



INAOE

Realización Eficiente de Conversión de Frecuencia de Muestreo por un Factor Racional Mediante el Uso de Filtros CIC

Por

Arcesio Arbeláez Nieto

Ingeniero Electrónico, Universidad del Quindío.
Presentado como requisito parcial para la obtención del
grado de:

**MAESTRO EN CIENCIAS CON ESPECIALIDAD EN
ELECTRÓNICA**

en el

Instituto Nacional de Astrofísica Óptica y Electrónica
Octubre 2012
Tonantzintla, Puebla

Asesor:

Dra. Gordana Jovanovic Dolecek
Investigador titular C
Departamento de Electrónica
INAOE

©INAOE 2012

El autor otorga al INAOE el permiso de reproducir copias
totales o parciales de esta tesis.



AGRADECIMIENTOS

Agradezco a la Dra. Gordana Jovanovic Dolecek por su paciencia y consejos que fueron de vital importancia para la realización de este trabajo.

Agradezco a mi familia, aquellos que siempre me han brindado su apoyo incondicional y el amor necesario para la culminación de esta etapa de mi vida, a mi tía Margarita Arbeláez, porque sin su apoyo todas estas metas hubiesen sido mucho más difíciles de lograr. A la familia Cortés Idárraga por su apoyo, cariño y motivación para luchar por salir adelante.

Agradezco a la familia Álvarez Fernández por la especial acogida que han tenido conmigo, por abrirme las puertas de su casa como si fuera parte de su familia. Especial agradecimiento a Alejandra Álvarez Fernández por su amor incondicional y por darme luz en los días más oscuros.

Gracias al Instituto Nacional de Astrifísica, Óptica y Electrónica porque han demostrado que tienen, además de calidad académica, la calidad humana que definitivamente proporciona un entorno acogedor y pacífico para crecer en el ámbito académico y personal.

Agradezco al CONACyT por el apoyo económico recibido de su parte para la realización de mis estudios de maestría, gracias por la oportunidad que me han brindado a mi y a mis compatriotas para ver nuevos horizontes en una tierra que nos ha recibido de una inmejorable manera.

RESUMEN

Este trabajo presenta una nueva estructura para la realización eficiente de filtros para la conversión de frecuencia de muestreo por un factor racional.

En primer lugar se muestran algunos conceptos fundamentales de los sistemas multi-razón, que son aquellos que requieren diferentes frecuencias de muestreo para su funcionamiento. Estos conceptos son claves para el análisis de las estructuras propuestas en este trabajo.

Posteriormente, se hace una revisión de los filtros usados a lo largo del documento, incluyendo una corta descripción de los mismos y ejemplos que permiten la comprensión de su funcionamiento.

En tercer lugar se presentan algunos de los métodos que permiten una conversión por un factor racional de forma eficiente. Estos se basan principalmente en la descomposición polifase y permiten realizar las operaciones necesarias en una frecuencia de muestreo reducida, generando un ahorro en el consumo de potencia.

Después se presenta una revisión de varios métodos encontrados en la literatura para la realización de los filtros usados para conversión racional. Algunos de ellos se usan para sistemas que soportan múltiples estándares. Sin embargo, este trabajo se ocupa de aquellos diseñados para valores de conversión específicos, que se usan para tareas particulares buscando la mayor eficiencia.

Además se presentan algunas condiciones y consideraciones que permitirán la definición de una propuesta para la conversión racional de frecuencias de muestreo.

Luego se presenta la propuesta para el diseño de los filtros para conversión racional. Este método resulta de la combinación de dos métodos existentes en la literatura. Adicionalmente, se presenta una modificación de la estructura propuesta que permite la realización eficiente de la estructura. Finalmente, se presentan las respectivas implementaciones para comprobar que efectivamente, estos filtros son realizables. Además, se incluyen sus respuestas al impulso tanto en *Simulink* como en *ModelSim*.

ABSTRACT

This work presents a novel structure which improves the efficiency in filters for sampling rate conversion by a rational factor.

Firstly, the fundamental concepts of multi-rate systems are presented. This type of systems require different sampling rates, and the related concepts are very important for the analysis of the structures proposed in this work.

Also, the filters used in this document are described.

Some methods to realize an efficient sample rate conversion are also presented. These methods are mainly based in polyphase decomposition, which allows to implement the required operations in reduced sample rate achieving a power saving.

Later, a review of several methods found in the literature is shown. Some of them are used for systems with multi-standard support. However, this work is concerned to those designed for individual conversion values, which are used for specific tasks to reach maximum efficiency.

Subsequently, the proposed method for rational sample rate conversion is presented. This method is the result of two methods found in the literature. The characteristics, constraints and conditions for their use are widely described. Additionally, a modification for these methods is presented allowing an efficient realization.

Finally, two examples are shown in order to verify that the implementation is possible. The simulations were realized in *Simulink* and *ModelSim*.

PREFACIO

El filtro CIC (Cascada Integrador-Comb) es usado en tareas de conversión de frecuencia de muestreo, debido al hecho de poseer ceros en los dobles de la banda (folding-bands), que es justamente donde se presenta el *aliasing*, y por su baja complejidad computacional.

Este filtro puede ser implementado sin multiplicadores y por lo tanto puede operar a altas frecuencias. La estructura CIC en forma recursiva necesita menos sumadores y por lo tanto permite un ahorro en el área de chip necesaria para la implementación de estos filtros. Por otra parte, la implementación de la función de transferencia equivalente de estos filtros en forma no-recursiva favorece el empleo de descomposición polifase y, como consecuencia, es posible obtener un menor consumo de potencia.

Dos características importantes en un sistema de conversión de frecuencia de muestreo, sobre las cuales se trabaja en esta tesis, son la Relación Señal a Ruido (o Signal to Noise Ratio, SNR, en inglés) y la cantidad de operaciones por muestra de salida. El objetivo de esta tesis es obtener, con base en filtros CIC, una estructura eficiente de realización de filtros para la conversión de frecuencia de muestreo por un factor racional, que permita la mejora de la SNR y que a su vez tenga un mínimo incremento en la cantidad de operaciones por muestra de salida. Debido a que la estructura no tiene multiplicadores, la cantidad de operaciones es medida en número de sumas por muestra de salida o Additions Per Output Sample (APOS) en inglés.

Para lograr este objetivo primero se realizó un estudio detallado de la teoría sobre los sistemas multi-razón, lo cual es presentado en el Capítulo 1. Luego se realizó un amplio estudio sobre ciertos filtros que aparecen recurrentemente en este campo. Esto se presenta en el Capítulo 2. Posteriormente, se revisaron con detenimiento las estructuras propuestas en la literatura con respecto a la conversión racional de frecuencias de muestreo y lo concerniente a su realización eficiente. Esto es presentado en los Capítulos 3 y 4. El marco de referencia provisto por el estudio realizado con anterioridad provee la capacidad de definir las condiciones y consideraciones requeridas para elaborar una propuesta. Estas condiciones se presentan en el Capítulo 5.

En el Capítulo 6 se describe en detalle la estructura propuesta y se muestran cada uno de los pasos necesarios para obtenerla, se ejemplifica y se compara con algunos otros métodos

encontrados en la literatura. Adicionalmente, se presenta la implementación en *Simulink* y en VHDL de dos ejemplos. Finalmente, se presentan las conclusiones y el posible trabajo futuro relacionado con esta tesis.

Índice general

Agradecimientos	III
Resumen	IV
Abstract	V
Prefacio	VI
Índice de Figuras	XVI
Índice de Tablas	XVII
Acrónimos	XVIII
1. Introducción	1
1.1. Conversión de la Frecuencia de Muestreo	1
1.1.1. Decimación	1
1.1.1.1. Submuestreo	2
1.1.1.2. Filtrado pasa-bajas <i>anti-aliasing</i>	5
1.1.2. Interpolación	6
1.1.2.1. Sobremuestreo	6
1.1.2.2. Filtrado pasa-bajas <i>anti-imaging</i>	8
1.1.3. Conversión por un factor racional	8
1.2. Identidades, descomposición polifase e intercambio de conversores de frecuencia de muestreo	10
1.2.1. Identidades nobles	10
1.2.1.1. Primera identidad	10
1.2.1.2. Segunda identidad	10
1.2.1.3. Tercera identidad	11
1.2.1.4. Cuarta identidad	11
1.2.1.5. Quinta identidad	12
1.2.1.6. Sexta identidad	12
1.2.2. Decimación e interpolación polifase	13
1.2.2.1. Descomposición polifase	13

1.2.2.2.	Decimación polifase	15
1.2.2.3.	Interpolación polifase	16
1.2.2.4.	Intercambio de conversores de frecuencia de muestreo	17
2.	Filtros usados en este trabajo	19
2.1.	Filtro CIC	19
2.1.1.	Ventajas y desventajas del filtro CIC	22
2.2.	Filtro Coseno	23
2.3.	Filtro CIC triangular escalonado	24
2.4.	Filtro de media banda	31
2.5.	Interpolador lineal	33
2.6.	Interpolador Farrow	34
3.	Métodos para la realización eficiente de descomposición polifase	38
3.1.	Realización polifase	38
3.2.	Realización eficiente de conversión racional de frecuencia de muestreo	39
3.3.	Implementación de filtros FIR de fase lineal para conversión racional de frecuencias de muestreo usando la simetría de los coeficientes	42
3.3.1.	Propiedades de la matriz de entrada-salida $\mathbf{H}_{L \times (p+q+1)}$	47
3.3.2.	Caso A	49
3.3.3.	Caso B	51
4.	Revisión métodos para RSRC	54
4.1.	Métodos basados en el filtro CIC	54
4.1.1.	Filtro CIC modificado para conversión de frecuencias de muestreo para sistemas de Radio definido por Software	55
4.1.2.	Filtro CIC Triangular Escalonado para conversión de frecuencia de muestreo racional	56
4.1.3.	Arquitectura CIC intercalada de bajo consumo de potencia para receptores de Radio Definido por Software	58
4.2.	Métodos basados en filtros CIC e interpoladores	61
4.2.1.	Conversión eficiente de frecuencias de muestreo para Radio Definido por Software Multi-Estándar	61
4.2.2.	Decimación por un factor irracional usando filtros CIC e interpolador lineal	64
4.2.3.	Estructura eficiente en consumo de potencia para conversión entre frecuencias de muestreo arbitrarias	67

4.3. Métodos basados en filtros CIC y filtros auxiliares	73
4.3.1. Inclusión de un filtro IIR y modificación de la ubicación de los ceros del filtro CIC	73
4.3.2. Implementación Eficiente de Conversores de Frecuencia de Muestreo Multi-etapa	75
5. Consideraciones y condiciones para el método propuesto	78
5.1. Selección de los retardos para el filtro CIC modificado	78
5.2. Variación de la mejora del filtro CIC triangular escalonado con respecto al factor de decimación	81
5.3. Análisis de la atenuación de los subfiltros del método del filtro CIC modificado	84
5.4. Mejora de la SNR usando el filtro coseno expandido	86
6. Método propuesto	87
6.1. Descripción del método	87
6.2. Estructura	91
6.3. Estructura polifase	92
6.4. Comparación	98
6.5. Implementación	102
6.5.1. Implementación en <i>Simulink</i>	103
6.5.2. Implementación en VHDL	107
6.5.2.1. Generación de los coeficientes de la estructura eficiente y tamaños de palabra requeridas en cada etapa	109
6.5.2.2. Generación de las señales de reloj	111
6.5.2.3. Generación de los coeficientes de la estructura eficiente y tamaños de palabra requeridas en cada etapa	113
6.5.2.4. Generación de las señales de reloj	115
7. Conclusiones y trabajo futuro	116
Bibliografía	118

Índice de Figuras

1.1. Diagrama de bloques de la operación de decimación.	2
1.2. Representación de la operación de submuestreo.	2
1.3. Ilustración del proceso de submuestreo en el tiempo, ($M = 2$).	3
1.4. Ilustración del proceso de submuestreo en el dominio de la frecuencia.	5
1.5. Ilustración del efecto del <i>aliasing</i> en una señal debido al submuestreo.	6
1.6. Estructura en diagrama de bloques para la interpolación.	6
1.7. Ilustración del proceso de sobremuestreo en el tiempo.	7
1.8. Ilustración del proceso de sobremuestreo en el dominio de la frecuencia ($L = 4$).	9
1.9. Cascada decimación-interpolación.	9
1.10. Estructura para conversión racional de frecuencia de muestreo	10
1.11. Estructuras equivalentes para la primera identidad.	11
1.12. Estructuras equivalentes para la segunda identidad.	11
1.13. Estructuras equivalentes para la tercera identidad.	11
1.14. Estructuras equivalentes para la cuarta identidad.	12
1.15. Estructuras equivalentes para la quinta identidad.	12
1.16. Estructuras equivalentes para la sexta identidad.	12
1.17. Implementación de la estructura polifase.	14
1.18. Decimación con filtro polifase.	15
1.19. Decimación eficiente con filtro polifase.	16
1.20. Interpolación con filtro polifase.	16
1.21. Interpolación eficiente con filtro polifase transpuesto.	17
1.22. Cascada de sobremuestreador y submuestreador.	17
1.23. Cascada de submuestreador y sobremuestreador.	18
2.1. Estructura del filtro CIC recursivo.	20
2.2. Estructura recursiva.	20
2.3. Propiedades en magnitud del filtro CIC como filtro <i>anti-aliasing</i>	21

2.4. Respuestas en magnitud para el filtro CIC con $K = 1$ y sus versiones con factores de cascada $K = 2$ y $K = 3$	22
2.5. Filtro coseno y filtro coseno expandido	24
2.6. Comparación de las respuestas en magnitud para diferentes aproximaciones a respuestas escalonadas de un filtro pasa-bajas.	27
2.7. Respuestas en magnitud del filtro CIC Triangular Escalonado con diferentes selecciones de los parámetros.	28
2.8. Ubicaciones de los ceros para $M_1 = 7$ y $M_2 = 3$	30
2.9. Ubicaciones de los ceros para $M_1 = 3$ y $M_2 = 7$	30
2.10. Características de la respuesta en magnitud del filtro de media banda.	32
2.11. Filtro de media banda.	32
2.12. Características de la respuesta en magnitud del interpolador lineal con $L = 4$	34
2.13. Diagrama de bloques para el filtro de interpolación.	34
2.14. Modelo analógico de un filtro de interpolación.	35
2.15. Estructura Farrow.	36
2.16. Interpolador de Lagrange usando la estructura Farrow.	37
2.17. Respuesta en magnitud del interpolador basado en la estructura Farrow.	37
3.1. Conversión racional usando decimación polifase.	39
3.2. Conversión racional usando interpolación polifase.	39
3.3. Descomposición polifase del filtro <i>anti-aliasing/anti-imaging</i>	40
3.4. Primer paso para la derivación de la estructura eficiente de [3].	40
3.5. Segundo paso para la derivación de la estructura eficiente de [3].	41
3.6. Desarrollo de la $k^{\text{ésima}}$ rama.	42
3.7. Cuarto paso para la derivación de la estructura eficiente de [3].	42
3.8. Estructura final.	43
3.9. Ejemplo de la estructura descrita en [3] para $L = 2$ y $M = 3$	43
4.1. Espectro de magnitud para el ejemplo del filtro CIC modificado.	56
4.2. Estructura general para el filtro CIC triangular escalonado.	58
4.3. Espectro de magnitud para el ejemplo del filtro CIC triangular escalonado.	59
4.4. Estructura general para el filtro propuesto en [16].	60
4.5. Estructura para conversión de frecuencia de muestreo arbitraria propuesta en [17].	62
4.6. Estructura propuesta en [17].	62
4.7. Estructura del Interpolador Farrow.	63

4.8. Filtro CIC $M = 16$, $K = 4$	65
4.9. Respuesta deseada y respuesta aproximada para el interpolador.	65
4.10. Estructura general para el filtro propuesto en [18].	66
4.11. Ejemplo de conversión de frecuencia de muestreo por un factor $R = 3.25$	68
4.12. Respuesta en magnitud para conversión de frecuencia de muestreo por un factor $R = 10.1$	68
4.13. Estructura Farrow a) Modificada b) Modificada transpuesta.	70
4.14. Estructura para el caso I.	72
4.15. Estructura para el caso II.	72
4.16. Respuesta en magnitud del ejemplo para $R = 10\frac{4}{11}$	73
4.17. Estructura usada para la modificación de los ceros del filtro CIC.	74
4.18. Respuesta en magnitud para el ejemplo de diseño usando el método [14].	76
4.19. Decimación.	76
4.20. Interpolación.	77
4.21. Respuesta en magnitud para el ejemplo del interpolador multi-etapa de [13].	77
5.1. Respuestas en magnitud para algunas selecciones de los retardos para el filtro CIC modificado de orden 4 con $M = 10$	79
5.2. Respuestas en magnitud para ilustrar la selección de los retardos D_i	82
5.3. Variación de la mejora obtenida con el factor de decimación.	83
5.4. Variación de la mejora obtenida con el factor de decimación.	83
5.5. Comparación de algunos subfiltros para el método CIC modificado con $M = 20$ y $K = 5$	85
5.6. Ejemplo del filtro Coseno Expandido para $\omega_1 = 0.125$	86
6.1. Comparación de la ubicación para el primer cero para el filtro coseno ex- pandido con $R = 8$ y el subfiltro con $D_4 = 16$	90
6.2. Comparación de la ubicación para el primer cero para el filtro coseno ex- pandido con $R = 16$ y el subfiltro con $D_4 = 30$, zoom de los primeros ceros de cada filtro.	91
6.3. Estructura general del método propuesto.	93
6.4. Estructura modificada para realización polifase.	94
6.5. Estructura para el ejemplo 6.6.	95
6.6. Estructura eficiente para el ejemplo 6.6.	96
6.7. Estructura para el ejemplo 6.7.	97
6.8. Estructura eficiente para el ejemplo 6.7.	99
6.9. Comparacion de las respuestas en magnitud para el ejemplo 6.8.	100

6.10. Estructura resultante para el ejemplo 6.8.	100
6.11. Respuesta en magnitud para la segunda parte del ejemplo 6.8.	101
6.12. Comparacion de las respuestas en magnitud para el ejemplo 6.9.	101
6.13. Estructura resultante para el ejemplo 6.9.	102
6.14. Estructura del ejemplo 6.10.	102
6.15. Estructura del ejemplo 6.11.	103
6.16. Estructuras equivalentes para el ejemplo 6.10.	104
6.17. Respuesta al impulso para los esquemas equivalentes de la Fig. 6.16. . . .	105
6.18. Esquema en bloques de <i>Simulink</i> para el ejemplo 6.10.	105
6.19. Esquema en bloques de <i>Simulink</i> para el ejemplo 6.10 usando la estructura eficiente.	105
6.20. Respuesta al impulso para los esquemas equivalentes de las Figuras 6.18 y 6.19.	106
6.21. Estructuras equivalentes para el ejemplo 6.11.	106
6.22. Respuesta al impulso para los esquemas equivalentes de las Figuras 6.21. .	107
6.23. Esquema en bloques de <i>Simulink</i> para el ejemplo 6.11.	107
6.24. Esquema en bloques de <i>Simulink</i> para el ejemplo 6.11 usando la estructura eficiente.	108
6.25. Respuesta al impulso para los esquemas equivalentes de las Figuras 6.18 y 6.19.	108
6.26. Esquema de realización eficiente de las multiplicaciones a base de corri- mientos y sumas.	109
6.27. Incremento en el tamaño de palabra en la realización eficiente del ejemplo 6.10.	110
6.28. Diagramas de bloques para las etapas del filtro del ejemplo 6.10.	110
6.29. Señales reloj utilizadas para cada etapa del filtro del ejemplo 6.10.	111
6.30. Implementación del sobremuestreador.	112
6.31. Respuesta al impulso del filtro del ejemplo 6.10.	113
6.32. Esquema de realización eficiente de las multiplicaciones a base de corri- mientos y sumas.	113
6.33. Incremento del tamaño de palabra para la primera etapa.	114
6.34. Diagramas de bloques para las etapas del filtro del ejemplo 6.11.	114
6.35. Señales de reloj generadas para el ejemplo 6.11.	115
6.36. Respuesta al impulso para ejemplo 6.11.	115

Índice de Tablas

4.1. Variación de la SNR con la selección de los retardos D_i	56
4.2. Cantidad de APOS y memoria requerida para el filtro CIC triangular escalonado usando $M_1 < M_2$	58
4.3. Escalamiento del voltaje de alimentación para algunos estándares de comunicación inalámbrica.	61
4.4. Factorización propuesta para diferentes estándares de comunicación inalámbrica.	62
4.5. Factorización del factor de decimación para diferentes estándares de comunicación inalámbrica	64
4.6. Magnitudes importantes para diferentes retardos en el último filtro <i>comb</i> .	74
6.1. Comparación ejemplo 6.8.	99
6.2. Comparación ejemplo 6.9.	102

Acrónimos

APOS Additions Per Output Sample.

CIC Cascade Integrator Comb.

FIR Finite Impulse Response.

IIR Infinite Impulse Response.

RSCR Rational Sample Rate Conversion.

SDR Software Defined Radio.

SNR Signal to Noise Ratio.

Capítulo 1

Introducción

En este capítulo se presenta una revisión de los fundamentos de sistemas de conversión de frecuencia de muestreo, incluyendo su descripción en el dominio del tiempo y de la frecuencia. También se presentan algunos conceptos claves usados en el procesamiento digital de múltiples frecuencias de muestreo o multi-razón. Estos conceptos serán de vital importancia para el desarrollo posterior de este trabajo.

1.1. Conversión de la Frecuencia de Muestreo

El proceso de conversión de frecuencias de muestreo puede realizarse directamente en el dominio digital. Allí se definen dos casos. El primero es la reducción de la frecuencia de muestreo mejor conocida como decimación y su incremento o interpolación. También es posible realizar una conversión racional mediante la combinación de las dos operaciones. Se describirán en esta sección las características temporales y frecuenciales de estos procesos.

1.1.1. Decimación

El proceso de decimación consiste en la reducción de la frecuencia de muestreo de una señal digital. Este proceso puede describirse en dos etapas (Fig. 1.1), las cuales modifican el espectro de la señal de entrada. En primer lugar se debe realizar un filtrado de la señal de entrada con el fin de limitarla en banda y evitar la presencia de *aliasing* en la señal de salida, luego es necesario realizar la operación de submuestreo que se describirá en detalle a continuación.

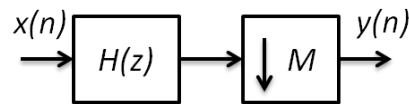


Figura 1.1: Diagrama de bloques de la operación de decimación.

1.1.1.1. Submuestreo

Para un factor entero de submuestreo M , las muestras múltiplos de M son mantenidas, haciendo las que no lo son cero. Posteriormente, se eliminan las muestras de valor cero de la señal. Este proceso en el dominio del tiempo equivale a la expresión presentada a continuación y se representa de la forma mostrada en la Fig. 1.2.

$$y[n] = x[nM]. \quad (1.1)$$

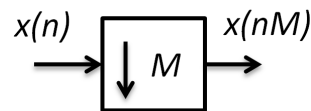


Figura 1.2: Representación de la operación de submuestreo.

Debido al proceso de submuestreo la frecuencia de muestreo es reducida en un factor M . En el ejemplo 1.1 se ilustra este proceso.

Ejemplo 1.1: En la Fig. 1.3 se presenta un ejemplo del submuestreo de una señal senoidal por un factor $M = 2$.

Para examinar los efectos del submuestreo en el dominio de la frecuencia se utilizará una función auxiliar que en el dominio del tiempo corresponderá a una señal periódica definida por la ecuación (1.2), que multiplica a la señal de entrada haciendo que únicamente las muestras con índices múltiplos de M se conserven.

$$C(n) = \begin{cases} 1, & n = 0, M, 2M, \dots, \\ 0 & \text{para otro caso} \end{cases} \quad (1.2)$$

En el dominio de la frecuencia esta función puede ser representada por su serie de Fourier dada por la ecuación (1.3).

$$C(n) = \frac{1}{M} \sum_{k=0}^{M-1} c(k) e^{\frac{j2\pi kn}{M}}. \quad (1.3)$$

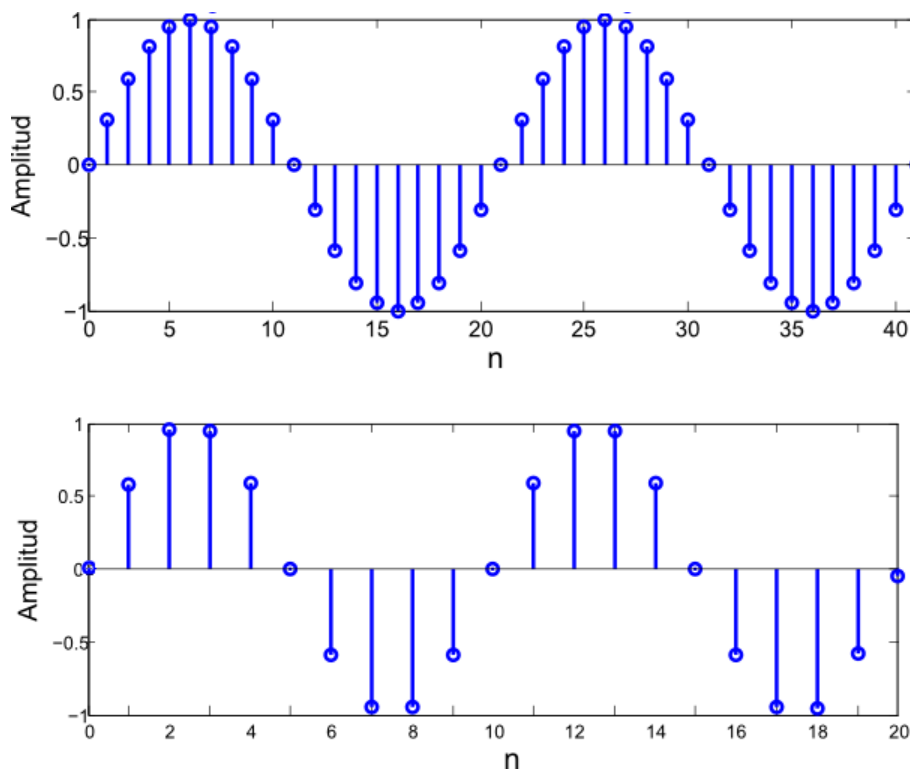


Figura 1.3: Ilustración del proceso de submuestreo en el tiempo, ($M = 2$).

Los coeficientes de la serie de Fourier de (1.3) están dados por

$$c(k) = \frac{1}{M} \sum_{k=0}^{M-1} C(n) e^{-\frac{j2\pi kn}{M}}. \quad (1.4)$$

Usando la ecuación (1.4) se obtiene la forma final de la función auxiliar en el dominio del tiempo.

$$C(n) = \frac{1}{M} \sum_{k=0}^{M-1} e^{\frac{j2\pi kn}{M}}. \quad (1.5)$$

Con el fin de facilitar la comprensión del proceso de submuestreo se definirá una salida intermedia $y'(n)$ que será el resultado del producto de la señal de entrada $x(n)$ y la función auxiliar $C(n)$.

$$y'(n) = x(n)C(n) = x(n) \frac{1}{M} \sum_{k=0}^{M-1} e^{\frac{j2\pi kn}{M}}. \quad (1.6)$$

Para obtener la representación en el dominio de la frecuencia de la ecuación (1.6), se utiliza la Transformada Discreta de Fourier definida por la ecuación (1.7) [1], obteniendo

$$A(e^{j\omega}) = \sum_{n=-\infty}^{\infty} a(n)e^{-j\omega n}, \quad (1.7)$$

$$Y'(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x(n) \frac{1}{M} \sum_{k=0}^{M-1} e^{j\frac{2\pi kn}{M}} e^{-j\omega n}. \quad (1.8)$$

Luego de algo de álgebra se obtiene una ecuación que permite identificar claramente el efecto de la multiplicación por la función auxiliar.

$$Y'(e^{j\omega}) = \frac{1}{M} \sum_{k=0}^{M-1} \sum_{n=-\infty}^{\infty} x(n) e^{-jn(\omega - \frac{2\pi k}{M})}. \quad (1.9)$$

El término dentro de la segunda sumatoria corresponde a una versión desplazada en frecuencia de la señal de entrada. Finalmente, la ecuación de la señal de salida auxiliar está dada por

$$Y'(e^{j\omega}) = \frac{1}{M} \sum_{k=0}^{M-1} X\left(e^{j(\omega - \frac{2\pi k}{M})}\right). \quad (1.10)$$

En el dominio de la frecuencia el hecho de hacer las muestras intermedias cero produce la periodicidad del espectro original, que se repetirá cada $2\pi/M$. Además, cuando la señal de entrada posee componentes espectrales por encima de la frecuencia π/M , estas réplicas se traslaparán, lo que se conoce comúnmente como *aliasing*.

Ahora se analizará el efecto de quitar las muestras con valor cero. Esto significa que la señal de entrada sufre una compresión en el dominio del tiempo dada por la ecuación (1.1), en el dominio de la frecuencia se tendrá una expansión. Finalmente, el espectro de la señal de salida está dado por la ecuación (1.11):

$$Y(e^{j\omega}) = \frac{1}{M} \sum_{k=0}^{M-1} X\left(e^{j(\omega - 2\pi k)/M}\right). \quad (1.11)$$

Usando $z = e^{j\omega}$ se obtiene la ecuación de la salida en el dominio z :

$$Y(z) = \frac{1}{M} \sum_{k=0}^{M-1} X(zW_M^k), \quad (1.12)$$

donde

$$W_M^k = e^{-j\frac{2\pi k}{M}} \quad \text{para } k = 0, 1, \dots, M-1. \quad (1.13)$$

Es importante mencionar que la presencia de *aliasing* en la salida del submuestreador distorsiona la señal de entrada imposibilitando su recuperación.

Ejemplo 1.2: En la Fig.1.4 se presenta el caso para $M = 5$, donde se incluyen las respectivas respuestas al impulso y su espectro en frecuencia. Se observa que la señal de entrada esta limitada en banda y su máxima componente de frecuencia no sobrepasa la frecuencia π/M , por lo que no se presenta *aliasing*.

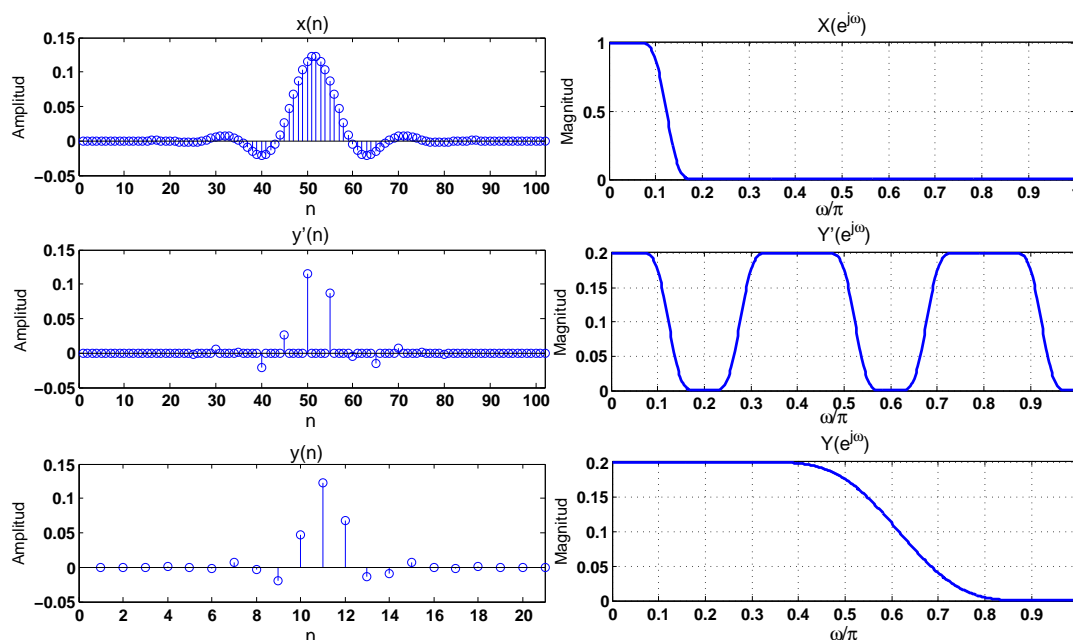


Figura 1.4: Ilustración del proceso de submuestreo en el dominio de la frecuencia.

1.1.1.2. Filtrado pasa-bajas *anti-aliasing*

Por otro lado, cuando la señal no está limitada en banda (Ejemplo 1.3), se tiene traslape de las réplicas. Es por esa razón por la que la señal de entrada es sometida a un proceso de filtrado cuya respuesta en magnitud se presenta en la ecuación (1.14). Este filtrado se encarga de limitar la banda de frecuencias de la señal de entrada eliminando el *aliasing*.

$$|H(e^{j\omega})| = \begin{cases} 1, & |\omega| \leq \omega_c/M \\ 0, & \text{en otro caso.} \end{cases} \quad (1.14)$$

donde ω_c es la frecuencia más alta de la señal de entrada que necesita preservarse.

Ejemplo 1.3: En la Fig. 1.5 se presenta el proceso de submuestreo. La componente de frecuencia máxima sobrepasa $\pi/M = \pi/5$. Se observa la presencia de *aliasing* resaltada por el círculo rojo.

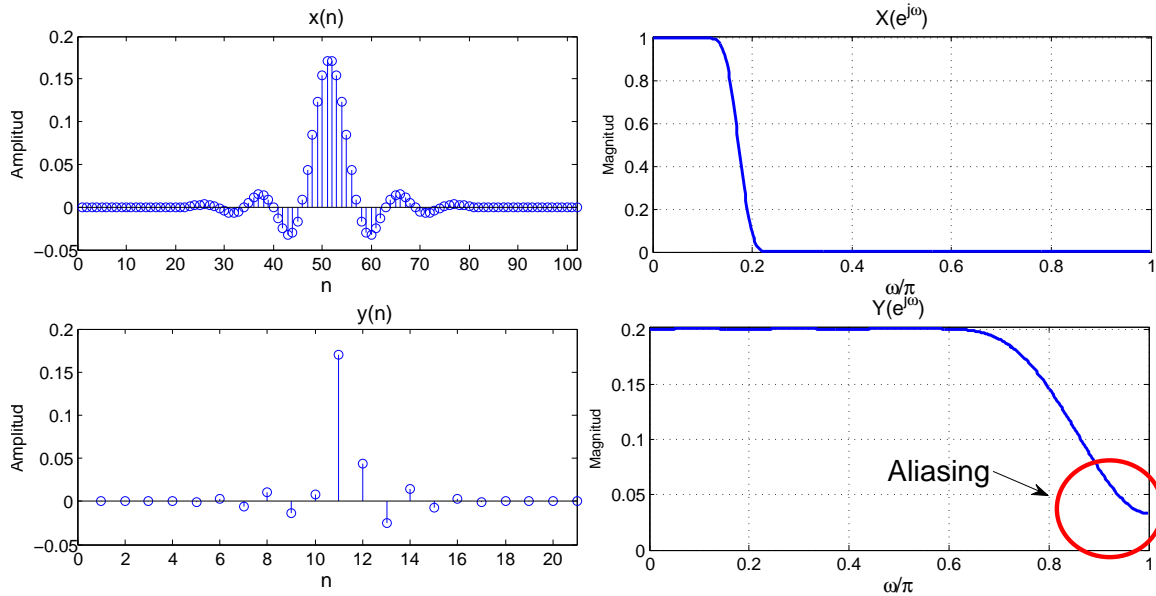


Figura 1.5: Ilustración del efecto del *aliasing* en una señal debido al submuestreo.

1.1.2. Interpolación

El proceso de interpolación se implementa mediante dos etapas, la primera es el sobremuestreo y la segunda es el filtrado, que se requiere para eliminar las réplicas del espectro original que se introducen en el proceso de sobremuestreo. El esquema en diagrama de bloques del proceso de interpolación se presenta en la Fig. 1.6.

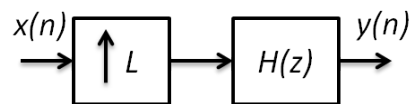


Figura 1.6: Estructura en diagrama de bloques para la interpolación.

1.1.2.1. Sobremuestreo

El sobremuestreo consiste en la inserción de ceros entre las muestras originales. Para el caso de un factor de interpolación entero L se deben introducir $L - 1$ ceros equidistantes entre las muestras de la señal original, como lo define la ecuación (1.15):

$$y[n] = \begin{cases} x[n/L] & \text{para } n = 0, L, 2L, \dots \\ 0 & \text{otros casos} \end{cases} \quad (1.15)$$

donde $n > 0$. El sobremuestreo incrementa la frecuencia de muestreo por un factor L . En el ejemplo 1.4 se ilustra este proceso.

Ejemplo 1.4: En la Fig. 1.7 se presenta un ejemplo del sobremuestreo de una señal senoidal por un factor $L = 3$.

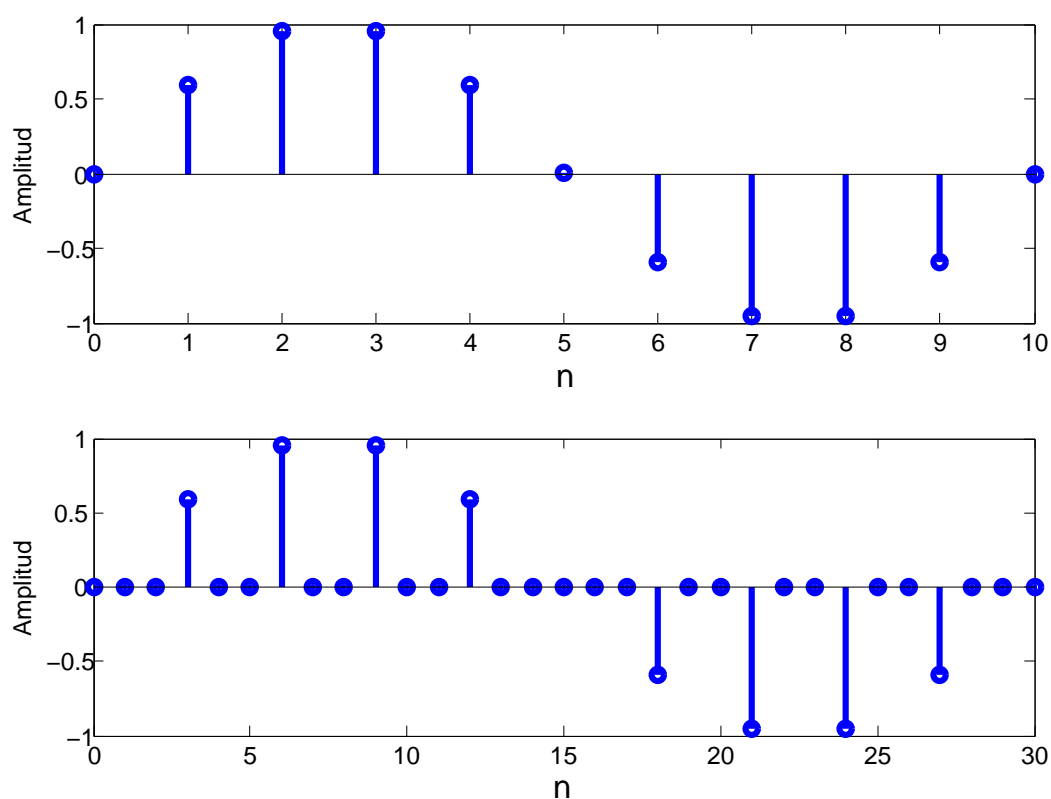


Figura 1.7: Ilustración del proceso de sobremuestreo en el tiempo.

Para analizar su comportamiento en frecuencia se usará la Transformada Discreta de Fourier:

$$Y(e^{j\omega}) = \sum_{k=-\infty}^{\infty} y(n)e^{-j\omega n}, \quad (1.16)$$

$$Y(e^{j\omega}) = \sum_{k=-\infty}^{\infty} x(n/L)e^{-j\omega L(\frac{n}{L})}. \quad (1.17)$$

Si en la ecuación (1.17), n/L se reemplaza por n entonces se tiene que la ecuación de salida está dada por la ecuación (1.18) y en el dominio z su equivalente está dado por la ecuación (1.19).

$$Y(e^{j\omega}) = X(e^{j\omega L}). \quad (1.18)$$

$$Y(z) = X(z^L). \quad (1.19)$$

El efecto del sobremuestreo en el dominio de la frecuencia lo describe la ecuación (1.18). De allí se observa que esta operación no produce un escalamiento en magnitud pero introduce $L - 1$ réplicas del espectro original. Este fenómeno se conoce como *imaging* y se observa en el ejemplo 1.5.

Ejemplo 1.5: En la Fig 1.8 se presenta la aparición de imágenes producidas por el sobremuestreo para $L = 4$. Las réplicas aparecen en los múltiplos enteros de $2\pi/L = \pi/2$.

1.1.2.2. Filtrado pasa-bajas *anti-imaging*

Después de hacer el sobremuestreo de la señal se realiza un proceso de filtrado pasabajas con el filtro definido por la ecuación (1.20), que permite recuperar la señal original eliminando las imágenes introducidas por el proceso de sobremuestreo.

$$|H(e^{j\omega})| = \begin{cases} L, & |\omega| \leq \omega_c/L \\ 0, & \text{en otro caso.} \end{cases} \quad (1.20)$$

donde ω_c corresponde a la frecuencia más alta de la señal de entrada que se desea conservar.

1.1.3. Conversión por un factor racional

Es posible obtener un factor de conversión de frecuencia de muestreo racional si se combinan las estructuras utilizadas en conversiones enteras. Al realizar una interpolación y una decimación en cascada se tendría un factor de conversión L/M . Para ello se pueden tener dos diferentes estructuras: la primera realiza la decimación y posteriormente la interpolación (Fig. 1.9) y en la segunda primero se lleva a cabo la interpolación y después

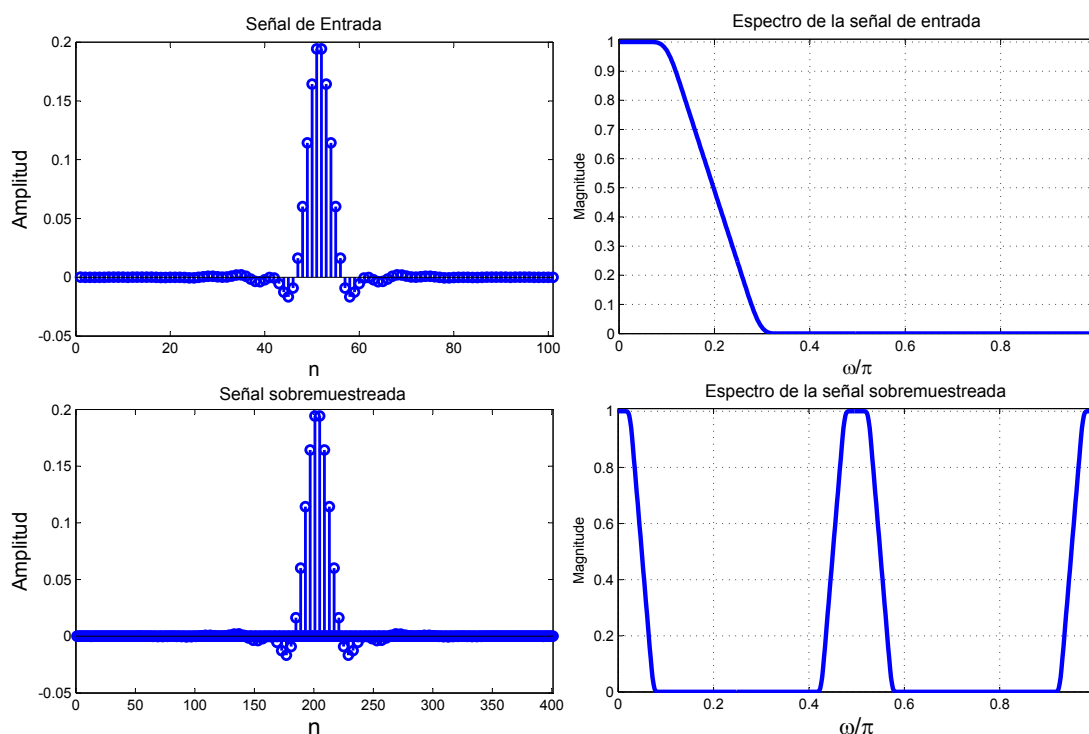


Figura 1.8: Ilustración del proceso de sobremuestreo en el dominio de la frecuencia ($L = 4$).

la decimación (Fig. 1.10).

