

Máster Universitario en Ingeniería de Sistemas Electrónicos



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA

Práctica E4.1: Filtro FIR **Compensador del CIC**

Jose Luis, Rocabado Rocha

Gianmarco Leopoldo, Sangoi Da Roza

04/28/2022

ÍNDICE

ÍNDICE	2
ÍNDICE DE FIGURAS.....	3
Descripción del módulo	4
Interfaz	6
Recursos Hardware	8
Frecuencia Máxima.....	11
Camino Critico	11
Verificación.....	12
Simulaciones con MATLAB y SIMULINK	12
Testbenchs	14
Resolución problemas encontrados	17

ÍNDICE DE FIGURAS

<i>Ilustración 1. Respuesta en frecuencia del filtro CIC de tres etapas implementado en la práctica anterior.</i>	<i>4</i>
<i>Ilustración 2. Respuesta en frecuencia del filtro compensador de 17 coeficientes.....</i>	<i>5</i>
<i>Ilustración 3. Recursos del filtro compensador (TOP)</i>	<i>8</i>
<i>Ilustración 4. Recursos del módulo de control</i>	<i>9</i>
<i>Ilustración 5. Recursos del módulo de memoria ROM</i>	<i>9</i>
<i>Ilustración 6. Recursos del módulo de celda básica (multiplicador + acumulador)</i>	<i>10</i>
<i>Ilustración 7. Recursos del módulo de desplazamiento y multiplexor.</i>	<i>10</i>
<i>Ilustración 8. Frecuencia máxima de operación del sistema (Fmax).</i>	<i>11</i>
<i>Ilustración 9. Camino crítico del sistema generado por el Technology Map Viewer.</i>	<i>11</i>
<i>Ilustración 10. Zoom de la simulación de Simulink que compara el modelo ideal con el modelo cuantificado con precisión completa.....</i>	<i>12</i>
<i>Ilustración 11. Arriba: Plot de los coeficientes del filtro cuantificados. Abajo: Resultado de la respuesta ante el impulso en</i>	<i>12</i>
<i>Ilustración 12. Resultados de la señal filtrada para una señal senoidal de frecuencia 1kHz, 10kHz y 15 kHz.</i>	<i>13</i>
<i>Ilustración 13. Verificación de la memoria ROM.</i>	<i>14</i>
<i>Ilustración 14. Verificación del módulo de control</i>	<i>15</i>
<i>Ilustración 15. Verificación del multiplicador + acumulador</i>	<i>15</i>
<i>Ilustración 16. Verificación del registro de desplazamiento y el multiplexor de salida</i>	<i>16</i>
<i>Ilustración 17. Respuesta al impulso de nuestro filtro (TOP).....</i>	<i>16</i>
<i>Ilustración 18. Verificación de nuestro filtro compensador (entrada: sinusoidal 10KHz)</i>	<i>17</i>

Descripción del módulo

Cuando implementamos un filtro CIC interpolador, como el realizado en la práctica 3 para nuestro sistema de modulador AM/FM normalmente queremos un paso de banda plano y estrecho en la región de transición, estas son cualidades deseadas que sin embargo no son propias de un filtro CIC debido a la caída prolongada en la banda de paso, es por esto por lo que se implementa un filtro compensador que en conjunto con el filtro CIC nos permita obtener una banda de paso con ganancia constante.

Si observamos la respuesta en frecuencia del filtro CIC y del filtro compensador (figura 1 y figura 2 respectivamente) podemos comprobar que obtenemos el efecto deseado.

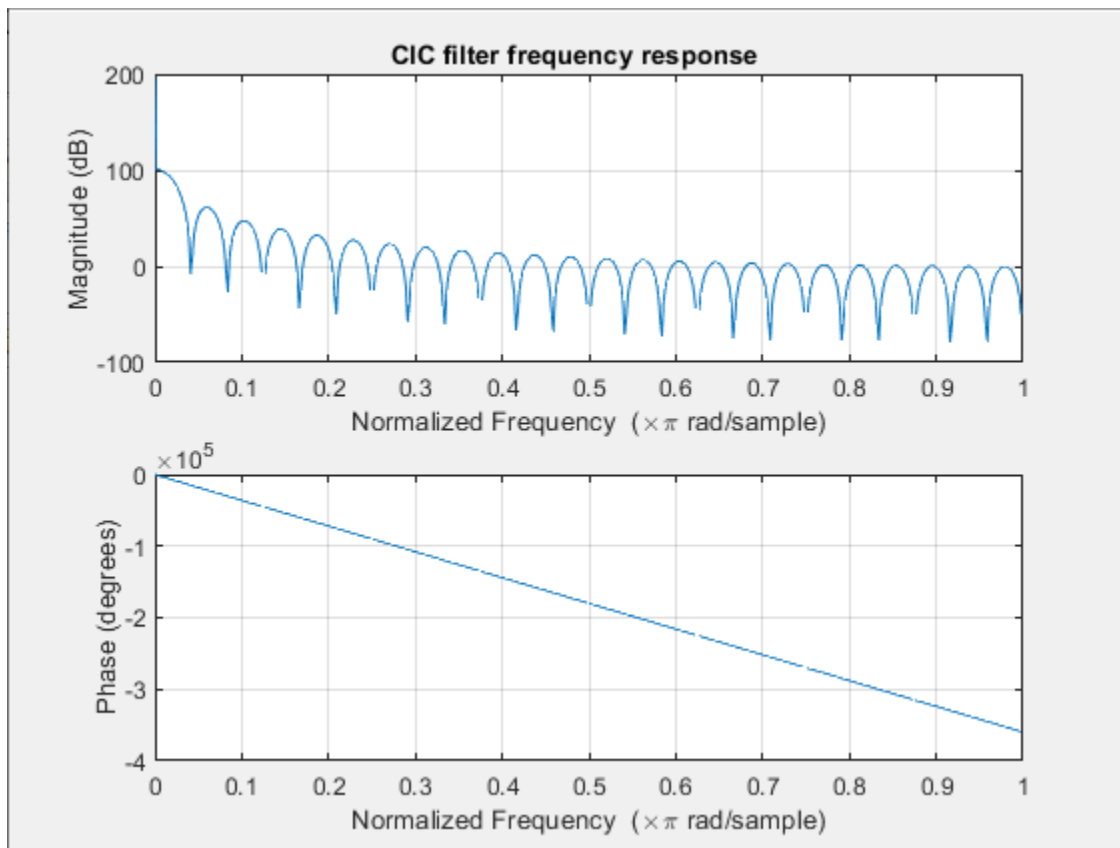


Ilustración 1. Respuesta en frecuencia del filtro CIC de tres etapas implementado en la práctica anterior.

