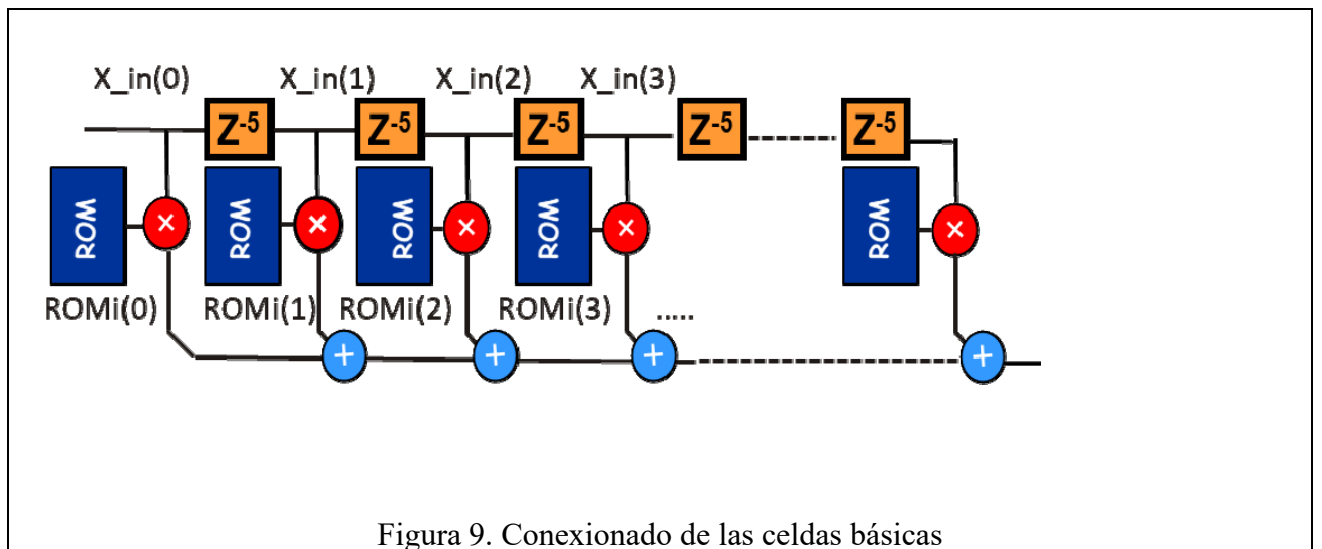
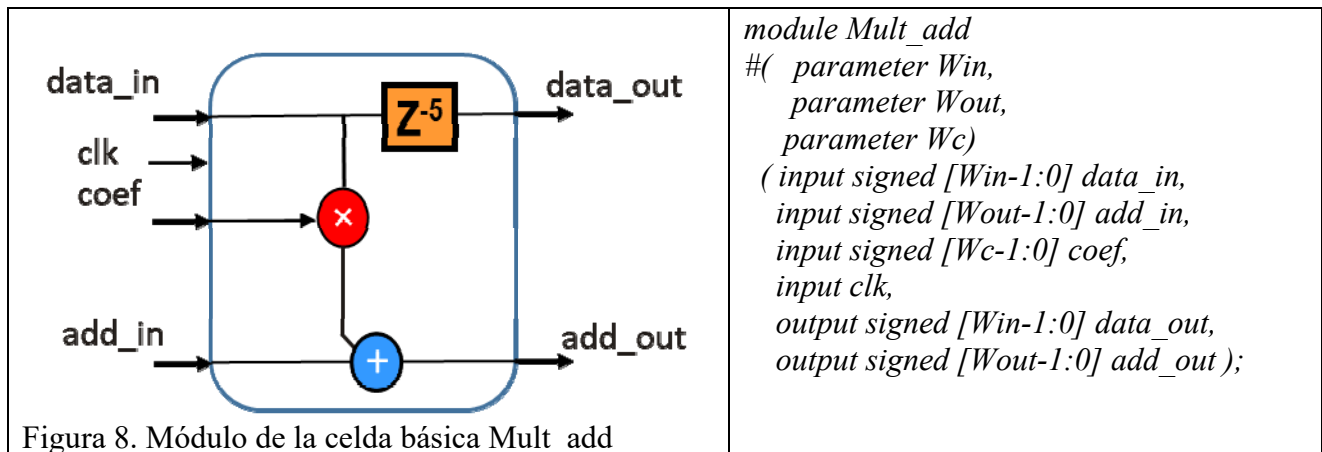


Figura 7. Estructura del filtro diezmador

7) Suponiendo que tenemos codificado en Verilog el módulo mostrado en la figura 8, ¿cómo realizaría la instanciación de M módulos de forma eficiente? Complete el código verilog proporcionado para instanciar los M módulos. (1.5 puntos)



```

wire signed [      ] X_in[      ];
wire signed [      ] ROMi[      ];
wire signed [      ] addi[      ];

```

```

genvar i;

```

```

    for (
        begin: Madds

```

```

        end
    endgenerate

```

8) Codificar, utilizando el lenguaje de descripción hardware Verilog, la celda básica de la figura 10. (1 punto)

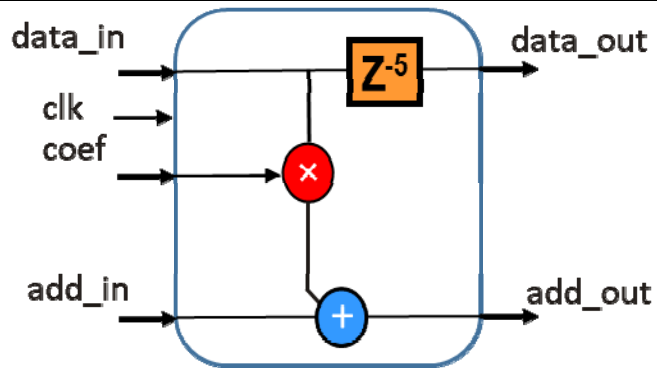


Figura 10. Módulo de la celda básica Mult_add

```
module Mult_add
#( parameter Win,
  parameter Wout,
  parameter Wc)
( input signed [Win-1:0] data_in,
  input signed [Wout-1:0] add_in,
  input signed [Wc-1:0] coef,
  input clk,
  output signed [Win-1:0] data_out,
  output signed [Wout-1:0] add_out );
```

```
endmodule
```