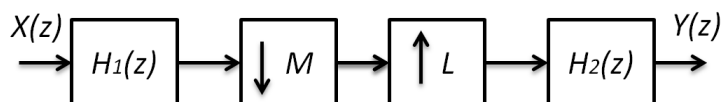
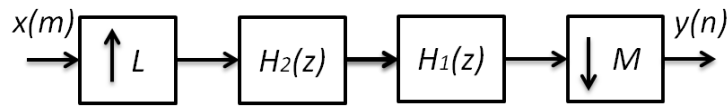


Figura 1.9: Cascada decimación-interpolación.

Más eficiente es el esquema de la Fig. 1.10, es decir, primero la interpolación seguida por la decimación, porque se pueden combinar los filtros *anti-aliasing* y *anti-imaging* en un solo filtro, esto se presenta en la parte (b) de la Fig. 1.10.

Además, la frecuencia de corte del nuevo filtro pasa-bajas que se encarga de eliminar tanto el *aliasing* como el *imaging* introducido por los dos procesos de cambio de frecuencia de muestreo, está definida por la ecuación (1.21):

$$\omega_s = \min\left(\frac{\pi}{L}, \frac{\pi}{M}\right). \quad (1.21)$$



(a) Cascada interpolación-decimación.

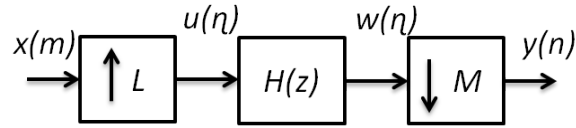
(b) Estructura resultante con filtro *anti-aliasing/anti-aliasing*.

Figura 1.10: Estructura para conversión racional de frecuencia de muestreo

1.2. Identidades, descomposición polifase e intercambio de conversores de frecuencia de muestreo

En esta sección se presentan algunos de los conceptos fundamentales requeridos para la comprensión y el desarrollo de sistemas que incluyan algún tipo de conversión de frecuencia de muestreo. Se definirán las identidades multi-razón o identidades nobles [1], [2], descomposición polifase [1] y finalmente el intercambio de conversores de frecuencias de muestreo [2].

1.2.1. Identidades nobles

Existen 6 identidades básicas en el campo de los sistemas de conversión de frecuencia de muestreo. Su importancia radica en que son poderosas herramientas que permiten la manipulación de estructuras multi-razón para su análisis o realización de forma eficiente.

1.2.1.1. Primera identidad

La primera identidad corresponde a la linealidad del proceso de submuestreo, es decir, el submuestreo de la suma de n señales escaladas es igual a la suma de esas n señales escaladas submuestreadas. Esta identidad se presenta en la Fig. 1.11.

1.2.1.2. Segunda identidad

La segunda identidad establece que un retardo de orden M que precede a un submuestreador es equivalente a un retardo de orden 1 después del submuestreador (Fig. 1.12).

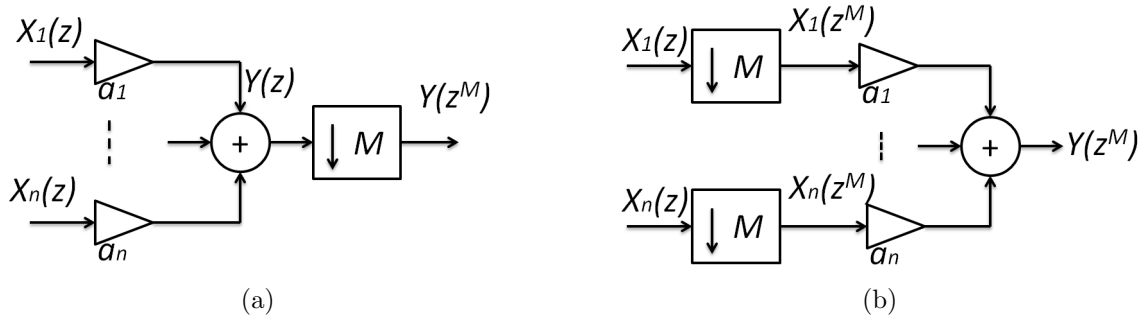


Figura 1.11: Estructuras equivalentes para la primera identidad.

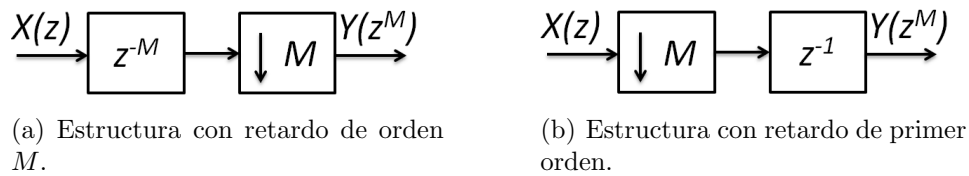


Figura 1.12: Estructuras equivalentes para la segunda identidad.

1.2.1.3. Tercera identidad

La tercera identidad corresponde a la generalización de la segunda identidad. En este caso se tiene un filtro expandido $G(z^M)$ precediendo al submuestreador, lo que es equivalente a tener el submuestreador precediendo al filtro no expandido $G(z)$ (Fig. 1.13).

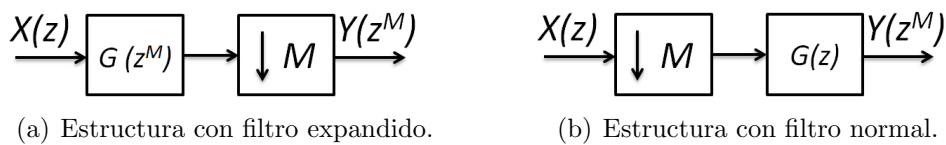


Figura 1.13: Estructuras equivalentes para la tercera identidad.

1.2.1.4. Cuarta identidad

La cuarta identidad es la propiedad de linealidad del sobremuestreador, de la salida del sobremuestreador se tienen n ramificaciones escaladas entonces el submuestreador puede moverse de la entrada al final de cada rama escalada (Fig. 1.14).

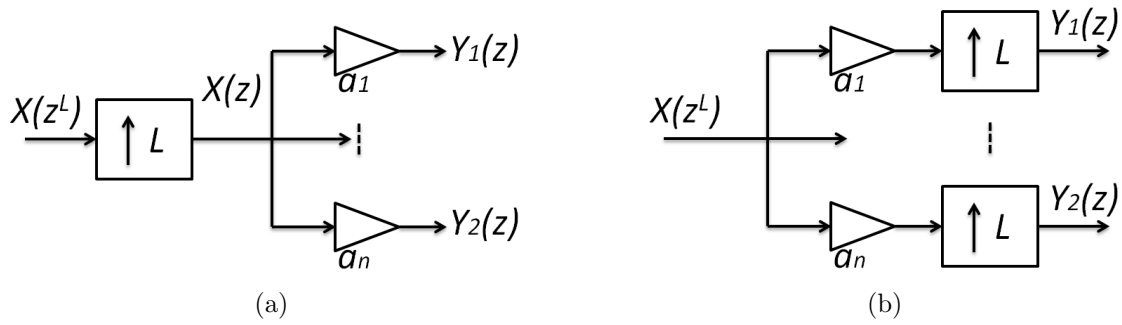
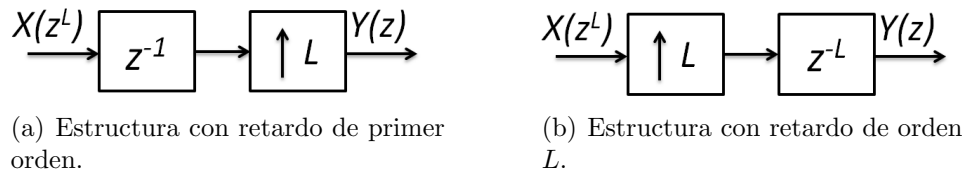


Figura 1.14: Estructuras equivalentes para la cuarta identidad.

1.2.1.5. Quinta identidad

La quinta identidad establece que un retardo de primer orden precediendo al sobremuestreador es equivalente a un retardo de orden L después del sobremuestreador (Fig. 1.15).



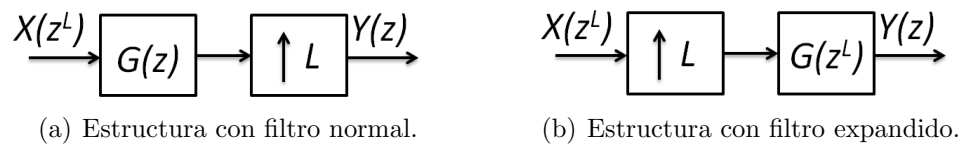
(a) Estructura con retardo de primer orden.

(b) Estructura con retardo de orden L .

Figura 1.15: Estructuras equivalentes para la quinta identidad.

1.2.1.6. Sexta identidad

La sexta identidad corresponde a la generalización de la quinta identidad y establece que un filtro que precede a un sobremuestreador es equivalente al sobremuestreador precediendo a la versión expandida de ese filtro (Fig. 1.16).



(a) Estructura con filtro normal.

(b) Estructura con filtro expandido.

Figura 1.16: Estructuras equivalentes para la sexta identidad.

1.2.2. Decimación e interpolación polifase

Usando la descomposición polifase y aplicando las identidades multi-razón, es posible obtener estructuras eficientes para la implementación de ambos procesos [2], [3] y [4].

1.2.2.1. Descomposición polifase

Se considera la función de transferencia descrita por la ecuación (1.22).

$$X(z) = \sum_{n=-\infty}^{\infty} x(n)z^{-n}. \quad (1.22)$$

La ecuación anterior puede ser reescrita de la siguiente manera para una descomposición en Q ramas.

$$X(z) = \sum_{k=0}^{Q-1} z^{-k} E_k(z^Q), \quad (1.23)$$

donde

$$E_k(z) = \sum_{n=0}^{\lfloor (N+1)/Q \rfloor} h[Qn+k]z^{-n} \quad \text{para } 0 \leq k \leq Q-1. \quad (1.24)$$

El uso de la ecuación (1.23) se conoce como la descomposición polifase y para su implementación se tienen dos posibles estructuras que se presentan en la Fig. 1.17.

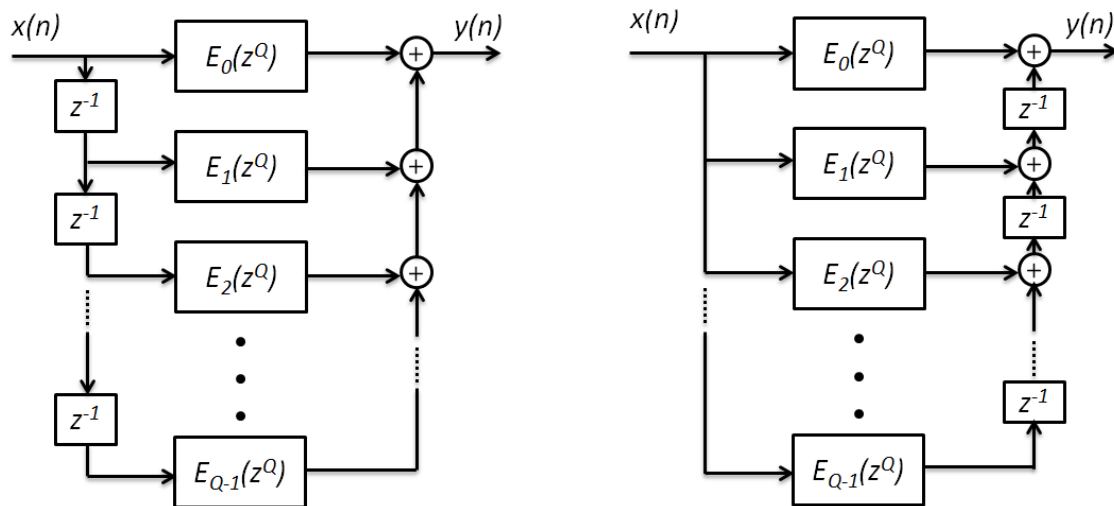
Se considera ahora la complejidad de la implementación de un filtro polifase, suponiendo que la longitud del filtro $H_k(z)$ es N_k :

$$N = \sum_{k=0}^{Q-1} N_k. \quad (1.25)$$

La cantidad de multiplicaciones para el $k^{\text{ésimo}}$ filtro son N_k , mientras que la cantidad de sumas es $N_k - 1$. Ambas cantidades corresponden a operaciones por muestra de salida. Para la estructura completa se tiene:

$$\sum_{k=0}^{Q-1} N_k = N, \quad (1.26)$$

y



(a) Estructura para la descomposición polifase.

(b) Estructura para la descomposición polifase transpuesta.

Figura 1.17: Implementación de la estructura polifase.

$$\sum_{k=0}^{Q-1} (N_k - 1) + (Q - 1) = N - 1. \quad (1.27)$$

La ecuación (1.26) corresponde a la cantidad total de multiplicaciones por muestra de salida y la ecuación (1.27) a la cantidad de sumas por muestra de salida. Se puede concluir que la cantidad de operaciones no se ve reducida, pero como se verá a continuación para la realización de decimación o interpolación, se obtendrá una estructura eficiente gracias a la descomposición polifase.

Ejemplo 1.6: Para clarificar el tema de la descomposición polifase se presenta un ejemplo para $Q = 3$. En este caso la función de transferencia está dada por la ecuación (1.28).

$$X(z) = h[0] + h[1]z^{-1} + h[2]z^{-2} + h[3]z^{-3} + h[4]z^{-4} + h[5]z^{-5} + h[6]z^{-6} + h[7]z^{-7} + h[8]z^{-8}. \quad (1.28)$$

La función de transferencia anterior puede escribirse usando la ecuación (1.23) como se presenta a continuación.

$$X(z) = E_0(z^3) + z^{-1}E_1(z^3) + z^{-2}E_2(z^3), \quad (1.29)$$

donde

$$E_0(z) = h[0] + h[3]z^{-1} + h[6]z^{-2}, \quad (1.30)$$

$$E_1(z) = h[1] + h[4]z^{-1} + h[7]z^{-2}, \quad (1.31)$$

$$E_2(z) = h[2] + h[5]z^{-1} + h[8]z^{-2}. \quad (1.32)$$

1.2.2.2. Decimación polifase

Como resultado de aplicar la descomposición polifase se obtienen ramas de filtros de Respuesta Finita al Impulso o Finite Impulse Response (FIR) expandidos. Si se realiza la decimación conjuntamente con esta descomposición es posible obtener una estructura eficiente para la realización de la decimación. Retomando la estructura presentada en la Fig. 1.1, si el filtro $H(z)$ es implementado usando la realización polifase con M ramas, se obtiene la estructura presentada en la Fig. 1.18.

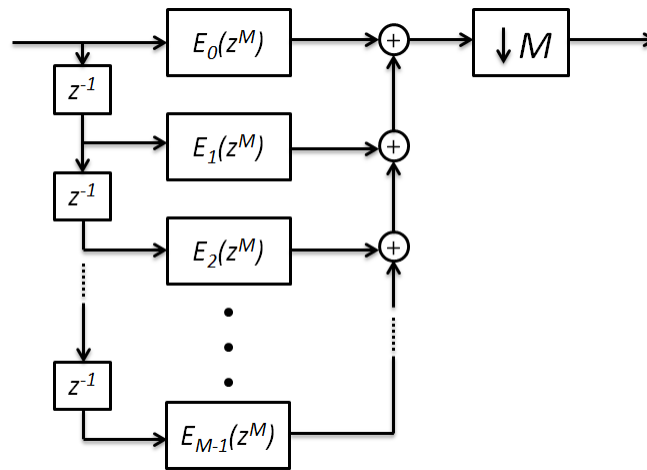


Figura 1.18: Decimación con filtro polifase.

Al usar la primera identidad, el submuestreador puede moverse a cada rama del filtro polifase. Además, usando la tercera identidad estos submuestreadores pueden moverse antes de los subfiltros, cambiando los filtros expandidos por los filtros originales como se muestra en la Fig. 1.19. En esta estructura, los subfiltros de cada rama funcionan en la frecuencia de entrada reducida por un factor M . Esta reducción de la frecuencia de muestreo representa un ahorro significativo en el número de operaciones aritméticas por muestra de salida generando un ahorro en el consumo de potencia.

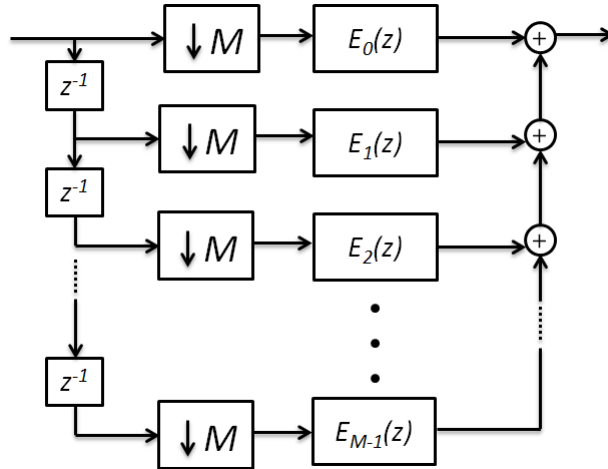


Figura 1.19: Decimación eficiente con filtro polifase.

1.2.2.3. Interpolación polifase

De forma similar a como se obtuvo una estructura eficiente para la decimación, es posible obtener una estructura eficiente para la interpolación. Usando la estructura presentada en la Fig. 1.6 y la descomposición polifase en L ramas del filtro *anti-imaging*, se obtiene la estructura de la Fig. 1.20.

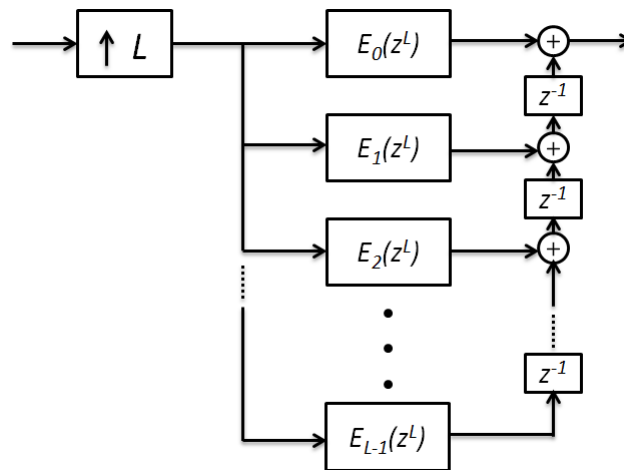


Figura 1.20: Interpolación con filtro polifase.

Utilizando la estructura polifase transpuesta de la Fig. 1.17(b) y usando la cuarta identidad es posible mover el sobremuestreador a cada rama del filtro polifase. Además, si se utiliza la sexta identidad estos sobremuestreadores pueden moverse después de los filtros expandidos, quedando antes de ellos los filtros originales. Tras aplicar las dos identidades

anteriormente nombradas se obtiene la estructura eficiente para la interpolación mostrada en la Fig. 1.21.

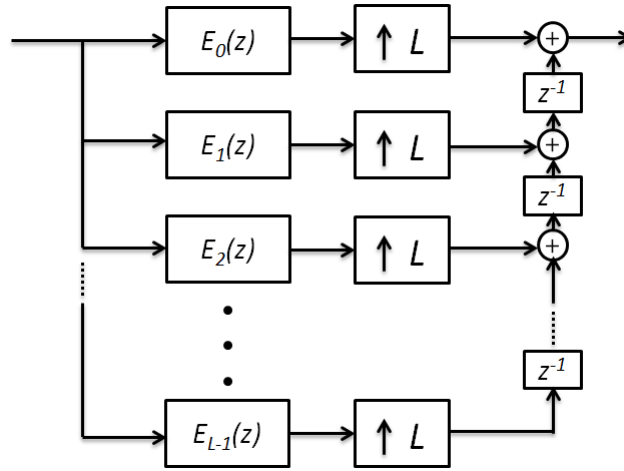


Figura 1.21: Interpolación eficiente con filtro polifase transpuesto.

Como se observa, en esta estructura eficiente los subfiltros de cada rama que antes funcionaban en alta frecuencia, ahora funcionan en la frecuencia de entrada que es L veces menor que la frecuencia de salida, obteniendo una realización con menor número de operaciones aritméticas por muestra de salida y con menor consumo de potencia.

1.2.2.4. Intercambio de conversores de frecuencia de muestreo

Cuando se realizan conversiones de frecuencia de muestreo se hace interesante y útil saber cuando es posible realizar el intercambio de los bloques de conversión.

Primero se considera el diagrama de bloques de la Fig. 1.22, donde se realiza primero el submuestreo y luego el sobremuestreo.

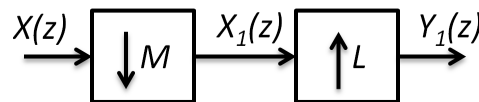


Figura 1.22: Cascada de sobremuestreador y submuestreador.

La señal $X_1(z)$ es:

$$X_1(z^M) = \frac{1}{M} \sum_{K=0}^{M-1} X(zW_M^K). \quad (1.33)$$

La señal de salida está dada por,

$$Y_1(z) = X_1(z^L) = \frac{1}{M} \sum_{K=0}^{M-1} X(z^{L/M} W_M^K). \quad (1.34)$$

Ahora se considera el caso contrario, es decir, primero el sobremuestreo y después el submuestreo como se presenta en la Fig. 1.23.

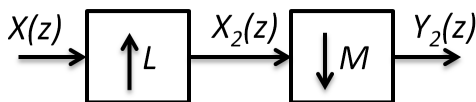


Figura 1.23: Cascada de submuestreador y sobremuestreador.

La señal $X_2(z)$ es:

$$X_2(z) = X(z^L). \quad (1.35)$$

La señal de salida está definida por la ecuación (1.36),

$$Y_2(z^M) = \frac{1}{M} \sum_{K=0}^{M-1} X_2(z W_M^K) = \frac{1}{M} \sum_{K=0}^{M-1} X(z^L W_M^{KL}), \quad (1.36)$$

donde z^L , debido a la decimación, se cambia por:

$$(z W_M^K)^L = z^L W_M^{KL}, \quad (1.37)$$

lo que resulta en:

$$Y_2(z^M) = \frac{1}{M} \sum_{K=0}^{M-1} X(z^{L/M} W_M^{KL}). \quad (1.38)$$

Comparando las ecuaciones (1.34) y (1.38), se tiene que se pueden intercambiar los conversores de frecuencia de muestreo cuando se cumpla la condición (1.39).

$$W_M^K = W_M^{KL}. \quad (1.39)$$

Esta condición se cumple cuando M y L son números primos relativos [2], lo que significa que no existe un factor común entero entre ellos excepto 1. En este trabajo se consideran valores de M y L que cumplen la propiedad de ser números primos relativos, ya que una conversión para factores que no lo son resulta en una conversión entera (decimación o interpolación). Por ejemplo con $M = 27$ y $L = 3$ se tiene un equivalente a una decimación por un factor 9.

Capítulo 2

Filtros usados en este trabajo

En este capítulo se presentarán algunos de los filtros usados para conversión de frecuencia de muestreo, empezando por el filtro CIC. También se presentarán estructuras para realizar interpolación.

2.1. Filtro CIC

El filtro Cascade Integrator Comb (CIC) fue propuesto por Hogenauer en [5]. Su nombre se deriva de su estructura, que se compone de una cascada de una etapa integradora y una etapa *comb*¹(*Cascaded-Integrator-Comb*).

La ecuación (2.1) corresponde a la respuesta al impulso del filtro CIC.

$$h[n] = \begin{cases} \frac{1}{M}, & 0 \leq n \leq M - 1 \\ 0, & \text{otros.} \end{cases} \quad (2.1)$$

La transformada z se calcula fácilmente usando la ecuación (2.2) [1].

$$H(z) = \sum_{n=-\infty}^{\infty} h[n]z^{-n}. \quad (2.2)$$

De (2.1) y (2.2) se tiene:

$$H(z) = \frac{1}{M} \sum_{n=0}^{M-1} z^{-n} = \frac{1}{M} (1 + z^{-1} + z^{-2} + \dots + z^{-(M-2)} + z^{-(M-1)}). \quad (2.3)$$

¹Se le conoce a este filtro como filtro comb o peine (en español), debido a que su respuesta al impulso exhibe un comportamiento similar a un peine para cabello

A esta ecuación se le conoce como ecuación no recursiva. Sin embargo, es posible obtener una representación alternativa del filtro al presentarlo con su ecuación equivalente en forma recursiva, con lo que se obtiene la expresión presentada a continuación. La estructura del filtro CIC en forma recursiva se presenta en la Fig. 2.1.

$$H(z) = \frac{1}{M} \left[\frac{1 - z^{-M}}{1 - z^{-1}} \right]. \quad (2.4)$$

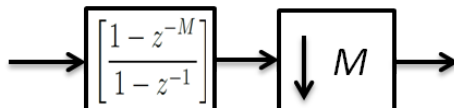


Figura 2.1: Estructura del filtro CIC recursivo.

Usando la tercera identidad multi-razón es posible mover a baja frecuencia la parte *comb.* Obteniendo la estructura de la Fig. 2.2.

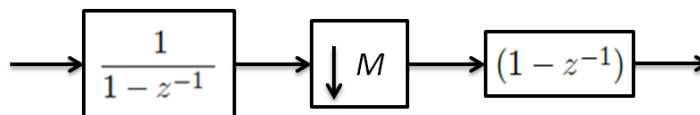


Figura 2.2: Estructura recursiva.

Con el fin de mostrar la características en el dominio de la frecuencia del filtro CIC se obtendrá la respuesta en magnitud. Partiendo de la ecuación (2.2), reemplazando z por el término de frecuencia compleja $e^{j\omega}$ se obtiene:

$$\begin{aligned} H(e^{j\omega}) &= \frac{1}{M} \left[\frac{1 - e^{-jM\omega}}{1 - e^{-j\omega}} \right] = \frac{1}{M} \left[\frac{e^{-j\omega M/2} (e^{j\omega M/2} - e^{-j\omega M/2})}{e^{-j\omega/2} (e^{j\omega/2} - e^{-j\omega/2})} \right] \\ &= \frac{1}{M} \left[\frac{e^{-jM\omega} 2j \operatorname{sen}(j\omega M/2)}{e^{-j\omega/2} 2j \operatorname{sen}(j\omega/2)} \right]. \end{aligned} \quad (2.5)$$

Luego de realizar algunas operaciones matemáticas se obtiene la ecuación presentada a continuación:

$$H(e^{j\omega}) = \frac{1}{M} \left[\frac{\operatorname{sen}\left(\frac{\omega M}{2}\right)}{\operatorname{sen}\left(\frac{\omega}{2}\right)} \right] e^{j\omega \frac{(M-1)}{2}}, \quad (2.6)$$

donde el término exponencial tiene que ver con la fase y el término que incluye las funciones seno determina la respuesta en magnitud. De allí es posible observar que el filtro CIC posee ceros en los múltiplos enteros de $2\pi/M$. En el caso de un M par se tienen $\frac{M}{2}$ ceros y para M impar se tienen $\frac{M-1}{2}$ ceros. La respuesta en magnitud se caracteriza por un comportamiento del tipo de una función sinc. Las características en magnitud

del filtro CIC se presentan en la Fig. 2.3. Allí se observa que el *aliasing* se presenta justamente donde se encuentran los ceros del filtro. Sin embargo, el filtro CIC presenta una pobre atenuación en los dobles de la banda, aunque esta atenuación puede ser mejorada incluyendo cascadas del filtro CIC original, lo que resulta en la función de transferencia:

$$H(z) = \frac{1}{M} \left[\frac{1 - z^{-M}}{1 - z^{-1}} \right]^K, \quad (2.7)$$

donde K es la cantidad de etapas del filtro CIC y a partir de este momento será referido con el nombre de factor de cascada.

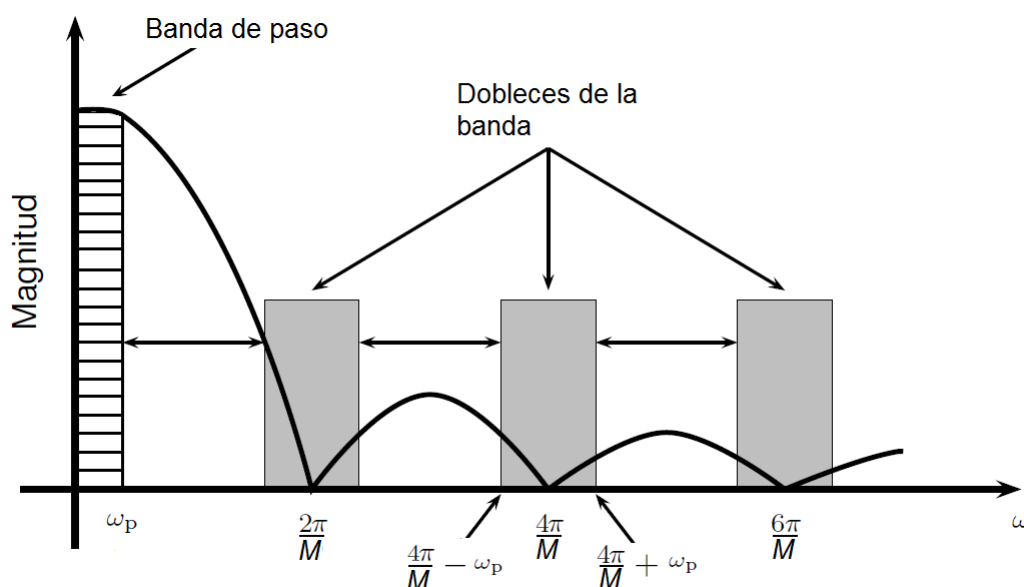


Figura 2.3: Propiedades en magnitud del filtro CIC como filtro *anti-aliasing*.

Ejemplo 2.1: Se presenta a continuación un ejemplo de un filtro CIC para un factor de decimación $M = 16$, donde la función de transferencia corresponde a la ecuación (2.8):

$$H(z) = \frac{1}{16} \left[\frac{1 - z^{-16}}{1 - z^{-1}} \right], \quad (2.8)$$

También se presenta la versión mejorada del filtro usando cascadas, cuya función de transferencia corresponde a la ecuación (2.9):

$$H(z) = \frac{1}{16} \left[\frac{1 - z^{-16}}{1 - z^{-1}} \right]^K. \quad (2.9)$$

Las respectivas respuestas en magnitud se presentan en la Fig. 2.4. Allí se observa que la atenuación es mejorada gracias a las cascadas de filtros CIC. La principal desventaja que

presenta la inclusión de cascadas, es el aumento en la caída en la banda de paso, lo que resulta en un ancho de banda útil muy pequeño.

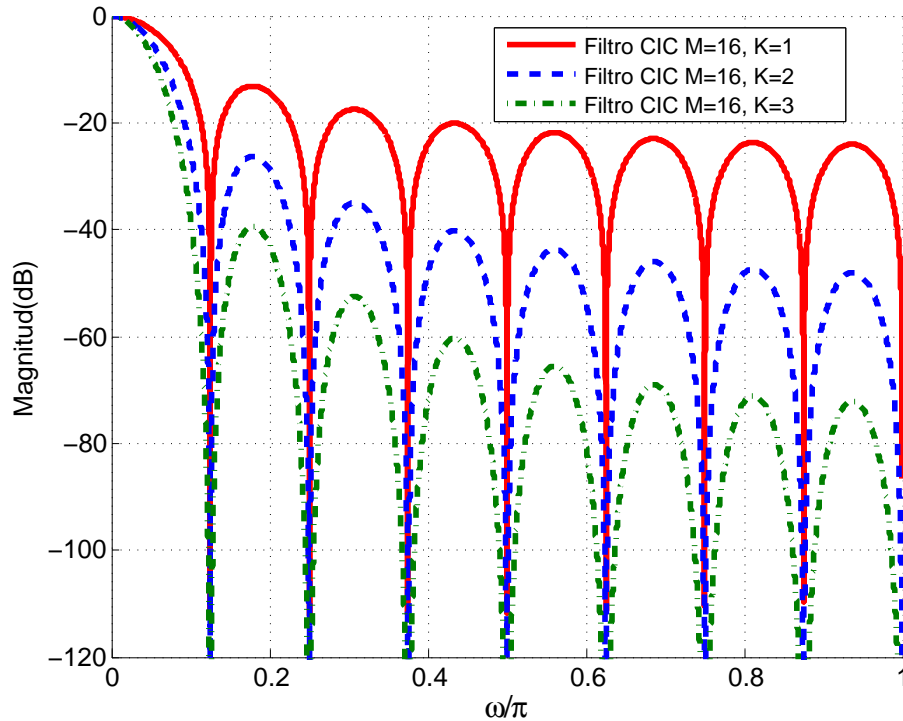


Figura 2.4: Respuestas en magnitud para el filtro CIC con $K = 1$ y sus versiones con factores de cascada $K = 2$ y $K = 3$.

2.1.1. Ventajas y desventajas del filtro CIC

Se presentan en esta sección las principales ventajas y desventajas del filtro CIC. Algunas características lo hacen realmente deseable en aplicaciones de Software Defined Radio (SDR), pero, también posee características negativas que deben mejorarse en la medida de lo posible para su uso en SDR.

Ventajas

- No posee multiplicaciones.
- Al ser un filtro FIR no presenta problemas de estabilidad.
- Posee dos formas de implementación que permiten lograr ahorro de potencia (estructura no recursiva) o conservación y simpleza de la estructura (estructura recursiva).

- Pueden realizarse eficientemente usando la descomposición polifásica y las identidades multi-razón.

Desventajas

- Baja atenuación en los dobleces de las bandas.
- Distorsión de la señal de entrada por la caída en la banda de paso.
- Pequeños anchos de banda útil.
- Limitados parámetros de control para las características de su respuesta en magnitud.

2.2. Filtro Coseno

El filtro coseno es un filtro digital que presenta una respuesta en frecuencia en forma de coseno. Su función de transferencia está dada por la ecuación (2.10):

$$H_{cos}(z) = (1 + z^{-1})/2. \quad (2.10)$$

A partir de la función de transferencia del filtro coseno es posible hallar una versión expandida del mismo. Esta expansión resultará en una compresión del eje de frecuencia, lo que generará que la respuesta en magnitud original se replique una cantidad de veces que depende del factor de expansión. La función de transferencia expandida por un factor entero N está dada como:

$$H_{cex}(z^N) = (1 + z^{-N})/2. \quad (2.11)$$

La respuesta en magnitud está dada como:

$$|H_{cex}(e^{j\omega N})| = |\cos(N\omega/2)|. \quad (2.12)$$

Los ceros del filtro coseno expandido se encuentran en los múltiplos impares de π/N y la respuesta en magnitud aparecerá replicada $N - 1$ veces entre 0 y 2π .

Ejemplo 2.2: Se presenta en la Fig. 2.5 la respuesta en magnitud para el filtro coseno y su versión expandida por un factor $N = 6$.

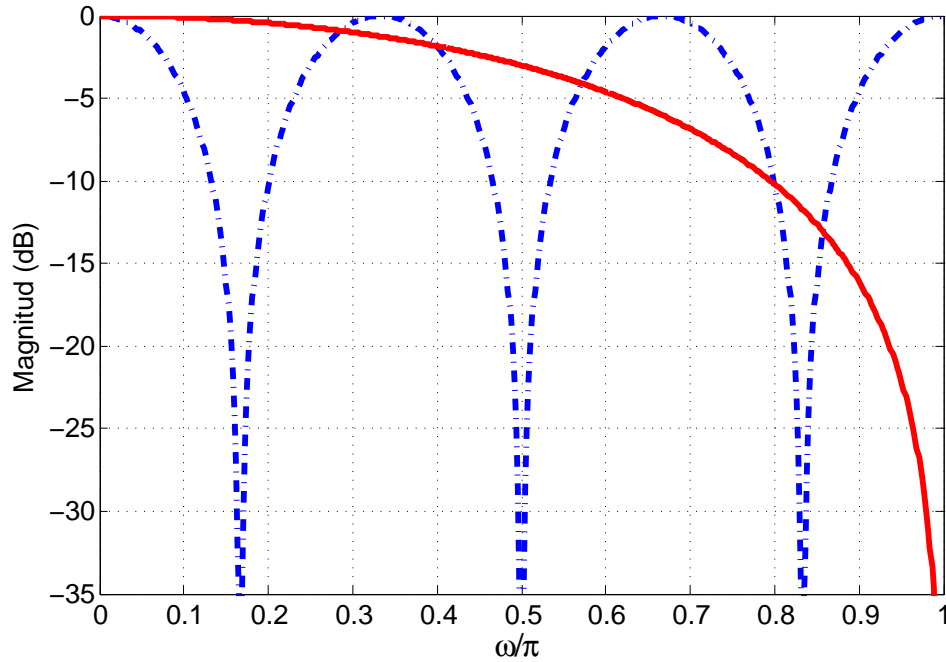


Figura 2.5: Filtro Coseno (línea continua), Filtro Coseno expandido con $N = 6$ (línea segmentada).

La ecuación (2.13) corresponde a la frecuencia a la que se encuentra el primer cero de este filtro.

$$\omega_N = \pi/N. \quad (2.13)$$

El primer cero del filtro coseno expandido puede ubicarse en la frecuencia deseada ω_d , y sabiendo que N debe corresponder a un número entero, se tiene:

$$N = \text{int}[\pi/\omega_d], \quad (2.14)$$

donde $\text{int}[\cdot]$ corresponde a la parte entera del argumento.

2.3. Filtro CIC triangular escalonado

El filtro CIC Triangular Escalonado [6], consiste en una versión con respuesta al impulso triangular [7] del filtro CIC usado para decimación por un factor M . Se requiere que el factor de decimación pueda descomponerse como el producto de dos números M_1 y M_2 .

$$M = M_1 M_2. \quad (2.15)$$

Como se presenta en [7], se consideran tres casos cuyas funciones de transferencia son presentadas a continuación.

$$X_1(z) = \left[\frac{(1 - z^{-(sN-1)})}{(1 - z^{-1})} \right] \left[\frac{(1 - z^{-(N-1)})}{(1 - z^{-s})} \right], \quad (2.16)$$

$$X_2(z) = \left[\frac{(1 - z^{-(s(N-1)+a-1)})}{(1 - z^{-1})} \right] \left[\frac{(1 - z^{-(N-1)})}{(1 - z^{-s})} \right], \quad (2.17)$$

$$X_3(z) = \left[\frac{(1 - z^{-(sN+R-1)})}{(1 - z^{-1})} \right] \left[\frac{(1 - z^{-(N-1)})}{(1 - z^{-s})} \right], \quad (2.18)$$

donde N es el número de niveles de la respuesta al impulso, s es el número de muestras por nivel y B es la cantidad de muestras en el último nivel. Para el primer caso $B = s$, para el segundo $B = a$ y $a < s$ y para el tercero $B = s + R$. Nótese que las funciones de transferencia (2.16)-(2.18) corresponden a un filtro CIC y un filtro CIC expandido. Al incluir una nueva notación para los parámetros se define la función de transferencia general para un filtro CIC triangular escalonado, como se muestra a continuación.

$$H_{CICTE}(z) = \frac{1}{M} \frac{(1 - z^{-M})}{(1 - z^{-1})} \frac{1}{N_2} \frac{(1 - z^{-N_1 N_2})}{(1 - z^{-N_1})}. \quad (2.19)$$

Usando las ecuaciones (2.15) y (2.19) se obtiene,

$$H_{CICTE}(z) = \frac{1}{M_2} \frac{(1 - z^{-M})}{(1 - z^{-M_1})} \frac{1}{M_1} \frac{(1 - z^{-M_1})}{(1 - z^{-1})} \frac{1}{N_2} \frac{(1 - z^{-N_1 N_2})}{(1 - z^{-N_1})}. \quad (2.20)$$

N_1 y N_2 se eligen de forma que se obtenga la mejor respuesta en magnitud. Es posible hacer la selección de estos valores en relación directa con M_1 y M_2 ($N_2 = M_1$, $N_1 = 1$ o $N_2 = M_2$, $N_1 = M_1$), conservando la forma de la respuesta en magnitud del filtro original, es decir, la ubicación de los ceros originales. Las funciones de transferencia para las posibles selecciones de estos parámetros son:

$$H_{CICTE}(z) = \left[\frac{1}{M_2} \frac{(1 - z^{-M})}{(1 - z^{-M_1})} \left[\frac{1}{M_1} \frac{(1 - z^{-M_1})}{(1 - z^{-1})} \right]^2 \right] \text{ para } N_2 = M_1 \text{ y } N_1 = 1, \quad (2.21)$$

$$H_{CICTE}(z) = \left[\left[\frac{1}{M_2} \frac{(1 - z^{-M})}{(1 - z^{-M_1})} \right]^2 \frac{1}{M_1} \frac{(1 - z^{-M_1})}{(1 - z^{-1})} \right] \quad \text{para } N_2 = M_2 \text{ y } N_1 = M_1 , \quad (2.22)$$

donde N_1 y N_2 corresponden a la cantidad de muestras por nivel y a la cantidad de niveles de la respuesta al impulso, respectivamente. Se observa que las ecuaciones (2.21) y (2.22) poseen dos factores: el de la derecha corresponde a un filtro CIC con factor de decimación M_1 , cuyos ceros están en los múltiplos enteros de $2\pi/M_1$, el de la izquierda corresponde a un filtro CIC expandido cuyos ceros se encuentran en $2\pi i/M$ para $i = 1, 2, \dots$, excepto en aquellos i que son múltiplos de M_2 .

Una mejor respuesta del filtro pasa-bajas se obtiene cuando N_2 es más grande, es decir, la respuesta al impulso presenta mayor número de niveles, ya que un N_2 más grande representa una copia más fiel de la respuesta original. Este efecto puede apreciarse en el ejemplo presentado en la Fig. 2.6. En la parte superior se presentan las respuestas al impulso y en magnitud de un filtro pasa-bajas de orden 23 y frecuencia de paso normalizada de 0.1. En la zona intermedia se presenta el filtro aproximado usando una constante de redondeo $r = 0.03$, lo que resulta en una respuesta al impulso con 4 niveles y que presenta peores características en magnitud con respecto al filtro original. Finalmente, se presenta el filtro aproximado usando una constante de redondeo $r = 0.06$, la cual genera solamente dos niveles y aún peores características en su respuesta en magnitud. Es posible llegar a la conclusión si se usa el mínimo número de niveles el filtro pasa-bajas será un filtro CIC, ya que poseerá un solo nivel resultante del redondeo.