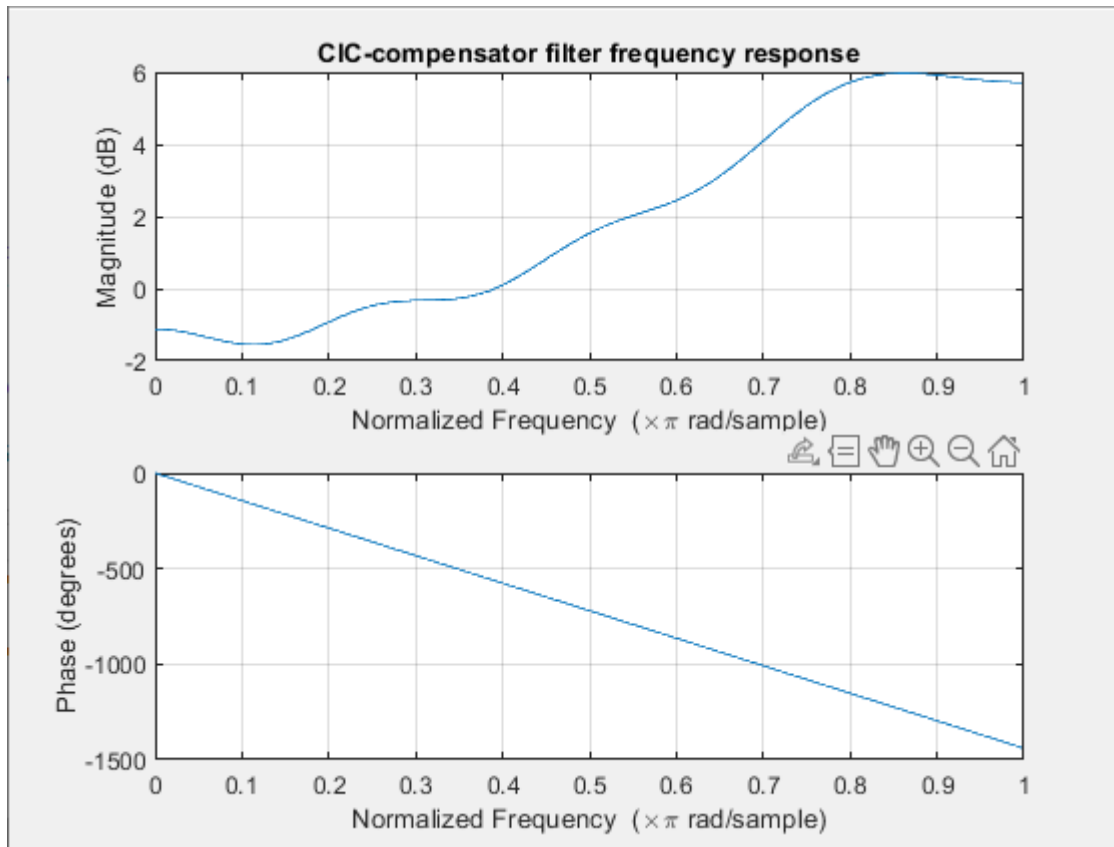


Ilustración 2. Respuesta en frecuencia del filtro compensador de 17 coeficientes.

Es necesario remarcar que Matlab tiene una normalización [0,1] donde 0.5 correspondería con la mitad de la frecuencia de muestreo. Por otro lado, se ha de tener en cuenta que el filtro CIC introduce un *upsampling* cambiando la frecuencia de muestreo. Por esta razón, si analizamos las dos respuestas en frecuencia respecto a frecuencias analógicas podremos comprobar como el filtro compensador afectará únicamente a la banda pasante del filtro CIC permitiéndonos obtener el resultado deseado.

En esta práctica se realizará un módulo de filtro FIR secuencial compensador para el filtro CIC realizado con anterioridad. La frecuencia de muestreo de los datos de entrada será de 50kHz, mientras que la frecuencia de reloj del filtro será de 100MHz.

El filtro se constituirá de 4 bloques:

- 1) Un registro de desplazamiento y multiplexor para la salida (REG_MUX).
- 2) Un multiplicador más acumulador (celda básica en filtros secuenciales) (MULT_ACC).
- 3) Una ROM (ROM).
- 4) Una máquina de estados para direccionar la memoria ROM, la celda del multiplicador acumulador y el registro de salida (CONTROL).

