

Prueba 2: Procesado digital de la señal con FPGAs

Alumno:

14/5/2021

Ejercicio 1 (10 Puntos)

Se desea diseñar un filtro paso-banda de 400 coeficientes simétricos, con ganancia 18.05dB. Los requisitos del filtro especifican que deberá trabajar a una frecuencia de muestreo $f_s = 1$ MHz. Para ello se propone utilizar una arquitectura secuencial similar a la que se muestra en la figura 1.

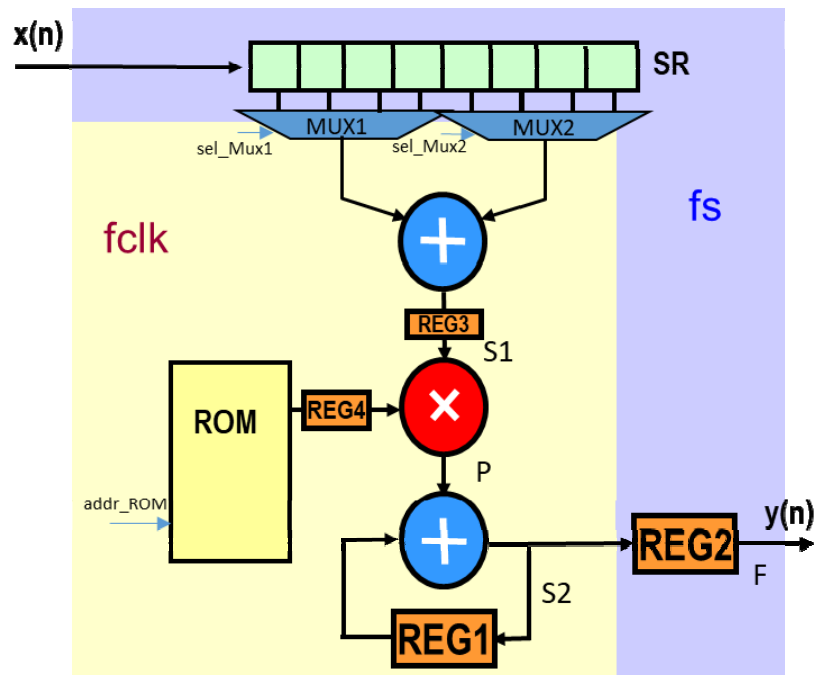


Figura 1. Arquitectura Secuencial

- a) (2 puntos) Explique el funcionamiento del sistema, detallando claramente:
- La **ecuación en diferencias** que está procesando
 - Cómo se consigue computar esta ecuación en diferencias
 - La relación f_{clk}/f_s que deberá cumplirse para poder utilizar esta arquitectura

Ecuación en diferencias:

$$y(n) = h_0 \cdot [x(n) + x(n - 399)] + h_1 \cdot [x(n - 1) + x(n - 398)] + h_2 \cdot [x(n - 2) + x(n - 397)] + \dots + h_{199} \cdot [x(n - 199) + x(n - 200)]$$

Esta arquitectura utiliza un solo multiplicador-acumulador, de manera que el cálculo de la ecuación en diferencias del filtro se realiza iterando sobre este operando. La arquitectura secuencial propuesta aprovecha que los coeficientes son simétricos, de manera que realiza la suma de aquellos datos que se multiplicarán por el mismo coeficiente, tras lo cual realiza la multiplicación y por último acumula los productos. Se deberá poner a cero la acumulación cada vez que comienza el cómputo de una nueva muestra.

Dado que tenemos 200 coeficiente diferentes, se necesitarán 200 ciclos de reloj para computar una muestra de salida del filtro, por lo tanto, la relación mínima que debe existir entre f_{clk} y f_s es $f_{clk}/f_s = 200$.

b) (2 puntos) Describa cuál será el **camino crítico** del circuito y halle:

- la máxima frecuencia de funcionamiento
- el throughput máximo que se podrá alcanzar con este dispositivo
- la carga computacional del algoritmo a implementar
- la capacidad computacional de la arquitectura de la figura 1

(NOTA: considere los tiempos de propagación de los operadores $t_{mux}=4.1ns$, $t_{mult}=5 ns$, $t_{add}=3.6 ns$ y $t_{ROM}=2.1ns$)

¿Cumple la arquitectura propuesta con los requisitos del filtro? En caso de que no cumpla los requisitos temporales: proponga algún cambio que permita cumplir con los requisitos temporales.