La respuesta en magnitud de los filtros de las ecuaciones (2.21) y (2.22) son:

$$|H_{CICTE}(e^{j\omega})| = \left| \left[\frac{1}{M_2} \frac{\text{sen}(\omega M/2)}{\text{sen}(\omega M_1/2)} \right] \left[\frac{1}{M_1} \frac{\text{sen}(\omega M_1/2)}{\text{sen}(\omega/2)} \right] \right|^2 \quad \text{para } N_2 = M_1 \text{ y } N_1 = 1 , \quad (2.23)$$

$$|H_{CICTE}(e^{j\omega})| = \left| \left[\frac{1}{M_2} \frac{\text{sen}(\omega M/2)}{\text{sen}(\omega M_1/2)} \right]^2 \left[\frac{1}{M_1} \frac{\text{sen}(\omega M_1/2)}{\text{sen}(\omega/2)} \right] \right| \quad \text{para } N_2 = M_2 \text{ y } N_1 = M_1 . \quad (2.24)$$

Los ceros de las ecuaciones (2.23) y (2.24) se encuentran en la misma ubicación. El filtro CIC original tiene ceros en los múltiplos enteros de $2\pi/M$, mientras el filtro CIC triangu-

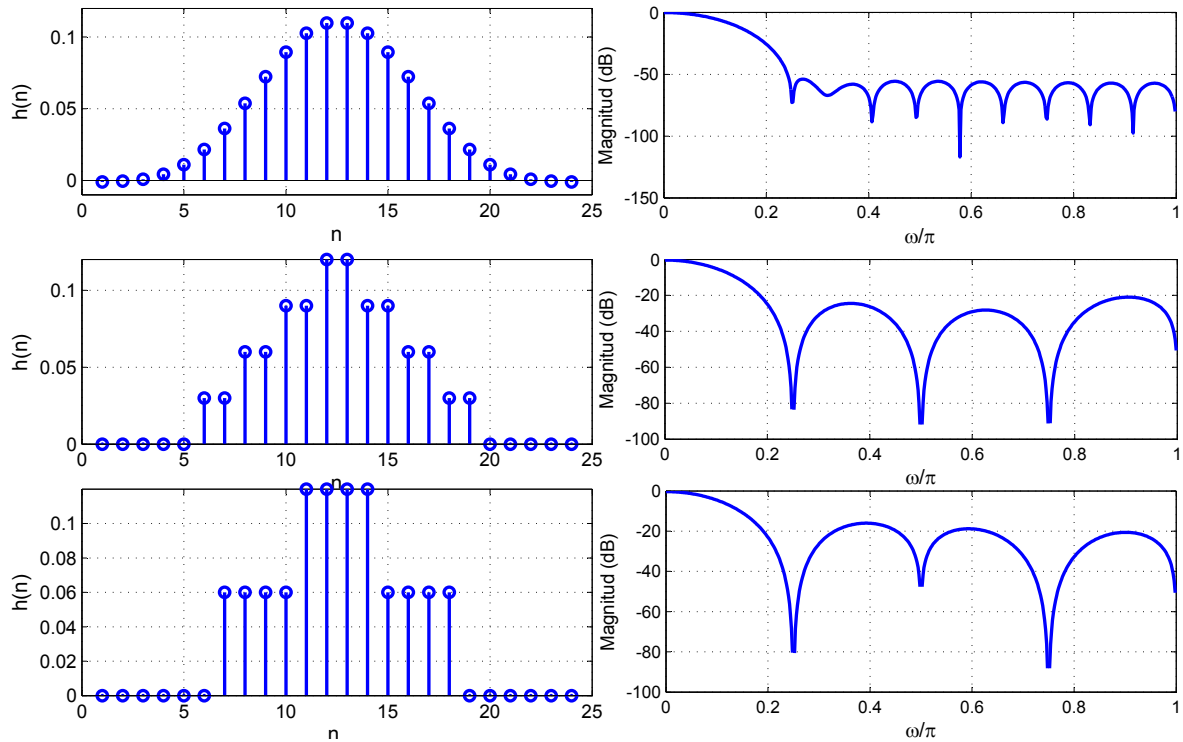


Figura 2.6: Comparación de las respuestas en magnitud para diferentes aproximaciones a respuestas escalonadas de un filtro pasa-bajas.

lar escalonado tiene ceros en los múltiplos enteros de $2\pi/M_1$ y en $2\pi i/M$ para $i = 1, 2, \dots$ excepto en aquellos i que son múltiplos de M_2 . Los ceros adicionales gracias al exponente dos en uno de los términos del filtro permiten que se obtenga una mejor respuesta en magnitud en comparación al filtro CIC original.

Ejemplo 2.3: Para ilustrar el efecto de tener una mayor cantidad de niveles en el filtro CIC triangular escalonado, se presentan a continuación las funciones de transferencia y sus respectivas respuestas en magnitud (Fig. 2.7), cuando $M = 15$. Usando la ecuación (2.21) y definiendo $N_2 = M_1 = 5$, $N_1 = 1$ y $M_2 = 3$ se obtiene:

$$|H_{CICTE}(e^{j\omega})|^2 = \left[\frac{1 \operatorname{sen}(15\omega/2)}{3 \operatorname{sen}(5\omega/2)} \right] \left[\frac{1 \operatorname{sen}(5\omega/2)}{5 \operatorname{sen}(\omega/2)} \right]^2. \quad (2.25)$$

Al usar la ecuación (2.22) con $N_2 = M_2 = 5$, $N_1 = M_1 = 3$ se obtiene:

$$|H_{CICTE}(e^{j\omega})|^2 = \left[\frac{1 \operatorname{sen}(\omega 5/2)}{5 \operatorname{sen}(\omega 3/2)} \right]^2 \left[\frac{1 \operatorname{sen}(\omega 3/2)}{3 \operatorname{sen}(\omega/2)} \right]^2. \quad (2.26)$$

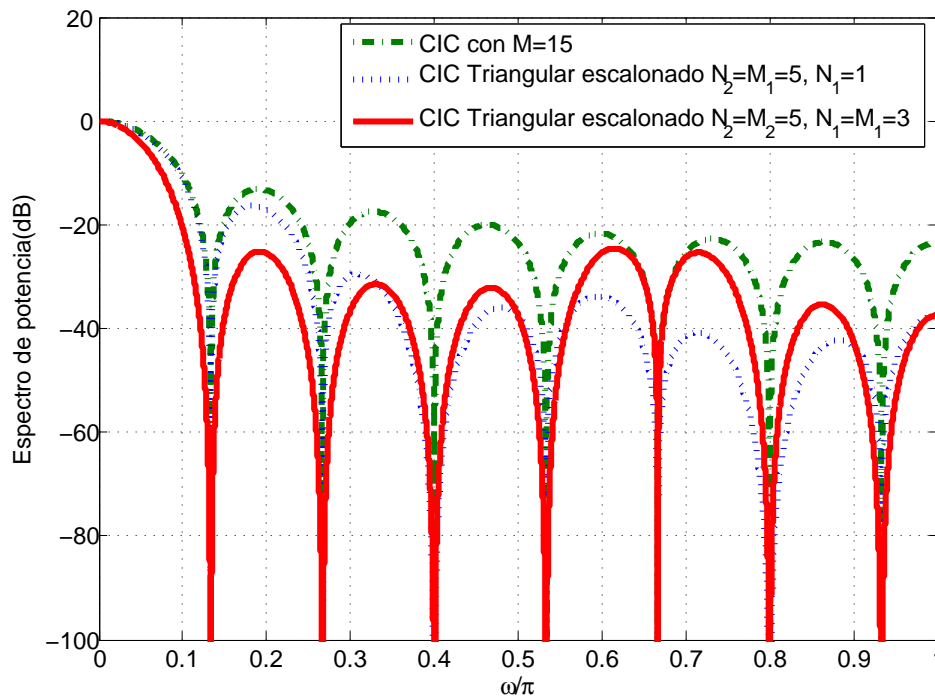


Figura 2.7: Respuestas en magnitud del filtro CIC Triangular Escalado con diferentes selecciones de los parámetros.

Bajo las condiciones anteriormente descritas, buscando que la cantidad de niveles sea la mayor posible, usando las ecuaciones (2.21) y (2.22) e incluyendo como parámetros de diseño el número de cascadas de cada uno de los términos del filtro, se obtiene la ecuación que define la forma final del filtro CIC triangular escalado de interés.

$$H_{TE}(z) = \left[\frac{1}{M_2} \frac{(1 - z^{-M})}{(1 - z^{-M_1})} \right]^{k_1} \left[\frac{1}{M_1} \frac{(1 - z^{-M_1})}{(1 - z^{-1})} \right]^{k_2}. \quad (2.27)$$

La ecuación anterior permite definir dos casos dependiendo de cual de los factores M_1 o M_2 sea mayor, para encontrar que caso provee una mejor respuesta en magnitud. Se presentará a continuación un breve análisis con respecto a los ceros del filtro (2.27).

El subfiltro CIC obtenido para el factor de decimación M_1 es representado por la ecuación (2.28):

$$H_{CIC}(z) = \frac{1}{M_1} \frac{(1 - z^{-M_1})}{(1 - z^{-1})}. \quad (2.28)$$

Para un mayor valor de M_1 se tendrá una mayor cantidad ceros distribuidos en toda la banda, lo que resulta en una mejor atenuación en la banda de rechazo. Además su

ubicación corresponde con algunos de los ceros del filtro original. La cantidad de ceros para este filtro está dada por la ecuaciones (2.29) y (2.30).

$$C_{CIC} = M_1/2 \text{ para } M \text{ par,} \quad (2.29)$$

$$C_{CIC} = (M_1 - 1)/2 \text{ para } M \text{ impar,} \quad (2.30)$$

La función de transferencia del filtro CIC expandido está dada por la ecuación (2.31).

$$H_{ECIC}(z) = \frac{1}{M_2} \frac{(1 - z^{-M})}{(1 - z^{-M_1})}. \quad (2.31)$$

$$C_{ECIC} = M/2 - M_1 \text{ para } M \text{ par,} \quad (2.32)$$

$$C_{ECIC} = (M_1 - 1)/2 - (M_1 - 1) \text{ para } M \text{ impar,} \quad (2.33)$$

Se observa que de forma contraria a como sucede con el filtro CIC para M_1 , el filtro CIC expandido tiene menor cantidad de ceros para valores grandes de este parámetro. Sin embargo, es importante mencionar que estos ceros se encuentran en las mismas posiciones que para el filtro original. Esto indica que cuando se elige $M_1 > M_2$, mayor cantidad de ceros del filtro expandido corresponderán a los ceros del filtro original. Además, aunque el filtro CIC para M_1 tenga menor cantidad de ceros, estos estarán ubicados justamente en las frecuencias donde el filtro expandido no tiene los ceros correspondientes al filtro CIC original.

Ejemplo 2.4: Se presenta un ejemplo en el que se comparan las posiciones de los ceros en las respuestas en magnitud (Figuras 2.8 y 2.9) de los filtros de las ecuaciones (2.28) y (2.31), usando $M = 21$, cuando $M_1 = 7$ y $M_2 = 3$ y cuando $M_1 = 3$ y $M_2 = 7$, se observa que en las frecuencias donde el filtro CIC expandido no tiene ceros, el filtro CIC si los posee, garantizando que se tenga una buena atenuación en la banda de rechazo.

Debido a que la cantidad del ceros del filtro CIC expandido es mayor para M_1 más pequeño y conservando la relación de duplicidad de los parámetros de cascada k_1 y k_2 en las ecuaciones (2.21) y (2.22), que garantiza la atenuación uniforme en la banda de rechazo del filtro, la función de transferencia seleccionada para este trabajo es la representada por la ecuación (2.34).

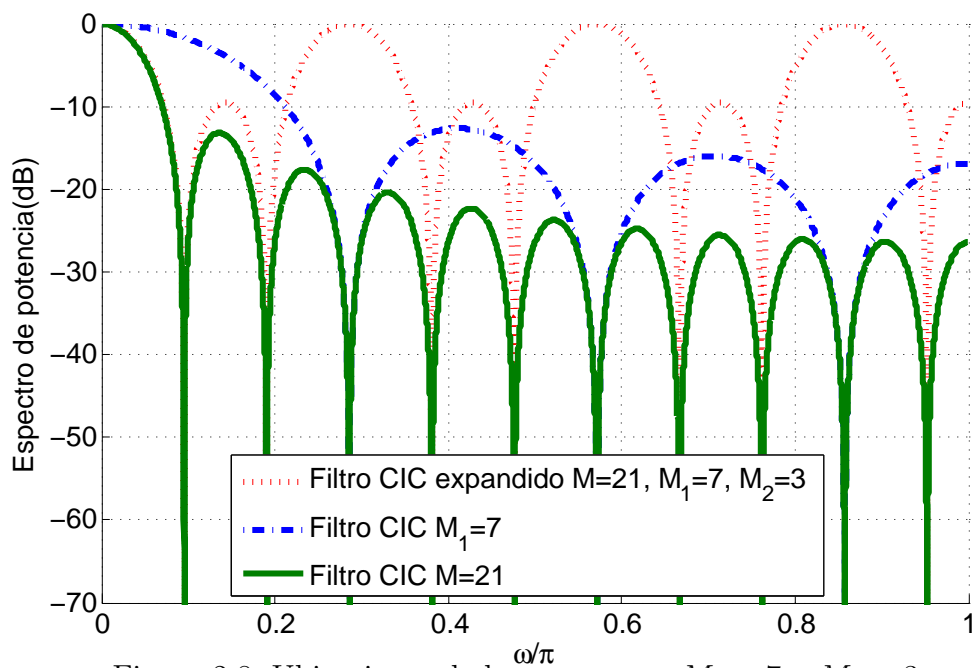


Figura 2.8: Ubicaciones de los ceros para $M_1 = 7$ y $M_2 = 3$.

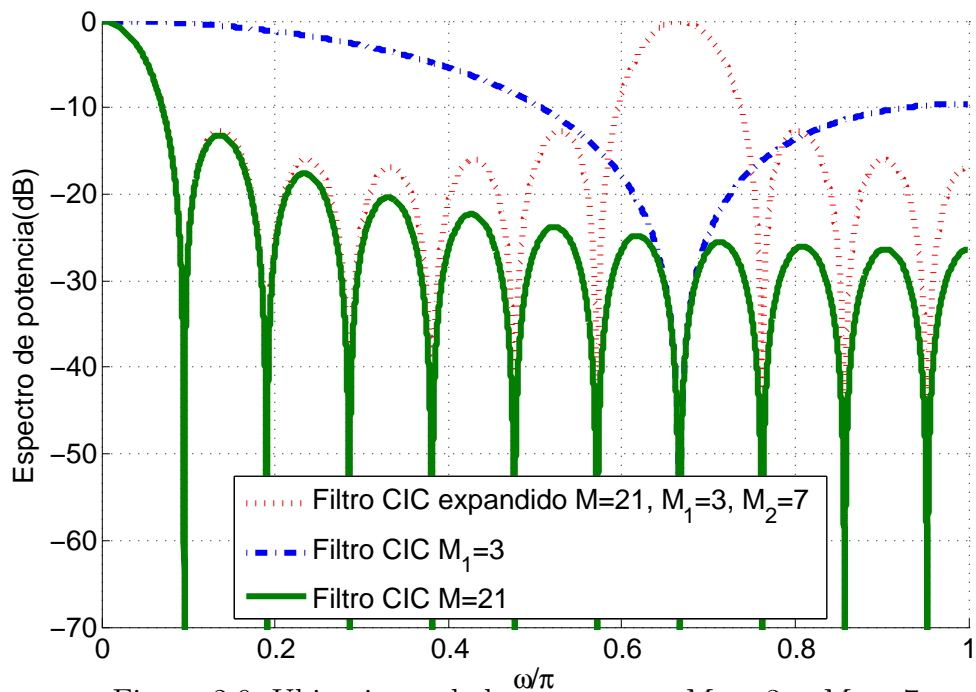


Figura 2.9: Ubicaciones de los ceros para $M_1 = 3$ y $M_2 = 7$.

$$H_{TE}(z) = \left[\frac{1}{M_2} \frac{(1 - z^{-M})}{(1 - z^{-M_1})} \right]^{k_1} \left[\frac{1}{M_1} \frac{(1 - z^{-M_1})}{(1 - z^{-1})} \right]^{k_2} \quad \text{para } M_1 < M_2, \quad (2.34)$$

donde

$$k_1 \geq 2k_2. \quad (2.35)$$

2.4. Filtro de media banda

La función de transferencia del filtro de media banda [4] está dada como:

$$H(z) = \sum_{n=0}^{2K} h[n]z^{-n}, \quad (2.36)$$

donde K es un número impar y los coeficientes $h[n]$ presentan simetría con respecto al coeficiente central $h[K]$, como lo define la ecuación (2.37).

$$h[2K - n] = h[n], \quad \text{para } n = 0, 1, \dots, 2K. \quad (2.37)$$

La longitud del filtro es un número impar

$$N = 2K + 1, \quad \text{para } K = 1, 3, 5, \dots \quad (2.38)$$

Las condiciones que debe cumplir el filtro de media banda en el dominio del tiempo se presentan a continuación.

$$h[K] = \frac{1}{2}, \quad (2.39)$$

$$h[K \pm 2r] = 0, \quad \text{para } r = 0, 1, \dots, \left\lfloor \frac{K}{2} \right\rfloor. \quad (2.40)$$

Se observa que el coeficiente central $h[K]$ es diferente de cero, mientras los coeficientes impares son cero y los coeficientes $h[K - 1]$ y $h[K + 1]$ son diferentes de 0.

Por otro lado en el dominio de la frecuencia un filtro de media banda presenta simetría definida por la ecuación (2.41):

$$H(e^{j\omega}) + H(e^{j(\pi-\omega)}) = 1, \quad (2.41)$$

lo que indica que la respuesta en magnitud del filtro de media banda presenta simetría de la banda de paso y la banda de rechazo con respecto a la frecuencia de media banda $\pi/2$. Debido a esto los rizados en la banda de paso y rechazo son iguales como se presenta

en la Fig. 2.10. Un filtro de media banda, por sus características en frecuencia, es usado como filtro de decimación por un factor de 2, debido a que la máxima componente de frecuencia permitida se encuentra en $\pi/2$, garantizando que sea eliminada por el filtro de media banda.

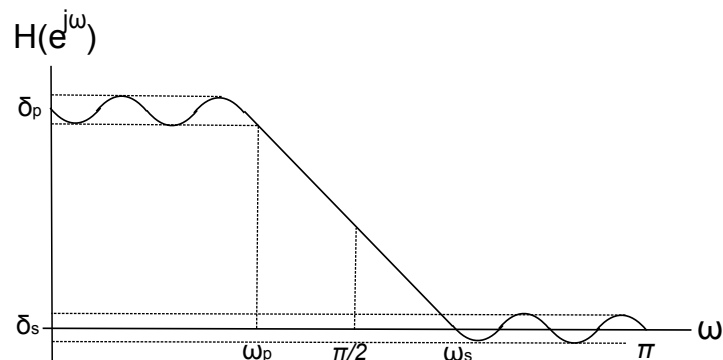
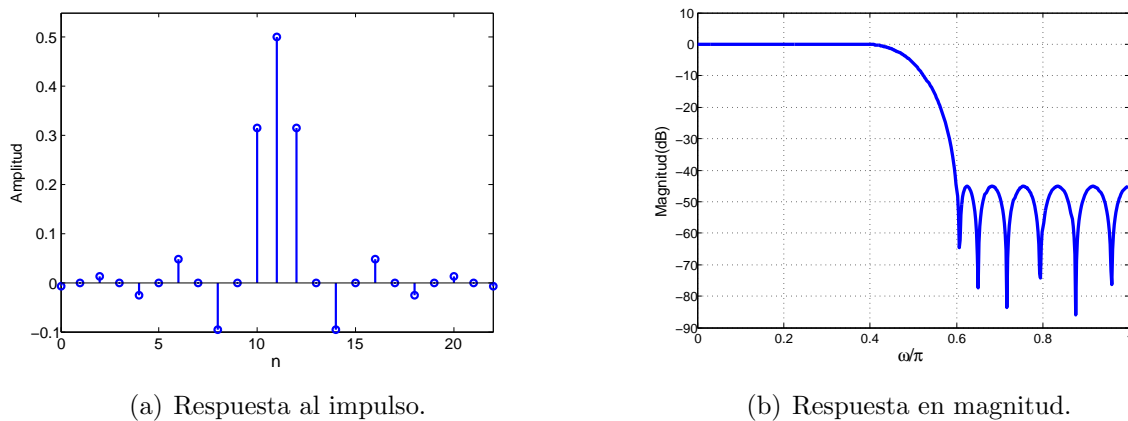


Figura 2.10: Características de la respuesta en magnitud del filtro de media banda.

Ejemplo 2.5: Se presenta a continuación un ejemplo para el diseño de un filtro de media banda cuyos parámetros son: mínima atenuación en la banda $a_s = 40dB$, frecuencia de rechazo $\omega_s = 0.6$, frecuencia de paso $\omega_p = 1 - \omega_s = 0.4$. La correspondiente respuesta al impulso y su respectiva respuesta en magnitud se presentan en la Fig. 2.11.



(a) Respuesta al impulso.

(b) Respuesta en magnitud.

Figura 2.11: Filtro de media banda.

2.5. Interpolador lineal

Como ha sido mencionado anteriormente, en el dominio de la frecuencia un filtro interpolador es un filtro pasa-bajas que se encarga de eliminar las imágenes introducidas por el proceso de sobremuestreo. En el dominio del tiempo este proceso consiste en calcular valores interpolados para reemplazar las muestras de valor cero introducidas previamente, usando muestras adyacentes. En general, un interpolador lineal, con factor de interpolación L , tiene la siguiente respuesta al impulso:

$$h_{IL}[n] = \begin{cases} 1 - \frac{|n|}{L}, & |n| \leq (L - 1), \\ 0, & \text{otros casos.} \end{cases} \quad (2.42)$$

La transformada z de la ecuación (2.42) está dado por

$$\begin{aligned} H_{IL}(z) &= \frac{1}{L} \sum_{n=-(L-1)}^{(L-1)} \left(1 - \frac{|n|}{L}\right) z^{-n} \\ &= \frac{1}{L} \left[\left(1 - \frac{L-1}{L}\right) z^{-(L-1)} + \dots + 1 + \dots + \left(1 - \frac{L-1}{L}\right) z^{(L-1)} \right], \end{aligned} \quad (2.43)$$

$$H_{IL}(z) = \frac{1}{L} \left[1 + \left(1 - \frac{L-1}{L}\right) (z^{-(L-1)} + z^{(L-1)}) + \dots + \left(1 - \frac{L-2}{L}\right) (z^{-(L-2)} + z^{(L-2)}) \right]. \quad (2.44)$$

Como se observa en la ecuación (2.42) un interpolador es un filtro FIR no causal. Sin embargo, para su implementación causal, es suficiente retrasar la salida $L - 1$ veces. Por otro lado, es claro que un interpolador lineal aumenta la frecuencia de muestreo por un factor entero, ya que introduce nuevas muestras interpoladas, siendo un elemento de alteración de frecuencia de muestreo cuya complejidad computacional depende de su factor de interpolación.

Ejemplo 2.6: Se presenta como ejemplo un interpolador lineal con $L = 4$. Su ecuación de diferencias corresponde a la ecuación (2.45). La correspondiente respuesta en magnitud se muestra en la Fig. 2.12.

$$\begin{aligned} y[n] &= \frac{1}{4}x[n] + \frac{1}{4}(x[n-3] + x[n+3]) \\ &\quad + \frac{1}{2}(x[n-2] + x[n+2]) + \frac{3}{4}(x[n-1] + x[n+1]). \end{aligned} \quad (2.45)$$

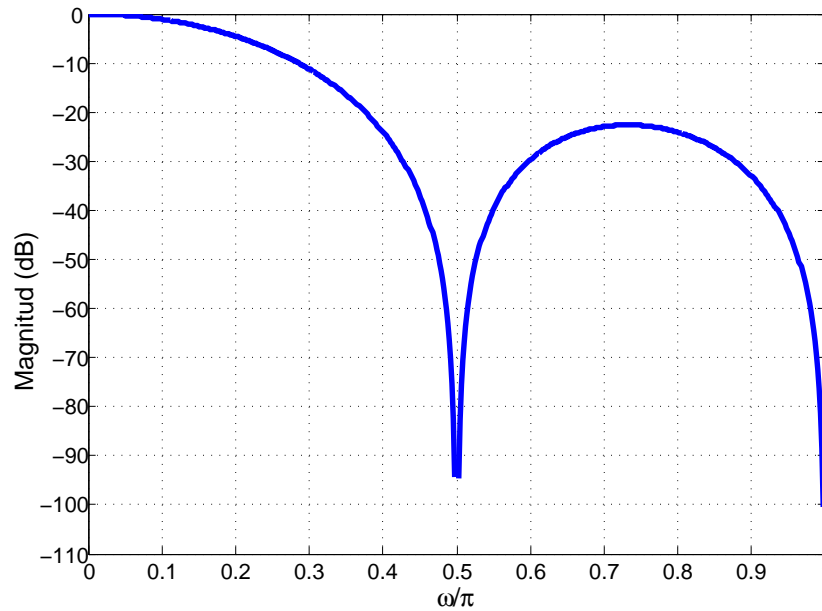


Figura 2.12: Características de la respuesta en magnitud del interpolador lineal con $L = 4$.

2.6. Interpolador Farrow

En la Fig. 2.13 se presenta el modelo analógico del filtro de interpolación. Los parámetros n_l y μ_l se usan para determinar el instante de tiempo t_l para la $l^{\text{ésima}}$ muestra de salida.

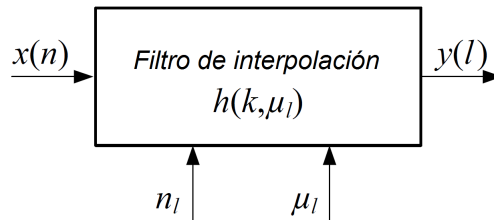


Figura 2.13: Diagrama de bloques para el filtro de interpolación.

$$t_l = (n_l + \mu_l)T_{in}. \quad (2.46)$$

Para un dado t_l , estos parámetros se calculan usando las siguientes ecuaciones:

$$n_l = \lfloor t_l/T_{in} \rfloor, \quad (2.47)$$

$$\mu_l = t_l/T_{in} - \lfloor t_l/T_{in} \rfloor. \quad (2.48)$$

La salida se calcula realizando la convolución definida por la ecuación (2.49).

$$y(l) = \sum_{k=-N/2}^{N/2-1} x(n_l - k)h(k, \mu_l), \quad (2.49)$$

donde N es la longitud y $h(k, \mu_l)$ es la respuesta al impulso discreta del filtro de interpolación.

A partir del modelo de analógico para el interpolador mostrado en la Fig. 2.14, es posible comprender la implementación de los filtros de interpolación basados en polinomios.

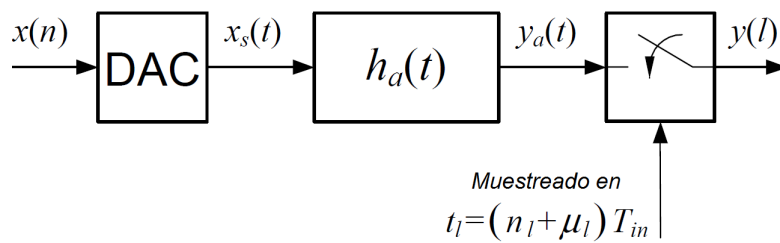


Figura 2.14: Modelo analógico de un filtro de interpolación.

En la Fig. 2.14, la señal $y(l)$ se obtiene al muestrear $y_a(t)$ en t_l resultanto en:

$$y(l) = y_a(t_l) = \sum_{k=-N/2}^{N/2-1} x(n_l - k)h_a((k + \mu_l)T_{in}), \quad (2.50)$$

donde $h_a(t)$ es diferente de cero en el intervalo $-NT_{in} \leq t \leq NT_{in}/2$. Al comparar las ecuaciones (2.49) y (2.50) se encuentra que las respuestas al impulso analógica y digital están relacionadas como se presenta a continuación.

$$h(k, \mu_l) = h_a((k + \mu_l)T_{in}) \quad \text{para } k = -N/2, -N/2 + 1, \dots, N/2 - 1. \quad (2.51)$$

Para que la ecuación (2.51) sea causal $h_a(t)$ debe ser retrasada $NT_{in}/2$, obteniendo:

$$y(l) = \sum_{k=0}^{N-1} x(n_l - k + N/2)h_a((k + \mu_l - N/2)T_{in}). \quad (2.52)$$

De esta forma la respuesta al impulso es expresada en intervalos de longitud T_{in} por medio de polinomios:

$$h_a((k + \mu_l)T_{in}) = \sum_{m=0}^M \hat{c}_m(k)\mu_l^m, \quad k = -N/2, \dots, N/2 - 1, \quad (2.53)$$

donde los $\hat{c}_m(k)$ son los coeficientes y M es el grado de los polinomios.

Reemplazando la ecuación (2.53) en la ecuación (2.52) se obtiene:

$$v_m(n_l) = \sum_{k=0}^{N-1} x(n_l - k + N/2) \hat{c}_m(k - N/2). \quad (2.54)$$

Se observa en la ecuación (2.54) que las señales $v_m(n_l)$ corresponden a la salida de $M + 1$ filtros FIR cuyas funciones de transferencia son:

$$\hat{C}_m(z) = \sum_{k=0}^{N-1} \hat{c}_m(k - N/2) z^{-k}. \quad (2.55)$$

En resumen las muestras de salida pueden ser generadas basadas en la muestras existentes mediante $M + 1$ filtros FIR. Luego se multiplican por μ_l^m y posteriormente son sumadas. La estructura resultante es conocida como la estructura Farrow y se presenta en la Fig. 2.15.

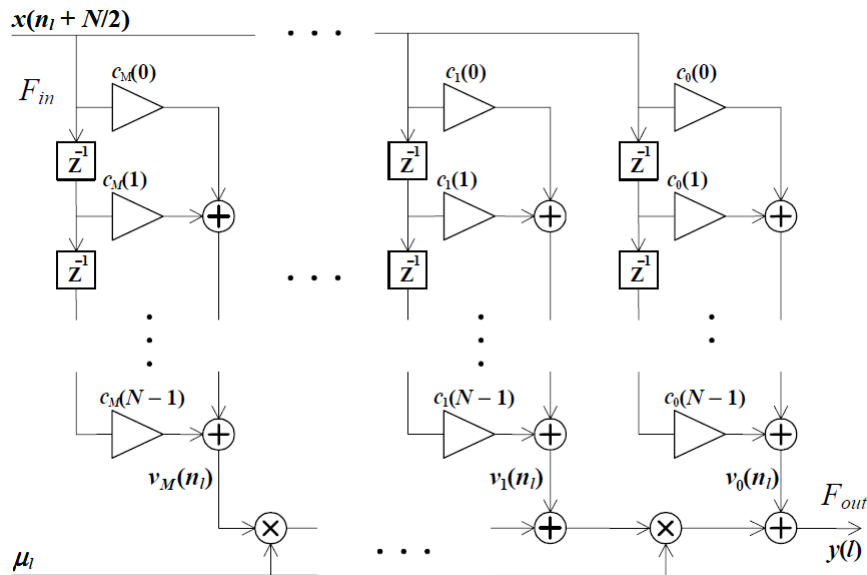


Figura 2.15: Estructura Farrow.

Ejemplo 2.7: Se presenta a continuación un ejemplo de un interpolador de Lagrange para un factor racional $D = 0.5$. Se usa la estructura Farrow con tres ramas y dos filtros de tercer orden y un tercer filtro que equivale a un retardo de segundo orden. La estructura se presenta en la Fig. 2.16 y su respuesta en magnitud en la Fig. 2.17. Las funciones de transferencia de los filtros usados son:

$$C_0(z) = z^{-2}, \quad (2.56)$$

$$C_1(z) = -\frac{1}{4} + \frac{5}{4}z^{-1} - \frac{3}{4}z^{-2} - \frac{1}{4}z^{-3}, \quad (2.57)$$

$$C_2(z) = \frac{1}{4} - \frac{1}{4}z^{-1} - \frac{1}{4}z^{-2} + \frac{1}{4}z^{-3}. \quad (2.58)$$

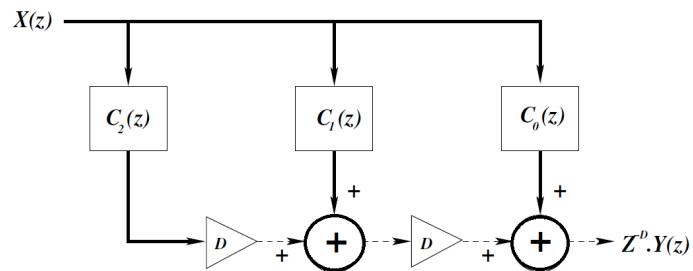


Figura 2.16: Interpolador de Lagrange usando la estructura Farrow.

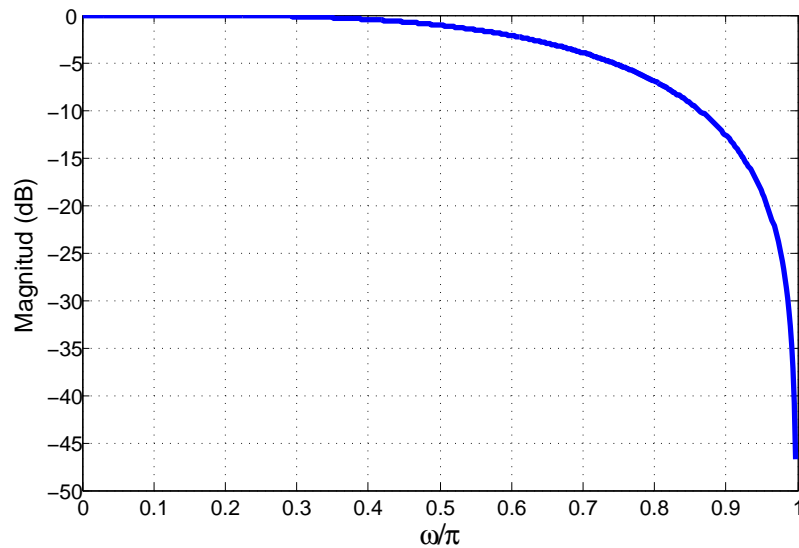


Figura 2.17: Respuesta en magnitud del interpolador basado en la estructura Farrow.

Capítulo 3

Métodos para la realización eficiente de descomposición polifase

En este capítulo se presentan algunos métodos para la descomposición polifase que permiten reducir la frecuencia de funcionamiento de los filtros.

3.1. Realización polifase

La realización polifase consiste en utilizar la descomposición polifase en el filtro *anti-aliasing/anti-imaging* de la Fig. 1.10(b). Como se observa en la Fig. 1.10(b), esta estructura permitiría la descomposición polifase del filtro en M ramas o fases cambiando la frecuencia de muestreo de todos los filtros a una frecuencia reducida M veces. Sin embargo, el sobremuestreador incrementa la frecuencia desde la entrada. La otra opción es realizar la descomposición del filtro en L ramas y reducir la frecuencia de funcionamiento L veces. Estas dos opciones corresponden a la decimación polifase e interpolación polifase, respectivamente. Esta estructura presenta mayor eficiencia dependiendo de cual de los dos factores el de decimación o el de interpolación sea más grande.

La estructura resultante para conversión racional de frecuencia de muestreo usando la decimación polifase se presenta en la Fig. 3.1. Se observa que se produce un aumento de la frecuencia de muestreo en la entrada y luego se reduce M veces. Si $M > L$ se logrará una reducción de la frecuencia de funcionamiento de cada filtro.

Cuando se usa la interpolación polifase se obtiene la estructura mostrada en la Fig. 3.2. Para este caso los filtros funcionan en la frecuencia de entrada. Si se considera que $M > L$, entonces la estructura basada en interpolación resulta menos eficiente que la basada en

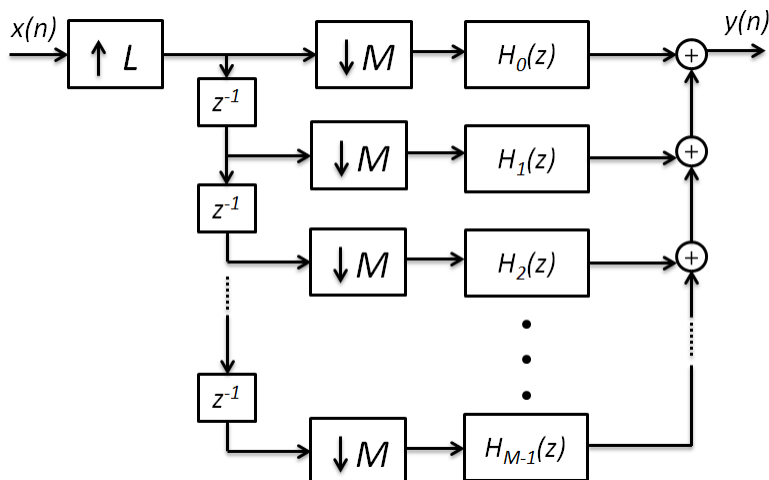


Figura 3.1: Conversión racional usando decimación polifase.

decimación, ya que no se reduce la frecuencia de funcionamiento de los filtros polifase. Sin embargo, para el caso en que $M < L$, resulta más eficiente la estructura basada en interpolación polifase.

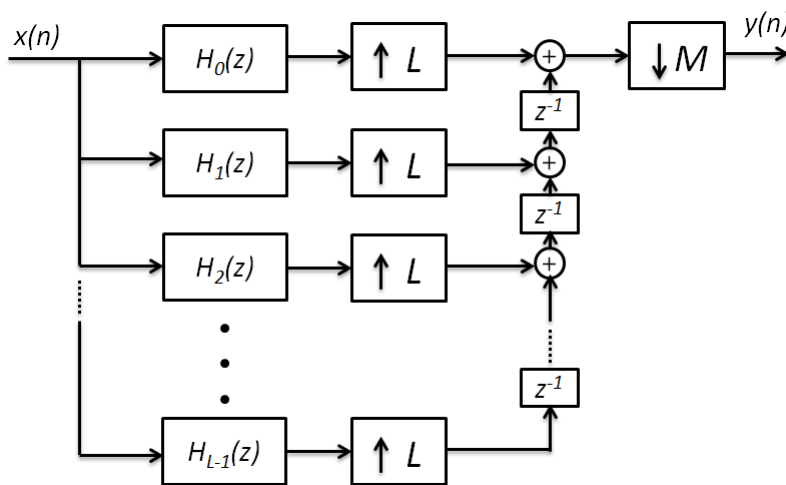


Figura 3.2: Conversión racional usando interpolación polifase.

3.2. Realización eficiente de conversión racional de frecuencia de muestreo [3]

Retomando la estructura para conversión de frecuencia de muestreo por un factor racional de la Fig. 1.10(b) y usando la estructura polifase transpuesta de la Fig. 1.17(b) para L ramas del filtro $H(z)$, dada por la ecuación (3.1), se obtiene la estructura de la Fig. 3.3.

$$H(z) = \sum_{k=0}^{L-1} z^{-k} H_k(z^L). \quad (3.1)$$

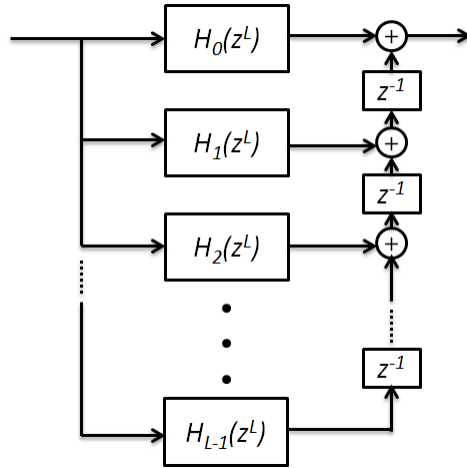


Figura 3.3: Descomposición polifase del filtro *anti-aliasing/anti-imaging*.

Usando la cuarta identidad multi-razón es posible mover el sobremuestreador a cada rama obteniendo la estructura de la Fig. 3.4. A continuación se utiliza la primera identidad para mover el bloque submuestreador a cada rama después de los respectivos retardos, obteniendo la estructura de la Fig. 3.5.

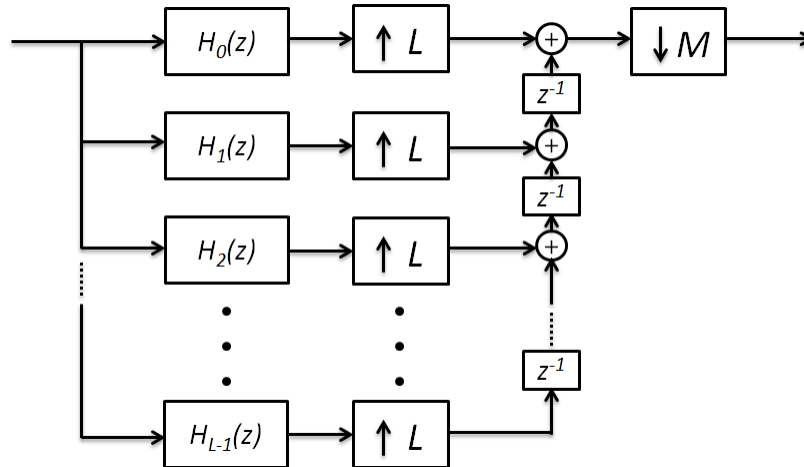


Figura 3.4: Primer paso para la derivación de la estructura eficiente de [3].

Si L y M son mutuamente primos o primos relativos se tiene,

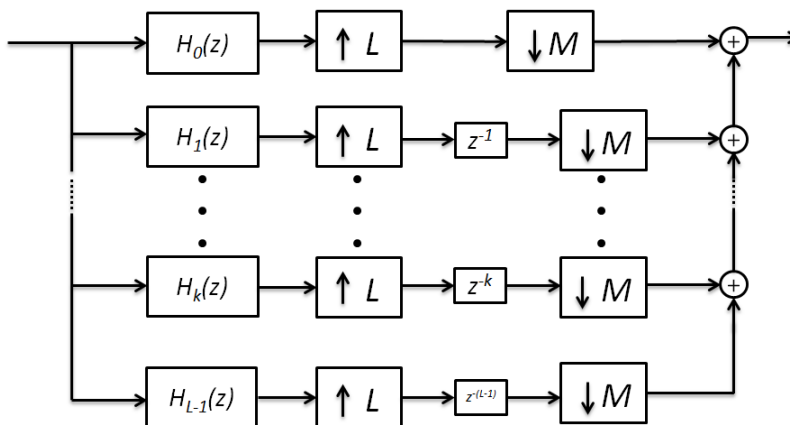


Figura 3.5: Segundo paso para la derivación de la estructura eficiente de [3].

$$\mu M - \lambda L = 1, \tag{3.2}$$

donde μ y λ son números enteros.

Por la propiedad anterior el retardo z^{-k} se puede descomponer como se presenta en la ecuación (3.3).

$$z^{-k} = z^{-k(\mu M - \lambda L)} \quad \text{para } 0 \leq k \leq L - 1. \tag{3.3}$$

Aplicando esta propiedad en la rama k se obtienen los pasos ilustrados en la Fig. 3.6.