Interfaz

- Módulo SEC_FILTER (top):

PARÁMETROS			
Nombre	Descripción		
Win	Cuantificación de la entrada y salida		
Wc	Cuantificación de los coeficientes		
Num_coef	Número de coeficientes		
INTERFAZ			
Nombre	Tipo	Formato	Descripción
clk	in	bit	Entrada de reloj
rst	in	bit	Reset síncrono
val_in	in	bit	Entrada de validación de datos
din	in	S[Win, Win-1]	Entrada de datos filtro CIC
dout	out	S[19, 16]	Salida de datos truncada a 19 bits.
val_out	out	bit	Salida de la señal de validación

- Módulo CONTROL:

PARÁMETROS			
Nombre	Descripción		
Num_coef	Número de coeficientes		
INTERFAZ			
Nombre	Tipo	Formato	Descripción
clk	in	bit	Entrada de reloj
rst	in	bit	Reset síncrono
val_in	in	bit	Entrada de validación de datos
addr	out	U[log2(Num_coef), 0]	Salida de direccionamiento
rst_Acc	out	bit	Reset del acumulador
ce_Acc	out	bit	Salida de la señal de habilitación del acumulador
val_out	out	bit	Salida de la señal de validación

- Módulo REG_MUX (top):

PARÁMETROS	
Nombre	Descripción
Win	Cuantificación de la entrada y salida
Num_coef	Número de coeficientes

INTERFAZ			
Nombre	Tipo	Formato	Descripción
clk	in	bit	Entrada de reloj
ce	in	bit	Entrada de habilitación de datos
sel	in	$U[\log_2(\text{Num_coef}), 0]$	Selector del mux
din	in	$S[\text{Win}, \text{Win}-1]$	Entrada
dout	out	$S[\text{Win}, \text{Win}-1]$	Salida

- Módulo MULT_ACC:

PARÁMETROS			
Nombre	Descripción		
Win	Cuantificación de la entrada y salida		
Wc	Cuantificación de los coeficientes		
INTERFAZ			
Nombre	Tipo	Formato	Descripción
clk	in	bit	Entrada de reloj
rst	in	bit	Reset síncrono del acumulador
ce	in	bit	Habilitador del acumulador
din	in	S[Win, Win-1]	Entrada del acumulador
coef	in	S[Wc, Wc-1]	Coeficiente del acumulador
dout	out	S[Win+Wc, Win+Wc-3]	Salida filtrada con 3 bits enteros para evitar desbordamiento

- Módulo ROM:

PARÁMETROS				
Nombre		Descripción		
Wc		Cuantificación de los coeficientes		
Num_coef		Número de coeficientes		
INTERFAZ				
Nombre		Tipo	Formato	Descripción
clk		in	bit	Entrada de reloj
addr		in	$U[\log_2(\text{Num_coef}), 0]$	Direccionamiento de la memoria
data		out	$S[Wc, Wc-1]$	Salida de la ROM

Recursos Hardware

- Recursos de nuestro sistema SEC_FILTER:

	Resource	Usage
1	▼ Total logic elements	496 / 114,480 (< 1 %)
1	-- Combinational with no register	126
2	-- Register only	175
3	-- Combinational with a register	195
2		
3	▼ Logic element usage by number of LUT inputs	
1	-- 4 input functions	181
2	-- 3 input functions	84
3	-- <=2 input functions	56
4	-- Register only	175
4		
5	▼ Logic elements by mode	
1	-- normal mode	284
2	-- arithmetic mode	37
6		
7	▼ Total registers*	370 / 117,053 (< 1 %)
1	-- Dedicated logic registers	370 / 114,480 (< 1 %)
2	-- I/O registers	0 / 2,573 (0 %)
8		
9	Total LABs: partially or completely used	37 / 7,155 (< 1 %)
10	Virtual pins	0
11	▼ I/O pins	39 / 529 (7 %)
1	-- Clock pins	1 / 7 (14 %)
2	-- Dedicated input pins	0 / 9 (0 %)
12		
13	M9Ks	0 / 432 (0 %)

Ilustración 3. Recursos del filtro compensador (TOP)

La estimación de recursos para esta práctica es un tanto complejo, por eso, para calcular los recursos utilizados es necesario tener en cuenta los bits utilizados en cada modulo del sistema y sus registros:

- En el módulo de control se estiman 13 registros correspondientes al contador de direccionamiento, 4 de las variables de los estados y 4 de las señales de control. Un sumador de 5 bits del contador.
- La memoria ROM se quería implementar mediante una memoria M9K, sin embargo, tal y como se verá en el apartado de problemas encontrados, se ha tenido que implementar mediante registros para poder cumplir con las especificaciones de frecuencia. Por lo tanto, se añaden 18 registros correspondientes a los bits de los coeficientes más la lógica del selector.
- El registro de desplazamiento se tienen 16 bits de entrada registrados 17 que nos dan 272 registros, además se registra la salida del multiplexor por lo que se estiman 288 registros.

- En el módulo de multiplicación y acumulador se estema un multiplicador de 16x16 dedicado, un sumador de 34 bits y 2 registros de 34 bits (68) debidos al registro del acumulador y el registro de segmentación.
- Finalmente, en SEC_FILTER un registro de 19 bits para la salida más el registro del val_out.
- Recursos del módulo de CONTROL:

	Resource	Usage
1	▼ Total logic elements	14 / 114,480 (< 1 %)
1	-- Combinational with no register	2
2	-- Register only	3
3	-- Combinational with a register	9
2		
3	▼ Logic element usage by number of LUT inputs	
1	-- 4 input functions	2
2	-- 3 input functions	1
3	-- <=2 input functions	8
4	-- Register only	3
4		
5	▼ Logic elements by mode	
1	-- normal mode	7
2	-- arithmetic mode	4
6		
7	▼ Total registers*	12 / 117,053 (< 1 %)
1	-- Dedicated logic registers	12 / 114,480 (< 1 %)
2	-- I/O registers	0 / 2,573 (0 %)
8		
9	Total LABs: partially or completely used	1 / 7,155 (< 1 %)
10	Virtual pins	0
11	▼ I/O pins	12 / 529 (2 %)
1	-- Clock pins	1 / 7 (14 %)
2	-- Dedicated input pins	0 / 9 (0 %)
12		
13	M9Ks	0 / 432 (0 %)

Ilustración 4. Recursos del módulo de control

- Recursos del módulo ROM:

	Resource	Usage
1	▼ Total logic elements	23 / 114,480 (< 1 %)
1	-- Combinational with no register	5
2	-- Register only	3
3	-- Combinational with a register	15
2		
3	▼ Logic element usage by number of LUT inputs	
1	-- 4 input functions	17
2	-- 3 input functions	2
3	-- <=2 input functions	1
4	-- Register only	3
4		
5	▼ Logic elements by mode	
1	-- normal mode	20
2	-- arithmetic mode	0
6		
7	▼ Total registers*	18 / 117,053 (< 1 %)
1	-- Dedicated logic registers	18 / 114,480 (< 1 %)
2	-- I/O registers	0 / 2,573 (0 %)
8		
9	Total LABs: partially or completely used	3 / 7,155 (< 1 %)
10	Virtual pins	0
11	▼ I/O pins	24 / 529 (5 %)
1	-- Clock pins	1 / 7 (14 %)
2	-- Dedicated input pins	0 / 9 (0 %)
12		
13	M9Ks	0 / 432 (0 %)

Ilustración 5. Recursos del módulo de memoria ROM

Cabe remarcar que se decide implementar este módulo mediante una memoria M9K para poder reducir en todo lo posible la lógica del sistema.

- Recursos del módulo MULT_ACC:

	Resource	Usage
1	▼ Total logic elements	114 / 114,480 (< 1 %)
1	-- Combinational with no register	46
2	-- Register only	11
3	-- Combinational with a register	57
2		
3	▼ Logic element usage by number of LUT inputs	
1	-- 4 input functions	0
2	-- 3 input functions	67
3	-- <=2 input functions	36
4	-- Register only	11
4		
5	▼ Logic elements by mode	
1	-- normal mode	70
2	-- arithmetic mode	33
6		
7	▼ Total registers*	68 / 117,053 (< 1 %)
1	-- Dedicated logic registers	68 / 114,480 (< 1 %)
2	-- I/O registers	0 / 2,573 (0 %)
8		
9	Total LABs: partially or completely used	9 / 7,155 (< 1 %)
10	Virtual pins	0
11	▼ I/O pins	71 / 529 (13 %)
1	-- Clock pins	3 / 7 (43 %)
2	-- Dedicated input pins	0 / 9 (0 %)
12		
13	M9Ks	0 / 432 (0 %)

Ilustración 6. Recursos del módulo de celda básica (multiplicador + acumulador)

- Recursos del módulo REG_MUX:

	Resource	Usage
1	▼ Total logic elements	320 / 114,480 (< 1 %)
1	-- Combinational with no register	32
2	-- Register only	160
3	-- Combinational with a register	128
2		
3	▼ Logic element usage by number of LUT inputs	
1	-- 4 input functions	160
2	-- 3 input functions	0
3	-- <=2 input functions	0
4	-- Register only	160
4		
5	▼ Logic elements by mode	
1	-- normal mode	160
2	-- arithmetic mode	0
6		
7	▼ Total registers*	288 / 117,053 (< 1 %)
1	-- Dedicated logic registers	288 / 114,480 (< 1 %)
2	-- I/O registers	0 / 2,573 (0 %)
8		
9	Total LABs: partially or completely used	20 / 7,155 (< 1 %)
10	Virtual pins	0
11	▼ I/O pins	39 / 529 (7 %)
1	-- Clock pins	1 / 7 (14 %)
2	-- Dedicated input pins	0 / 9 (0 %)
12		
13	M9Ks	0 / 432 (0 %)

Ilustración 7. Recursos del módulo de desplazamiento y multiplexor.

Frecuencia Máxima

La placa utilizada en el laboratorio de prácticas consiste en una Cyclone IV DE-115 cuya frecuencia máxima es de 250MHz.

Slow 1200mV 85C Model				
	Fmax	Restricted Fmax	Clock Name	Note
1	151.45 MHz	151.45 MHz	clk	

Ilustración 8. Frecuencia máxima de operación del sistema (Fmax).

Mediante la herramienta *TimeQuest Timing Analyzer*, podremos calcular la frecuencia máxima a la que trabajará el modelo diseñado. Utilizando el *wrapper* (instanciación con registros en las entradas y salidas) creado, hemos sido capaces de obtener una frecuencia máxima de trabajo de ~151.45 MHz tal y como se observa en la figura 8.

Camino Critico

El camino crítico es aquel en el que la señal requiere un tiempo mayor desde su salida de un registro hasta su destino en otro registro. Mediante el *Technology Map Viewer* podremos encontrar en que parte del circuito se encuentra en nuestro diseño tal y como se observa en la siguiente figura.

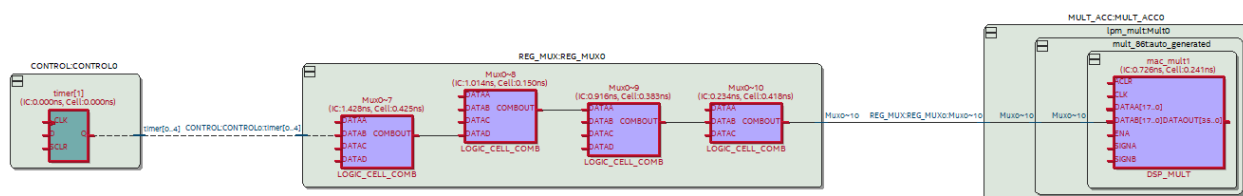


Ilustración 9. Camino crítico del sistema generado por el Technology Map Viewer.

El camino crítico de nuestro diseño del filtro secuencial se encuentra en la señal de direccionamiento del módulo de control que nos permite seleccionar la muestra deseada del registro de desplazamiento.

Verificación

Simulaciones con MATLAB y SIMULINK

Para el desarrollo y verificación del filtro, lo primero que se hizo fue entender las señales de entrada y salida de nuestro sistema. Esto se realizó simulando con un fichero en *MATLAB* y *Simulink* las entradas y salidas que debería tener nuestro filtro. Desde comparar el modelo ideal con el modelo cuantificado con precisión completa, a obtener la respuesta ante el impulso y realizar un barrido en frecuencia.