*El **camino crítico** será el de mayor tiempo de propagación. Observando la figura 1 se obtienen dos candidatos:*

1- Multiplexor más un sumador:

$$t_{crit1} = t_{mux} + t_{add} = 4.1e-9 + 3.6e-9 = 7.7 ns$$

2- Sumador más un multiplicador:

$$t_{crit2} = t_{add} + t_{mux} = 3.6e-9 + 5e-9 = 8.6 ns$$

Vemos que el camino crítico viene dado por la opción 2 (1 multiplicador más un sumador)

La frecuencia máxima se calculará teniendo en cuenta el camino crítico

$$\rightarrow f_{clk_max} = 1/t_{crit2} = 1/8.6e-9 = 116.28 MHz$$

$$\text{El throughput máximo será } T = f_{clk}/200 = 116.28 MHz/200 = 581.4 kHz$$

$$\text{La carga computacional del algoritmo será } C_L = 200 \text{ mult} * 1 MHz = 200 MMult/s$$

$$\text{La capacidad computacional de la arquitectura será } C_p = 116.28 MMult/s$$

Para averiguar si cumple con los requisitos temporales deberemos comparar la carga computacional del algoritmo (C_L) con la capacidad computacional de la arquitectura (C_p).

$$C_L = 200 MMult/s$$

$$C_p = 116.28 MMult/s < 200 MMult/s \rightarrow \text{NO cumple los requisitos temporales}$$

Para que cumpla con los requisitos temporales habría que segmentar a la salida de todos los operadores, de este modo el camino crítico sería 1 multiplicador, por lo que la frecuencia máxima de reloj será $f_{max} = 1/(5e-9) = 200 MHz$.

$$f_{clk}/f_s = 200 MHz/1MHz = 200 \rightarrow \text{Cumple los requisitos de velocidad}$$

c) (2 puntos) Rellene la tabla indicando cómo se direcciona la memoria ROM y los multiplexores en cada uno de los ciclos indicados en la arquitectura modificada según lo contestado en el apartado b. Suponga que en el ciclo 0 entra un dato al registro de desplazamiento.

Dado que tenemos el circuito completamente segmentado los datos de salida de los multiplexores tardarán dos ciclos en llegar al multiplicador, mientras que el dato de salida de la ROM sólo tarda un ciclo. Por lo tanto, habrá un desfase de un ciclo entre el direccionamiento de los multiplexores y la ROM

Direccionamiento de la memoria ROM y de los multiplexores

Ciclo (fclk)	addr_ROM	sel_MUX1	sel_MUX2
0	X	X	X
1	X	0	199
2	0	1	198
3	1	2	197
4	2	3	196
⋮			
199	197	198	1
200	198	199	0
201	199	0	199

- d) (1.5 puntos) Se desea cuantificar la entrada de datos y los coeficientes en complemento a dos con formato [16,15] y [20,19] respectivamente. Se debe cuantificar la salida de forma que se evite el desbordamiento y se mantengan 15 bits fraccionales de precisión a la salida, mientras que internamente el filtro deberá operar con precisión completa. Cuantifique todos los operadores internos del filtro (puntos S1, S2, P y F).

La ganancia del filtro es 18.05 dB $\rightarrow g_{lin} = 10^{(18.05/20)} = 7.98$

Rango de entrada [-1, +1] \rightarrow Rango de salida [-7.98, 7.98]

Se necesitan 4 bits de parte entera para representar el rango de salida sin desbordamiento.

Entrada	Coeficientes	S1	P	S2	F
sfix[16,15]	sfix[20,19]	sfix[17,15]	sfix[37,34]	sfix[38,34]	sfix[19,15]

- e) (1.5 puntos) Se ha codificado una función Matlab que computa la salida de un filtro FIR con precisión completa. La llamada a la función toma la forma $y = \text{FIR_filter}(x, c)$; siendo x los datos de entrada y c los coeficientes del filtro, ambos se proporcionan en formato double.

Escriba el script de Matlab que modela el comportamiento del filtro cuantificando todos los datos y coeficientes tal y como se indica en el **apartado d**. Utilice para ello las funciones floor y round de Matlab.

```
x_q = floor(x*2^15)*2^-15;
c_q = round(x*2^19)*2^-19;
y = FIR_filter(x_q,c_q);
y_q = floor(y*2^15)*2^-15;
```

- f) (1 punto) Rellene la tabla siguiente indicando el número de flip-flops que hacen falta para implementar cada uno de los bloques indicados en la figura 1.

Bloque	Número de flip-flops (FFs)
SR	400 FFs x 16 bits = 6400 FFs
REG1	38 FFs
REG2	19 FFs
REG3	17 FFs
REG4	20 FFs

Seleccione la configuración más adecuada para la memoria ROM. Se proporciona las posibles configuraciones para una FPGA Cyclone IV.

Table 3-1. Summary of M9K Memory Features

Feature	M9K Blocks
Configurations (depth x width)	8192 x 1
	4096 x 2
	2048 x 4
	1024 x 8
	1024 x 9
	512 x 16
	512 x 18
	256 x 32
	256 x 36

Figura 2. Configuraciones memoria embebida M9K Cyclone IV

La configuración más adecuada es 256x32.