Usando el desarrollo anterior en cada rama y eliminando los elementos no causales resultantes de la identidad (3.2), mediante la introducción de retardos en la entrada, se obtiene la estructura de la Fig. 3.7. Dado que M y L son primos relativos, como se ha presentado con en el capítulo 1, se pueden intercambiar los bloques de submuestreo y sobremuestreo, permitiendo usar la descomposición polifase de $H_k(z)$ en M fases o ramas, obteniendo la estructura definitiva de la Fig. 3.8.

Ejemplo 3.1: Se presenta en la Fig. 3.9 un ejemplo de la estructura para $L = 2$ y $M = 3$. Se observa que todos los filtros resultantes de las dos descomposiciones polifase funcionan en la más baja frecuencia posible que corresponde a la frecuencia de muestreo de entrada reducida por un factor de 3.

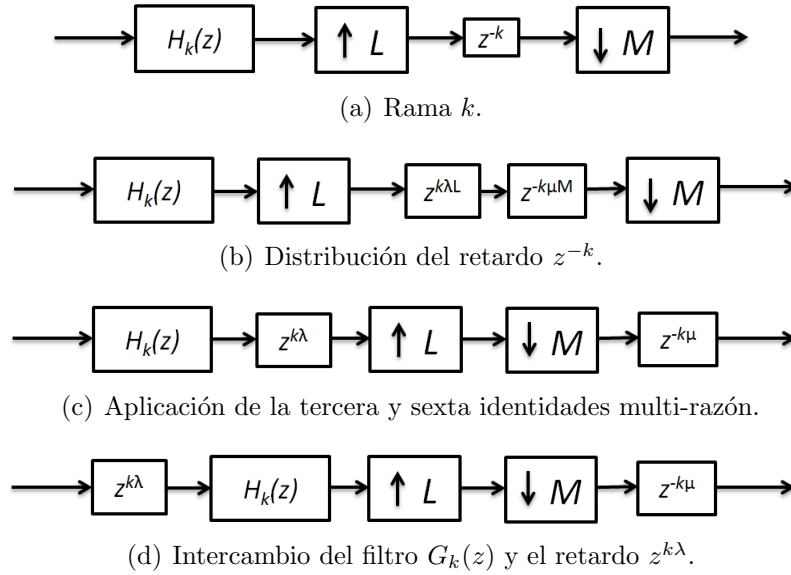


Figura 3.6: Desarrollo de la k -ésima rama.

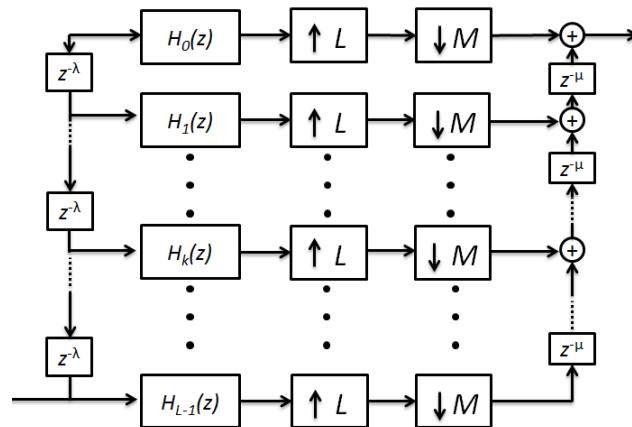


Figura 3.7: Cuarto paso para la derivación de la estructura eficiente de [3].

3.3. Implementación de filtros FIR de fase lineal para conversión racional de frecuencias de muestreo usando la simetría de los coeficientes [9]

Este método se basa en determinar la relación entre la entrada y salida del sistema en forma matricial, aprovechando la simetría de los coeficientes de filtro FIR encargado de eliminar el *imaging* y el *aliasing* resultantes de los procesos de interpolación y decimación respectivamente.

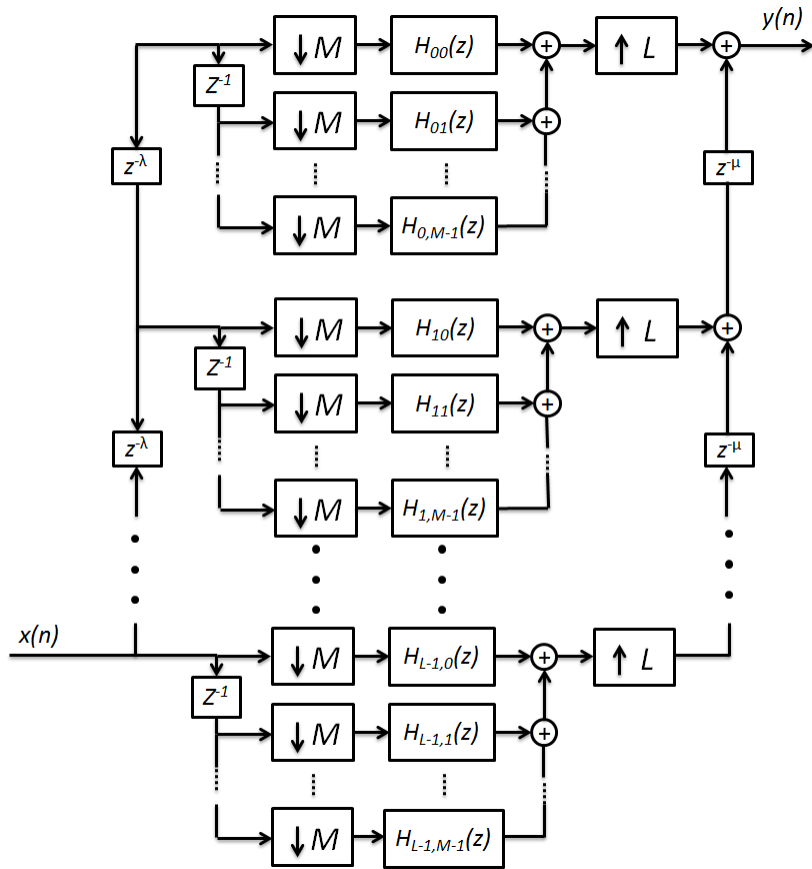


Figura 3.8: Estructura final.

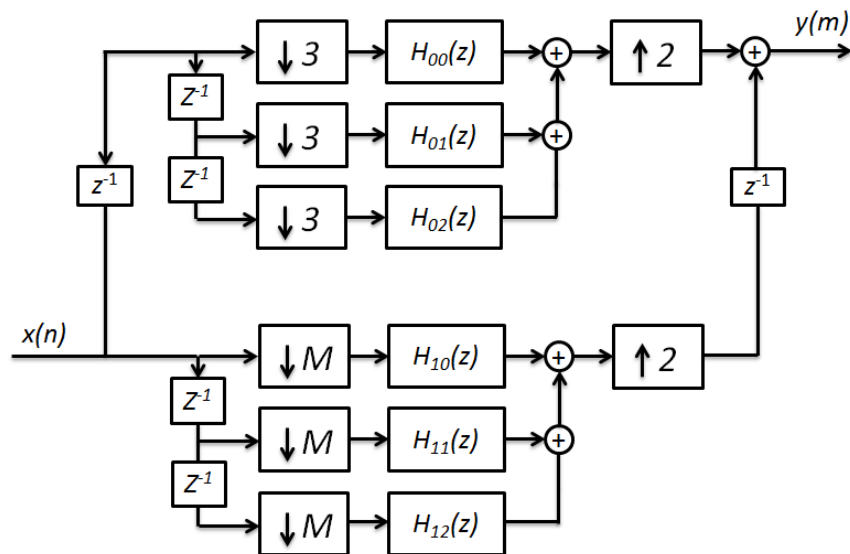


Figura 3.9: Ejemplo de la estructura descrita en [3] para $L = 2$ y $M = 3$.

Se considera que el filtro $H(z)$ en la Fig. 1.10(b) es un filtro FIR de orden N que posee la simetría dada por la ecuación (3.5).

$$H(z) = \sum_{k=0}^N h_k z^{-k}. \quad (3.4)$$

$$h_k = h_{N-k}, \quad (3.5)$$

donde $k = 0, 1, \dots, N$, solamente $\lfloor N/2 \rfloor + 1$ de los coeficientes son diferentes.

Las relaciones entre las señales de entrada y salida en la Fig. 1.10(b) están definidas por las ecuaciones presentadas a continuación.

$$u[\eta] = \begin{cases} x \left[\frac{\eta}{L} \right], & \text{para } \eta = 0, L, 2L, \dots \\ 0, & \text{en otro caso.} \end{cases} \quad (3.6)$$

$$w[\eta] = \sum_{k=0}^N h_k u[\eta - k], \quad (3.7)$$

$$y[n] = w[Mn], \quad (3.8)$$

Al combinar las ecuaciones (3.6), (3.7) y (3.8) se tiene:

$$y[n] = w[Mn] = \sum_{k=0}^N h_k u[Mn - k] = \sum_{k=0}^N h_k x \left[\frac{nM - k}{L} \right], \quad (3.9)$$

$$y[n] = \sum_{k=0}^N h_k x \left[\frac{nM - k}{L} \right]. \quad (3.10)$$

donde $x[m]$ existe cuando $\left(\frac{nM - k}{L} \right)$ tiene un valor entero.

Para obtener la salida $(n + KL)$, donde K es un entero se tiene:

$$y[n + KL] = \sum_{k=0}^N h_k x \left[\frac{M(n + KL) - k}{L} \right] = \sum_{k=0}^N h_k x \left[\frac{Mn - k}{L} + MK \right]. \quad (3.11)$$

Se observa de la ecuación (3.11), que la señal de salida con un retardo KL corresponde a la señal de salida desplazada KM . Es posible obtener un conjunto de L ecuaciones que relacionan la entrada y la salida:

$$y[n + \ell] = \sum_{k=0}^N h_k x \left[\frac{M(n + \ell) - k}{L} \right], \quad \text{para } \ell = 0, 1, \dots, L - 1. \quad (3.12)$$

Solamente cuando $\left(\frac{M(n+\ell)-k}{L} \right)$ tiene un valor entero el término correspondiente entrará en la sumatoria. Es posible escribir de una forma alternativa el argumento de la ecuación (3.12), como se presenta a continuación.

$$\frac{M(n + \ell) - k}{L} = \frac{Mn}{L} - \frac{k - \ell M}{L}. \quad (3.13)$$

Si se elige que el parámetro n tenga los valores $0, L, 2L, \dots$, se puede definir un parámetro m dado por la ecuación (3.14), que tomará valores enteros.

$$m = \frac{Mn}{L} \quad \text{para } \ell = 0, M, 2M, \dots \quad (3.14)$$

El término de la derecha de la ecuación (3.13) también debe tomar un valor entero. A su vez debe existir un valor K_ℓ que cumpla con esta condición, resultando en:

$$k = \text{mod}(\ell M, L) + K_\ell, \quad (3.15)$$

donde

$$\text{mod}(\ell M, L) = K_\ell L. \quad (3.16)$$

Sabiendo que k toma valores enteros entre 0 y k , el parámetro K_ℓ puede tomar valores enteros entre 0 y K_ℓ^{max} , los cuales pueden ser hallados usando la ecuación (3.17).

$$K_\ell = \left\lfloor \frac{k - \text{mod}(\ell M, L)}{L} \right\rfloor \quad \text{para } 0 \leq K_\ell \leq K_\ell^{max}, \quad (3.17)$$

donde

$$K_\ell^{max} = \left\lfloor \frac{N - \ell M}{L} \right\rfloor + \left\lfloor \frac{\ell M}{L} \right\rfloor. \quad (3.18)$$

Ahora es posible reescribir la ecuación (3.12), obteniendo la ecuación (3.19). Además incluyendo la relación entre M y N dada por la ecuación (3.14), se obtiene la representación matricial de la ecuación (3.20).

$$y[n + \ell] = \sum_{K_\ell=0}^{K_\ell} h_{\ell M + (K_\ell - \lfloor \frac{\ell M}{L} \rfloor)L} x \left[\frac{Mn}{L} + \left\lfloor \frac{\ell M}{L} \right\rfloor - K_\ell \right]. \quad (3.19)$$

$$y[n + \ell] = \begin{bmatrix} h_{\ell M - \lfloor \frac{\ell M}{L} \rfloor L} \\ h_{\ell M + L - \lfloor \frac{\ell M}{L} \rfloor L} \\ \vdots \\ h_{\ell M + \lfloor \frac{N - \ell M}{L} \rfloor L} \end{bmatrix}^T \begin{bmatrix} x \left[m + \lfloor \frac{\ell M}{L} \rfloor \right] \\ x \left[m - 1 + \lfloor \frac{\ell M}{L} \rfloor \right] \\ \vdots \\ x \left[m - \lfloor \frac{N - \ell M}{L} \rfloor \right] \end{bmatrix} \quad \ell = 0, 1, \dots, L - 1. \quad (3.20)$$

La ecuación anterior puede escribirse en forma compacta como se presenta a continuación.

$$\mathbf{Y}_{n,L} = \mathbf{H}_{L \times (p+q+1)} \mathbf{X}_{m+p,m-q}. \quad (3.21)$$

La matriz \mathbf{H} se conoce como la matriz de entrada-salida. Está definida por la ecuación (3.22) y los vectores \mathbf{X} y \mathbf{Y} que corresponden a salidas consecutivas y a entradas consecutivas en orden descendente, respectivamente.

$$\mathbf{H}_{L \times (p+q+1)} = \begin{bmatrix} h_{-pL} & \dots & h_{-L} & h_0 & h_L & \dots & h_{(q-1)L} & h_{qL} \\ h_{M-pL} & \dots & h_{M-L} & h_M & h_{M+L} & \dots & h_{M+(q-1)L} & h_{M+qL} \\ h_{2M-pL} & \dots & h_{2M-L} & h_{2M} & h_{2M+L} & \dots & h_{2M+(q-1)L} & h_{2M+qL} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ h_{(L-1)M-pL} & \dots & h_{(L-1)M-L} & h_{(L-1)M} & h_{(L-1)M+L} & \dots & h_{(L-1)M+(q-1)L} & h_{(L-1)M+qL} \end{bmatrix}, \quad (3.22)$$

$$\mathbf{Y}_{n,L} = \left[y[n], y[n+1], y[n+2], \dots, y[n+L-1] \right]^T, \quad (3.23)$$

$$\mathbf{X}_{m+p,m-q} = \left[x[m+p], x[m+p-1], x[m+p-2], \dots, x[m-q] \right]^T, \quad (3.24)$$

donde

$$q = \left\lfloor \frac{N}{L} \right\rfloor, \quad (3.25)$$

$$p = \left\lfloor \frac{(L-1)M}{L} \right\rfloor, \quad (3.26)$$

Antes de presentar las propiedades de la matriz de entrada-salida, se presentará de forma resumida la descomposición polifásica con una notación alternativa que será usada a lo largo de esta descripción para el filtro $H(z)$.

$$H(z) = \sum_{\mu=0}^{L-1} z^{-\mu} H_{\mu}(z^L), \quad (3.27)$$

donde

$$H_\mu = \sum_{n=0}^{N_\mu} h_{\mu+nL} z^{-n} = \mathbf{h}_\mu [1 \quad z^{-1} \quad z^{-2} \quad \dots \quad z^{-N_\mu}], \quad (3.28)$$

$$\mathbf{h}_\mu = [h_\mu \quad h_{\mu+L} \quad h_{\mu+2L} \quad \dots \quad h_{\mu+N_\mu L}], \quad (3.29)$$

$$N_\mu = \left\lfloor \frac{N - \mu}{L} \right\rfloor \quad \text{para} \quad \mu = 0, 1, \dots, L - 1. \quad (3.30)$$

3.3.1. Propiedades de la matriz de entrada-salida $\mathbf{H}_{L \times (p+q+1)}$ [9]

Se presentan en esta sección algunas propiedades importantes de la matriz de entrada-salida que permitirán obtenerla a partir de las componentes polifásicas del filtro $H(z)$.

Propiedad I

Como puede observarse de la ecuación (3.22), la $p^{\text{ésima}}$ columna está dada como:

$$h^p = [h_0 \quad h_M \quad h_{2M} \quad \dots \quad h_{(L-1)M}]^T. \quad (3.31)$$

Propiedad II

La fila $\ell^{\text{ésima}}$ contiene la $\mu_\ell^{\text{ésima}}$ componente polifásica de $H(z)$ que posee los coeficientes $h_{\text{mod}(\ell M, L) + rL}$ para $r = 0, 1, \dots, N_{\text{mod}(\ell M, L)}$. Existe un $R \in 0, 1, \dots, N_{\text{mod}(\ell M, L)}$ para el cual $\text{mod}(\ell M, L) = \ell M$, donde

$$R = \left\lfloor \frac{\ell M}{L} \right\rfloor, \quad (3.32)$$

lo que significa que el $R^{\text{ésimo}}$ coeficiente está en la columna P . Es posible concluir, gracias a la propiedad I, que existen $p - R$ coeficientes cero antes del primer coeficiente diferente de cero en la $\ell^{\text{ésima}}$ fila. Usando la notación de un exponente para clarificar la cantidad de coeficientes con valor cero antes (a) y después (d) de la \mathbf{h}_{μ_ℓ} componente polifásica en la $\ell^{\text{ésima}}$ fila.

$$\mu_\ell^{(a)} = p - \left\lfloor \frac{\ell M}{L} \right\rfloor, \quad (3.33)$$

$$\mu_\ell^{(d)} = q - \left\lfloor \frac{(N - \ell M)}{L} \right\rfloor. \quad (3.34)$$

La matriz de entrada-salida puede calcularse de la siguiente manera:

$$\mathbf{H}_{L \times (p+q+1)} = \begin{bmatrix} \mathbf{h}_0^{(H)} \\ \mathbf{h}_1^{(H)} \\ \dots \\ \mathbf{h}_\ell^{(H)} \\ \dots \\ \mathbf{h}_{L-1}^{(H)} \end{bmatrix}, \quad (3.35)$$

$$\mathbf{h}_\ell^{(H)} = [\mathbf{0}_{1, \mu_\ell^{(a)}} \quad \mathbf{h}_{\mu_\ell} \quad \mathbf{0}_{1, \mu_\ell^{(d)}}], \quad (3.36)$$

donde \mathbf{h}_{μ_ℓ} es el vector que contiene los coeficientes que pertenecen a la $\mu_\ell^{\text{ésima}}$ componente polifásica.

Ejemplo 3.2: Para clarificar el cálculo de la matriz de entrada-salida se presenta un ejemplo que corresponde a la conversión de frecuencia de muestreo por un factor $\frac{L}{M} = \frac{2}{3}$ usando un filtro de orden $N = 11$. En este caso se tienen los siguientes parámetros:

$$q = \left\lfloor \frac{11}{2} \right\rfloor = 5, \quad (3.37)$$

$$p = \left\lfloor \frac{(2-1)3}{2} \right\rfloor = 1, \quad (3.38)$$

$$\mu_0^{(a)} = 1, \quad (3.39)$$

$$\mu_0^{(d)} = 5 - \left\lfloor \frac{11}{2} \right\rfloor = 0, \quad (3.40)$$

$$\mu_1^{(a)} = 1 - \left\lfloor \frac{3}{2} \right\rfloor = 0, \quad (3.41)$$

$$\mu_1^{(d)} = 5 - \left\lfloor \frac{11-3}{2} \right\rfloor = 1. \quad (3.42)$$

Usando las ecuaciones (3.37)-(3.42) se puede formar la matriz de entrada-salida $\mathbf{H}_{2 \times 7}$, resultando en la ecuación (3.43).

$$H_{2 \times 7} = \begin{bmatrix} 0 & h_0 & h_2 & h_4 & h_6 & h_8 & h_{10} \\ h_1 & h_3 & h_5 & h_7 & h_9 & h_{11} & 0 \end{bmatrix}. \quad (3.43)$$

La factorización de una matriz centro-simétrica se presentará en detalle más adelante.

Dependiendo de los valores de M , L y K se definen dos casos que se presentarán a continuación.

3.3.2. Caso A

Si existe un K entero para el que se cumple la condición (3.44), la matriz de entrada-salida estará dada como:

$$N = M(L - 1) + (2K + 1)L. \quad (3.44)$$

$$\mathbf{Y}_{n,L} = \mathbf{H}_{L \times (p+q+1)} \mathbf{X}_{m+p,m-q} = \mathbf{H}_A \mathbf{X}_{m+p,m-q}. \quad (3.45)$$

La matriz \mathbf{H}_A tiene dimensiones $L_H \times 2\lambda$ definidas como:

$$L_H = L, \quad (3.46)$$

$$\lambda = (p + q + 1)/2. \quad (3.47)$$

Finalmente

$$\mathbf{H}_A = [\mathbf{H}_1 \quad \mathbf{H}_2], \quad (3.48)$$

donde

$$\mathbf{H}_2 = \mathbf{J}_{L_H} \mathbf{H}_1 \mathbf{J}_\lambda, \quad (3.49)$$

Los elementos de \mathbf{H}_A son $h_{k,\ell}^{(A)}$ para $k = 0, 1, \dots, L_H - 1$ y $\ell = 0, 1, \dots, 2\lambda - 1$. Además \mathbf{H}_A es una matriz centro-simétrica y puede descomponerse como lo presenta la ecuación (3.50) y (3.51) para λ impar y par, respectivamente.

$$\mathbf{H}_A = \begin{bmatrix} \mathbf{I}_{\lfloor L_H/2 \rfloor} & 0 & \mathbf{J}_{\lfloor L_H/2 \rfloor} \\ \mathbf{0}_{1, \lfloor L_H/2 \rfloor} & 1 & \mathbf{0}_{1, \lfloor L_H/2 \rfloor} \\ \mathbf{J}_{\lfloor L_H/2 \rfloor} & 0 & -\mathbf{I}_{\lfloor L_H/2 \rfloor} \end{bmatrix} \begin{bmatrix} \mathbf{C} & \mathbf{0}_{\lfloor L_H/2 \rfloor, \lambda} \\ \mathbf{e} & \mathbf{0}_{1, \lambda} \\ \mathbf{0}_{\lfloor L_H/2 \rfloor, \lambda} & \mathbf{D} \end{bmatrix} \begin{bmatrix} \mathbf{I}_\lambda & \mathbf{J}_\lambda \\ \mathbf{J}_\lambda & -\mathbf{I}_\lambda \end{bmatrix}, \quad (3.50)$$

$$\mathbf{H}_A = \begin{bmatrix} \mathbf{I}_{L_H/2} & \mathbf{J}_{L_H/2} \\ \mathbf{J}_{L_H/2} & -\mathbf{I}_{L_H/2} \end{bmatrix} \begin{bmatrix} \mathbf{C} & \mathbf{0}_{L_H/2, \lambda} \\ \mathbf{0}_{L_H/2, \lambda} & \mathbf{D} \end{bmatrix} \begin{bmatrix} \mathbf{I}_\lambda & \mathbf{J}_\lambda \\ \mathbf{J}_\lambda & -\mathbf{I}_\lambda \end{bmatrix}, \quad (3.51)$$

donde

$$\mathbf{C} = \begin{bmatrix} c_{0,0} & c_{0,1} & \cdots & c_{0,\lambda-1} \\ \vdots & \vdots & \vdots & \vdots \\ c_{\lfloor L_H/2 \rfloor - 1, 0} & c_{\lfloor L_H/2 \rfloor - 1, 1} & \cdots & c_{\lfloor L_H/2 \rfloor - 1, \lambda - 1} \end{bmatrix}, \quad (3.52)$$

$$\mathbf{D} = \begin{bmatrix} d_{\lfloor L_H/2 \rfloor - 1, \lambda - 1} & \cdots & d_{\lfloor L_H/2 \rfloor - 1, 1} & d_{\lfloor L_H/2 \rfloor - 1, 0} \\ \vdots & \vdots & \vdots & \vdots \\ d_{0, \lambda - 1} & \cdots & d_{0, 1} & d_{0, 0} \end{bmatrix}, \quad (3.53)$$

$$\mathbf{e} = [e_0 \ e_1 \ \dots \ e_{\lambda-1}]. \quad (3.54)$$

Es importante notar que cada uno de los coeficientes de la matriz \mathbf{H}_A depende solamente de los coeficientes del filtro $H(z)$ y se obtienen con las siguientes ecuaciones:

$$c_{k,\ell} = \frac{h_{k,\ell}^{(A)} + h_{L_H-1-k,\ell}^{(A)}}{2}, \quad (3.55)$$

$$d_{k,\ell} = \frac{h_{k,\ell}^{(A)} - h_{L_H-1-k,\ell}^{(A)}}{2}, \quad (3.56)$$

$$e_\ell = h_{\lfloor L_H/2 \rfloor, \ell}^{(A)}, \quad (3.57)$$

para $k = 0, 1, \dots, \lfloor L_H/2 \rfloor - 1$ y $\ell = 0, 1, \dots, \lambda - 1$. El término de la derecha de las ecuaciones (3.50), (3.51) puede combinarse con el vector de entradas según la ecuación (3.58).

$$\mathbf{x}_{m+p, m-q}^{(\lambda-1)} = \begin{bmatrix} \mathbf{I}_\lambda & \mathbf{J}_\lambda \\ \mathbf{J}_\lambda & -\mathbf{I}_\lambda \end{bmatrix} \mathbf{x}_{m+p, m-q}. \quad (3.58)$$

Ejemplo 3.3: Retomando el ejemplo 3.2, y usando la condición de simetría de $H(z)$ y separando el coeficiente h_4 que multiplica a la entrada $x[m-2]$, la ecuación (3.43) puede ser reescrita como en (3.59).

$$\begin{bmatrix} y[n] \\ y[n+1] \end{bmatrix} = h_4 x[m-2] \begin{bmatrix} 1 \\ 1 \end{bmatrix} + \begin{bmatrix} 0 & h_0 & h_2 & h_5 & h_3 & h_1 \\ h_1 & h_3 & h_5 & h_2 & h_0 & 0 \end{bmatrix} \begin{bmatrix} x_{m+1, m-1} \\ x_{m-3, m-5} \end{bmatrix}. \quad (3.59)$$

La parte del lado derecho de la ecuación (3.59) tiene la representación alternativa dada por la ecuación (3.60).

$$\begin{bmatrix} 0 & h_0 & h_2 & h_5 & h_3 & h_1 \\ h_1 & h_3 & h_5 & h_2 & h_0 & 0 \end{bmatrix} \begin{bmatrix} x_{m+1,m-1} \\ x_{m-3,m-5} \end{bmatrix} = \begin{bmatrix} 1 & -1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} c_0 & c_1 & c_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & d_2 & d_1 & d_0 \end{bmatrix} \mathbf{X}_{m+1,m-5}^{(2)}, \quad (3.60)$$

donde

$$\begin{aligned} c_0 &= h_1/2, & d_0 &= -h_1/2, \\ c_1 &= (h_0 + h_3)/2, & d_1 &= (h_0 - h_3)/2, \\ c_2 &= (h_2 + h_5)/2, & d_2 &= (h_2 - h_5)/2, \end{aligned} \quad (3.61)$$

$$\mathbf{X}_{m+1,m-5}^{(2)} = \begin{bmatrix} x[m+1] + x[m-5] \\ x[m] + x[m-4] \\ x[m-1] + x[m-3] \\ x[m-1] - x[m-3] \\ x[m] - x[m-4] \\ x[m+1] - x[m-5] \end{bmatrix}. \quad (3.62)$$

Finalmente se tienen las ecuaciones que permiten la implementación.

$$\begin{bmatrix} y[n] \\ y[n+1] \end{bmatrix} = \begin{bmatrix} h_4 x[m-2] \\ h_4 x[m-2] \\ c_0(x[m+1] + x[m-5]) + c_1(x[m] + x[m-4]) + c_2(x[m-1] + x[m-3]) \\ c_0(x[m+1] + x[m-5]) + c_1(x[m] + x[m-4]) + c_2(x[m-1] + x[m-3]) \\ + d_2(x[m-1] - x[m-3]) + d_1(x[m] - x[m-4]) + d_0(x[m+1] - x[m-5]) \\ - (d_2(x[m-1] - x[m-3]) + d_1(x[m] - x[m-4]) + d_0(x[m+1] - x[m-5])). \end{bmatrix} \quad (3.63)$$

3.3.3. Caso B

Este caso contempla cualquier combinación de los parámetros N , L y M . Se tiene que:

$$\mathbf{H}_{L \times (p+q+1)} = \begin{bmatrix} \mathbf{0}_{L_1, p-p_1} & \mathbf{H}_{c_1} \\ \mathbf{H}_{c_2} & \mathbf{0}_{L_2, q-q_2} \end{bmatrix}. \quad (3.64)$$

Existe un entero r tal que la r ésima fila sea la última fila de la matriz \mathbf{H}_{c_1} que contiene la $\mu_r^{\text{ésima}}$ componente polifásica.

$$\mu_r = \mu_{N-qL}, \quad (3.65)$$

$$\text{mod}(rM, L) = N - qL, \quad (3.66)$$

Conociendo el valor numérico de la ecuación (3.66), el valor de r se determina evaluando $\ell = 0, 1, \dots, L - 1$, lo que permite obtener las dimensiones de las matrices \mathbf{H}_{c_1} y \mathbf{H}_{c_2} .

$$L_1 = r + 1, \quad (3.67)$$

$$L_2 = L + L_1, \quad (3.68)$$

$$p_1 = \frac{(L_1 - 1)M - (N - qL)}{L}, \quad (3.69)$$

$$q_2 = \left\lfloor \frac{N - \text{mod}(L_1M, L)}{L} \right\rfloor - \left\lfloor \frac{L_1M}{L} \right\rfloor, \quad (3.70)$$

$$L_H = \begin{cases} L_1 & \text{para } \mathbf{H}_{c_1}. \\ L_2 & \text{para } \mathbf{H}_{c_2}. \end{cases} \quad (3.71)$$

$$\lambda = \begin{cases} \left\lfloor \frac{(q+p_1+1)}{2} \right\rfloor & \text{para } \mathbf{H}_{c_1}. \\ \left\lfloor \frac{(q_2+p+1)}{2} \right\rfloor & \text{para } \mathbf{H}_{c_2}. \end{cases} \quad (3.72)$$

Ejemplo 3.4: Se presenta a continuación un ejemplo para la descomposición mostrada de la matriz H para el caso B, considerando la función de transferencia presentada a continuación.

$$\begin{aligned} H(z) &= (1 + z^{-1})^7 \\ &= [1 + 7z^{-1} + 21z^{-2} + 35z^{-3} + 35z^{-4} + 21z^{-5} + 7z^{-6} + z^{-7}]. \end{aligned} \quad (3.73)$$

Usando la ecuación (3.27) se obtienen las componentes polifase de la $H(z)$ obteniendo:

$$\begin{aligned} H_1(z) &= h_0 + h_7z^{-1}, & H_4(z) &= h_4, & H_7(z) &= h_7, \\ H_2(z) &= h_2, & H_5(z) &= h_5, \\ H_3(z) &= h_3, & H_6(z) &= h_6. \end{aligned} \quad (3.74)$$

La matriz de entrada-salida obtenida es:

$$H_{7 \times 3} = \begin{bmatrix} 0 & h_0 & h_7 \\ 0 & h_2 & 0 \\ 0 & h_4 & 0 \\ 0 & h_6 & 0 \\ h_1 & 0 & 0 \\ h_3 & 0 & 0 \\ h_5 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & h_0 & h_0 \\ 0 & h_2 & 0 \\ 0 & h_3 & 0 \\ 0 & h_1 & 0 \\ h_1 & 0 & 0 \\ h_3 & 0 & 0 \\ h_2 & 0 & 0 \end{bmatrix} \quad (3.75)$$

También se tiene que: $L_1 = 1$, $L_2 = 6$, $p_1 = 0$, $q_2 = 0$, con lo que se puede dividir la matriz $H_{7 \times 3}$ según la ecuación (3.64) como se muestra a continuación. La matriz H_{c_2} se implementa usando el caso A.

$$H_{c_1} = H_{1 \times 2} = \begin{bmatrix} h_0 & h_0 \end{bmatrix}, \quad (3.76)$$

$$H_{c_2} = H_{6 \times 2} = \begin{bmatrix} 0 & h_2 \\ 0 & h_3 \\ 0 & h_1 \\ h_1 & 0 \\ h_3 & 0 \\ h_2 & 0 \end{bmatrix}. \quad (3.77)$$

Capítulo 4

Revisión de métodos para conversión de frecuencia de muestreo por un factor racional

En este capítulo se presenta una revisión de los métodos para realizar la Conversión de Frecuencia de Muestreo Racional o en inglés Rational Sample Rate Conversion (RSCR). Estos se pueden clasificar en dos grandes grupos, los que permiten la selección arbitraria del factor de conversión y aquellos que permiten la conversión por factores específicos y requieren rediseño para diferentes estándares. Sin embargo, en este trabajo se consideran los métodos del segundo grupo.

La mayoría del trabajo sobre este tema se centra en el uso de filtros CIC: [6]- [10], por otro lado se tienen combinaciones entre filtros CIC e interpoladores [11]- [12]; también se encuentran esquemas que usan filtros adicionales que no son CIC, ya sea filtros Infinite Impulse Response (IIR) o FIR como los propuestos en [13] y [14].

4.1. Métodos basados en el filtro CIC

En esta sección se describen algunos métodos que se basan únicamente en el filtro CIC para la realización de los filtros *anti-aliasing/imaging*. Estos métodos son libres de multiplicaciones, lo que indica que poseen una baja complejidad computacional, presentan una buena relación señal a ruido o Signal to Noise Ratio (SNR) y pueden incluir mejoras en la respuesta en magnitud que garanticen las propiedades requeridas para su óptimo desempeño.

4.1.1. Filtro CIC modificado para conversión de frecuencias de muestreo para sistemas de Radio definido por Software [15]

El método consiste en la modificación al filtro convencional de la ecuación (4.1). En esta ecuación L es el factor de interpolación y D generalmente se elige como 1. Esta modificación se presenta en la ecuación (4.2).

$$H(z) = \frac{(1 - z^{-DM})^{K_1} (1 - z^{-DL})^{K_2}}{(1 - z^{-1})^K}, \quad (4.1)$$

$$H(z) = \frac{(1 - z^{-D_1})(1 - z^{-D_2}) \dots (1 - z^{-D_K})}{(1 - z^{-1})^K}. \quad (4.2)$$

La respuesta en magnitud del filtro modificado se presenta a continuación.

$$|H(e^{j\omega})| = \left| \left[\frac{1}{D_1} \frac{\text{sen}(\omega D_1/2)}{\text{sen}(\omega/2)} \right] \left[\frac{1}{D_2} \frac{\text{sen}(\omega D_2/2)}{\text{sen}(\omega/2)} \right] \dots \left[\frac{1}{D_K} \frac{\text{sen}(\omega D_K/2)}{\text{sen}(\omega/2)} \right] \right|. \quad (4.3)$$

A diferencia del filtro convencional que tiene ceros en los múltiplos de $\frac{2\pi}{DL}$ y $\frac{2\pi}{DM}$, el filtro modificado presenta ceros en $\frac{2\pi}{D_i}$ con $i = 1, 2, \dots, K$, lo que provee una atenuación más uniforme para todas las imágenes. La SNR se mide en las frecuencias definidas por la ecuaciones (4.4). Allí se asume que la banda de paso corresponde a las $\frac{3}{4}$ partes de toda la banda disponible.

$$\omega_p = \frac{3}{4M}, \quad \omega_1 = \frac{2}{M} - \omega_p \quad (4.4)$$

Los retardos D_i se seleccionan en el rango $\text{máx}(DM, DL)$ y $\text{máx}(2DM, 2DL)$, dependiendo del espectro de potencia de la señal de entrada, buscando que la atenuación de las imágenes sea lo más uniforme posible.

Ejemplo 4.1: El factor de conversión de frecuencia de muestreo es $\frac{9}{10}$. Los retardos correspondientes para el CIC modificado son: 16, 14, 12, 10 respectivamente. En la Fig. 4.1 se puede apreciar la diferencia entre las SNR's del filtro CIC convencional y el filtro CIC modificado. Con el fin de comparar la respuesta del filtro CIC modificado usando diferentes valores para los retardos D_i , en la Tabla 4.1 se presentan algunas selecciones alternativas que permitan tener una referencia de la variación de la SNR.

En general el filtro CIC modificado presenta una Additions Per Output Sample (APOS) dada por la ecuación (4.5).

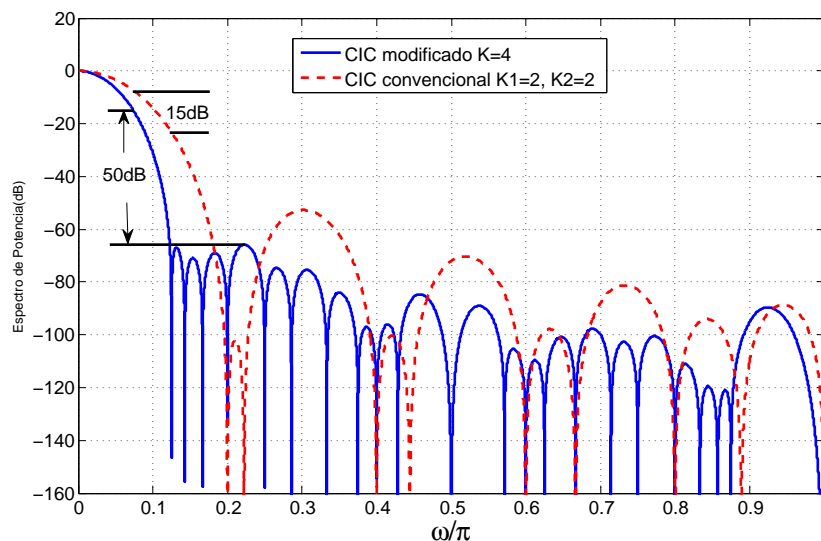


Figura 4.1: Espectro de magnitud para el ejemplo del filtro CIC modificado.

Selección	Retardos	SNR
Opción 1	$D_i = 10, 11, 12, 14$	46dB
Opción 2	$D_i = 14, 16, 18, 20$	31dB
Opción 3	$D_i = 10, 11, 12, 14$	37dB

Tabla 4.1: Variación de la SNR con la selección de los retardos D_i .

$$APOS = 2KM, \quad (4.5)$$

donde los integradores y los filtros *comb* tienen KM APOS, y funcionan en alta frecuencia intermedia. La cantidad de elementos de memoria requeridos es $(D_1 + D_2 + \dots + D_K + K)$.

4.1.2. Filtro CIC Triangular Escalonado para conversión de frecuencia de muestreo racional [6]

El filtro CIC triangular escalonado utiliza la aproximación triangular escalonada, propuesta en [10] para el diseño de filtros FIR pasa-bajas. Consiste en reemplazar el filtro original por una versión cuantizada del mismo. Esta versión del filtro puede implementarse como una cascada de un filtro *comb* y un filtro *comb* expandido.

Se supone que el factor de decimación se puede presentar como el producto de dos factores M_1 y M_2 . La estructura no posee multiplicaciones. Presenta una baja complejidad y un mejor comportamiento comparado con otros métodos. Se asume que la banda que se

usará corresponde a $\frac{3}{4}$ de toda la banda usable.

Sustituyendo el filtro CIC por un filtro aproximado resulta en las ecuaciones (4.6) y (4.7).

$$H_{TE}(z) = \left[\frac{1}{M_2} \frac{(1 - z^{-M})}{(1 - z^{-M_1})} \left[\frac{1}{M_1} \frac{(1 - z^{-M_1})}{(1 - z^{-1})} \right]^2 \right] \quad \text{para} \quad M_1 > M_2, \quad (4.6)$$

$$H_{TE}(z) = \left[\left[\frac{1}{M_2} \frac{(1 - z^{-M})}{(1 - z^{-M_1})} \right]^2 \frac{1}{M_1} \frac{(1 - z^{-M_1})}{(1 - z^{-1})} \right] \quad \text{para} \quad M_1 < M_2. \quad (4.7)$$

El filtro coseno expandido tiene la función de transferencia:

$$H_{cos}(z) = (1 + z^{-R})/2, \quad (4.8)$$

donde R es el factor de expansión. El rango de selección de R está dado como

$$R_{min} \leq R \leq M, \quad (4.9)$$

donde

$$R_{min} = \text{int}[1/\omega_1], \quad (4.10)$$

$$\omega_1 = \frac{2}{M} - \omega_p; \quad \omega_p = \frac{3}{4M}. \quad (4.11)$$

El filtro de decimación está dado por la ecuación (4.12).

$$H_p(z) = \left[\frac{1}{M_2} \frac{(1 - z^{-M})}{(1 - z^{-M_1})} \right]^{K_1} \left[\frac{1}{M_1} \frac{(1 - z^{-M_1})}{(1 - z^{-1})} \right]^{K_2} [(1 + z^{-R})/2]^{K_3}, \quad (4.12)$$

La ecuación (4.13) corresponde a la respuesta en magnitud del filtro.

$$|H_p(e^{j\omega})| = \left| \left[\frac{1}{M_2} \frac{\text{sen}(\omega M/2)}{\text{sen}(\omega M_1/2)} \right]^{K_1} \left[\frac{1}{M_1} \frac{\text{sen}(\omega M_1/2)}{\text{sen}(\omega/2)} \right]^{K_2} [\cos(R\omega/2)]^{K_3} \right|, \quad (4.13)$$

donde

$$K_2 \geq 2K_1 \quad \text{para} \quad M_1 > M_2, \quad (4.14)$$

$$K_1 \geq 2K_2 \quad \text{para} \quad M_1 < M_2. \quad (4.15)$$

Usando las identidades multirazón es posible obtener la estructura general mostrada en

la Fig. 4.2.

$$H_{p1} = [1 + z^{-R}]^{K_3} \left[\frac{1}{1 - z^{-1}} \right]^{K_2}, \quad (4.16)$$

$$H_{p2} = [1 - z^{-1}]^{K_2 - K_1}, \quad (4.17)$$

$$H_{p3} = [1 - z^{-1}]^{K_1}. \quad (4.18)$$

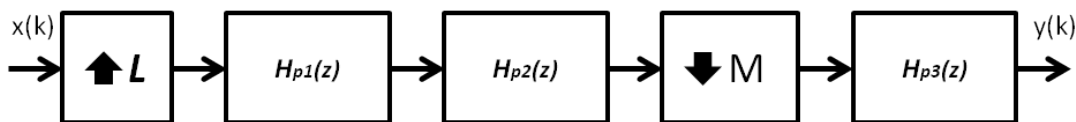


Figura 4.2: Estructura general para el filtro CIC triangular escalonado.

Solo se considera el caso en el que $M_1 < M_2$, ya que este presenta mejores características [6]. La cantidad de APOS para este método depende de la selección del parámetro R , K_1 , K_2 y K_3 como se presenta en la Tabla 4.2.

R	Memoria requerida	APOS
$R = R_{min}$	$4K_2 + RK_3$	$2K_2 + K_2M_2 + (K_3 + K_2)M$
$R = R_1M_1$	$4K_2 + R_1K_3$	$2K_2 + (K_2 + K_3)M_2 + K_2M$
$R = M$	$4K_2 + K_3$	$2K_2 + K_3 + K_2M_2 + K_2M$

Tabla 4.2: Cantidad de APOS y memoria requerida para el filtro CIC triangular escalonado usando $M_1 < M_2$.

Ejemplo 4.2: Usando el filtro de [6] se diseña un filtro decimador para un factor de conversión de frecuencia de $\frac{9}{10}$. Se comparan los espectros de potencia usando $R_1 = 8$ y $R_2 = 10$ y el filtro CIC convencional. La SNR del filtro convencional es 15dB, para el filtro de [6] con $R = 8$ es 58.9dB y usando $R = 10$ es 54.9dB.