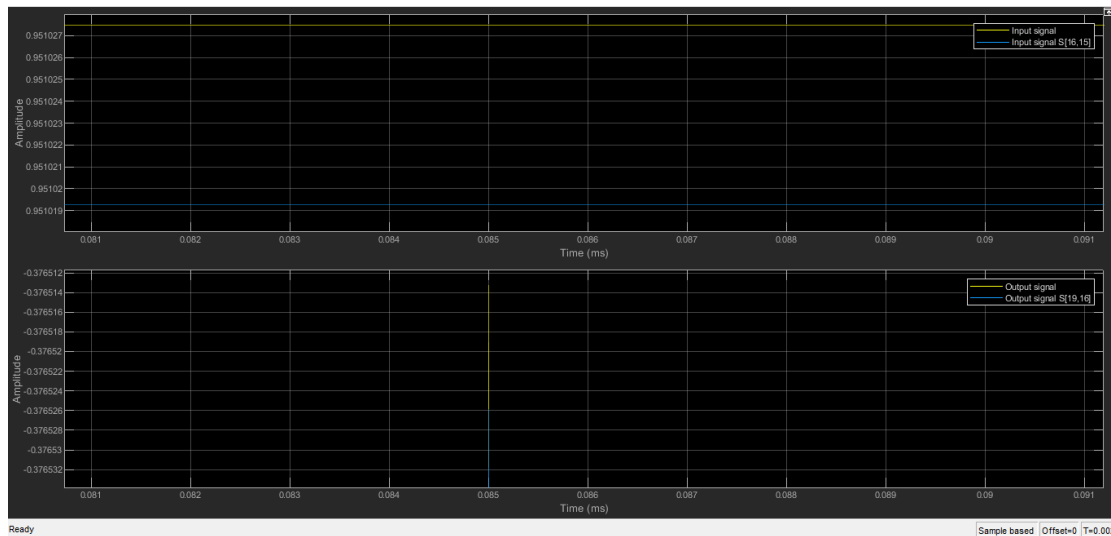


Ilustración 10. Zoom de la simulación de Simulink que compara el modelo ideal con el modelo cuantificado con precisión completa.

Cuando comparamos ambos modelos (ilustración 10), se puede observar que se obtiene un error prácticamente negligible debido a la cuantificación. En cuanto a la respuesta al impulso, se espera obtener por definición los coeficientes del filtro tal y como se observa en la siguiente figura.

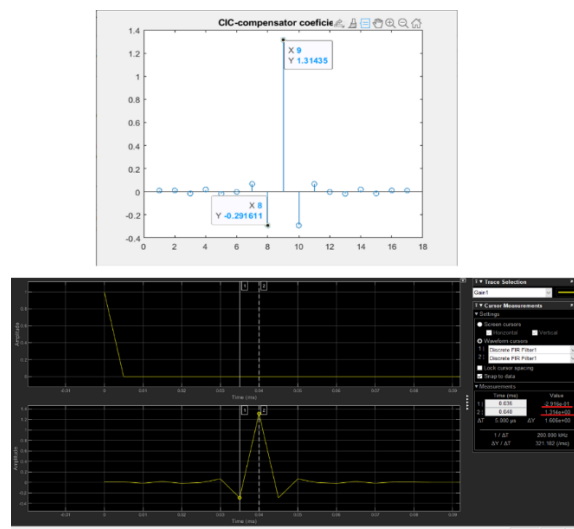


Ilustración 11. Arriba: Plot de los coeficientes del filtro cuantificados. Abajo: Resultado de la respuesta ante el impulso en

Por último, al observar la salida ante una señal senoidal de 1V de amplitud y diferentes frecuencias se pretende comprobar cómo, para frecuencias mayores de señal, realmente se amplifican para compensar el filtro CIC permitiéndonos obtener una banda pasante con una ganancia constante (figura 5).

