

Guía Práctica 2

Ejercicio 1 - Sugerido.

- El objetivo es diseñar un multiplicador de punto flotante de 13 bits con las siguientes características.
 - El número de punto flotante tiene 1 bit de signo s , 4 bits exponenciales e y 8 bits de la mantisa m .
 - Asumir un valor de sesgo de 7 para la representación. Usar un multiplicador no signado de 8 x 8 bits para multiplicar dos mantisas, un sumador y restador de 4 bits para sumar dos exponentes y luego restarle el valor de sesgo para eliminar el efecto de este sobre la suma.
 - Dibujar el Diagrama en bloques del diseño y codificarlo en verilog.
 - Escribir los estímulos para verificar el funcionamiento del diseño.

Ejercicio 2.

Realizar en papel la multiplicación los siguientes números de 8 bits considerándolos como U x U, U x S, S x U y S x S, donde U y S hacen referencia a números no signados (unsigned) y signados (signed), respectivamente:

```
1 A = 8'b1011_0011
  B = 8'b1100_0101
```

Las operaciones se deben realizar considerando:

- Ambos números como enteros.
- Ambos números con formato S(8,7) (1 bit entero, 7 bits fraccionales).
- A como un entero y B con formato S(8,7) (1 bit entero, 7 bits fraccionales).

Ejercicio 3.

Escribir el código RTL para el esquema de la Fig. 1, agregando redondeo y truncado (ambas operaciones) en el **multiplicador** de 17 x 17 bits. El diseño debe generar un resultado de 32 bits con una señal de control (flag) que avise si el truncado de la multiplicación realizó un overflow.

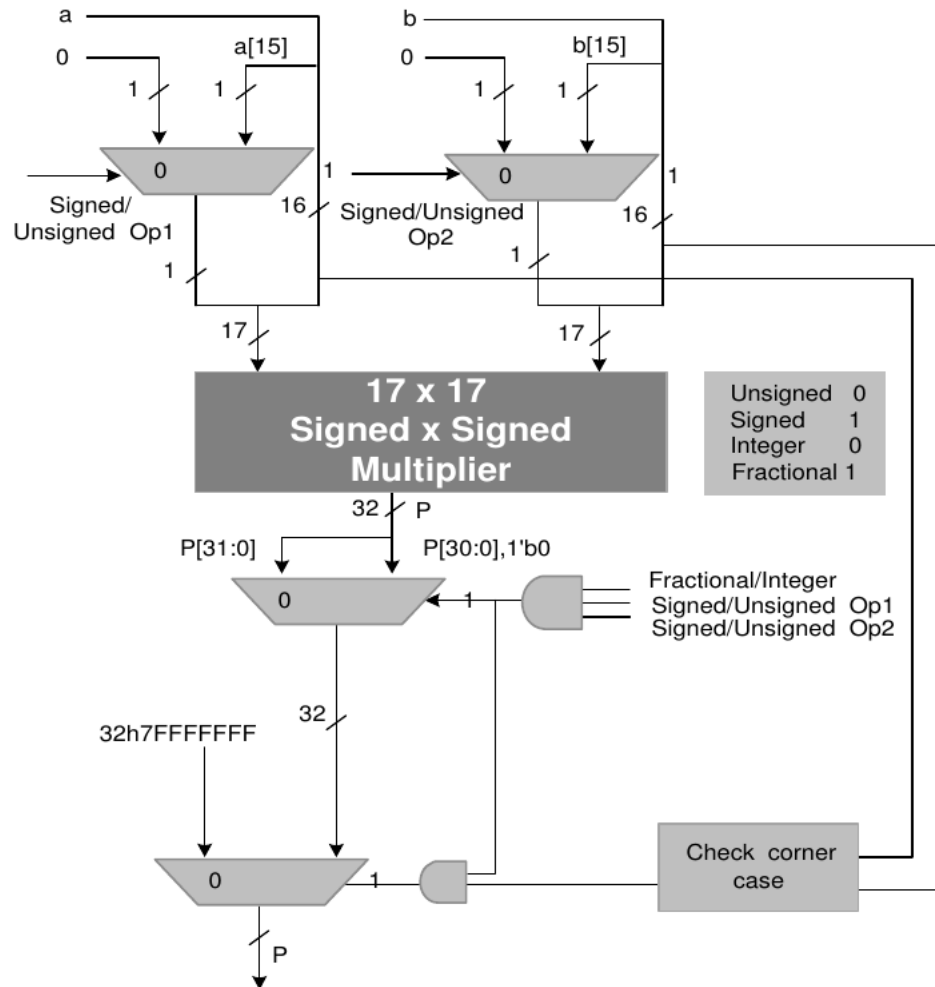


Figura 1: Multiplicador Uniforme (Libro Fig. 3.18)

Ejercicio 4 - Sugerido.

Escribir el código RTL Verilog para implementar el filtro FIR de cuatro coeficientes que se muestra en la figura 2.

- Asuma que todos los coeficientes ($h_0 \dots h_3$) y los datos de entrada $x[n]$ se encuentran en formato S(16,15).
- Realizar el truncado en el resultado de las multiplicaciones para implementar sumadores de 18 bits y defina el formato para $y[n]$.
- Proponga una optimización del diseño asumiendo que los coeficientes del filtro son simétricos.
- Re-escriba el código RTL Verilog pero reemplazando el truncado del apartado b) por redondeo.

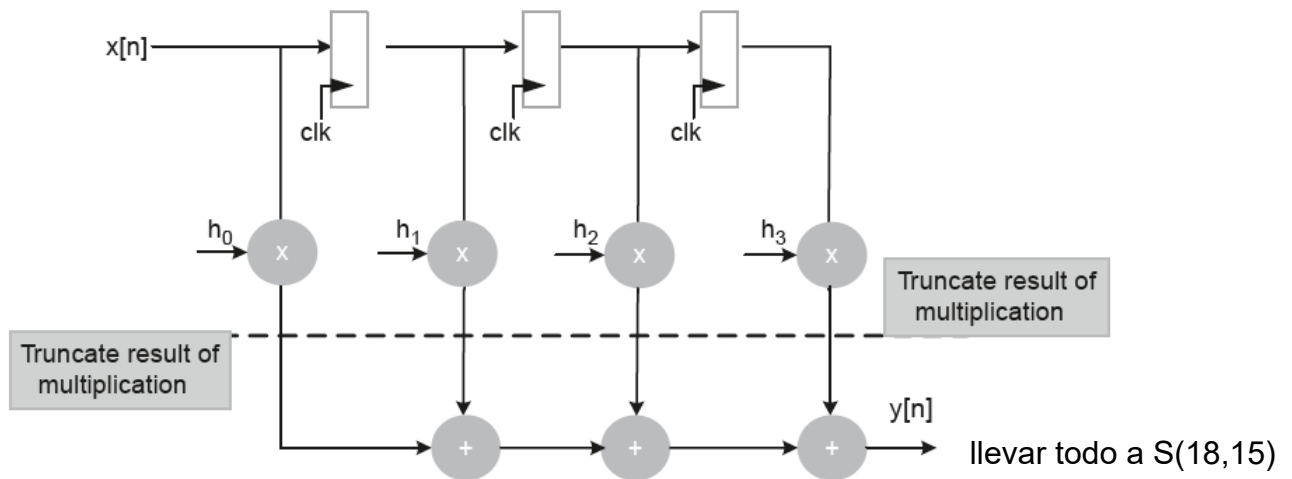


Figura 2: Filtro FIR de cuatro coeficientes