4.1.3. Arquitectura CIC intercalada de bajo consumo de potencia para receptores de Radio Definido por Software [16]

En este método se propone el uso de una estructura intercalada para la primera etapa de la decimación. De esta forma se pueden combinar los dos filtros que se usan usualmente para la señales en cuadratura en el receptor digital. Esta estructura utiliza la decimación polifásica y la técnica de intercalamiento en el tiempo, donde las señales pueden ser

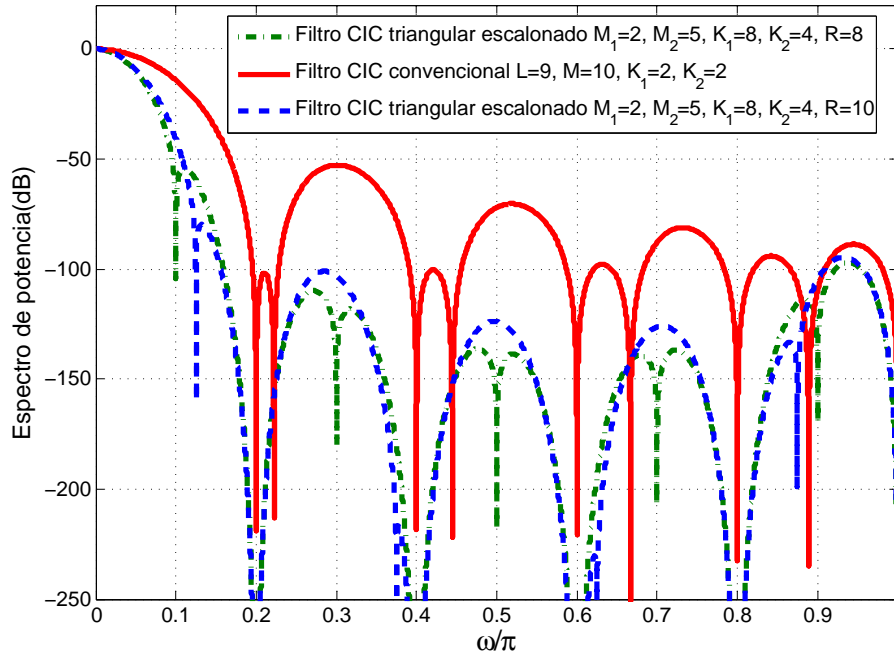


Figura 4.3: Espectro de magnitud para el ejemplo del filtro CIC triangular escalonado.

procesadas en una frecuencia de muestreo reducida, permitiendo además la reducción del voltaje de alimentación V_{dd} sin afectar el funcionamiento del filtro.

La función de transferencia del filtro a CIC se presenta a continuación.

$$H(z) = \left(\frac{1 - z^{-D}}{1 - z^{-1}} \right)^K, \quad (4.19)$$

$$H(z) = \left(\frac{1 - z^{-D}}{1 - z^{-R_1}} \right)^{N_1} \left(\frac{1 - z^{-R}}{1 - z^{-R_1}} \right)^{N_2} = H_1(z)H_2(z^{R_1}), \quad (4.20)$$

La ecuación (4.20) corresponde a la factorización de la forma recursiva del filtro CIC, además $K = N_1N_2$ y $M = R_1R_2$, con lo que se obtienen los dos filtros CIC presentados a continuación.

$$H_1(z) = (1 + z^{-1} + z^{-2} + \dots + z^{-(R_1-1)})^{N_1}, \quad (4.21)$$

$$H_2(z) = \left(\frac{1 - z^{-R_2}}{1 - z^{-1}} \right)^{N_2}. \quad (4.22)$$

El primer filtro $H_1(z)$ ha sido llevado a la forma no recursiva, ahora es posible aplicar la técnica de segmentación/intercalamiento. Lo primero es obtener la versión expandida de

los filtros $H_1(z)$ y $H_2(z)$.

$$H_{1p}(z^2) = (1 + z^{-1} + z^{-2} + \dots + z^{-2(R_1-1)})^{N_1}, \quad (4.23)$$

$$H_{2p}(z^2) = \left(\frac{1 - z^{-2R_2}}{1 - z^{-2}} \right)^{N_2}. \quad (4.24)$$

Se observa en la ecuación (4.23) que las componentes pares son cero, lo que resulta en el uso de retardos dobles para la implementación polifásica como se ve en la ecuación (4.25), donde se usa $R_1 = 8$ y $N_1 = 2$.

$$H_{1p}(z^2) = E_1(z^{16}) + 0 + z^{-2}E_3(z^{16}) + 0 + z^{-4}E_5(z^{16}) + \dots + 0 + z^{-6}E_7(z^{16}) + 0. \quad (4.25)$$

Las multiplicaciones requeridas para la implementación polifásica pueden hacerse usando corrimientos y sumas. La estructura resultante se presenta en la Fig. 4.4.

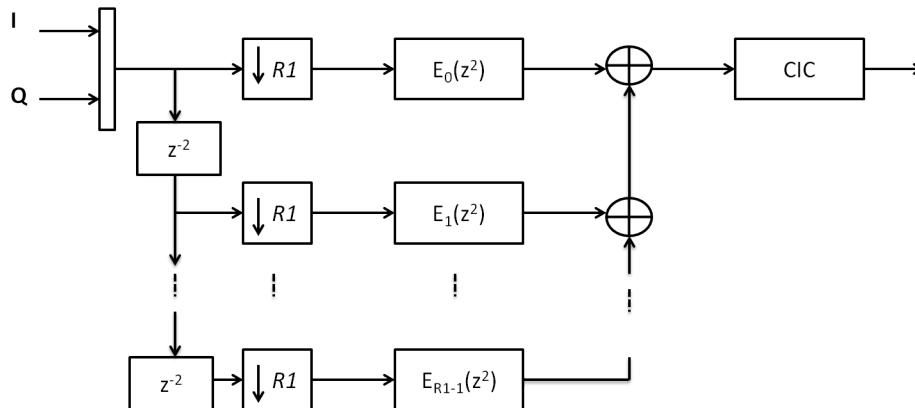


Figura 4.4: Estructura general para el filtro propuesto en [16].

El consumo de potencia de un circuito CMOS está dado por la siguiente ecuación.

$$P_{din} = C_{ef}V_{dd}^2f, \quad (4.26)$$

donde C_{ef} es la capacitancia de carga efectiva, que depende del tamaño del circuito, V_{dd} es el voltaje de alimentación y f es la frecuencia de operación, es decir, la frecuencia de muestreo. Debido a la paralelización derivada de la estructura de la Fig. 4.4, es posible el escalamiento del voltaje de alimentación como se presenta en la Tabla 4.3.

Algunas estimaciones del consumo de potencia para tres casos de la Tabla 4.3 son:

Estándar	Descomposición	V_{dd} escalado (V)	Número de sumas
GSM	8, 8	1.27	18
Mobitex	5, 107	1.48	7
Ardis	5, 107	1.48	7
IS-95	5	1.48	7
UMTS	5	1.48	7

Tabla 4.3: Escalamiento del voltaje de alimentación para algunos estándares de comunicación inalámbrica.

$$P_1 = \left(\frac{10C_{ef}}{6}\right) \left(\frac{1.48}{3.3}\right)^2 \left(\frac{2f}{5}\right) + \left(\frac{3C_{ef}}{6}\right) \left(\frac{1.48}{3.3}\right)^2 \left(\frac{(107/2)f}{535}\right) = 0.144P_0. \quad (4.27)$$

$$P_2 = \left(\frac{21C_{ef}}{6}\right) \left(\frac{1.27}{3.3}\right)^2 \left(\frac{2f}{8}\right) + \left(\frac{3C_{ef}}{6}\right) \left(\frac{1.27}{3.3}\right)^2 \left(\frac{f}{2}\right) = 0.166P_0. \quad (4.28)$$

$$P_3 = \left(\frac{10C_{ef}}{6}\right) \left(\frac{1.48}{3.3}\right)^2 \left(\frac{2f}{5}\right) = 0.134P_0. \quad (4.29)$$

En los tres casos se observa un ahorro en el consumo de potencia de aproximadamente 85 % de la potencia consumida por el sistema de referencia.

4.2. Métodos basados en filtros CIC e interpoladores

En esta sección se presentan algunos métodos que recurren al uso de interpoladores, ya sea apoyándose del filtro CIC o sin él para garantizar la respuesta en magnitud requerida. La inclusión de interpoladores generalmente produce la aparición de multiplicaciones, que a su vez representan un alto costo computacional.

4.2.1. Conversión eficiente de frecuencias de muestreo para Radio Definido por Software Multi-Estándar [11]

Este método consiste en la modificación del esquema para conversión arbitraria de frecuencias de muestreo presentado en la Fig. 4.5. Este esquema fue propuesto en [17].

Usando multiplexores y filtros de decimación predefinidos, es posible eliminar el filtro de decimación programable. Además la primera etapa que incluye la compensación de la caída en la banda de paso del filtro CIC se realiza en conjunto con la interpolación. Para

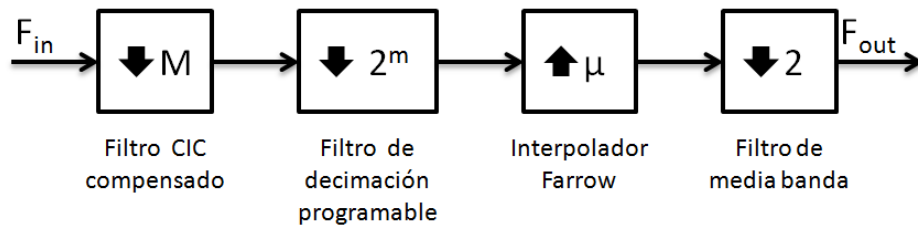


Figura 4.5: Estructura para conversión de frecuencia de muestreo arbitraria propuesta en [17].

ello la banda de paso del interpolador Farrow se reemplaza por una respuesta polinomial de segundo orden. El factor de conversión de frecuencia de muestreo se factoriza según la ecuación (4.30).

$$\frac{L}{D} = \mu \times \frac{1}{M} \times 2^{-k}. \quad (4.30)$$

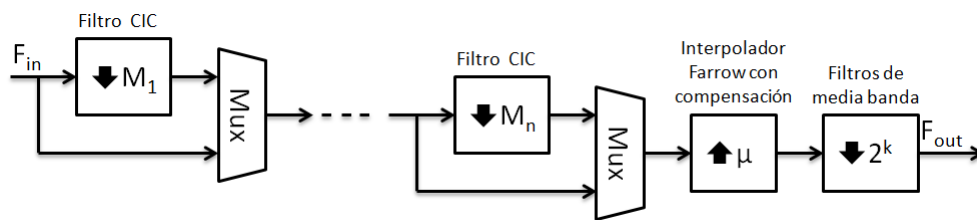


Figura 4.6: Estructura propuesta en [17].

Mediante los multiplexores es posible seleccionar los filtros CIC necesarios para el factor de decimación total. Además, se observa que el interpolador Farrow, que incluye la compensación en la banda de paso para los filtros CIC, funciona en baja frecuencia de muestreo y la última etapa corresponde a un filtro de media banda con el que se obtiene una menor carga computacional, ya que la mitad de sus coeficientes son cero.

Estándar	Tasa de datos	Factores de conversión	Ancho de banda	M	μ	k
GSM	270.8Kbps	677/200000	200KHz	256	1.733	1
IS-95	1.2288Mbps	48/3125	1.25MHz	64	1.966	1
UMTS	3.84Mbps	6/125	5MHz	16	1.54	1
D-AMPS	48.6Kbps	243/400000	30KHz	896	1.08	1
IEEE 802.11b	11Mbps	11/80	22MHz	2	1.1	2

Tabla 4.4: Factorización propuesta para diferentes estándares de comunicación inalámbrica.

El diseño del interpolador/compensador se hace de acuerdo a la siguiente respuesta deseada:

$$D(\omega) = \begin{cases} 1 + K_p \omega (\omega_c - \omega) & 0 \leq \omega \leq \omega_c \\ 1 - (\omega - \omega_c) / (\omega_s - \omega_c) & \omega_c \leq \omega \leq \omega_s \\ 0 & \omega \geq \omega_s \end{cases} \quad (4.31)$$

$$K_p = \frac{4 \max[D(\omega)] - 1}{\omega_c^2}, \quad (4.32)$$

donde ω_c es la frecuencia hasta donde se compensa el filtro CIC, ω_s es la frecuencia de rechazo del interpolador Farrow y el parámetro K_p controla la amplitud máxima del polinomio de segundo orden y se calcula usando la ecuación (4.32).

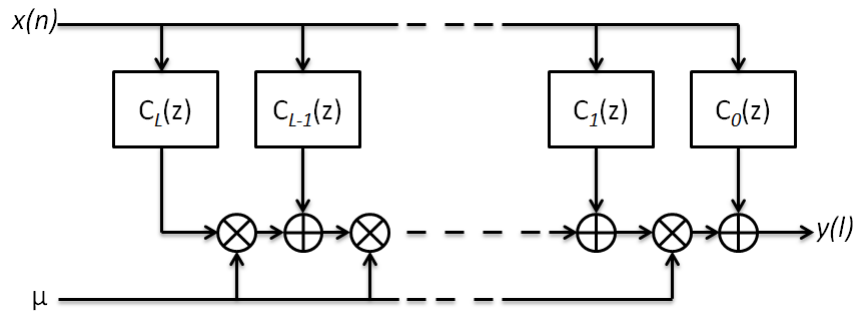


Figura 4.7: Estructura del Interpolador Farrow.

En la Fig. 4.7 se presenta la estructura del interpolador Farrow. Se tienen $L + 1$ ramas de orden N . El diseño del interpolador se realiza usando el método de optimización minimax, buscando que sea lo más parecido posible a la respuesta definida por la ecuación (4.31) usando las ecuaciones mostradas a continuación.

$$H(j\omega F_{in}) = \frac{e^{-j\omega N/2}}{F_{in}} \sum_{l=0}^L C_l(\omega) W_l(\omega), \quad (4.33)$$

$$C_l(\omega) = \begin{cases} 2 \sum_{n=0}^{N/2-1} c_l(n) \cos \left(\left(\frac{N-1}{2} - n \right) \omega \right) & \text{para } l \text{ par,} \\ 2 \sum_{n=0}^{N/2-1} c_l(n) \sin \left(\left(\frac{N-1}{2} - n \right) \omega \right) & \text{para } l \text{ impar,} \end{cases} \quad (4.34)$$

$$W_l(\omega) = \begin{cases} \sum_{k=0}^l k! \binom{l}{k} \frac{\sin(\omega/2 + k\pi/2)}{(\omega/2)^{k+1}} & \text{para } l \text{ par,} \\ \sum_{k=0}^l k! \binom{l}{k} \frac{\cos(\omega/2 + k\pi/2)}{-(\omega/2)^{k+1}} & \text{para } l \text{ impar,} \end{cases} \quad (4.35)$$

donde C_l representa cada rama del filtro y c_l son los respectivos coeficientes. Gracias a

la decimación previa al interpolador la cantidad de ramas L y la cantidad de coeficientes en cada rama N se ven reducidas. Además, no se requiere de una gran atenuación en la banda de rechazo porque ya se ha mejorado gracias al filtrado previo. Usando solamente 5 filtros CIC es posible obtener todas factores de decimación requeridos. Estos factores de decimación son potencias de dos consecutivas hasta 16, a excepción de 7 que se requiere para el estándar D-AMPS (Tabla 4.5).

Ejemplo 4.3: Se presenta el caso del estándar UMTS. Las características principales se presentan en la Tabla 4.5. Se requiere solamente un filtro CIC con factor de decimación de $M = 16$, un intepolador/compensador con $\mu = 1.08$ y un filtro de media banda para la decimación final por 2. Las respuestas en magnitud se presentan en las Figuras 4.8 y 4.9. La respuesta deseada y la aproximada para el interpolador/compesador se presentan en la misma gráfica con el fin de comparar la banda de paso que se encarga de compensar el filtro CIC.

Estándar	Factor de decimación (M)	Equivalente factorizado
GSM	256	$2 \times 8 \times 16$
IS-95	64	4×16
UMTS	16	$2 \times 8, 16$
D-AMPS	896	$7 \times 8 \times 16$
IEEE 802.11b	2	2

Tabla 4.5: Factorización del factor de decimación para diferentes estántares de comunicación inalámbrica

4.2.2. Decimación por un factor irracional usando filtros CIC e interpolador lineal [18]

Se propone la implementación de un sistema de conversión de frecuencia de muestreo en un factor racional o irracional, basado en filtros CIC conectados en paralelo, de los cuales dos se usan para realizar una interpolación lineal. Cuando se realiza una conversión racional de frecuencia de muestreo, los filtros CIC convencionales no pueden usarse directamente. Para este fin se utilizan filtros de interpolación lineal que son de fácil implementación y requieren una única multiplicación, como se observa en la ecuación (4.36),

$$y(l) = x(n_l) + [x(n_l + 1) - x(n_l)]\mu_l, \quad (4.36)$$

donde μ_l es un intervalo fraccional ajustable usado para la interpolación.

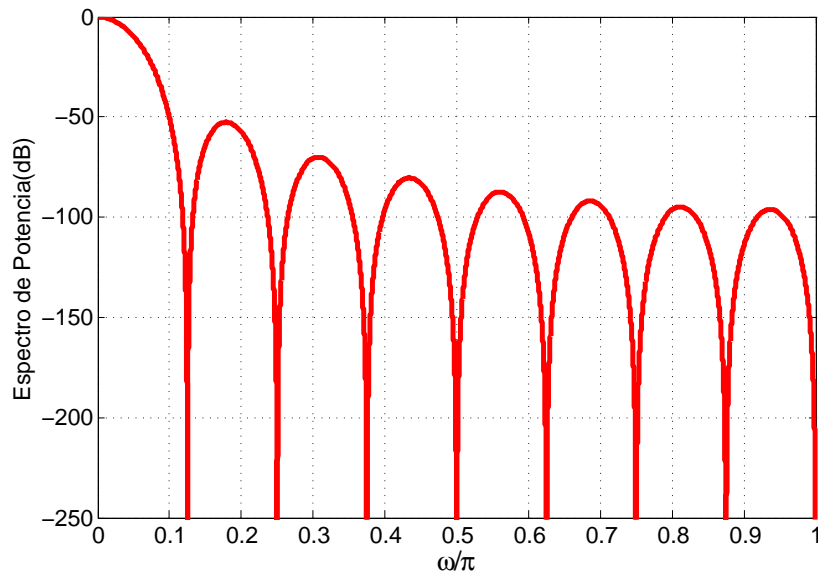


Figura 4.8: Filtro CIC $M = 16$, $K = 4$.

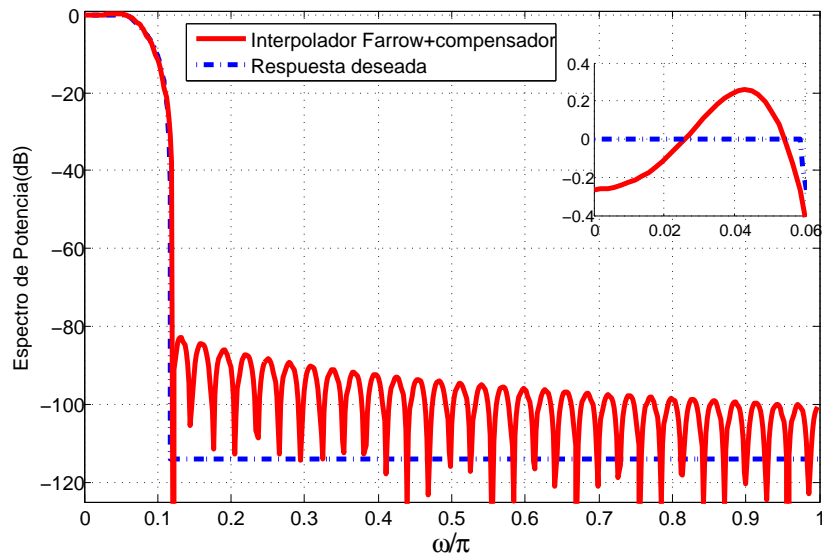


Figura 4.9: Respuesta deseada y respuesta aproximada para el interpolador.

La ecuación (4.42) define la respuesta en magnitud del interpolador lineal,

$$H_{IL}(w) = \left(\frac{\text{sen} \left(\frac{3w}{2} \right)}{2w} \right), \quad (4.37)$$

donde $w = 2\pi f/F_{in}$.

El factor de conversión de frecuencia de muestreo está dado por,

$$R = \frac{F_{in}}{F_{out}} = R_{int} + \epsilon, \quad (4.38)$$

donde R_{int} es la parte entera y ϵ es la parte decimal del factor de conversión.

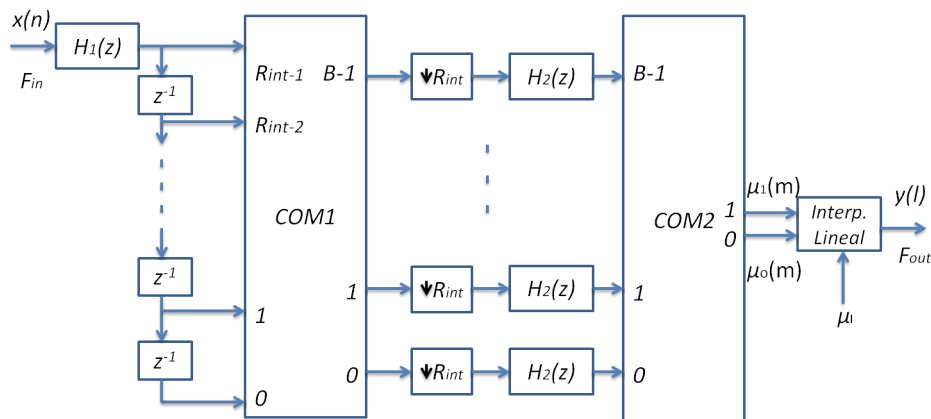


Figura 4.10: Estructura general para el filtro propuesto en [18].

El esquema de decimación se presenta en la Fig. 4.10. Los filtros $H_1(z)$ y $H_2(z)$ corresponden a las funciones de transferencia presentadas a continuación.

$$H_1(z) = \left[\frac{1}{R_{int}} \frac{1}{1 - z^{-1}} \right]^N, \quad (4.39)$$

$$H_2(z) = [1 - z^{-1}]^N. \quad (4.40)$$

Las ecuaciones (4.39) y (4.40) resultan de aplicar la identidad multirazón a la forma recursiva del filtro CIC de orden N . Después de realizar la descomposición polifásica (Fig. 4.10), el filtro (4.39) puede moverse a la entrada, mientras que a cada rama se le aplica el filtro (4.40) a una frecuencia de muestreo reducida R_{int} veces. Las ramas 0 y 1 se usan para realizar la interpolación. En la Fig. 4.11 se observa un ejemplo para decimación por un factor de 3.25. Los intervalos μ_i corresponden a la diferencia de tiempo entre la señal $x_0(n)$ y la señal interpolada. La cuarta muestra de la señal de salida $y(l)$ no cae en el intervalo entre $x_0(n)$ y $x_1(n)$. Entonces las ramas usadas para la interpolación son desplazadas, usándose para la siguiente interpolación las señales $x_1(n)$ y $x_2(n)$. Estos retardos pueden ser calculados usando un sumador en aritmética de punto fijo como lo muestra la ecuación (4.41), usando la condición inicial $\mu_0 = 0$. Cuando ocurre un sobreflujo al calcular μ_l se debe realizar el desplazamiento de las ramas con las cuales se calcula la interpolación. La implementación de estos corrimientos se realiza usando los conmutadores COM_1 y

COM_2 , como se observa en la Fig. 4.10.

$$\mu_l = \mu_{l-1} \oplus \epsilon. \quad (4.41)$$

Debido a la posibilidad de la selección arbitraria del factor racional de decimación, el costo computacional del método es relativamente alto, ya que se debe incluir una multiplicación para la interpolación. Por otro lado, la cantidad de ramas necesarias es $N + 2$, donde dos de ellas se usan para la interpolación y las demás para obtener resultados necesarios que son usados posteriormente para el cálculo de muestras de salida.

La respuesta en magnitud del sistema está definida como:

$$H_T(w) = \left(\frac{\text{sen}\left(\frac{\omega R_{int}}{2}\right)}{\text{sen}\left(\frac{\omega}{2}\right)} \right)^N \left(\frac{\text{sen}\left(\frac{\omega}{2}\right)}{\frac{\omega}{2}} \right)^2, \quad (4.42)$$

donde $\omega = 2\pi f/F_{in}$ y corresponde a la frecuencia normalizada relativa la frecuencia de entrada F_{in} .

Ejemplo 4.4: En la Fig. 4.11 se presenta un ejemplo del proceso requerido para la conversión de frecuencia de muestreo. Se observa que para la última interpolación se cambian las señales usadas para el cálculo, debido a que la muestra requerida no se encuentra entre el intervalo definido por $x_0(n)$ y $x_1(n)$.

Ejemplo 4.5: Este ejemplo presenta la respuesta en magnitud con un factor de decimación de 10.1, usando 3 etapas de filtros CIC ($N = 3$), se presenta en la Fig. 4.12.

4.2.3. Estructura eficiente en consumo de potencia para conversión entre frecuencias de muestreo arbitrarias [12]

Este método se basa en el uso del equivalente en tiempo continuo del filtro CIC. La propuesta permite introducir ceros en las frecuencias múltiplos de las frecuencia de entrada y de salida. Además posee una baja cantidad de multiplicadores enteros que reducen el consumo de potencia.

$$H_{CIC}(z) = H_I^N(z)H_C^N(z) = \frac{(1 - z^{-R})^N}{(1 - z^{-1})^N}. \quad (4.43)$$

La respuesta en frecuencia usando la frecuencia de entrada normalizada f/F_{in} está dada a continuación.

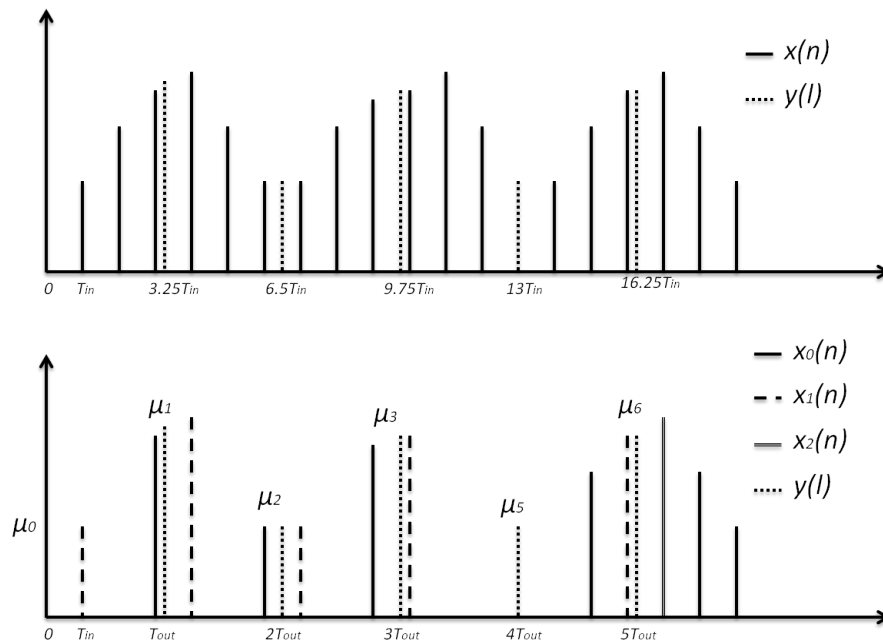


Figura 4.11: Ejemplo de conversión de frecuencia de muestreo por un factor $R = 3.25$.

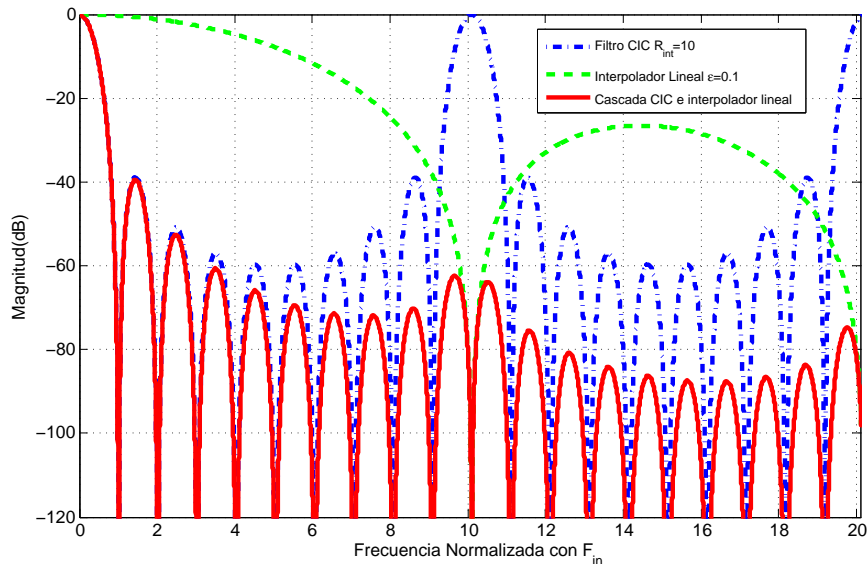


Figura 4.12: Respuesta en magnitud para conversión de frecuencia de muestreo por un factor $R = 10.1$.

$$H_{CIC}(e^{j2\pi f/F_{in}}) = \left(\frac{\text{sen}(\pi R f/F_{in})}{R \text{sen}(\pi R f/F_{in})} \right). \quad (4.44)$$

El equivalente en tiempo continuo de un filtro CIC es un integrador ideal. La respuesta al impulso del integrador ideal corresponde a un pulso rectangular de longitud T , así como

la respuesta al impulso de filtro CIC es un pulso rectangular en el dominio digital. Para este caso se selecciona esa longitud como el período de muestreo de la entrada o de la salida. La respuesta en frecuencia se presenta a continuación.

$$H_i(\omega) = \frac{\text{sen}(\omega T)}{\omega T}. \quad (4.45)$$

La cascada de N integradores ideales en tiempo continuo corresponde a un filtro CIC de orden N en tiempo discreto. Además, presenta una respuesta al impulso polinomial a tramos con N segmentos de polinomios de orden $M = N - 1$. La respuesta al impulso en tiempo continuo corresponde a la ecuación presentada a continuación.

$$h_a(t) = \sum_{n=0}^{N-1} \sum_{m=0}^M d_m(n) f_m(n, T, t), \quad (4.46)$$

donde

$$f_m(n, T, t) = \begin{cases} \left(\frac{(t-nT)}{T} \right)^m, & \text{para } nT \leq t < (n+1)T \\ 0, & \text{otros casos,} \end{cases} \quad (4.47)$$

$$d_m(k) = \sum_{i=-k}^{N/2} \frac{(-1)^{N/2-i} N! (i+k)^{M-m}}{(N/2+i)!(N/2-i)!(M-m)!m!}. \quad (4.48)$$

para $k = 0, 1, \dots, N - 1$ y $m = 0, 1, \dots, M$.

Usando la estructura Farrow es posible implementar la ecuación (4.46) cuando $T = T_{in}$ y cuando $T = T_{out}$ se usa la estructura Farrow transpuesta. Ambas estructuras se presentan en la Fig. 4.13.

Es posible obtener $h_a(t)$ usando versiones modificadas de $f_m(n, T, t)$:

$$h_a(t) = \sum_{n=0}^{N-1} \sum_{m=0}^M c_m(n) \bar{f}_m(n, T, t), \quad (4.49)$$

donde

$$\bar{f}_m(n, T, t) = \begin{cases} \left(2 \frac{(t-nT)}{T} - 1 \right)^m, & \text{para } nT \leq t < (n+1)T \\ 0, & \text{otros casos,} \end{cases} \quad (4.50)$$

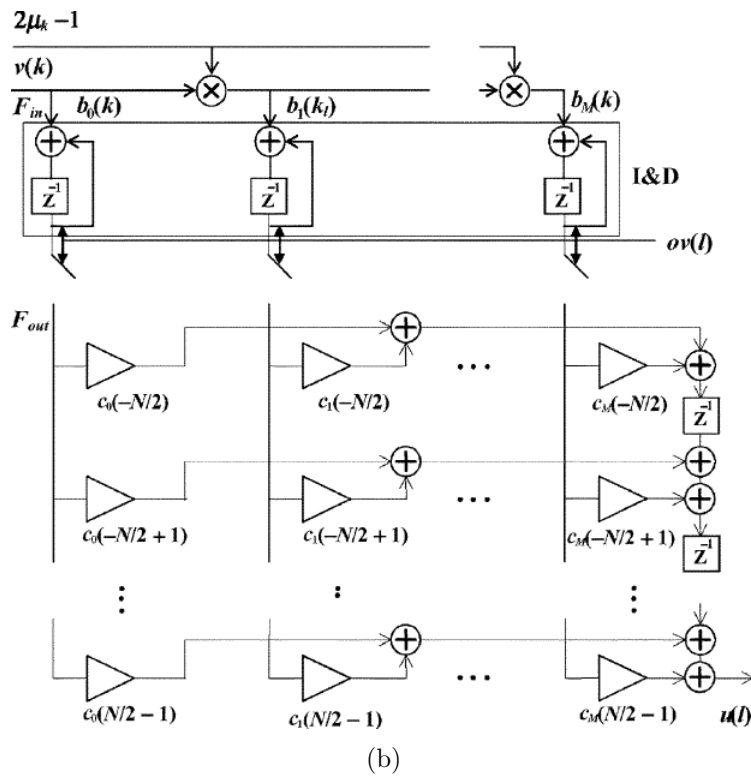
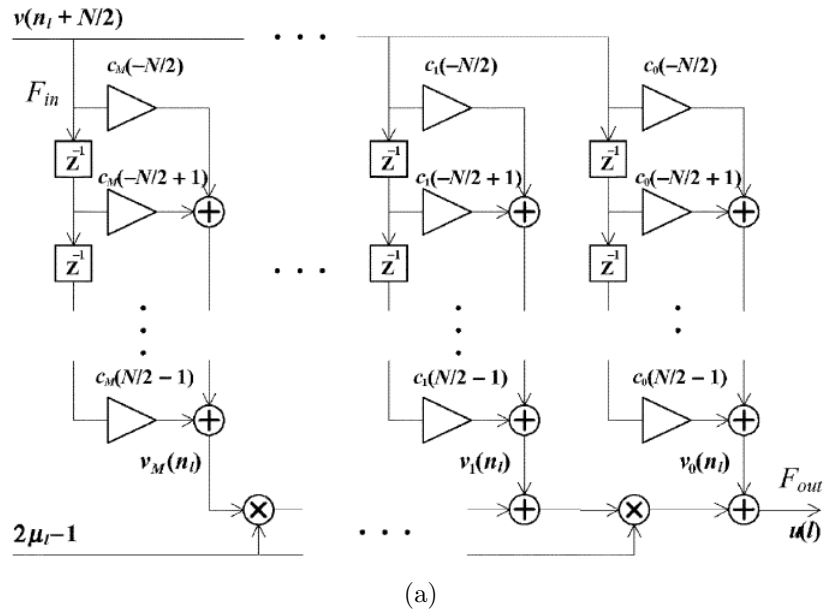


Figura 4.13: Estructura Farrow a) Modificada b) Modificada transpuesta.

$$c_m(k) = \sum_{l=m}^M \left(\frac{1}{2}\right)^l d_l(-N/2 + 1 + k), \quad (4.51)$$

$$c_m(N - 1 - n) = \begin{cases} c_m(n), & \text{para } m \text{ par} \\ -c_m(n), & \text{para } m \text{ impar,} \end{cases} \quad (4.52)$$

Los coeficientes $c_m(n)$ se obtienen a partir de los coeficientes $d_m(n)$ usando la ecuación (4.51) basándose en la estructura Farrow original. Además, los coeficientes presentan una relación entre ellos que permite obtener una simetría o antisimetría en los $M + 1$ filtros FIR requeridos para su implementación, lo que reducirá la cantidad de coeficientes a la mitad. La ecuación (4.50) puede ser implementada usando la estructura Farrow modificada para $T = T_{in}$, o usando la estructura Farrow modificada transpuesta para $T = T_{out}$.

La respuesta en frecuencia para un integrador ideal de orden N es:

$$H_a(\omega) = \left(\frac{\text{sen}(\omega T)}{\omega T} \right)^N. \quad (4.53)$$

Es posible ubicar una cantidad específica de ceros en las frecuencias múltiplos de la frecuencia de entrada. Además también es posible ubicar ceros en las frecuencias múltiplos de la frecuencia de salida. Basado en lo anterior, se desean ubicar L ceros en las frecuencias múltiplos de F_{in} y K ceros en las frecuencias múltiplos de F_{out} , lo que resulta en la siguiente respuesta en frecuencia deseada, que corresponde a la cascada de un interpolador Farrow modificado que funciona en frecuencia F_{in} y un interpolador Farrow transpuesto funcionando a frecuencia F_{out} :

$$H(\omega) = H_1(\omega)H_2(\omega) = \left(\frac{\text{sen}(\omega T_{in})}{\omega T_{in}} \right)^L \left(\frac{\text{sen}(\omega T_{out})}{\omega T_{out}} \right)^K. \quad (4.54)$$

La ecuación (4.54) puede ser reescrita de dos formas diferentes, lo que deriva en los dos casos presentados a continuación.

Estructura caso I

Este caso resulta de reescribir la ecuación (4.54) de la siguiente forma:

$$H(\omega) = \left(\frac{T_{in}}{T_{out}} \right)^K \frac{1}{\text{sen}(\omega T_{in})^K} \left(\frac{\text{sen}(\omega T_{in})}{\omega T_{in}} \right)^{L+K} \text{sen}(\omega T_{out})^K. \quad (4.55)$$

La estructura del caso I, presentada en la Fig. 4.14, tiene K etapas de integradores en cascada funcionando a la frecuencia de entrada F_{in} , la estructura Farrow modificada con $N = L + K$ y $M = L + K - 1$, y K etapas de filtros *comb* en cascada funcionando a la frecuencia de salida F_{out} . El interpolador Farrow modificado tiene $(L + K)^2/2$ multiplica-

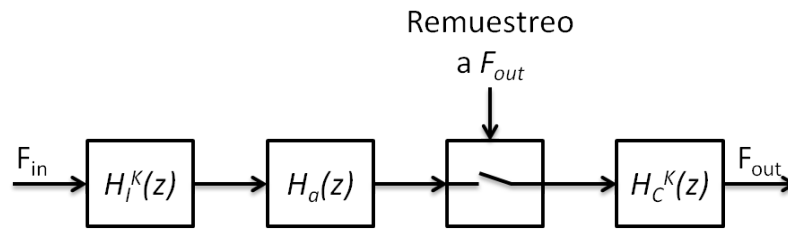


Figura 4.14: Estructura para el caso I.

dores de punto fijo funcionando a la frecuencia de entrada y $L + K$ multiplicadores por μ funcionando a la frecuencia de salida.

Estructura caso II

Este caso se obtiene al reescribir la ecuación (4.54) de la forma siguiente:

$$H(\omega) = \left(\frac{T_{out}}{T_{in}}\right)^L \text{sen}(\omega T_{in})^L \left(\frac{\text{sen}(\omega T_{out})}{\omega T_{out}}\right)^{L+K} \frac{1}{\text{sen}(\omega T_{in})^L}. \quad (4.56)$$

La estructura para el caso II, presentada en la Fig. 4.15, posee L etapas en cascada de filtros *comb* funcionando a la frecuencia de entrada F_{in} , la estructura Farrow modificada transpuesta con $N = L + K$ y $M = L + K - 1$, y L etapas integradoras en cascada funcionando a la frecuencia de salida F_{out} . Por parte del interpolador Farrow modificado transpuesto se tienen $(L + K)^2/2$ multiplicadores de punto fijo funcionando a la frecuencia de salida y $L + K$ multiplicadores por μ funcionando a la frecuencia de entrada.

Ejemplo 4.6: Se presenta la respuesta en magnitud para el caso de un factor de conversión $R = 10\frac{4}{11}$. Para el caso I se tiene un integrador y un filtro de interpolación lineal trabajando en frecuencia de entrada y una sección *comb* en frecuencia de salida. Solamente se requiere una multiplicación por μ en alta frecuencia. Para el caso II se requiere una etapa *comb* que funciona en alta frecuencia, un filtro de interpolación lineal transpuesto y un integrador que trabajan en alta frecuencia. Para este caso el multiplicador por μ trabaja en la más baja frecuencia de muestreo.

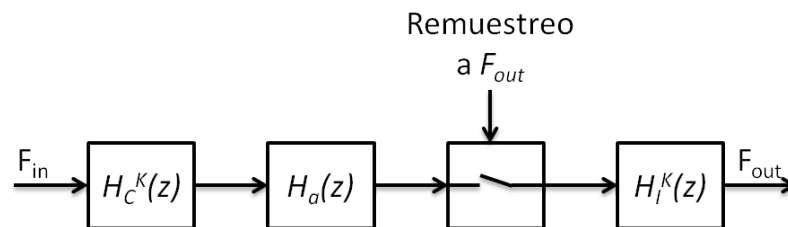


Figura 4.15: Estructura para el caso II.

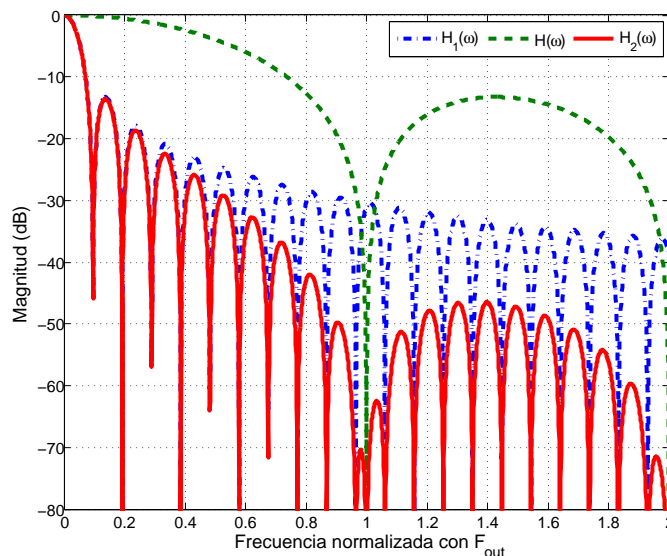


Figura 4.16: Respuesta en magnitud del ejemplo para $R = 10\frac{4}{11}$.

4.3. Métodos basados en filtros CIC y filtros auxiliares

En esta sección se presenta algunos métodos basados en el filtro CIC acompañado de filtros auxiliares FIR o IIR, que generalmente se encargan de proveer la atenuación requerida en la banda de rechazo o de disminuir la caída en la banda de paso. Estos métodos ofrecen una mejora en la SNR pero a la vez incrementan la complejidad computacional, incluso hasta el punto de introducir multiplicaciones.

4.3.1. Inclusión de un filtro IIR y modificación de la ubicación de los ceros del filtro CIC [14]

El método consiste en la inclusión de un filtro IIR auxiliar para mejorar la caída en la banda de paso y una modificación de la ubicación de los ceros del filtro CIC con el fin de aumentar la atenuación de las bandas donde se presenta el aliasing. Para ello se considera la respuesta en magnitud para el filtro CIC dada por la ecuación (4.64).

$$H(\omega) = \left[\frac{\text{sen}\left(\frac{\omega RM}{2}\right)}{\text{sen}(\omega/2)} \right]^N, \quad (4.57)$$

donde el factor de decimación total es RM , y M puede ser 1 ó 2.