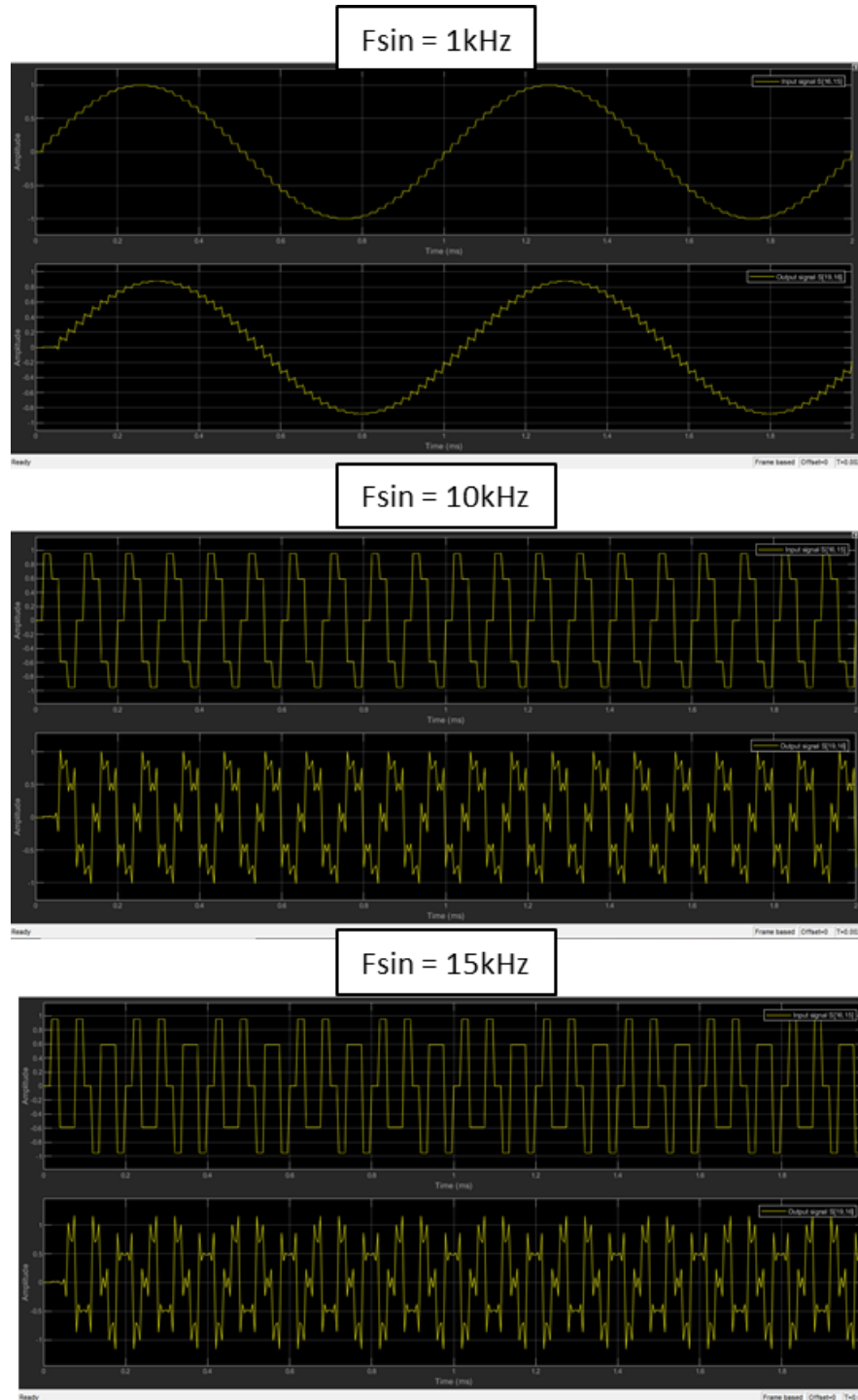


Ilustración 12. Resultados de la señal filtrada para una señal senoidal de frecuencia 1kHz, 10kHz y 15 kHz.

Testbenchs

Para la verificación de los módulos se generaron los *TestBench* correspondientes que de forma sencilla nos permiten validar la ejecución correcta de las operaciones. Desde el fichero en MATLAB elaborado podemos realizar las simulaciones con distintas frecuencias y ondas (sinusoidal, Chirp e Impulse). Principalmente para nuestras simulaciones empleamos una onda sinusoidal que iremos variando en frecuencias de 1KHz, 10KHz y 15KHz para ver el comportamiento de nuestro filtro al aumentar la frecuencia de señal.

- Testbench módulo de la memoria ROM:

Se comprobó que el direccionamiento de los coeficientes en nuestra memoria estaba bien realizado y los valores guardados en la ROM son correctos.

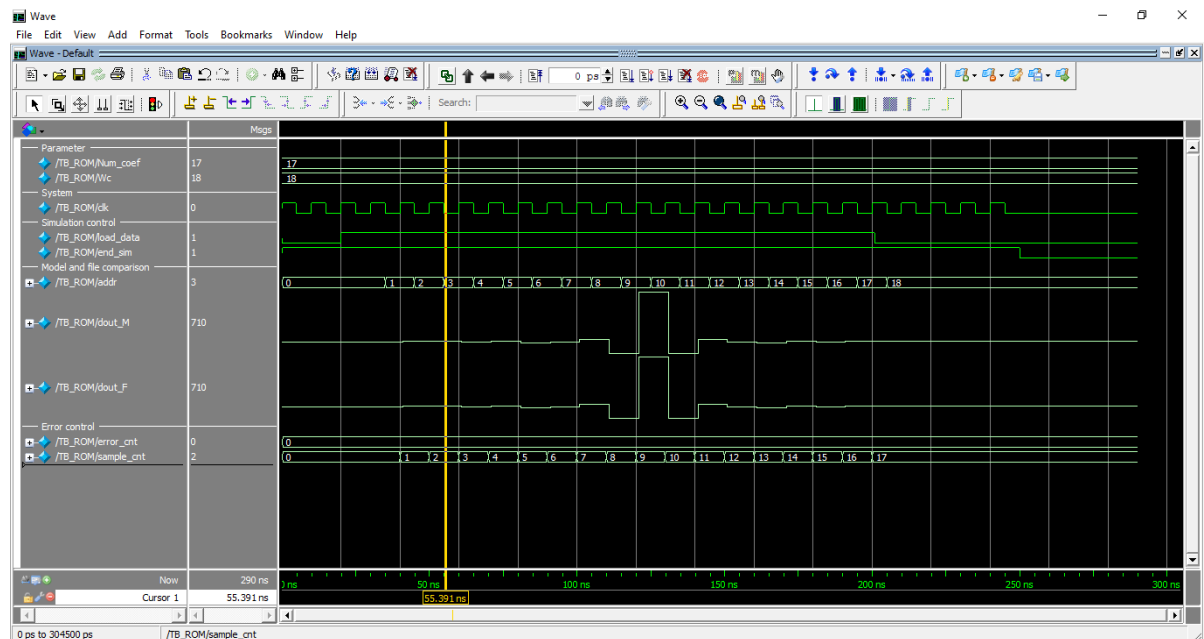


Ilustración 13. Verificación de la memoria ROM.

- Testbench módulo de Control:

En este módulo se comprobó el funcionamiento de nuestra máquina de estados.