Los ceros de la ecuación anterior se encuentran localizados en la frecuencia normalizada

$k/(RM)$ para $k = 1, 2, \dots, \lfloor RM/2 \rfloor$. Si la frecuencia de paso del filtro es ω_p , entonces las bandas de aliasing se encuentran entre las frecuencias $k/RM - \omega_p$ y $k/RM + \omega_p$.

Se propone modificar la respuesta del filtro obteniendo la función de transferencia dada como:

$$H(z) = \left(\frac{1 - bz^{-R} + z^{-2R}}{1 - az^{-1} + z^{-2}} \right). \quad (4.58)$$

Los ceros del filtro son movidos a la frecuencia $k/R \pm \alpha$ donde α tiene un valor cercano o más pequeño que ω_p . De esta forma las señales localizadas entre $k - k\omega_p$ y $k + k\omega_p$ tienen una mayor atenuación. La estructura modificada se presenta en la Fig. 4.17.

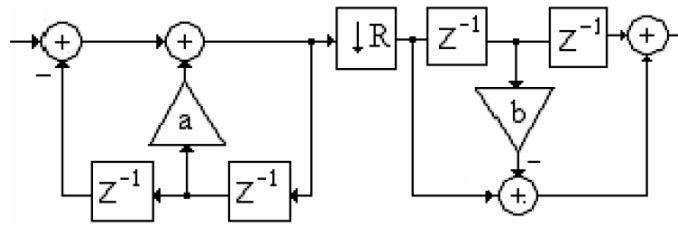


Figura 4.17: Estructura usada para la modificación de los ceros del filtro CIC.

El incremento de la cantidad de retardos en el filtro *comb* permite obtener mayor cantidad de ceros, generando un incremento en la atenuación en las bandas de rechazo del filtro.

Ejemplo 4.7: Para clarificar el uso del método se presenta el caso para $M = 25$ y $\omega_p = 1/8$. En la Tabla 4.6 se presentan las atenuaciones tanto en la banda de paso como en la frecuencia $1/R - \omega_p$, que es donde empieza la banda de rechazo del filtro. Se observa que para un filtro de orden 3, al agregar retardos adicionales en el último filtro *comb* se tiene una atenuación mayor, pero a su vez, la caída en la banda de paso aumenta. Estas magnitudes han sido calculadas usando la ecuación (4.59).

Retardos	Magnitud en ω_p (dB)	Magnitud en $(1/R) - \omega_p$ (dB)
$M = [111]$	-0.67	-51.33
$M = [112]$	-1.36	-52.01
$M = [113]$	-2.56	-53.21

Tabla 4.6: Magnitudes importantes para diferentes retardos en el último filtro *comb*

$$H_c(z) = \frac{\text{sen}\left(\frac{\omega RM_1}{2}\right) \dots \text{sen}\left(\frac{\omega RM_N}{2}\right)}{\text{sen}(\omega/2)^N}. \quad (4.59)$$

El cálculo de a y b se realiza usando las siguientes ecuaciones:

$$a = 2 \cos(2\pi\alpha/R), \quad (4.60)$$

$$b = 2 \cos(2\pi\alpha). \quad (4.61)$$

Ejemplo 4.8: Usando el ejemplo 4.7, se selecciona $\alpha = 0.85\omega_p$, obteniendo $a = 1.9993$ y $b = 1.5706$. Si además se selecciona un filtro *comb* con tres retardos se tendrá la función de transferencia dada por la ecuación (4.62).

$$H(z) = \left(\frac{1 - 1.5706z^{-25} + z^{-50}}{1 - 1.9993z^{-1} + z^{-2}} \right) \left(\frac{1 - z^{-75}}{1 - z^{-1}} \right), \quad (4.62)$$

El filtro auxiliar tendrá ceros en 1 y polos en $\pm 3\omega_p$ con una ganancia de $2.5dB$ en ω_p , lo que resulta en una función de transferencia dada por

$$H(z) = \left(\frac{1 - 2z^{-1} + z^{-2}}{1 - 0.5511z^{-1} + 0.5184z^{-2}} \right). \quad (4.63)$$

En la Fig. 4.18 se presenta la respuesta en magnitud obtenida para el ejemplo anterior. La atenuación ha aumentado pero la inclinación en la banda de paso también lo ha hecho. Para mejorar este aspecto puede usarse un filtro FIR que compense la caída en la banda de paso. Gracias a la inclusión de retardos adicionales en el filtro *comb*, el orden del filtro CIC puede reducirse generando un ahorro en el consumo de potencia, a pesar de la inclusión de los nuevos elementos. La principal desventaja del método es la necesidad de realizar multiplicaciones no enteras incrementando su complejidad de implementación.

4.3.2. Implementación Eficiente de Conversores de Frecuencia de Muestreo Multi-etapa [13]

Este método se basa en la realización de la conversión de la frecuencia de muestreo en varias etapas. Si el factor de decimación M es factorizable como $M = N \times R$, entonces la decimación puede ser realizada usando la cascada de dos decimaciones por factores N y R , donde la primera etapa la realiza un filtro CIC y la segunda etapa un filtro FIR como se muestra en la Fig. 4.19.

De igual forma, la interpolación puede realizarse en dos etapas (Fig. 4.20). Sin embargo a diferencia de la decimación, en la que el filtro CIC se encuentra en la etapa inicial, aquí se encuentra en la etapa final. El filtro $T(z)$ se encarga de proporcionar la banda de transi-

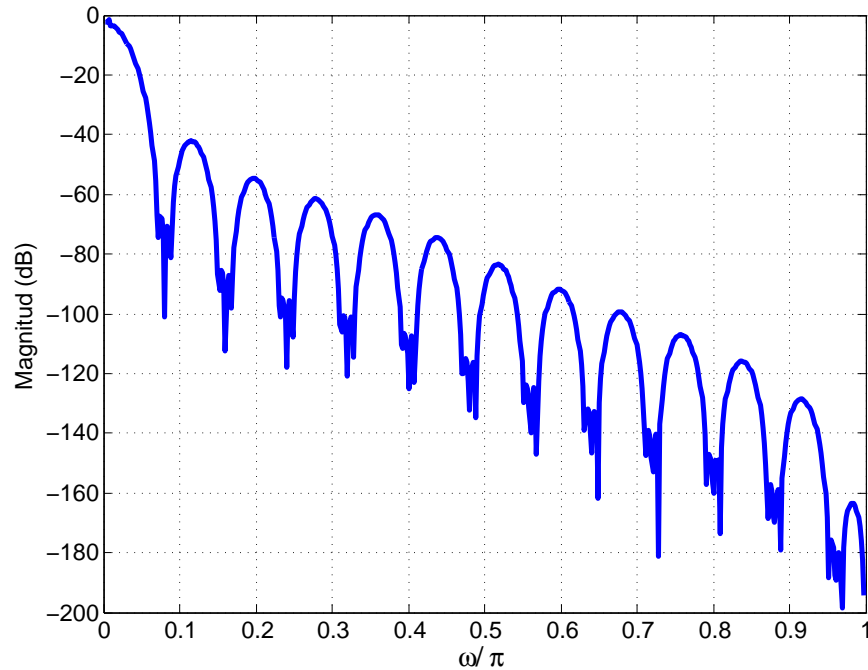
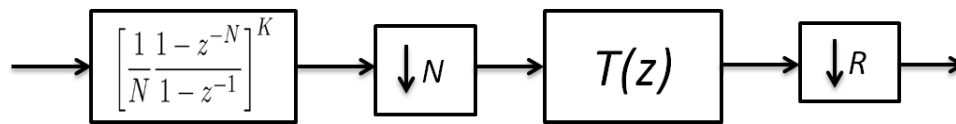
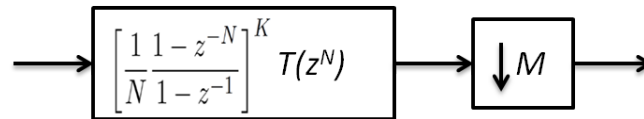


Figura 4.18: Respuesta en magnitud para el ejemplo de diseño usando el método [14].



(a) Implementación en cascada.



(b) Implementación en una única etapa.

Figura 4.19: Decimación.

ción y compensar la caída en la banda de paso. Por otro lado el filtro CIC posee ceros en las bandas de paso indeseadas del filtro expandido $T(z^N)$ que garantizan la atenuación del filtro.

Finalmente, la respuesta en magnitud del filtro de decimación de dos etapas está dado por:

$$|H_1(z)| = |H(z)T(z^N)|. \quad (4.64)$$

Ejemplo 4.9: Para el caso de $M = 10$, y usando $N = 2$ y $R = 5$, la respuesta en magnitud

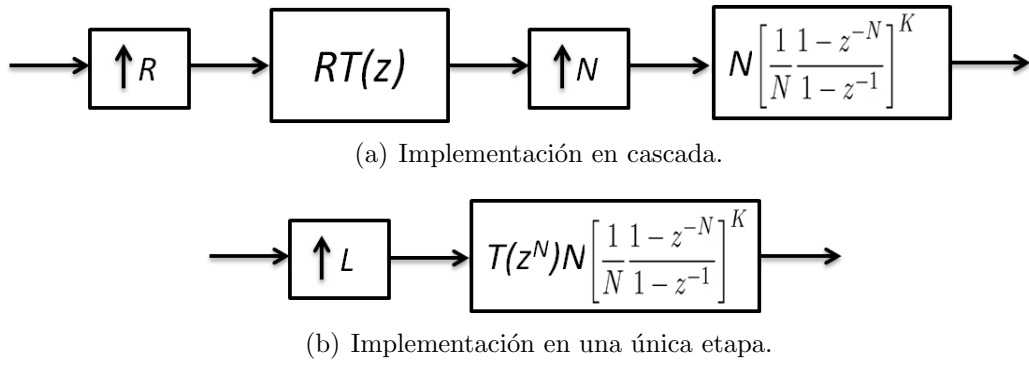


Figura 4.20: Interpolación.

se presenta en la Fig. 4.21.

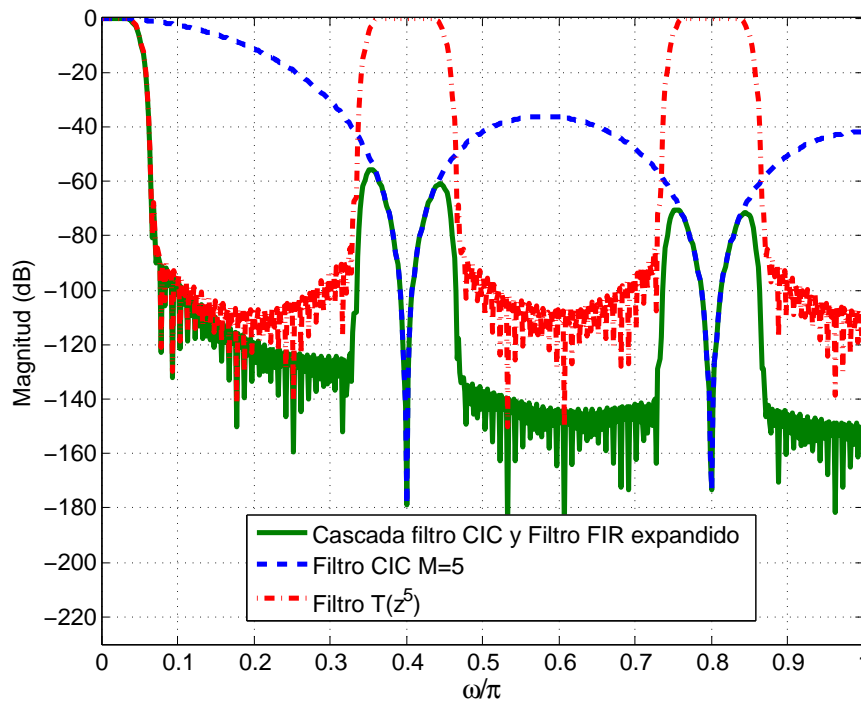


Figura 4.21: Respuesta en magnitud para el ejemplo del interpolador multi-etapa de [13].

Capítulo 5

Consideraciones y condiciones para el método propuesto

En este capítulo se presentará un algoritmo para la selección de los retardos usados para el filtro CIC modificado. Se muestra en que caso resulta más conveniente usar el filtro CIC triangular escalonado, con respecto al tamaño del factor de decimación. Además, se describe el uso del filtro coseno para mejorar la respuesta en magnitud con lo introducción de un cero adicional en la frecuencia, donde se presenta el peor caso de *aliasing*. Finalmente, se presentarán las condiciones que debe cumplir un filtro CIC para su realización eficiente basada en el método descrito en [9].

5.1. Selección de los retardos para el filtro CIC modificado [15]

En cuanto a la selección de los retardos usados para el filtro CIC modificado de [15] se busca que la selección se realice obteniendo la mejor respuesta en magnitud posible.

En primer lugar, basados en múltiples simulaciones en Matlab que mostraron que se obtiene una mejor respuesta en magnitud cuando estos retardos se distribuyen uniformemente (ejemplo 5.1). Además, es recomendable evitar usar retardos grandes, ya que introducen ceros cercanos a la frecuencia de paso ω_p , lo que empeora la SNR. En este caso se evitará tener ceros bajo la frecuencia donde se presenta el peor caso de *aliasing* ω_1 . Estas frecuencias están definidas por la ecuaciones (4.4).

Ejemplo 5.1: Se presenta en la Fig. 5.1 la respuesta en magnitud para diferentes selecciones de los retardos para el filtro presentado en [15] de orden 4 con $M = 10$. Como se observa el filtro con $D_1 = 14$, $D_2 = 16$, $D_3 = 18$ y $D_4 = 20$, posee un cero en 0.1, lo que genera una caída pronunciada en la banda de paso del filtro reduciendo, generando una atenuación en las componentes con más alta frecuencia dentro de la banda de interés.

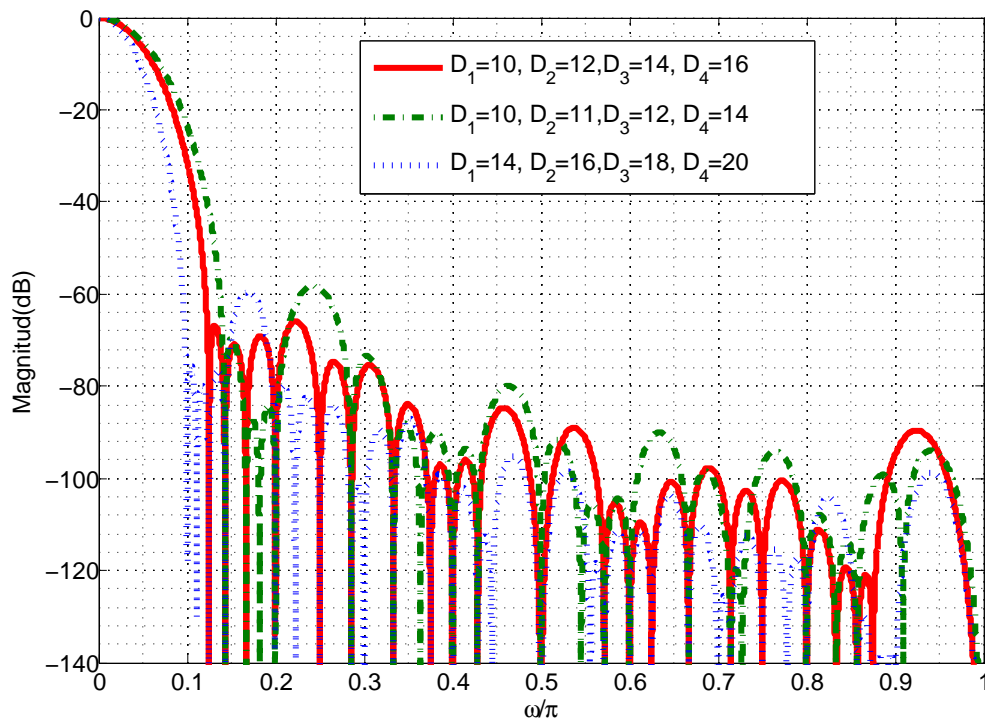


Figura 5.1: Respuestas en magnitud para algunas selecciones de los retardos para el filtro CIC modificado de orden 4 con $M = 10$.

A continuación se presenta el procedimiento de selección de los retardos para el caso en que el factor de decimación se puede obtener como el producto de dos factores M_1 y M_2 , lo que hace posible usar el filtro CIC triangular escalonado [6]. Se propone realizar la selección de la siguiente forma: Se eligen los retardos iniciando desde M , seguidos por múltiplos consecutivos del factor de decimación M_1 . Si el rango de selección de estos retardos es muy grande se utilizan múltiplos más grandes que de igual manera maximicen la uniformidad de la respuesta en magnitud. Se busca evitar que alguno de los subfiltros tenga ceros a frecuencias menores que la frecuencia ω_1 . Esta condición se obtiene cuando los primeros ceros de todos los subfiltros se encuentran por encima o en la frecuencia deseada. La frecuencia del primer cero de los subfiltros está dada por la ecuación (5.1).

$$\omega_{pc} = \frac{2\pi}{D_i}. \quad (5.1)$$

Se debe cumplir que,

$$\omega_{pc} \geq \omega_1. \quad (5.2)$$

Usando las ecuaciones (4.4), (5.1) y la condición (5.2), que evita que existan ceros por debajo de la frecuencia ω_1 , se tiene:

$$\frac{2\pi}{D_i} \geq \frac{2}{M} - \frac{3}{4M} \quad (5.3)$$

Despejando el valor de los retardos D_i , se llega a la condición que define el valor máximo para los mismos. Con este valor se garantiza que no existan ceros por debajo de la frecuencia de interés (en este caso ω_1).

$$D_i \leq 1.6M. \quad (5.4)$$

A partir de esta condición se recomienda que el retardo D_K tenga un valor máximo de $1.6M$. El nuevo rango de selección es desde M hasta $1.6M$, lo que permite un máximo orden del filtro CIC modificado usando múltiplos de M_1 , definido por la ecuación (5.5).

$$K_{max} = \text{int}[0.6M_2 + 1], \quad (5.5)$$

donde $\text{int}[\cdot]$ es la parte entera del argumento. Es posible elegir el orden del filtro CIC modificado conociendo el valor de K_{max} . La ecuación (5.6) define los valores para los retardos D_i .

$$D_i = M + (i - 1)PM_1, \quad (5.6)$$

donde $i = 1, \dots, K$ y P es un entero que permitirá la distribución uniforme de los retardos para ordenes del filtro CIC modificado más pequeños o más grandes dentro del rango permitido por la condición (5.2). Al hallar el valor del retardo D_K con la ecuación (5.6) se obtiene:

$$D_K = M + (K - 1)PM_1, \quad (5.7)$$

Al combinar la condición (5.6), que define al valor máximo del retardo más grande y la ecuación (5.7), se tiene:

$$M + (K - 1)PM_1 \leq 1.6M. \quad (5.8)$$

Después de realizar algunas operaciones matemáticas se obtiene la condición (5.9), que permite definir el valor adecuado del parámetro P :

$$P \leq 0.6 \frac{M_2}{(K - 1)}. \quad (5.9)$$

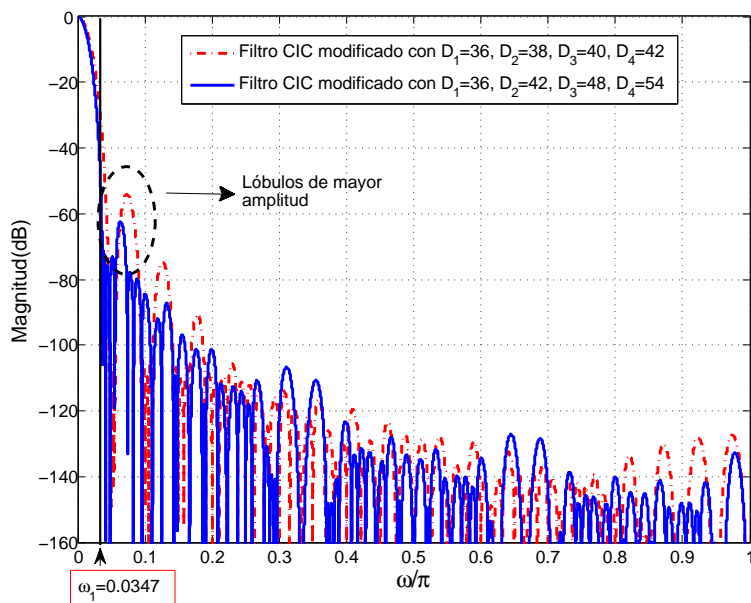
Ejemplo 5.2: Se presenta la selección de los retardos para el filtro CIC modificado con $L = 5$ y $M = 36$. Si se seleccionan $M_1 = 2$ y $M_2 = 18$, se tiene un $K_{max} = 11$. Conociendo este valor se procede a elegir K . Para este ejemplo se define como 4, lo que nos daría que P debe ser un entero menor o igual a 3.6, garantizando que para $P = 3$ se obtenga una mejor distribución de los retardos y una mejor respuesta en magnitud que al usar $P = 1$ o $P = 2$. Los retardos obtenidos al usar la ecuación (5.6) son: $D_1 = 36$, $D_2 = 42$, $D_3 = 48$, $D_4 = 54$. Las respuestas en magnitud usando $P = 1$ y $P = 3$ con el zoom que incluye los lóbulos de mayor amplitud y la frecuencia ω_1 se presentan en la Fig. 5.2. La línea vertical señala la frecuencia ω_1 . Se observa que los ceros de los subfiltros no se encuentran por debajo de esta frecuencia.

5.2. Variación de la mejora del filtro CIC triangular escalonado con respecto al factor de decimación [6]

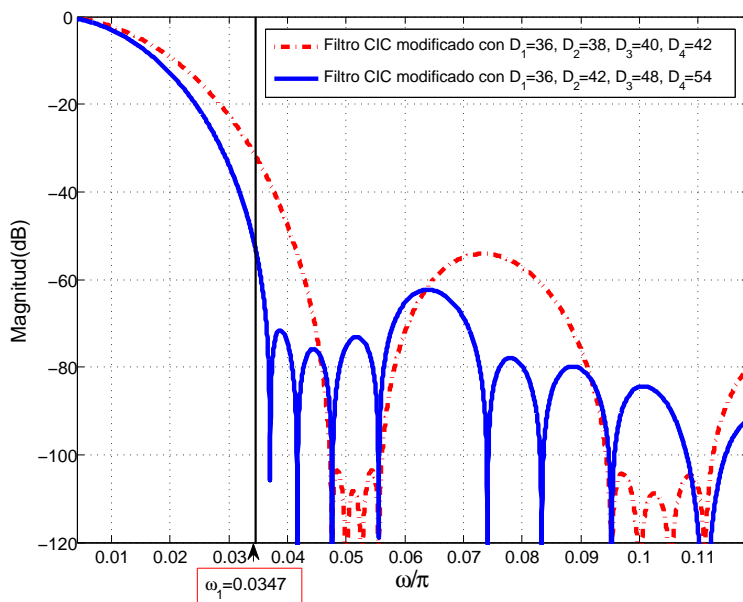
Es importante tener en cuenta que para el método descrito en [6] se obtienen mejores resultados para $M_1 < M_2$ y $K_1 \geq 2K_2$. Para analizar en que medida depende la mejora obtenida usando el método presentado en [6], se elige un M_1 y se varía el factor de decimación M , dejando los demás parámetros fijos. La mejora se mide con respecto a la atenuación del filtro CIC con factor de cascada 1.

En la Fig. 5.3 se presenta un barrido para M desde 10 hasta 60. Se usa $M_1 = 2$, $M_1 = 3$ y $M_1 = 4$. Se observa que se obtiene una mejora más grande para valores grandes del factor de decimación. Sin embargo, la diferencia entre las mejoras no resulta muy significativa. Para mostrar que este comportamiento se conserva para diferentes valores de K_1 y K_2 , se presentan los resultados obtenidos usando $K_1 = 10$ y $K_2 = 5$.

En la Fig. 5.4 se muestra que los resultados anteriores no dependen de la selección de los



(a) Respuestas en magnitud



(b) Zoom

Figura 5.2: Respuestas en magnitud para ilustrar la selección de los retardos D_i .

parámetros K_1 y K_2 . Es claro que para los parámetros de cascada usados, se tendrá una mejora alta en la SNR. Sin embargo, en términos generales la diferencia entre las mejoras

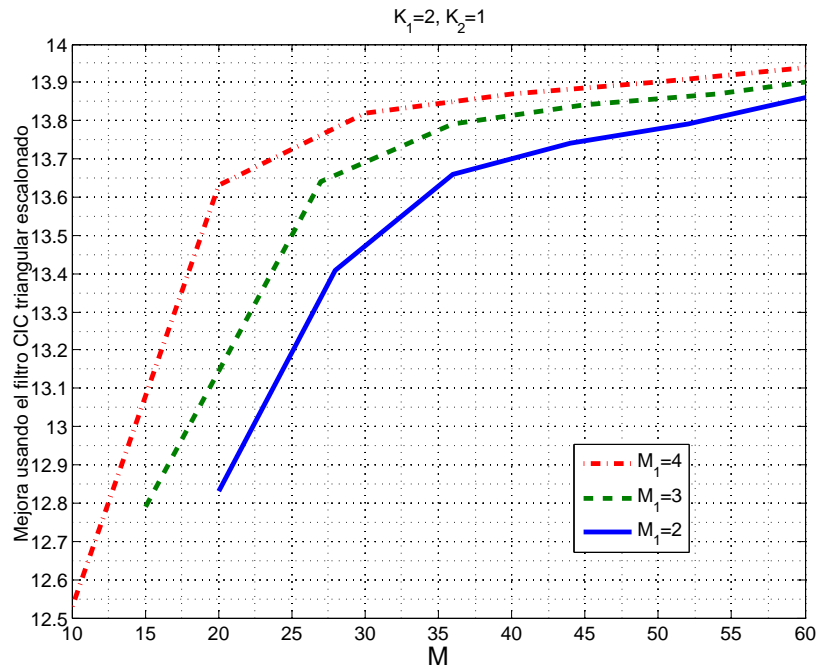


Figura 5.3: Variación de la mejora obtenida con el factor de decimación.

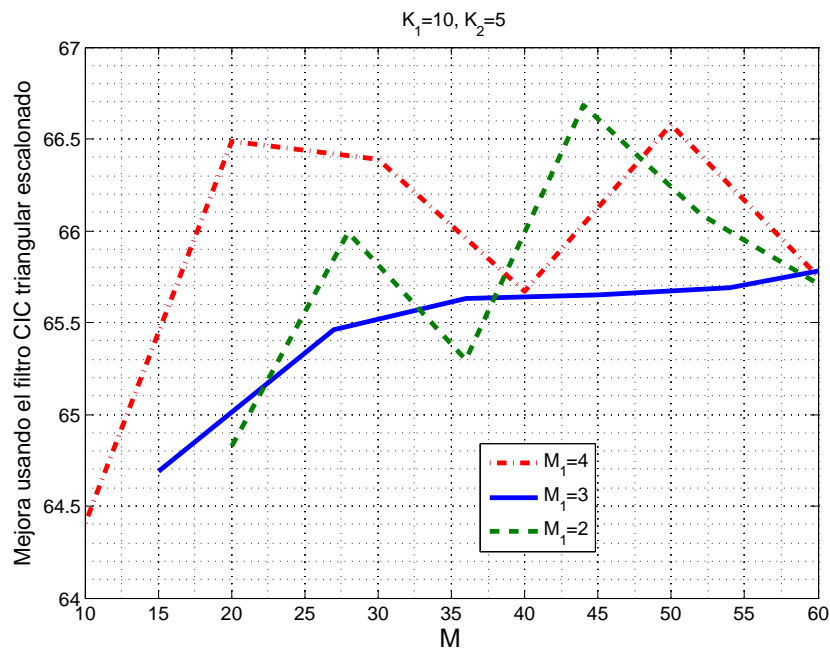


Figura 5.4: Variación de la mejora obtenida con el factor de decimación.

para los diferentes valores de M es pequeña.

Ejemplo 5.3: En la Fig. 5.3 se observa que para $M = 20$ con $M_1 = 2$, $M_2 = 10$, $K_1 = 2$

y $K_2 = 1$. El filtro CIC triangular escalonado presenta una mejora de 12.85dB, mientras para $M = 40$ con $M_1 = 2$, $M_2 = 20$, $K_1 = 2$ y $K_2 = 1$ se tiene una mejora de 13.7dB, aproximadamente 1dB de diferencia. También puede observarse que para $M = 40$ con $M_1 = 4$ y $M_2 = 10$, la mejora es de 13.8dB, aproximadamente 0.1dB de diferencia.

5.3. Análisis de la atenuación de los subfiltros del método del filtro CIC modificado [15]

Por otro lado, es importante analizar cuales de los subfiltros resultantes del método descrito en [15] presentan una mejor o peor atenuación en la banda de rechazo. De esta forma se tendrá certeza sobre cuales de ellos presentan una contribución más baja a la atenuación total del filtro CIC modificado. Para ello se realizará un breve análisis de la cantidad de ceros distribuidos en la banda de rechazo.

La ecuación (5.10) define la ubicación de los ceros de los subfiltros resultantes del método del filtro CIC.

$$\omega_i = 2\pi/D_i, \quad (5.10)$$

donde $i = 1, \dots, K$. La cantidad de ceros distribuidos en toda la banda para cada subfiltro es $\frac{D_i}{2}$ para D_i par y $\frac{D_i-1}{2}$ para D_i impar, dado que D_{i+1} es mayor a D_i . La cantidad de ceros para los primeros D_i es menor que para los últimos, lo que consecuentemente representa una peor atenuación para los primeros filtros. Matemáticamente se tiene que la cantidad de ceros para retardos consecutivos está dada por:

$$C_i = \frac{D_i}{2} \quad \text{para } D_i \text{ par.} \quad (5.11)$$

$$C_{i+1} = \frac{D_{i+1} - 1}{2} \quad \text{para } D_i \text{ par.} \quad (5.12)$$

Además

$$C_i = \frac{D_i - 1}{2} \quad \text{para } D_i \text{ impar.} \quad (5.13)$$

$$C_{i+1} = \frac{D_{i+1}}{2} \quad \text{para } D_i \text{ impar.} \quad (5.14)$$

Sabiendo que

$$D_{i+1} > D_i, \tag{5.15}$$

finalmente se tiene:

$$C_i < C_{i+1}. \tag{5.16}$$

Los C_{i+1} ceros del subfiltro con retardo D_{i+1} , se distribuyen uniformemente en toda la banda de frecuencia digital y se ubican en las frecuencias múltiplos de $2\pi/D_{i+1}$. Se puede concluir que un filtro CIC para un factor de decimación más grande presenta una mayor cantidad de ceros, lo que equivale a una atenuación mayor en la banda de rechazo.

Ejemplo 5.4: En la Fig. 5.5 se presenta una comparación para los subfiltros resultantes de aplicar el método del filtro CIC modificado a un filtro con factor de conversión $\frac{11}{20}$ y $K = 5$. Se observa que la peor atenuación la posee el subfiltro con $D_1 = 20$. Este posee 10 ceros en toda la banda, mientras que el subfiltro con $D_5 = 30$ posee 15 ceros que proporcionan una atenuación mayor.

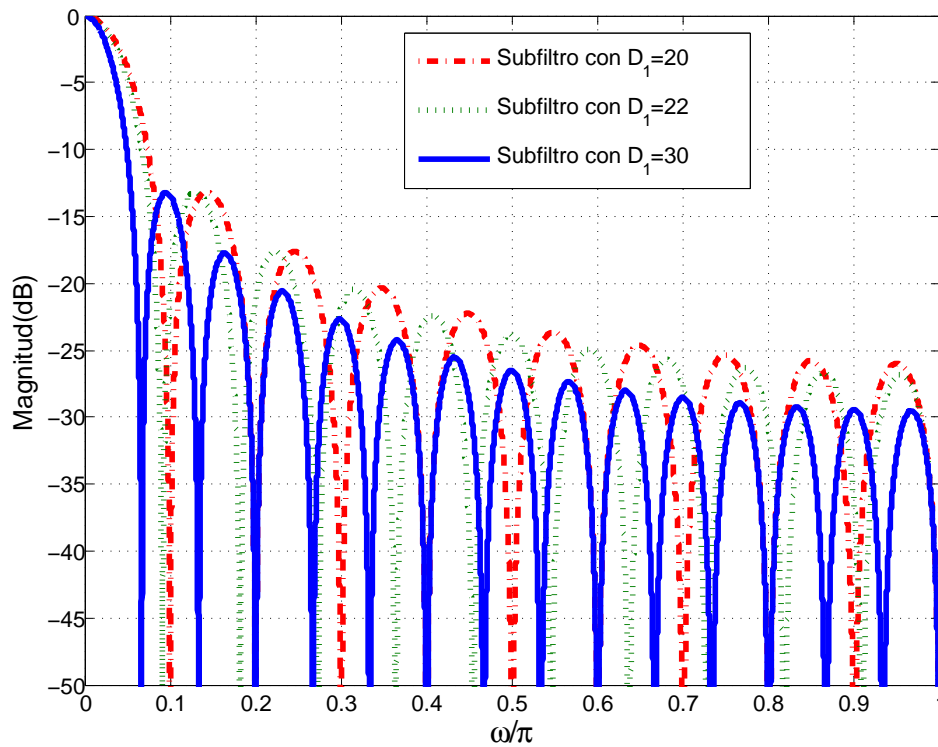


Figura 5.5: Comparación de algunos subfiltros para el método CIC modificado con $M = 20$ y $K = 5$.

5.4. Mejora de la SNR usando el filtro coseno expandido

El filtro coseno expandido es usado típicamente para realizar la inclusión de ceros en la frecuencia deseada. En este caso se usará para mejorar la atenuación en la frecuencia donde se presenta el peor caso de *aliasing*.

Usando las ecuaciones (2.14), es posible ubicar el primer cero del filtro coseno expandido en la frecuencia ω_1 , con el fin de mejorar la atenuación en la banda frecuencia donde se presenta el peor caso de aliasing. La ecuación (5.17) define el valor del factor de expansión R ,

$$R = \text{int}[\pi/\omega_1], \quad (5.17)$$

Ejemplo 5.5: Para mostrar la ubicación del primer cero en la frecuencia deseada y considerando frecuencia normalizada entre 0 y 1, si se desea obtener una $\omega_1 = 0.125$ entonces R debe tener un valor de 8. La respectiva respuesta en magnitud se observa en la Fig. 5.6.

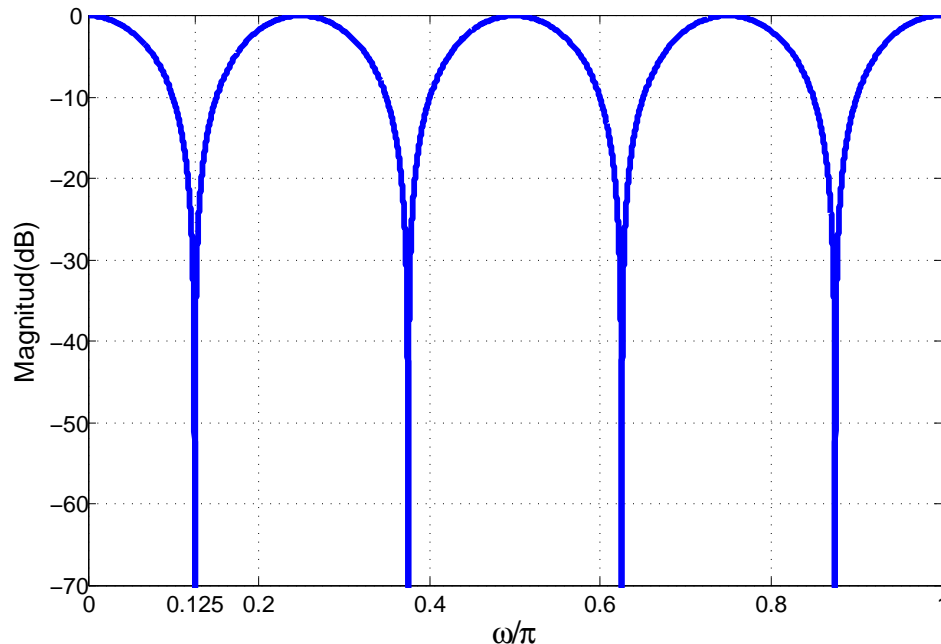


Figura 5.6: Ejemplo del filtro Coseno Expandido para $\omega_1 = 0.125$.

Capítulo 6

Método propuesto

Bajo las condiciones presentadas en el capítulo 5, es posible definir un procedimiento que permite el diseño de filtros para conversión racional de frecuencia de muestreo basado en las estructuras propuestas en [15] y [6], que permite mejorar la SNR con respecto a la propuesta presentada en [6], reduciendo o aumentando ligeramente la cantidad de APOS requeridas para su implementación.

6.1. Descripción del método

Primer paso

Consiste en utilizar el método del filtro CIC modificado partiendo del filtro CIC original de orden K , obteniendo la función de transferencia de la ecuación (6.1).

$$H_m(z) = \frac{(1 - z^{-D_1})(1 - z^{-D_2}) \dots (1 - z^{-D_K})}{(1 - z^{-1})^K}, \quad (6.1)$$

donde el retardo $D_1 = M = M_1 M_2$ y los retardos restantes se obtienen usando las ecuaciones (5.5)-(5.9).

La ecuación (5.5) permite obtener el máximo orden que podrá tenerse para el filtro cuando se usan múltiplos consecutivos del factor de decimación M_1 . Sin embargo, en general el K seleccionado será menor que K_{max} , así que los retardos pueden distribuirse mejor usando un factor entero P , que permitirá que estos sean múltiplos consecutivos de PM_1 en lugar de M_1 .

Ejemplo 6.1: Se tiene un factor de decimación $M = 20$. Si se elige $M_1 = 2$ y $M_2 = 10$,

$K_{max} = 7$, que es el máximo orden que puede tener el filtro con el método propuesto y corresponde a la cantidad de subfiltros usados. Si se selecciona $K = 4$ entonces $P = 2$. El primer retardo es $D_1 = 20$, el segundo es $D_1 + 1 \times P \times M_1 = 20 + 1 \times 2 \times 2 = 24$, el tercer retardo es $D_1 + 2 * P * M_1 = 20 + 2 * 2 * 2 = 28$ y el cuarto retardo es $D_1 + 2 * P * M_1 = 20 + 3 * 2 * 2 = 32$.

De esta forma usando las ecuaciones (5.6) y (6.1), se tendrá un conjunto de retardos que garantizarán una buena respuesta en magnitud del filtro CIC modificado.

Paso 2

Con los retardos para los subfiltros seleccionados, se procede a aplicar de forma selectiva la aproximación al filtro CIC triangular escalonado. Como se mostró en el capítulo 5, resulta conveniente aplicar la mejora en los filtros que presentan una peor respuesta en magnitud. Además se mostró que no existe una diferencia significativa si se aplica en los primeros o en los últimos filtros. Debe tenerse en cuenta que los filtros con factores de decimación más pequeños, poseen una pobre atenuación en la banda de rechazo, generando que la respuesta total de filtro CIC modificado tenga una peor atenuación y por consiguiente una baja SNR.

Aplicando la aproximación triangular escalonada sobre los primeros N_1 filtros se obtiene:

$$H_p(z) = \prod_{i=1}^{N_1} H_{TE_i}(z) \prod_{j=1}^{K-N_1} H_j(z). \quad (6.2)$$

Es importante clarificar el hecho de que cada subfiltro al que se le aplique la aproximación al filtro CIC triangular escalonado puede poseer diferentes parámetros de cascada K_1 y K_2 . Por esto se incluirá una nueva notación que permitirá reconocer el subíndice del subfiltro que se use y también los parámetros de cascada del filtro CIC triangular escalonado particular obtenido para cada uno de ellos. Como ejemplo de la notación para el segundo subfiltro que sería D_2 , se deben incluir los parámetros de cascada de su aproximación triangular escalonada correspondiente, lo que se logra incluyendo un segundo subíndice. Ahora se tienen dos parámetros de cascada para el subfiltro 2 que serían K_{21} y K_{22} . Usando esta notación se tiene:

$$H_{TE1}(z) = \left[\frac{1}{M} \frac{(1 - z^{-M})}{(1 - z^{-M_1})} \right]^{K_{11}} \left[\frac{1}{M_1} \frac{(1 - z^{-M_1})}{(1 - z^{-1})} \right]^{K_{12}}, \quad (6.3)$$

$$H_{TEi}(z) = \left[\frac{1}{D_i} \frac{(1 - z^{-D_i})}{(1 - z^{-M_1})} \right]^{K_{i1}} \left[\frac{1}{M_1} \frac{(1 - z^{-M_1})}{(1 - z^{-1})} \right]^{K_{i2}}, \quad (6.4)$$

$$H_j(z) = \frac{(1 - z^{-D_{N_1+j}})}{(1 - z^{-1})}. \quad (6.5)$$

Además

$$K_{i1} \geq 2K_{i2}. \quad (6.6)$$

Ejemplo 6.2: Usando el ejemplo 6.1 se aplica el método del filtro CIC triangular escalonado, únicamente sobre los primeros subfiltros con retardos D_1 y D_2 . Estos dos poseen factores de cascada independientes K_{11} , K_{12} , K_{21} y K_{22} . Estos factores permiten la modificación de la respuesta en magnitud del filtro obtenido para que se ajuste a las especificaciones requeridas. Para este caso se tiene:

$$H_{TE1}(z) = \left[\frac{1}{20} \frac{(1 - z^{-20})}{(1 - z^{-2})} \right]^{K_{11}} \left[\frac{1}{2} \frac{(1 - z^{-2})}{(1 - z^{-1})} \right]^{K_{12}}, \quad (6.7)$$

$$H_{TE2}(z) = \left[\frac{1}{24} \frac{(1 - z^{-24})}{(1 - z^{-2})} \right]^{K_{21}} \left[\frac{1}{2} \frac{(1 - z^{-2})}{(1 - z^{-1})} \right]^{K_{22}}. \quad (6.8)$$

Paso 3

Es posible obtener una mejora adicional si se utiliza el filtro coseno expandido para incluir un cero adicional en la banda de frecuencia donde ocurre el peor caso de aliasing ω_1 . La inclusión del filtro coseno expandido se hará cuando la frecuencia del primer cero del subfiltro con retardo D_K sea diferente de la frecuencia ω_1 . Además solo se incluirá el filtro coseno expandido para el primer subfiltro, ya que este es el único cuya frecuencia del peor caso de aliasing ω_1 coincide con la del filtro original. Se tiene entonces una nueva versión del primer subfiltro con retardo $D_1 = M$:

$$H_{TECOS}(z) = \left[\frac{1}{M} \frac{(1 - z^{-M})}{(1 - z^{-M_1})} \right]^{K_{11}} \left[\frac{1}{M_1} \frac{(1 - z^{-M_1})}{(1 - z^{-1})} \right]^{K_{12}} H_{cos}(z), \quad (6.9)$$

donde

$$H_{cos}(z) = \begin{cases} (1 + z^{-R})/2, & \text{para } \omega_R \neq \omega_{DK}, \\ 1, & \text{para } \omega_R = \omega_{DK}, \end{cases} \quad (6.10)$$

$$R = \text{int}[\pi/\omega_1]. \quad (6.11)$$

Como se observa en la ecuación (6.11), debido al redondeo a la parte entera en el cálculo de R , el cero que introduce el filtro coseno expandido puede no encontrarse justamente

en la frecuencia deseada. Sin embargo, también contribuye a la mejora de la SNR, ya que aumenta la atenuación en el peor caso de aliasing, quedando limitado, en el mejor de los casos por el lóbulo de mayor amplitud que se encuentre en la banda de rechazo.

Ejemplo 6.3: Se eligen los retardos $D_1 = 10$, $D_2 = 12$, $D_3 = 14$ y $D_4 = 16$. Para este caso $R = 8$. Como se observa en la Fig. 6.1, el primer cero del subfiltro con retardo $D_4 = 16$ se encuentra en la misma ubicación que para el filtro coseno expandido con $R = 8$. Para este caso no se incluirá el filtro coseno expandido porque ya se tiene un cero que mejora la SNR.

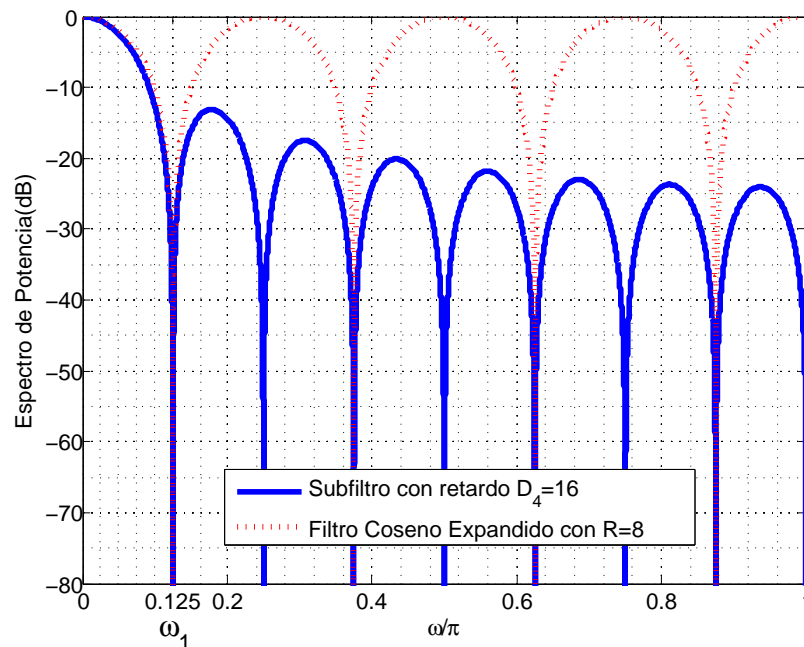


Figura 6.1: Comparación de la ubicación para el primer cero para el filtro coseno expandido con $R = 8$ y el subfiltro con $D_4 = 16$.

Ejemplo 6.4: Usando los retardos $D_1 = 21$, $D_2 = 24$, $D_3 = 27$ y $D_4 = 30$, se tiene $R = 16$. Como se observa en la Fig. 6.2, el primer cero del subfiltro con retardo $D_4 = 30$ no se encuentra en la misma ubicación que para el filtro coseno expandido con $R = 16$. Para este caso debe incluirse el filtro coseno expandido para mejorar la atenuación en la frecuencia donde se presenta el peor caso de *aliasing*.

Ejemplo 6.5: Se presenta el procedimiento completo para el caso de un factor de decimación $M = 21$, $M_1 = 3$ y $M_2 = 7$. Para este caso el $K_{max} = 5$, se elige $K = 4$, obteniendo un $P_{max} = 1$. Los retardos correspondientes usando la ecuación (5.6) son:

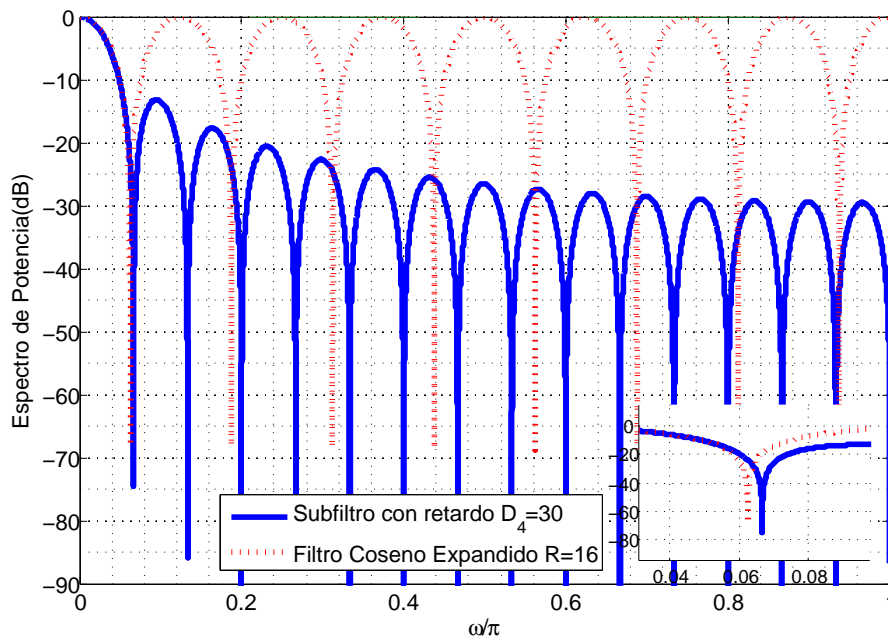


Figura 6.2: Comparación de la ubicación para el primer cero para el filtro coseno expandido con $R = 16$ y el subfiltro con $D_4 = 30$, zoom de los primeros ceros de cada filtro.

$D_1 = 21$, $D_2 = 24$, $D_3 = 27$, $D_4 = 30$. La aproximación triangular escalonada se aplica sobre los primeros subfiltros. En este caso se puede elegir si se aplica solamente a los dos o a los tres primeros según la SNR deseada. Finalmente, considerando que la frecuencia está normalizada, el primer cero del filtro coseno expandido se encuentra en $\omega_R = 1/16$ y el primer cero del filtro con retardo D_4 se encuentra en $\omega_{D_4} = 2/30 = 1/15$, es decir, se encuentran en frecuencias diferentes. Para este caso se incluirá el filtro coseno expandido con el fin de mejorar la SNR.

6.2. Estructura

La estructura propuesta se obtiene al combinar las ecuaciones (6.2), (6.5) y (6.9). El conjunto de parámetros de cascada $K, K_1, K_2, \dots, K_{N_1}$, permite modificar la respuesta en magnitud del filtro propuesto. Cuando estos parámetros poseen valores grandes, las especificaciones de SNR serán mejores y el costo computacional se incrementa. Dependerá de la aplicación y de si es requerida una buena SNR la decisión de sacrificar costo computacional para mejorar este parámetro.

Al usar la aproximación triangular escalonada a los filtros correspondientes y gracias a la selección anteriormente definida de los retardos D_i , es posible reducir a baja frecuencia intermedia los filtros $H_j(z)$ y $H_{TEi}(z)$. La parte *comb* del filtro $H_{TE1}(z)$ puede moverse a la más baja frecuencia y por cada subfiltro al que se le aplique la aproximación al filtro CIC triangular escalonado incrementará el APOS dependiendo del conjunto de parámetros de cascada que los caracterice.

La estructura general para el método propuesto se presenta en la Fig. 6.3. En la parte superior se tienen los integradores relacionados a los filtros que no son CIC triangulares escalonados, seguido por los que si lo son, cuyos parámetros de cascada son los K_{i2} . Estos integradores funcionan a la frecuencia más alta; en frecuencia intermedia se encuentran los integradores de los filtros CIC triangulares escalonados con parámetros de cascada $K_{i1} - K_{i2}$, seguidos por los filtros *comb* relacionados con los filtros CIC triangulares escalonados con parámetros de cascada K_{i1} , excepto el relacionado al primero, ya que este puede llevarse a baja frecuencia. Además en frecuencia intermedia se tienen los filtros *comb* de los subfiltros CIC que no se aproximaron a filtros CIC triangulares escalonados y el filtro coseno, en el caso que sea requerido. Finalmente, en baja frecuencia de muestreo se tiene el filtro *comb* relacionado al primer filtro CIC triangular escalonado.

En el caso general la APOS está definida por la ecuación (6.12).

$$APOS_p = \left[(K - N_1) + \sum_{k=1}^{N_1} K_{i2} \right] M + \left[\sum_{k=1}^{N_1} (K_{i1} - K_{i2}) + \sum_{k=2}^{N_1} K_{i1} + (K - N_1) + 1 \right] M_2 + K_{11}, \quad (6.12)$$

donde N_1 es la cantidad del subfiltros a los que se les aplica la aproximación al filtro CIC triangular escalonado. Si no es necesario aplicar la mejora en el peor caso de atenuación del aliasing la APOS se reduce a la ecuación (6.13).

$$APOS_p = \left[(K - N_1) + \sum_{k=1}^{N_1} K_{i2} \right] M + \left[\sum_{k=1}^{N_1} (K_{i1} - K_{i2}) + \sum_{k=2}^{N_1} K_{i1} + (K - N_1) \right] M_2 + K_{11}, \quad (6.13)$$

6.3. Estructura polifase

En la sección anterior se propuso un método para el diseño de filtros para la conversión racional de frecuencia de muestreo. Este método presenta una SNR mejorada y una re-

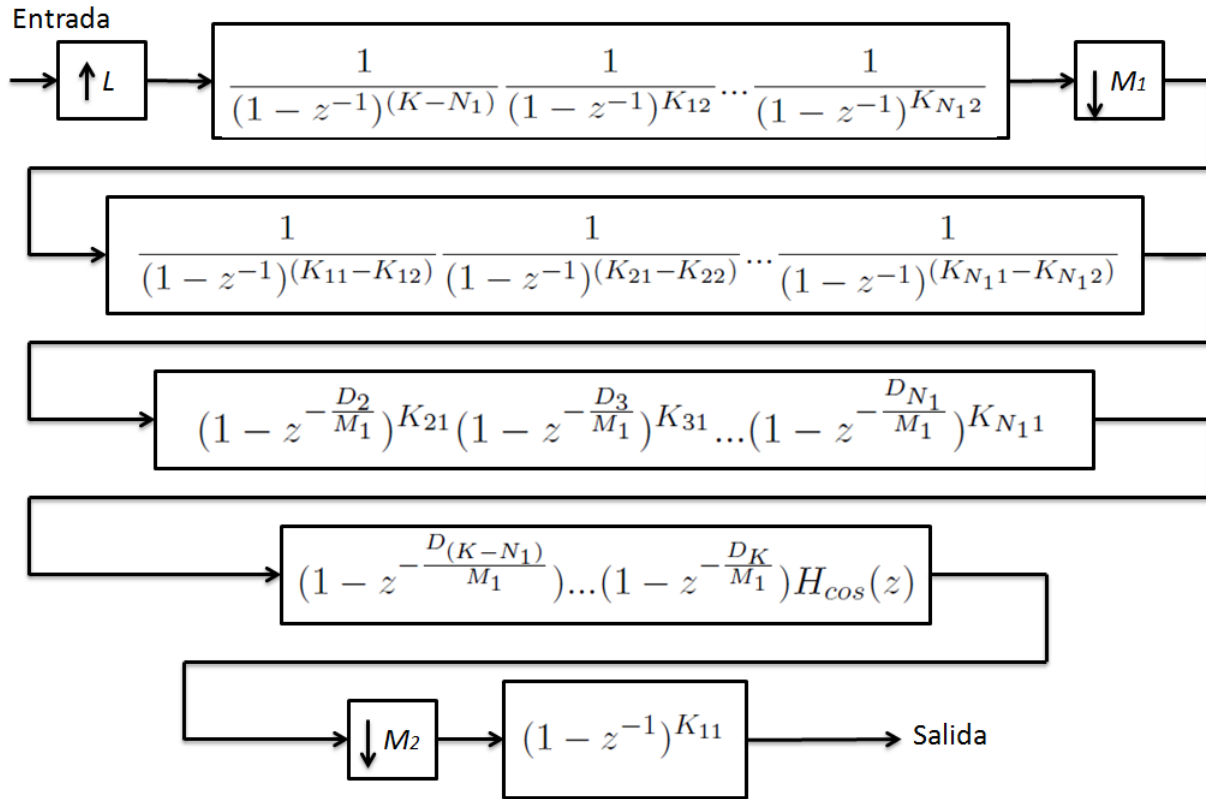


Figura 6.3: Estructura general del método propuesto.

ducción de la cantidad de operaciones aritméticas por muestra de salida.

Volviendo a la ecuación (6.4), el término de la derecha se mantendrá en alta frecuencia y el término de la izquierda se moverá a frecuencia intermedia usando las identidades nobles. Además (6.5) se factoriza de la forma presentada en la ecuación (6.14), manteniendo la parte del lado derecho en alta frecuencia y la del lado izquierdo se mueve después del primer submuestreador, la estructura resultante al introducir estos cambios se presenta a continuación.

$$H_j(z) = \left(\frac{1 - z^{-M_1}}{1 - z^{-1}} \right) \left(\frac{1 - z^{-D_{N_1+j}}}{1 - z^{-1}} \right). \quad (6.14)$$

La aplicación del método presentado en [9] se hace sobre la subestructura que se encuentra dentro de la línea roja punteada en la Fig. 6.4. El filtro entre los bloques sobremuestreador y submuestreador es un filtro CIC, que puede llevarse a su forma no recursiva, resultando en un filtro de orden $(M_1 - 1)Q$ con la simetría requerida para el uso del método descrito en [9], donde Q y P están dados por:

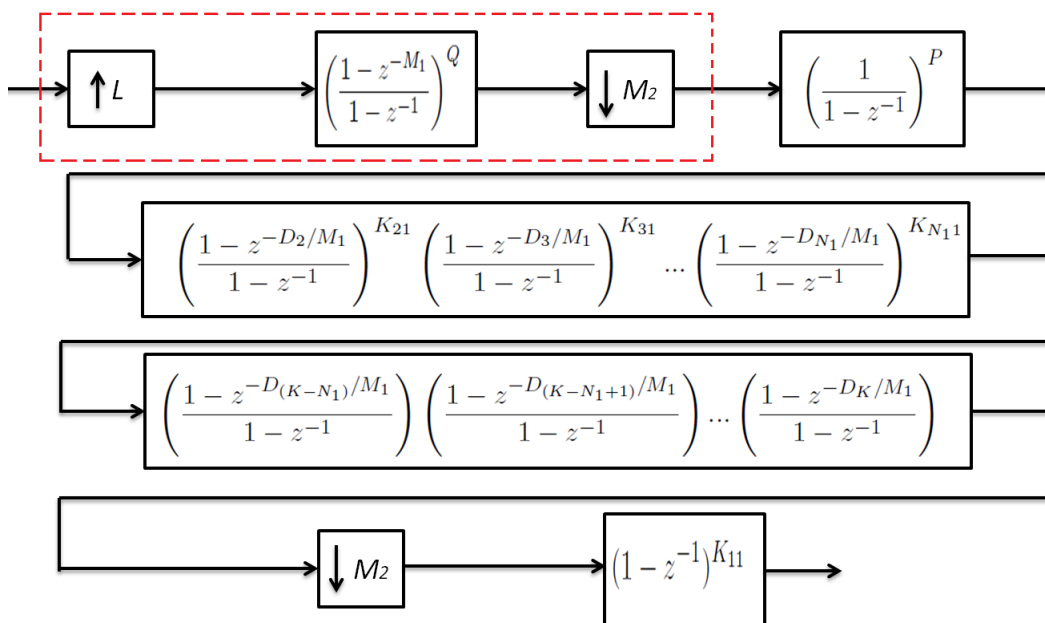


Figura 6.4: Estructura modificada para realización polifase.

$$Q = \sum_{i=1}^{N_1} K_{i2} + (K - N_1), \quad (6.15)$$

$$P = \sum_{i=1}^{N_1} (K_{i1} - K_{i2}). \quad (6.16)$$

La función de transferencia del filtro resultante es:

$$H(z) = [1 + z^{-1} + z^{-2} + \dots + z^{-(M_1-1)}]^{(M_1-1)Q}. \quad (6.17)$$

A este nuevo filtro se le aplicará el método de realización eficiente, con lo que todas las operaciones podrán realizarse en una frecuencia reducida, mejorando notablemente el consumo de potencia de la estructura propuesta. El procedimiento se muestra en el siguiente ejemplo.

Ejemplo 6.6: Se presenta un ejemplo para la estructura eficiente usando $L = 7$, $M = 10$, $M_1 = 2$, $M_2 = 5$, $K_{11} = 8$, $K_{12} = 4$, $K_{21} = 2$, $K_{22} = 1$ y $N_1 = 2$. Para este caso, usando la ecuación (6.16) se encuentra que $Q = 7$, con lo que se obtiene la estructura presentada en la Fig. 6.5 y sobre la cual se aplicará el método.

Como se mostró en el capítulo anterior la separación de la matriz H está definida por las ecuaciones (6.29) y (6.30). Usando estas ecuaciones se obtienen las formas matriciales

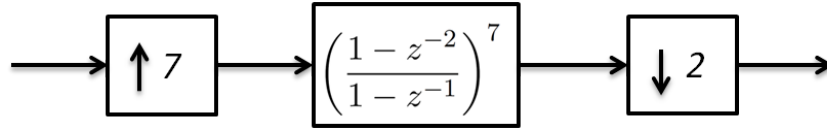


Figura 6.5: Estructura para el ejemplo 6.6.

que definen las salidas.

$$X_{m+1,m-1} = \begin{bmatrix} x[m+1] & x[m] & x[m-1] \end{bmatrix}. \quad (6.18)$$

$$y[n] = \begin{bmatrix} h_0 & h_0 \end{bmatrix} \begin{bmatrix} x[m] \\ x[m-1] \end{bmatrix} = h_0(x[m] + x[m-1]). \quad (6.19)$$

$$\begin{bmatrix} y[n+1] \\ y[n+2] \\ y[n+3] \\ y[n+4] \\ y[n+5] \\ y[n+6] \end{bmatrix} = \begin{bmatrix} 0 & h_2 \\ 0 & h_3 \\ 0 & h_1 \\ h_1 & 0 \\ h_3 & 0 \\ h_2 & 0 \end{bmatrix} X_{m+1,m}. \quad (6.20)$$

Gracias a la propiedad de centro-simetría de la matriz H_{c_2} , se puede obtener una forma factorizada alternativa como se muestra a continuación.

$$H_{c_2} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_{00} & 0 \\ c_{10} & 0 \\ c_{20} & 0 \\ 0 & d_{20} \\ 0 & d_{10} \\ 0 & d_{00} \end{bmatrix} X_{m+1,m}^{(0)}, \quad (6.21)$$

donde

$$\begin{aligned} c_{00} &= h_2/2, & d_{20} &= -h_2/2, \\ c_{10} &= h_3/2, & d_{10} &= -h_3/2, \\ c_{20} &= h_1/2, & d_{00} &= -h_1/2. \end{aligned} \quad (6.22)$$

$$X_{m+1,m}^{(0)} = \begin{bmatrix} x[m+1] + x[m] \\ x[m+1] - x[m] \end{bmatrix}. \quad (6.23)$$

Finalmente

$$\begin{bmatrix} y[n+1] \\ y[n+2] \\ y[n+3] \\ y[n+4] \\ y[n+5] \\ y[n+6] \end{bmatrix} = \begin{bmatrix} h_2x[m] \\ h_3x[m] \\ h_1x[m] \\ h_1x[m] \\ h_3x[m] \\ h_2x[m] \end{bmatrix}. \quad (6.24)$$

La estructura resultante de la implementación de las ecuaciones (6.35) y (6.32) se presenta en la Fig. 6.6.

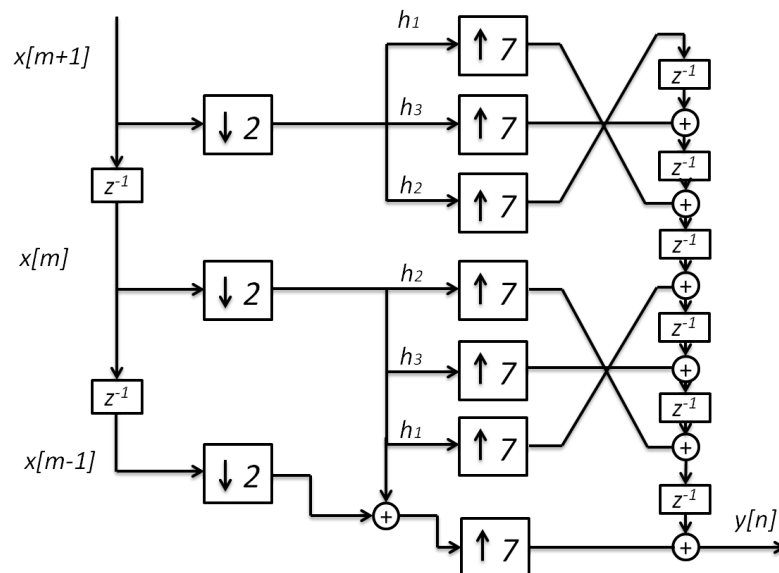


Figura 6.6: Estructura eficiente para el ejemplo 6.6.

Ejemplo 6.7: Se presenta un segundo ejemplo para la estructura eficiente usando $L = 5$, $M = 21$, $M_1 = 3$, $M_2 = 7$, $K_{11} = 2$, $K_{12} = 1$, $K_{21} = 2$, $K_2 = 1$, $K_{31} = 2$, $K_{32} = 1$ y $N_1 = 3$. Para este caso, usando la ecuación (6.16) se encuentra que $Q = 5$, con lo que se obtiene la estructura presentada en la Fig. 6.7 y sobre la cual se aplicará el método.

$$\begin{aligned} H(z) &= (1 + z^{-1} + z^{-2})^5 \\ &= [1 + 5z^{-1} + 15z^{-2} + 30z^{-3} + 45z^{-4} + 51z^{-5} + 45z^{-6} + 30z^{-7} + 15z^{-8} + 5z^{-9} + z^{-10}]. \end{aligned} \quad (6.25)$$

Usando la ecuación (3.27) se obtienen las componentes polifase de la $H(z)$ obteniendo:

$$\begin{aligned}
H_0(z) &= 1 + 51z^{-5} + z^{-10}, & H_2(z) &= 15 + 30z^{-1}, \\
H_1(z) &= 5 + 45z^{-1}, & H_3(z) &= 30 + 15z^{-1}, \\
H_4(z) &= 45 + 5z^{-1}, & H_5(z) &= 51 + z^{-1}.
\end{aligned} \tag{6.26}$$

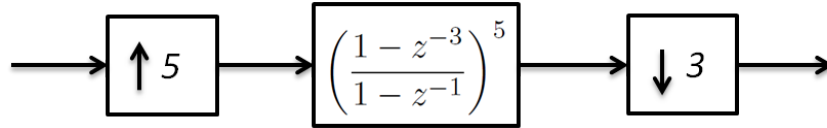


Figura 6.7: Estructura para el ejemplo 6.7.

La matriz de entrada-salida H se presenta a continuación.

$$H_{5 \times 5} = \begin{bmatrix} 0 & 0 & h_0 & h_5 & h_{10} \\ 0 & 0 & h_3 & h_8 & 0 \\ 0 & h_1 & h_6 & 0 & 0 \\ 0 & h_4 & h_9 & 0 & 0 \\ h_2 & h_7 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & h_0 & h_5 & h_{10} \\ 0 & 0 & h_3 & h_2 & 0 \\ 0 & h_1 & h_4 & 0 & 0 \\ 0 & h_4 & h_1 & 0 & 0 \\ h_2 & h_3 & 0 & 0 & 0 \end{bmatrix} \tag{6.27}$$

$$X_{m+2, m-2} = \begin{bmatrix} x[m+2] & x[m+1] & x[m] & x[m-1] & x[m-2] \end{bmatrix}. \tag{6.28}$$

La matriz H puede dividirse en las matrices presentadas a continuación.

$$H_{c_1} = H_{1 \times 3} = \begin{bmatrix} h_0 & h_5 & h_{10} \end{bmatrix}, \tag{6.29}$$

$$H_{c_2} = H_{4 \times 4} = \begin{bmatrix} 0 & 0 & h_3 & h_2 \\ 0 & h_1 & h_4 & 0 \\ 0 & h_4 & h_1 & 0 \\ h_2 & h_3 & 0 & 0 \end{bmatrix}. \tag{6.30}$$

$$y[n] = \begin{bmatrix} h_0 & h_5 & h_{10} \end{bmatrix} \begin{bmatrix} x[m] \\ x[m-1] \\ x[m-2] \end{bmatrix} = h_0(x[m] + x[m-2]) + h_5x[m-2]. \tag{6.31}$$

$$\begin{bmatrix} y[n+1] \\ y[n+2] \\ y[n+3] \\ y[n+4] \end{bmatrix} = \begin{bmatrix} 0 & 0 & h_3 & h_2 \\ 0 & h_1 & h_4 & 0 \\ 0 & h_4 & h_1 & 0 \\ h_2 & h_3 & 0 & 0 \end{bmatrix} X_{m+2, m-1}. \tag{6.32}$$

Se puede obtener una forma factorizada como se muestra a continuación.

$$\begin{bmatrix} y[n+1] \\ y[n+2] \\ y[n+3] \\ y[n+4] \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & -1 & 0 \\ 1 & 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} c_{00} & c_{01} & 0 & 0 \\ c_{10} & c_{11} & 0 & 0 \\ 0 & 0 & d_{11} & d_{10} \\ 0 & 0 & d_{01} & d_{00} \end{bmatrix} X_{m+2,m-1}^{(1)}, \quad (6.33)$$

donde

$$\begin{aligned} c_{00} &= h_2/2, & d_{00} &= -h_2/2, \\ c_{01} &= h_3/2, & d_{01} &= -h_3/2, \\ c_{10} &= 0, & d_{10} &= 0, \\ c_{11} &= (h_1 + h_4)/2, & d_{11} &= (h_1 - h_4)/2, \end{aligned} \quad (6.34)$$

$$X_{m+2,m-1}^{(1)} = \begin{bmatrix} x[m+2] + x[m-1] \\ x[m+1] + x[m] \\ x[m+1] - x[m] \\ x[m+2] - x[m-1] \end{bmatrix}. \quad (6.35)$$

Finalmente

$$\begin{bmatrix} y[n+1] \\ y[n+2] \\ y[n+3] \\ y[n+4] \end{bmatrix} = \begin{bmatrix} h_2x[m-1] + h_3x[m] \\ c_{11}(x[m+1] + x[m]) + d_{11}(x[m+1] - x[m]) \\ c_{11}(x[m+1] + x[m]) - d_{11}(x[m+1] - x[m]) \\ h_2x[m+2] + h_3x[m+1] \end{bmatrix}. \quad (6.36)$$

La estructura resultante de la implementación de las ecuaciones (6.31) y (6.36) se presenta en la Fig. 6.8.

6.4. Comparación

Se realiza en esta sección una comparación con del método propuesto con dos de las estructuras basadas en filtros CIC presentadas en el Capítulo 4.

La comparación se hace entre el método descrito en [6], el propuesto en [15] y el desarrollado en este trabajo. Para ello se usan dos ejemplos que incluyen los valores de SNR y APOS.

Ejemplo 6.8: Se desea diseñar un filtro para realizar la conversión de frecuencia de muestreo por el factor $\frac{7}{10}$. Se comparan los resultados obtenidos al aplicar la estructura mostrada en [6], la estructura de [15] y la estructura propuesta aplicando la aproximación

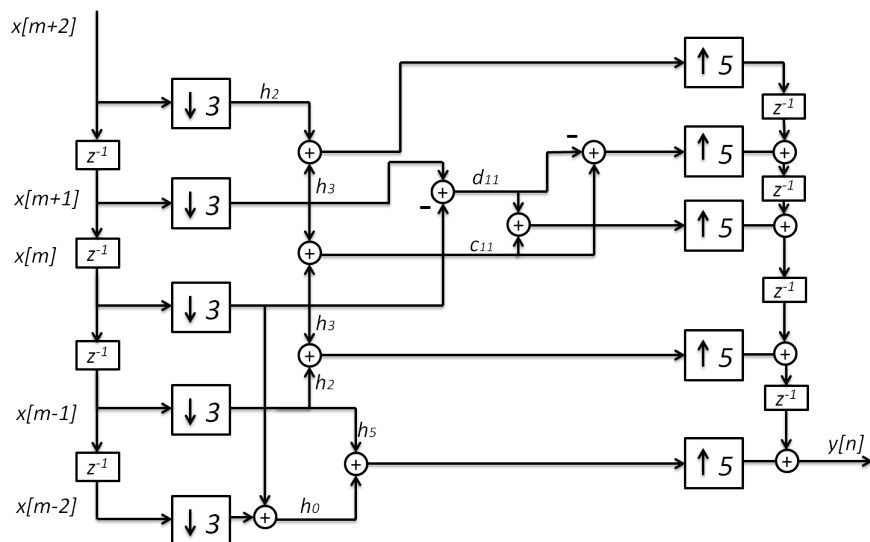


Figura 6.8: Estructura eficiente para el ejemplo 6.7.

al filtro CIC triangular escalonado en el primer y segundo subfiltro. El filtro CIC triangular escalonado con $M_1 = 2$, $M_2 = 5$, $K_1 = 8$, $K_2 = 4$, $K_3 = 1$, $R = 8$, posee una SNR de 58.9dB y requiere una APOS de 73. La estructura usando el filtro CIC modificado usando $D_1 = 10$, $D_2 = 12$, $D_3 = 14$, $D_4 = 16$ posee una SNR de 50dB (Fig. 4.1) y una APOS de 80, mientras la estructura propuesta seleccionando $D_1 = 10$, $D_2 = 12$, $D_3 = 14$, $D_4 = 16$, $K_{11} = 2$, $K_{12} = 1$, $K_{21} = 2$, $K_{22} = 1$, $N_1 = 2$ tiene una SNR de 72.25dB y requiere 72 APOS. Debido a que en este caso no se requiere la mejora en el peor caso de atenuación de aliasing, la APOS se calcula usando la ecuación (6.13). Las respectivas respuestas en magnitud son presentadas en la Fig. 6.9 y la estructura resultante en la Fig. 6.10. Usando diferentes valores para los parámetros de cascada es posible tener una mejor SNR pero se incrementará la APOS, por ejemplo, al usar $K_{11} = 8$, $K_{12} = 4$, $K_{21} = 2$, $K_{22} = 1$, la SNR se incrementa a 95.49dB pero la APOS consecuentemente se incrementa a 123. La Fig. 6.11 presenta la respuesta en magnitud para la segunda parte del ejemplo 6.6. La estructura se presenta en la Fig. 6.10. En la Tabla 6.1 se resumen los resultados para el ejemplo 6.8.

Método	APOS	SNR (dB)
[15]	80	50
[6]	73	58.9
Propuesto	72	72.25

Tabla 6.1: Comparación ejemplo 6.8.

Ejemplo 6.9: Se desea realizar la conversión de frecuencia por el factor $\frac{5}{21}$. De igual for-

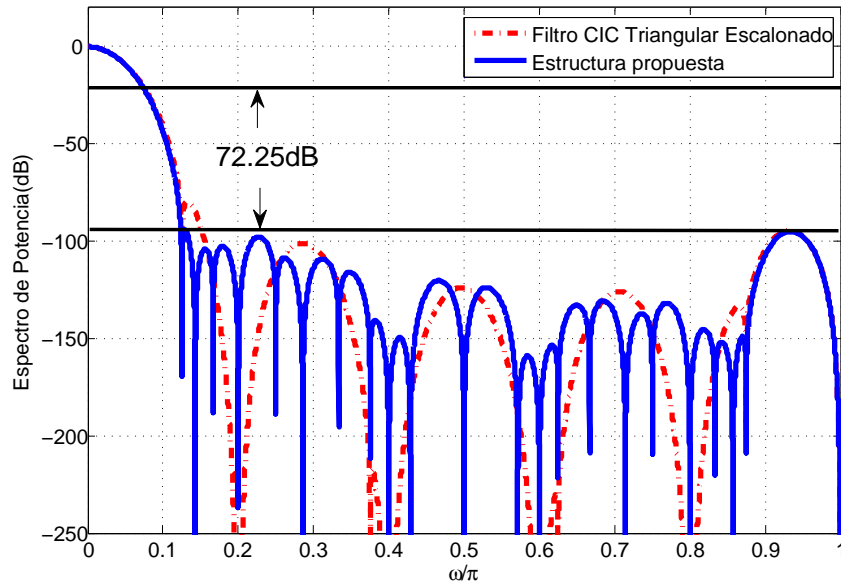


Figura 6.9: Comparacion de las respuestas en magnitud para el ejemplo 6.8.

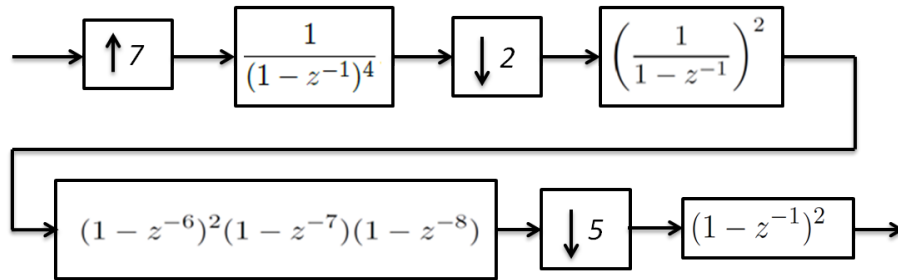


Figura 6.10: Estructura resultante para el ejemplo 6.8.

ma se comparan los resultados obtenidos al usar ambos métodos. El filtro CIC triangular escalonado con $M_1 = 3$, $M_2 = 7$, $K_1 = 10$, $K_2 = 5$, $K_3 = 1$, $R = 16$, posee una SNR de 63dB y requiere una APOS de 171. El filtro CIC modificado usando $D_1 = 21$, $D_2 = 24$, $D_3 = 27$, $D_4 = 30$, $D_5 = 33$ presenta una APOS de 210 y una SNR de 60dB. Por otro lado, la estructura propuesta seleccionando $D_1 = 21$, $D_2 = 24$, $D_3 = 27$, $D_4 = 30$, $D_5 = 33$, $K_{11} = 2$, $K_{12} = 1$, $K_{21} = 2$, $K_{22} = 1$, $K_{31} = 2$, $K_{32} = 1$ y $N_1 = 3$ tiene una SNR de 95.22dB y requiere 177 APOS. La gran diferencia de las SNR's se debe principalmente a que en este caso para el filtro CIC triangular escalonado la mejora debería encontrarse en la frecuencia $\omega_1 = 0.0595$ pero en realidad se encuentra en la frecuencia 0.0625. Las respectivas respuestas en magnitud son presentadas en la Fig. 6.12 y la estructura en la Fig. 6.13. En la Tabla 6.2 se resumen los resultados para la primera parte del ejemplo 6.9.

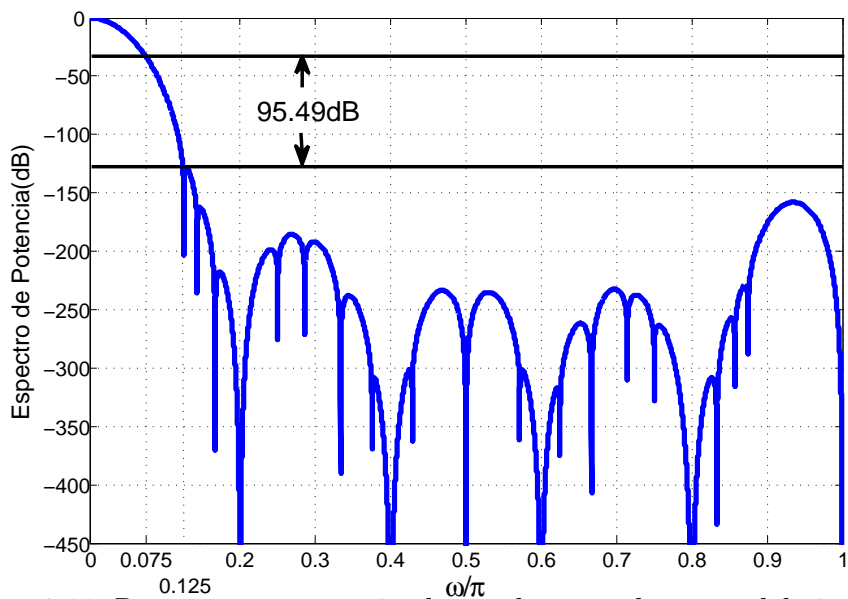


Figura 6.11: Respuesta en magnitud para la segunda parte del ejemplo 6.8.

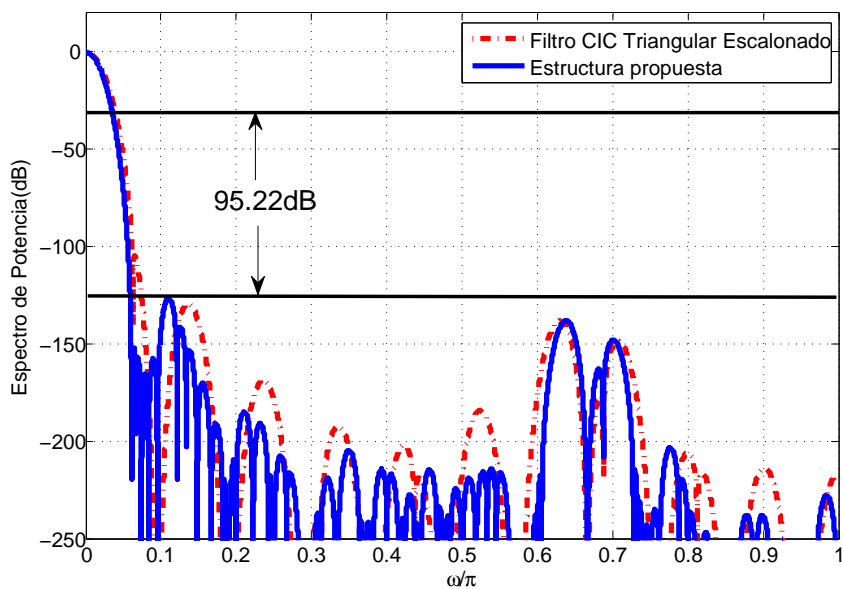


Figura 6.12: Comparación de las respuestas en magnitud para el ejemplo 6.9.

Como se observa en las Tablas 6.1 y 6.2, el método propuesto presenta una mejora considerable en SNR y una menor cantidad de APOS para ejemplo 6.8 y un leve incremento para ejemplo 6.9. Sin embargo, la especificación de SNR es relativamente alta en el ejemplo 6.9, y puede ser reducida usando el filtro propuesto de orden más bajo, es decir, un

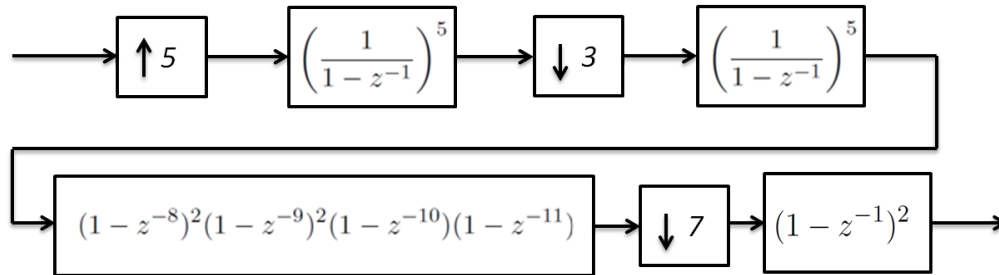


Figura 6.13: Estructura resultante para el ejemplo 6.9.

Método	APOS	SNR (dB)
[15]	210	60
[6]	171	63
Propuesto	177	95.22

Tabla 6.2: Comparación ejemplo 6.9.

filtro de menores requerimientos computacionales.

6.5. Implementación

Se presenta en esta sección la implementación en *Simulink* y Matlab usando el método propuesto. Para ello se presentan dos ejemplos que incluyen la simulación de la respuesta al impulso usando las herramientas correspondientes.

Ejemplo 6.10: Se considera un filtro usando la estructura propuesta con $K = 4$, $L/M = 7/10$, $M_1 = 2$, $M_2 = 5$, $K_{11} = 8$, $K_{12} = 4$, $K_{21} = 2$, $K_{22} = 1$ y $N_1 = 2$. La estructura del filtro se presenta en la Fig. 6.14.

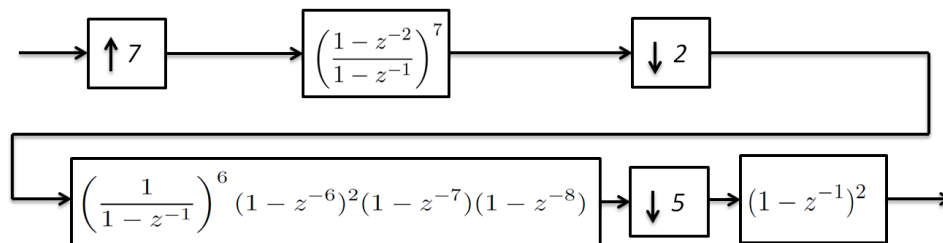


Figura 6.14: Estructura del ejemplo 6.10.

La estructura resultante al usar el método presentado en [9] para el filtro CIC que se encuentra después del primer submuestreador se obtiene de las ecuaciones (6.35) y (6.32) y es mostrado en la Fig. 6.6. Se observa que la estructura original presenta 7 sumas trabajando en alta frecuencia de muestreo, mientras la estructura eficiente tiene una suma

funcionando en la frecuencia de entrada disminuida por 2 y 6 sumas funcionando en la frecuencia de entrada aumentada 3.5 veces.

Ejemplo 6.11: En este ejemplo se considera un filtro usando la estructura propuesta para $K = 5$, $L/M = 5/21$, $M_1 = 3$, $M_2 = 7$, $K_{11} = 2$, $K_{12} = 1$, $K_{21} = 2$, $K_{22} = 1$, $K_{31} = 2$, $K_{32} = 1$ y $N_1 = 3$. La estructura del filtro se presenta en la Fig. 6.15.

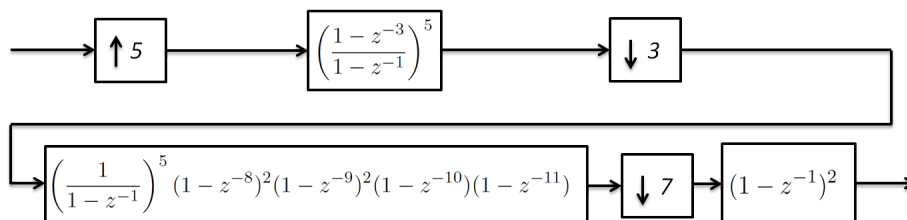


Figura 6.15: Estructura del ejemplo 6.11.

Para este caso la estructura eficiente para el filtro que se encuentra después del primer submuestreador se presenta en la Fig. 6.8. Para este se tienen 10 sumadores funcionando en alta frecuencia, mientras la versión eficiente requiere 7 sumadores funcionando en frecuencia disminuida 3 veces y 4 sumadores funcionando en la frecuencia de entrada aumentada 1.667 veces.