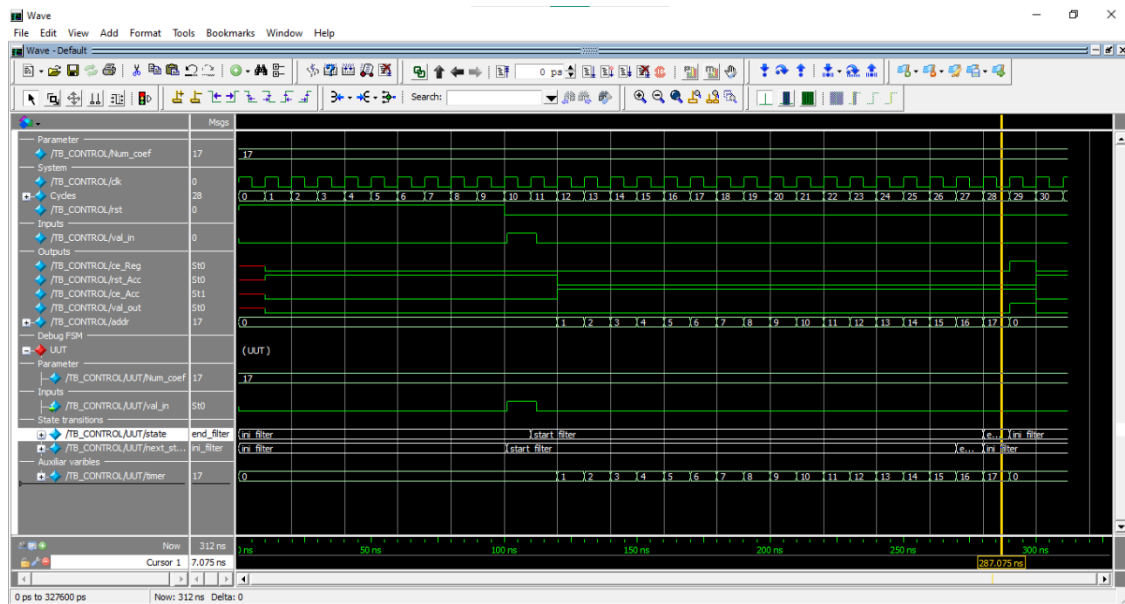


Ilustración 14. Verificación del módulo de control

- Testbench módulo de multiplicador más acumulador:

Esta es la celda básica de un filtro secuencial. Se verifico la respuesta de este módulo simulando las dos señales de control que se le inyectan. Una es la señal para realizar el reset del acumulador y la otra es para habilitar el acumulador y este aumente al recibir una muestra.

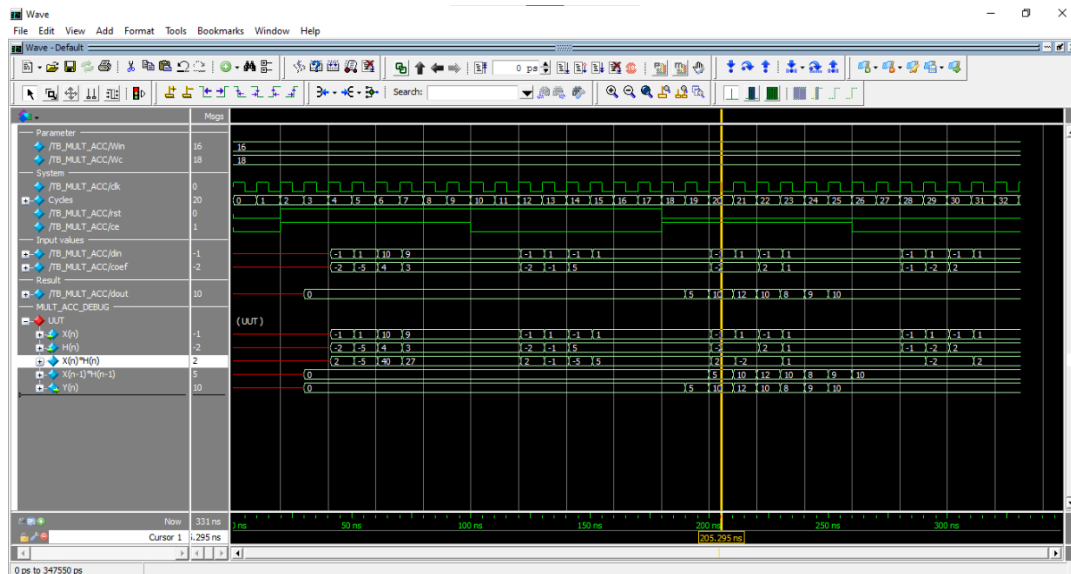


Ilustración 15. Verificación del multiplicador + acumulador

- Testbench módulo de desplazamiento y multiplexor:

En este módulo se verifico la carga del nuevo dato y que se pudiera direccionar el bus del registro de desplazamiento correctamente.

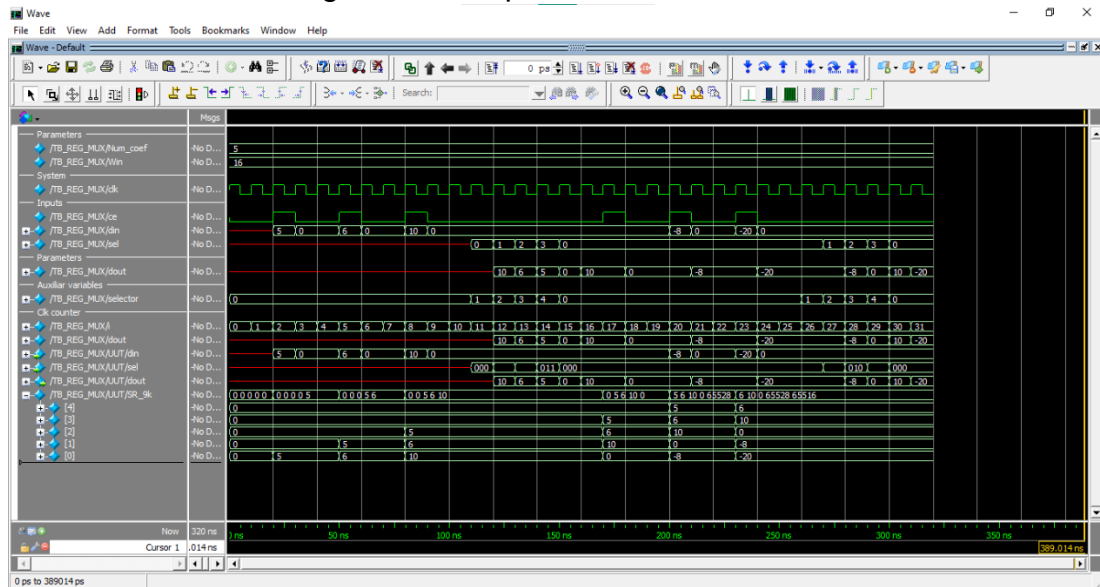


Ilustración 16. Verificación del registro de desplazamiento y el multiplexor de salida

- Testbench modulo filtro de compensación (TOP):

Por último, para la verificación final, se realizará una verificación a nivel de puertas para comprobar si el diseño podrá funcionar correctamente en la placa de desarrollo. Se simulará tanto la respuesta al impulso como señales senoidales o de tipo *chirp* de diferentes frecuencias. Se puede observar algunos ejemplos en las siguientes figuras.

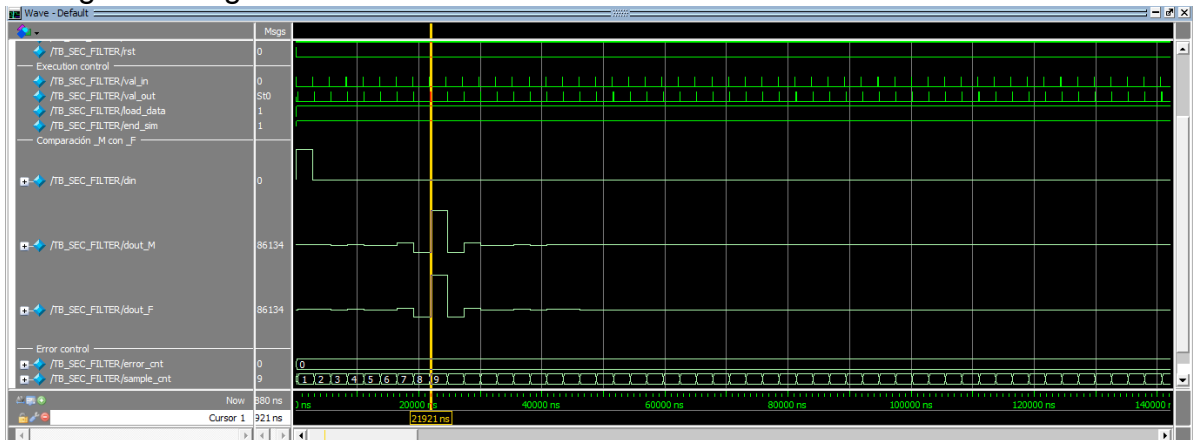


Ilustración 17. Respuesta al impulso de nuestro filtro (TOP)

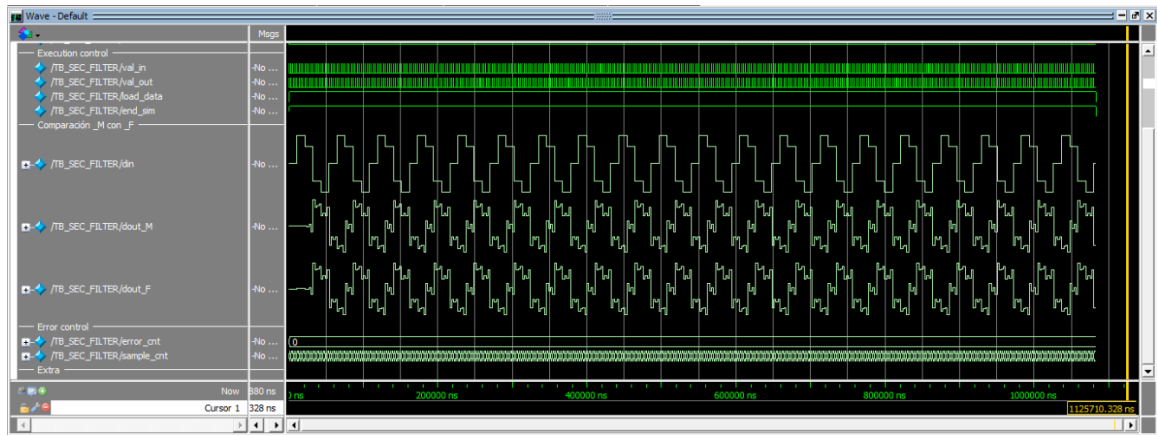


Ilustración 18. Verificación de nuestro filtro compensador (entrada: sinusoidal 10KHz)

Resolución problemas encontrados

En el desarrollo de nuestro modulo se tubo como principal problema el ajuste de las señales de control, estas ocasionaban que el módulo MULT_ACC no funcionase correctamente. Para solucionar esto simplemente se ajustaron las señales mediante la simulación.

Otro de los problemas que se encontró durante el desarrollo de la práctica fue durante la simulación *Gate Level* donde se comprobó que las señales no se capturaban correctamente. Tras un *debug* mediante *Modelsim* y *Time Quest Timing Analyzer* se encontró que era necesario realizar una segmentación en el módulo del multiplicador-acumulador.

Por último, se observó nuevamente mediante la herramienta de análisis temporal. Se observó que, si se realizaba una implementación de la memoria ROM que contiene los coeficientes del filtro en un bloque de memoria M9K, no era posible cumplir las especificaciones de frecuencia de una frecuencia máxima de 125MHz. Se configuró el compilador de *Quartus* para que realice un emplazamiento priorizando el rendimiento del sistema permitiéndonos aumentar la frecuencia máxima unos cuantos megahercios. Sin embargo, seguía sin cumplirse las especificaciones.

Finalmente, se comprobó que si se realizaba la implementación con registros en lugar de una memoria M9K era posible obtener la frecuencia que se observa en el apartado de Frecuencia Máxima.