6.5.1. Implementación en *Simulink*

Los filtros presentados en los ejemplos anteriores se implementaron directamente en *Simulink*, con el fin de comprobar su correcto funcionamiento al usar la estructura propuesta.

Ejemplo 6.10

Se presenta en primer lugar la respuesta al impulso de la estructura de la Fig. 6.5. Para ello se utilizan los esquemas en bloques de la Fig. 6.16. Esta respuesta es mostrada en la Fig. 6.17. Allí se usa un período de muestreo unitario. Las dos respuestas están sobrepuestas. Además, se observa que la frecuencia de muestreo ha sido modificada, consecuentemente el período de muestreo es $2/7$ veces el período original.

Adicionalmente, en la Fig. 6.18 se muestra el modelo en bloques usado para la simulación de la respuesta al impulso del filtro completo y el esquema en bloques del filtro propuesto en la Fig. 6.19, donde $H_b(z)$ es el resultado de aplicar el método presentado en [9] al filtro

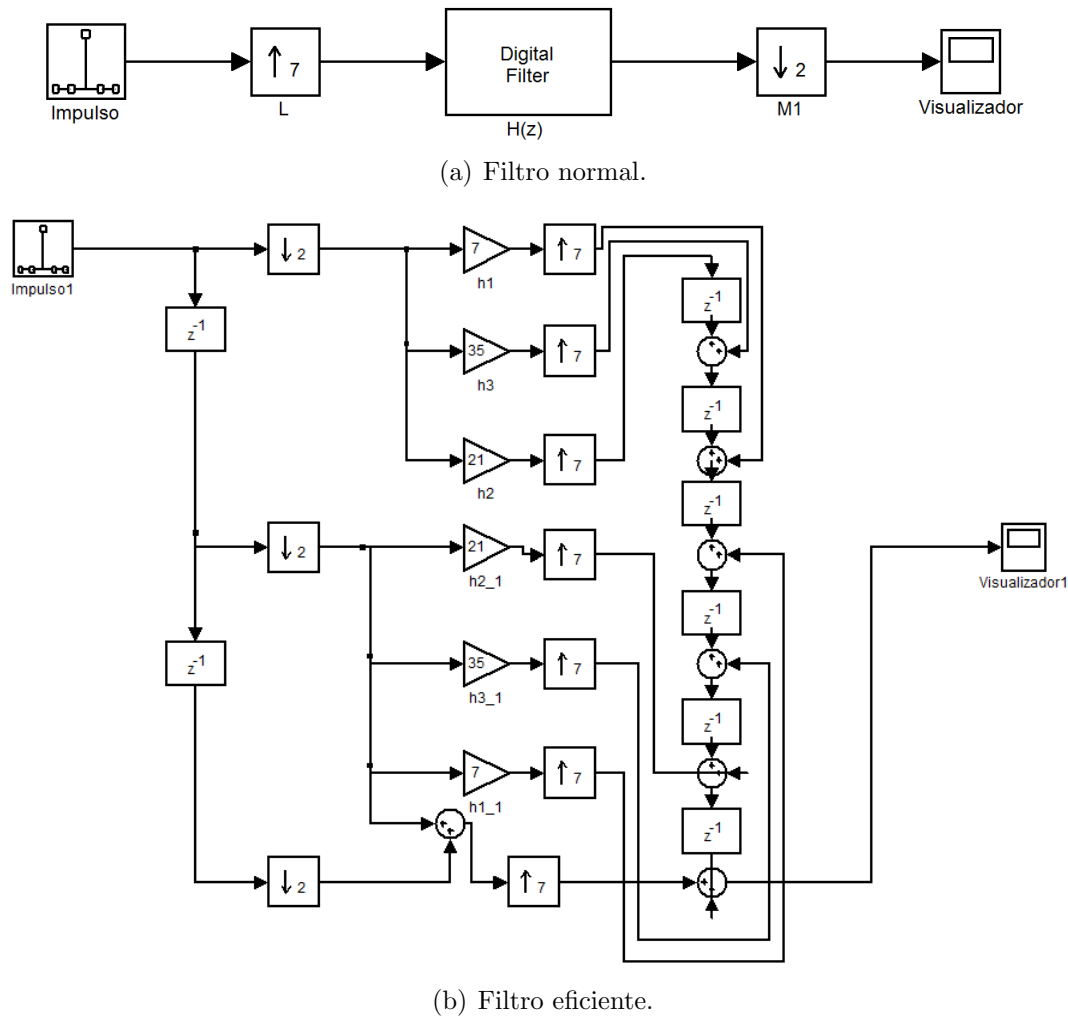


Figura 6.16: Estructuras equivalentes para el ejemplo 6.10.

$H(z)$ que corresponde al filtro que se encuentra después del sobremuestreador en la Fig. 6.14. Estos esquemas pueden compararse directamente con el esquema general del método propuesto presentado en la Fig. 6.4. Las respectivas respuestas al impulso se presentan en la Fig. 6.20. Se observa que estas son idénticas tanto en amplitudes como temporalmente. Además, el período de muestreo ha sido modificado a $10/7 = 1.4286$ veces el período original.

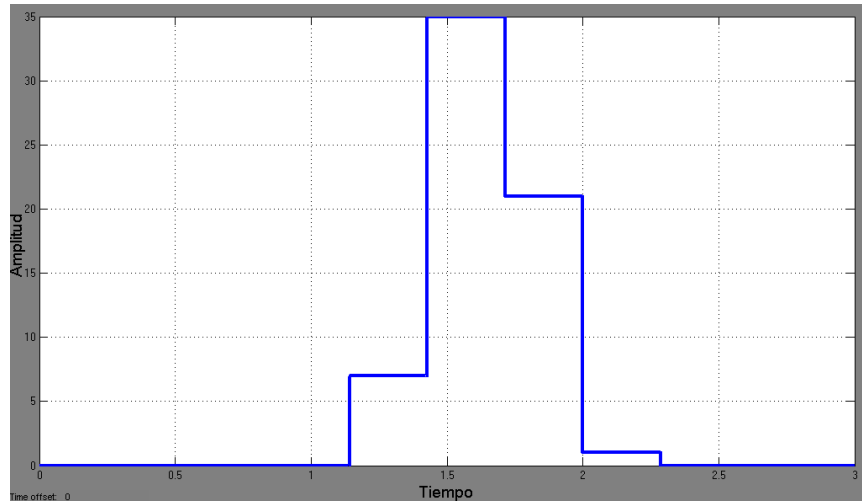


Figura 6.17: Respuesta al impulso para los esquemas equivalentes de la Fig. 6.16.

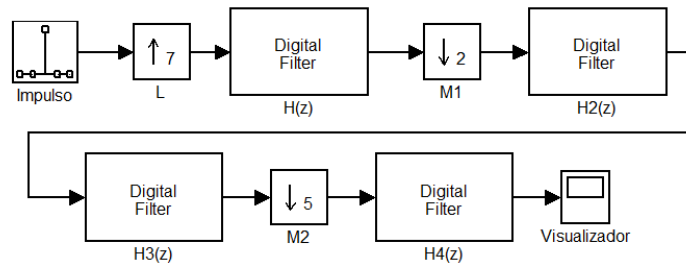


Figura 6.18: Esquema en bloques de *Simulink* para el ejemplo 6.10.

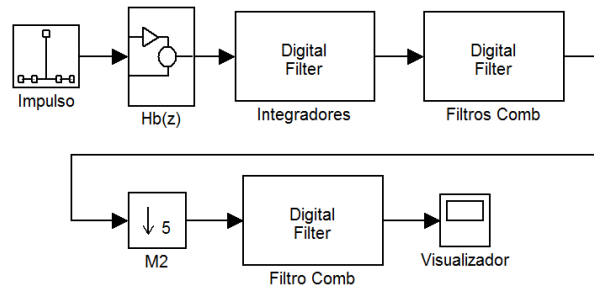


Figura 6.19: Esquema en bloques de *Simulink* para el ejemplo 6.10 usando la estructura eficiente.

Ejemplo 6.11

Primero se realiza la comparación de la respuesta al impulso de la estructura de la Fig. 6.7. Se utilizan los esquemas en bloques de la Fig. 6.21. La respectiva respuesta al impulso se presenta en la Fig. 6.22. Allí, como en el primer ejemplo se usa un período de muestreo unitario. En este caso el período de muestreo corresponde a $3/5$ partes del período original.

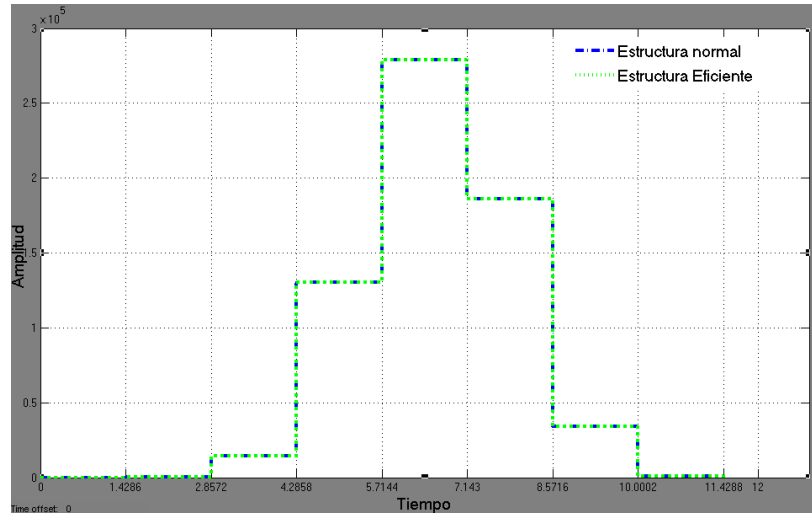
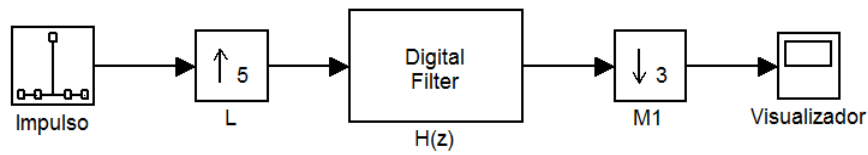
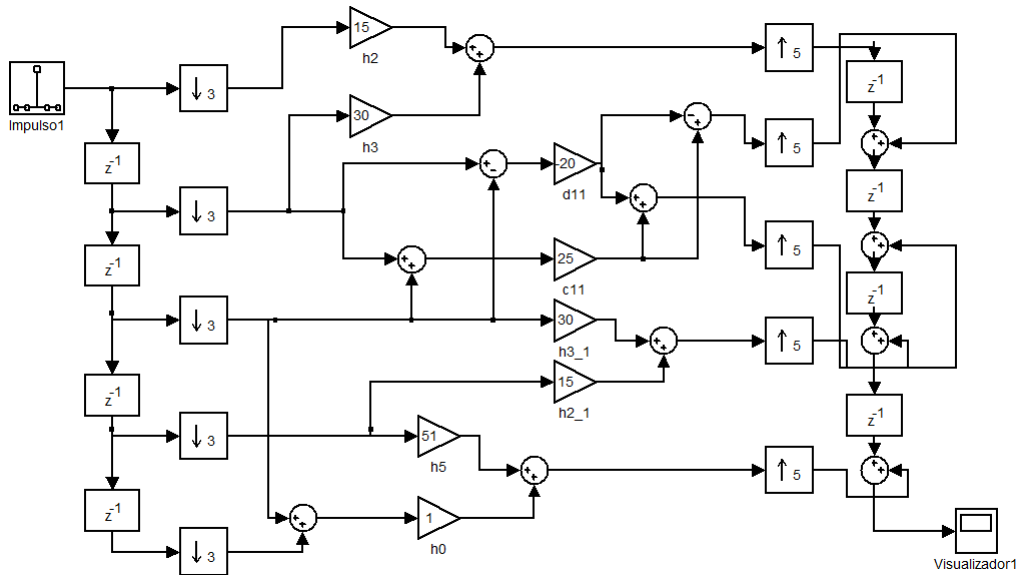


Figura 6.20: Respuesta al impulso para los esquemas equivalentes de las Figuras 6.18 y 6.19.



(a) Filtro normal.



(b) Filtro eficiente.

Figura 6.21: Estructuras equivalentes para el ejemplo 6.11.

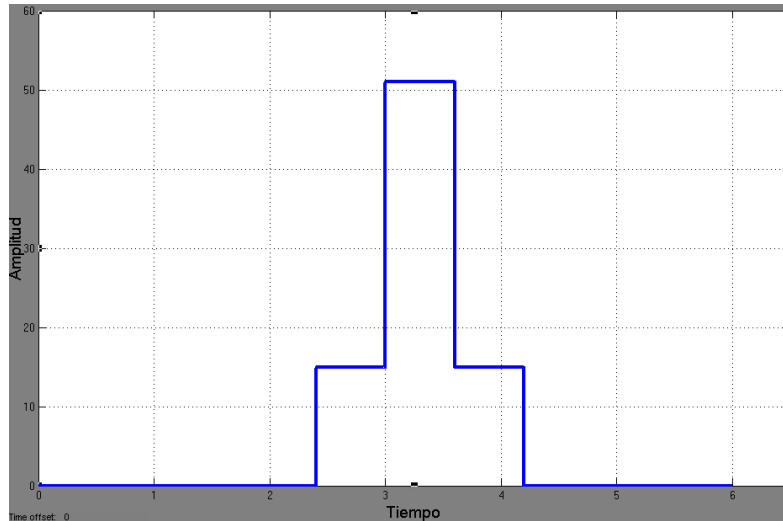


Figura 6.22: Respuesta al impulso para los esquemas equivalentes de las Figuras 6.21.

Se presenta en la Fig. 6.23 el modelo en bloques usado para la simulación de la respuesta al impulso del filtro completo. Además, se presenta el esquema en bloques del filtro propuesto en la Fig. 6.24, donde $H_c(z)$ es el resultado de aplicar el método presentado en [9] al filtro $H(z)$. Las respectivas respuestas al impulso se presentan en la Fig. 6.25. Como se observa estas son iguales, lo que confirma la equivalencia de los dos esquemas. También se puede observar que el nuevo período de muestreo corresponde a $21/5 = 4.2$ veces el período original.

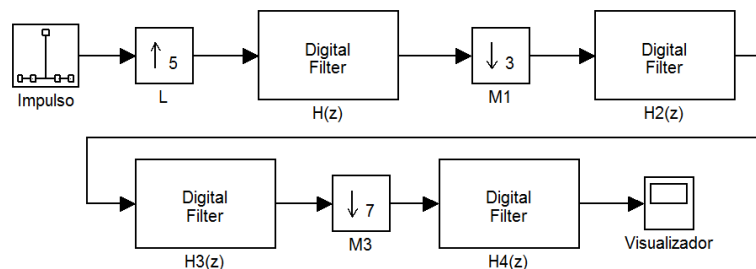


Figura 6.23: Esquema en bloques de *Simulink* para el ejemplo 6.11.

6.5.2. Implementación en VHDL

Se realiza una implementación estructural en VHDL que permita la realización en bloques de los filtros de los ejemplos 1 y 2. Para ello primero se debe buscar la forma de implementar los coeficientes del filtro de forma eficiente para evitar el uso de multiplicaciones. También es importante analizar el crecimiento del tamaño de palabra de los filtros en cada

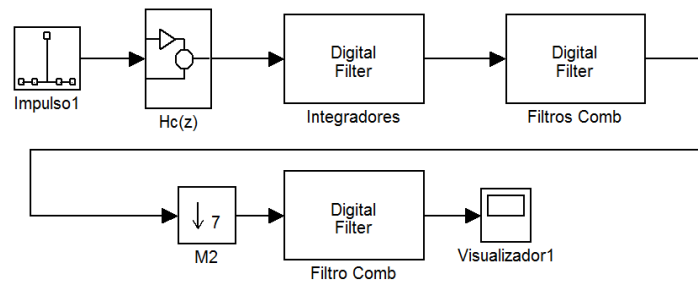


Figura 6.24: Esquema en bloques de *Simulink* para el ejemplo 6.11 usando la estructura eficiente.

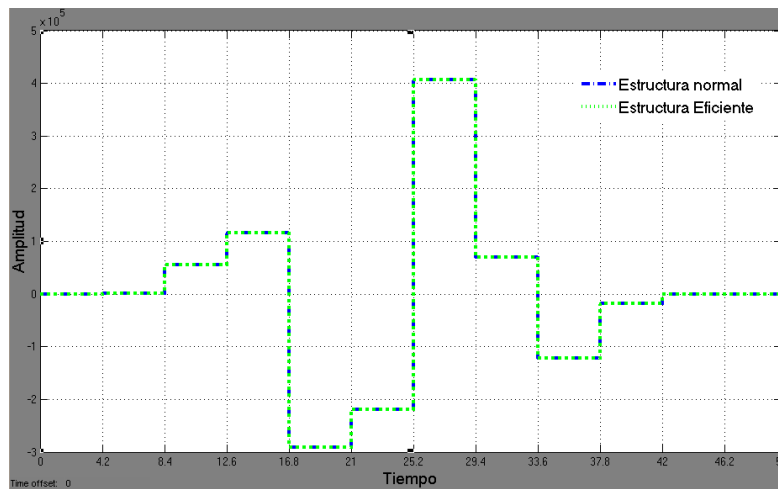


Figura 6.25: Respuesta al impulso para los esquemas equivalentes de las Figuras 6.18 y 6.19.

bloque para evitar tener cálculos erróneos a la salida de los mismos. Por último, se debe tener en cuenta las diferentes frecuencias de muestreo usadas en las etapas intermedias. Para ello se requiere de un hardware que se encargue de la generación de dichas señales de reloj.

Ejemplo 6.10

Se presentan a continuación algunas consideraciones para la implementación del ejemplo 6.10. Principalmente, se enfoca en el aumento del tamaño de palabra debido a las diferentes etapas del filtro, la realización eficiente de las multiplicaciones enteras resultantes de usar el método de [9], la generación de las señales de reloj con las frecuencias adecuadas y la estabilidad de la implementación de los integradores.

6.5.2.1. Generación de los coeficientes de la estructura eficiente y tamaños de palabra requeridas en cada etapa

Como ha sido mencionado anteriormente, la implementación se hace de forma estructural, lo que permite la realización en bloques de los subfiltros presentes en cada etapa del filtro. La primera etapa se observa en la Fig. 6.16b. Allí se observa que se presentan multiplicaciones enteras, sin embargo, estas pueden eliminarse usando la estructura presentada en la Fig. 6.26.

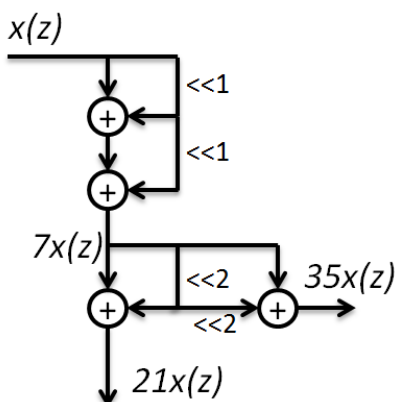


Figura 6.26: Esquema de realización eficiente de las multiplicaciones a base de corrimientos y sumas.

Se debe tener en cuenta que un corrimiento a la izquierda genera un aumento de las cantidad de bits usados en el valor de ese corrimiento. También la suma incluye un nuevo bit que corresponde al acarreo. Es posible analizar el incremento de la cantidad de bits utilizando únicamente las salidas multiplicadas. Para el producto por 7, que en binario sería 111, se tiene un incremento de tres bits, y para el producto por 35, que sería 100011, se tiene un incremento de 6 bits en la salida correspondiente. En la Fig. 6.27 se presenta la cantidad de bits necesarios para cada rama del filtro eficiente para el ejemplo 6.10, donde B es el tamaño en bits de la señal de entrada.

Para las otras etapas resulta evidente que al incluir una suma el incremento de bits será de uno multiplicado por la cantidad de cascadas de esa etapa. De la Fig. 6.14 se observa que se tienen, después del primer submuestreador, una cascada de 6 integradores, dos cascadas dobles de filtros *comb* y dos filtros *comb* simples. Además, después del segundo submuestreador se tiene una cascada de dos filtros *comb*. Las respectivas implementaciones de estas etapas se presentan en la Fig. 6.28. Para evitar el incremento excesivo del tamaño de palabra la sección integradora se mueve hasta el final de todas las cascadas y

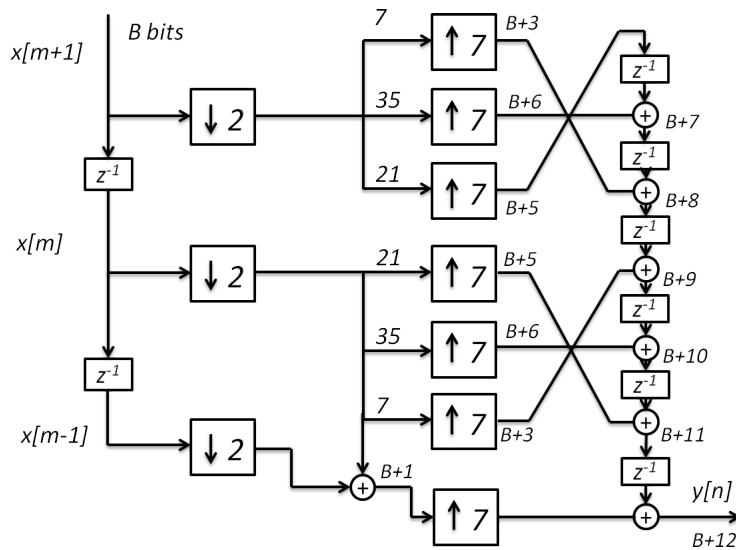


Figura 6.27: Incremento en el tamaño de palabra en la realización eficiente del ejemplo 6.10.

el filtro *comb*, que funciona en la más baja frecuencia de muestreo, se mueve a frecuencia intermedia.

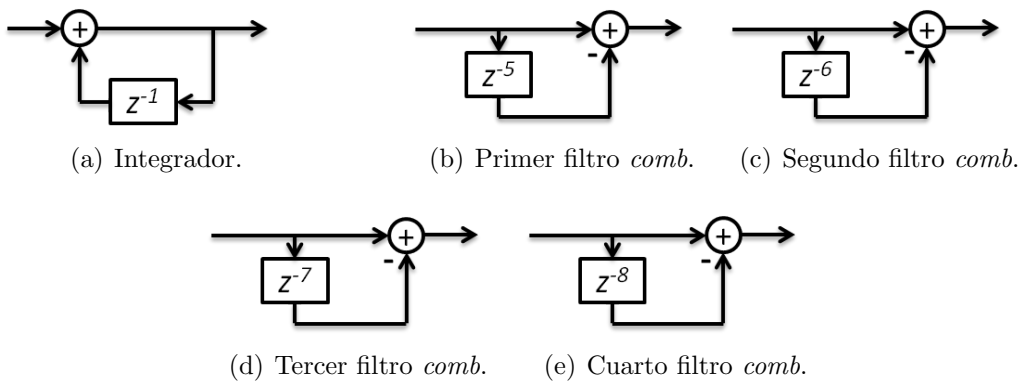


Figura 6.28: Diagramas de bloques para las etapas del filtro del ejemplo 6.10.

El tamaño de palabra de salida de la etapa integradora corresponde al tamaño de palabra de salida final del filtro. Sin embargo, para determinar el tamaño de palabra de la salida de la etapa integradora es necesario observar la simulación cuando se aplica la entrada de mayor amplitud posible, teniendo en cuenta que se usa la representación de complemento a dos. Para este caso si se una una entrada de dos bits en complemento a dos, la salida más grande posible la produce un escalón de entrada con valor -2, que produciría una

salida máxima de 3'686.400, que requiere 23 bits para su representación en complemento a dos. Sabiendo que la etapa inicial aumenta el tamaño de palabra como lo describe la Fig. 6.27. Además, se tienen dos filtros *comb* de dos etapas cada uno y dos simples, que incrementan al tamaño de palabra en un bit. Debido a la resta que incluyen, se tiene que el tamaño de palabra en la entrada de los integradores es de 18 bits. Además, se deben incluir otros 5 bits que permiten la representación de los valores obtenidos a la salida de la etapa integradora para un total de 23 bits requeridos a la salida.

6.5.2.2. Generación de las señales de reloj

Para este fin se requieren señales que controlen la actividad de los registros usados en cada etapa. Como se observa en la Fig. 6.14, se requieren tres modificaciones de la frecuencia de reloj original. La primera corresponde a una señal de reloj 7 veces más rápida que la original, luego, se requiere una señal que tenga la mitad de la frecuencia de muestreo de la primera y para la etapa final una señal que tenga una frecuencia de muestreo reducida 5 veces. Para el esquema presentado en la Fig. 6.19, se debe cambiar el orden de la modificación de las frecuencias de muestreo. Primero es necesario generar una señal de reloj que posea la frecuencia de entrada reducida 2 veces, lo que equivale a una señal que tiene el doble del período de muestreo original y a esa señal se le cambiará a frecuencia de muestreo por una que sea 7 veces su frecuencia de muestreo. Las señales de control de tiempo para las diferentes etapas se aprecian en la Fig. 6.29. Allí se observa el reloj principal clk , luego la señal con frecuencia reducida dos veces clk_1 , después la señal clk_2 que corresponde a la señal clk_1 con frecuencia aumentada 7 veces y por último la señal clk_2 correspondiente a la señal clk_1 con frecuencia reducida 5 veces.

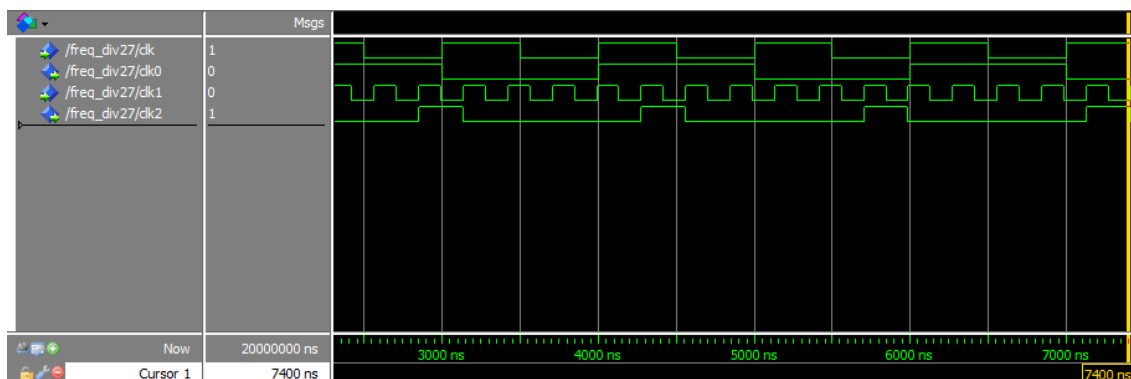


Figura 6.29: Señales reloj utilizadas para cada etapa del filtro del ejemplo 6.10.

Es importante mencionar que las señales clk y clk_1 son entradas al bloque generador, mientras clk_0 y clk_2 son generadas a partir de las entradas por medio de contadores y

comparaciones con las cuentas llevadas por los mismos, de los ciclos de reloj de las señales de entrada.

Los bloques submuestreadores se implementan con un registro que funciona con la frecuencia de entrada correspondiente, mientras que los sobremuestreadores corresponden a un contador que lleva la cuenta de los flancos de subida de la señal de alta frecuencia, en este caso clk_1 . Cuando ese contador llega a un múltiplo de 7 entonces en ese momento se cambia la señal de control de un multiplexor que tiene como entradas los ceros que se introducen entre las muestras de salida con frecuencia clk_0 . En la Fig. 6.30 se presenta un esquema de la introducción del sobremuestreador, además se incluye una parte del código de VHDL usado para su implementación.

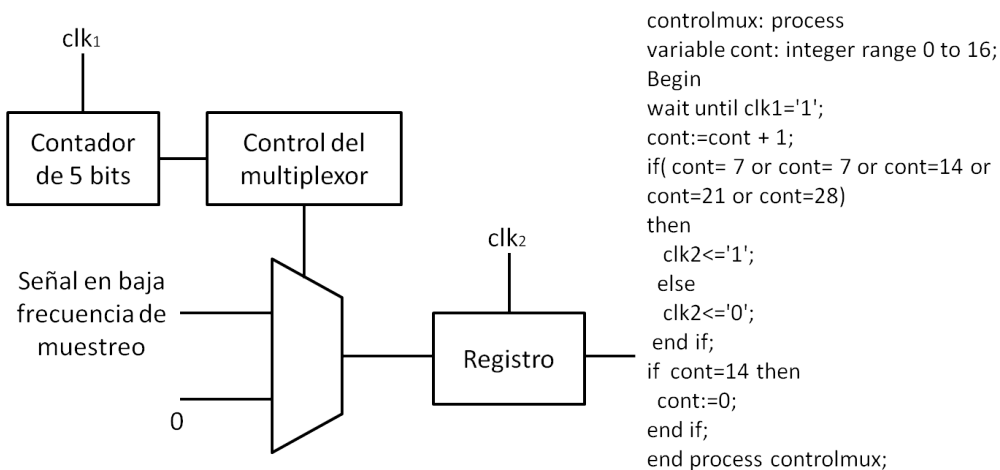


Figura 6.30: Implementación del sobremuestreador.

Por último se presenta la simulación del filtro completo en la Fig. 6.31. El software usado para dicha simulación fue *ModelSim* de Mentors Graphics Corporation. Como se observa es la misma respuesta encontrada en *Simulink*, algunos de los bits más significativos no fueron usados debido a que el tamaño de palabra de salida se consideró para una entrada de mayor tamaño. Se pueden apreciar las señales de reloj que controlan las diferentes etapas del filtro, que en el caso del submuestreo corresponden a registros con la señal de reloj correspondiente a la frecuencia reducida para esa etapa.

Ejemplo 6.11

Como para el ejemplo anterior, se presentan en esta sección las principales consideraciones para la implementación hardware del segundo ejemplo. Básicamente el problema

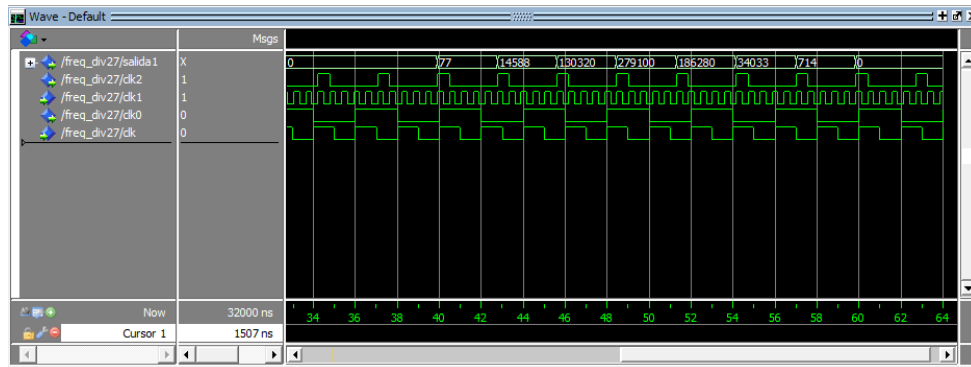


Figura 6.31: Respuesta al impulso del filtro del ejemplo 6.10.

de implementación es abordado de la misma forma, cambiando solamente el tamaño de palabra y las señales de reloj generadas para las respectivas etapas.

6.5.2.3. Generación de los coeficientes de la estructura eficiente y tamaños de palabra requeridas en cada etapa

Para la generación de las multiplicaciones requeridas para la primera etapa se utiliza el esquema basado en sumas y corrimientos de la Fig. 6.32, a diferencia del primer ejemplo, en el que todas las multiplicaciones se realizan sobre la misma rama. En este caso no todas son para la misma, debido a esto se replica deben usar réplicas del hardware dependiendo del coeficiente cuya multiplicación se requiera, por ejemplo para el producto por 51 no son requeridas las salidas de los productos por 20 y 25, requiriendo solamente 4 sumas, incluyendo una suma con la unidad que produce un incremento de 1 bit.

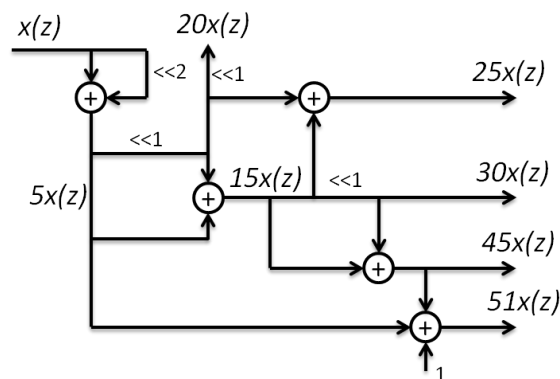


Figura 6.32: Esquema de realización eficiente de las multiplicaciones a base de corrimientos y sumas.

En la Fig. 6.33 se presenta el incremento del tamaño de palabra a lo largo de la primera

etapa del filtro. De la Fig. 6.15 se tiene un incremento para las etapas siguientes de 8 bits adicionales debido a las restas y a la cascada de los bloques que componen la etapa, lo que da un incremento total de 25 bits antes de entrar a la etapa integradora.

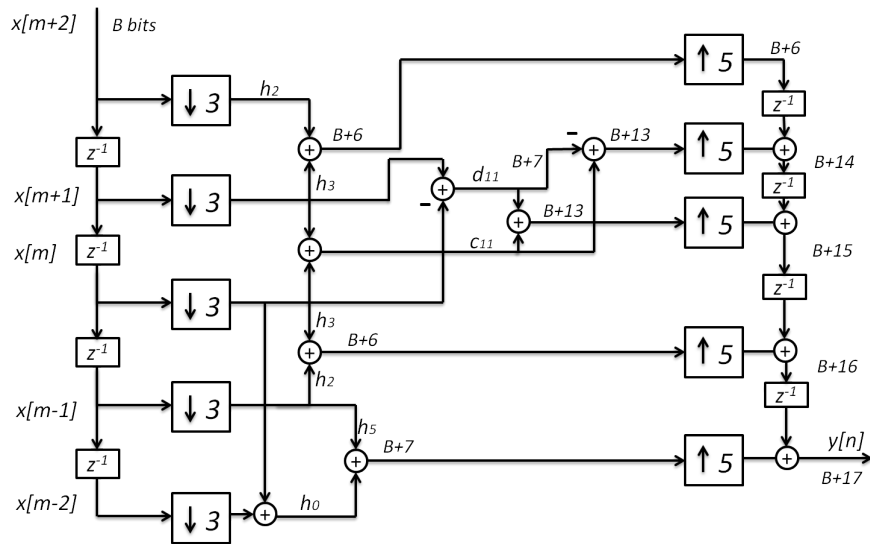


Figura 6.33: Incremento del tamaño de palabra para la primera etapa.

Los diagramas de bloques de los subfiltros utilizados se presentan en la Fig. 6.34. El primero, el segundo y el tercero tienen cascada de dos cada uno, mientras el cuarto y el quinto son filtros simples, que solo se usan una vez.

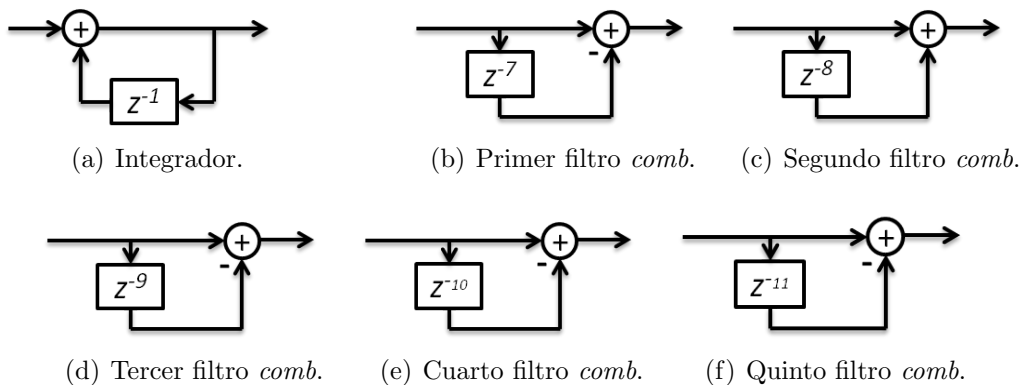


Figura 6.34: Diagramas de bloques para las etapas del filtro del ejemplo 6.11.

6.5.2.4. Generación de las señales de reloj

Primero se genera una señal de reloj con frecuencia reducida 3 veces, posteriormente, usando la señal anterior, se genera una señal con frecuencia de muestreo aumentada 5 veces y finalmente con esta señal se genera una señal con frecuencia reducida 7 veces, como se presenta en la Fig. 6.35. Es importante mencionar que la señales con frecuencia reducida poseen una forma con un ciclo útil modificado. Justamente esto permite que el esquema usado para el sobremuestreador funcione, ya que si tuviese un ciclo útil del 50% el registro a la salida en la Fig. 6.30 tomaría muestras después de los tiempos requeridos.

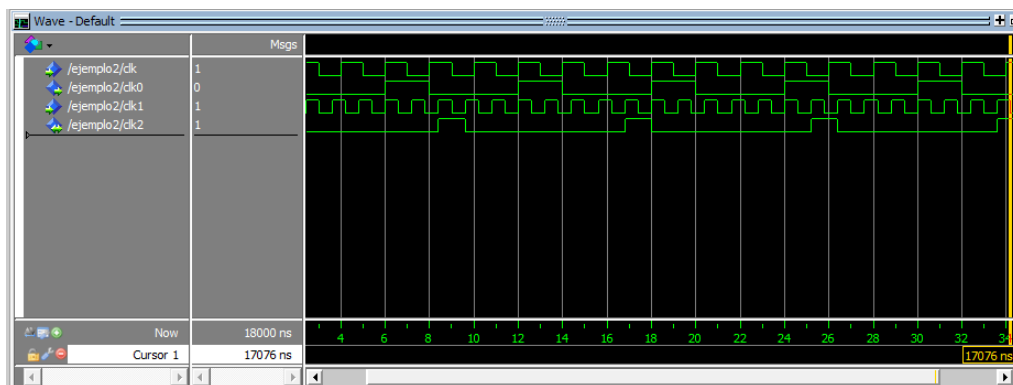


Figura 6.35: Señales de reloj generadas para el ejemplo 6.11.

Finalmente, se presenta la simulación en *ModelSim* de la respuesta al impulso del filtro del ejemplo 6.11. Se tiene una entrada de dos bits en complemento a dos, así que se usará un escalón de amplitud -3 para observar la amplitud de salida máxima a la salida y definir la cantidad de bits necesarios de modo que no se trunque, para este valor la salida máxima obtenida es 4'044.700 siendo requeridos 23 bits para su representación en complemento a dos. La respuesta al impulso obtenida se presenta en la Fig. 6.36.

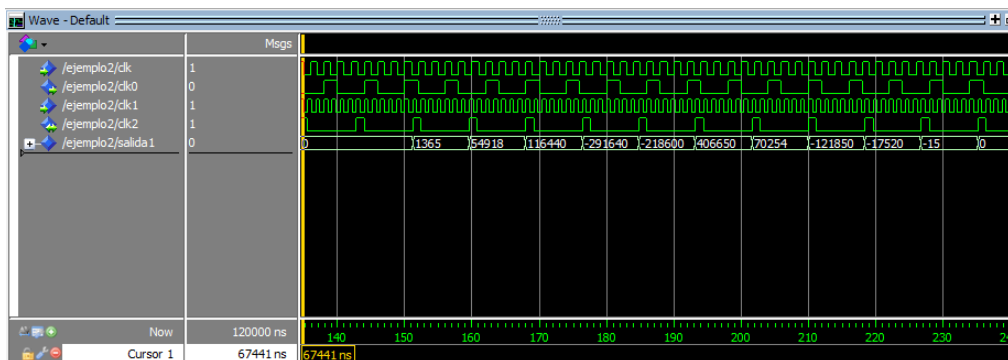


Figura 6.36: Respuesta al impulso para ejemplo 6.11.

Capítulo 7

Conclusiones y trabajo futuro

Conclusiones

Se propuso una estructura que cumple con los objetivos planteados (SNR mejorada y baja complejidad computacional). Además, se propuso una modificación de la misma que permitiera eliminar los integradores de la etapa de entrada que funcionan en la más alta frecuencia, y que en consecuencia poseen el mayor consumo de potencia toda la estructura.

La estructura propuesta fue obtenida mediante la combinación de dos métodos encontrados en la literatura: el filtro CIC modificado y el filtro CIC triangular escalonado. El primero permitía mejorar la atenuación en comparación al filtro CIC convencional. Para obtener una mejora adicional, se incluyó el segundo método, obteniendo una estructura con una SNR mejorada y un bajo incremento en la complejidad.

Se aprovechó la eficiencia del algoritmo que contempla la simetría de los coeficientes del filtro *anti-aliasing/anti-imaging* para obtener una estructura eficiente.

Trabajo futuro

Como trabajo futuro se plantean las siguientes propuestas:

- Mejorar la estructura propuesta en cuanto a la cantidad de APOS requeridas para su implementación y lograr un aumento de la SNR usando alguna técnica existente, por ejemplo *Sharpening*.
- Usar la estructura aquí propuesta para la generación de esquemas que permitan factores de conversión programables.

- Obtener una realización eficiente de los filtros *comb* que se encuentran después de la primera decimación mediante el uso de alguna técnica que lo permita, por ejemplo la decimación polifase.
- Desarrollo de un método que permita el diseño de compensadores libres de multiplicaciones para la corrección de la caída en la banda de paso del filtro.

Bibliografía

- [1] S. K. Mitra, *Procesamiento de senales digitales: Un enfoque basado en computadora*, 3rd ed. McGraw-Hill, 2007.
- [2] G. J. Dolecek, *Multirate systems: design and applications*. Idea Group Pub., 2002.
- [3] N. Fliege, *Multirate digital signal processing: multirate systems, filter banks, wavelets*. Wiley, 1994.
- [4] L. Milic, *Multirate filtering for digital signal processing*, 1st ed. Information Science Reference, 2009.
- [5] E. Hogenauer, “An Economical Class of Digital Filters for Decimation and Interpolation,” *IEEE Transactions on Acoustics, Speech and Signal Processing*, April 1981, vol. 29, no. 2, pp. 155–162.
- [6] G. J. Dolecek and S. K. Mitra, “Stepped Triangular CIC Filter for Rational Sample Rate Conversion,” Proc. of the *IEEE Asia Pacific Conference on Circuits and Systems, APCCAS 2006*, December 2006, pp. 916–919.
- [7] ———, “Design of FIR Lowpass Filters Using Stepped Triangular Approximation,” *5th Nordic Signal Processing Symposium. NORSIG-2002*, October 2002, pp. 215–219.
- [8] J. Vesma, *Optimization and Applications of Polynomial-based Interpolation Filters*, ser. Julkaisuja / Tampereen Teknillinen Korkeakoulu. Tampere University of Technology, 1999.
- [9] R. Bregovic, Y. J. Yu, Saramaki, T. ki, and Y. C. Lim, “Implementation of Linear-Phase FIR Filters for a Rational Sampling-Rate Conversion Utilizing the Coefficient Symmetry,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, March 2011, vol. 58, no. 3, pp. 548–561.

- [10] G. J. Dolecek, "Modified CIC Filter for Rational Sample Rate Conversion," Proc. of the *International Symposium on Communications and Information Technologies, ISCIT 2007*, October 2007, pp. 252–255.
- [11] F. Sheikh and S. Masud, "Efficient Sample Rate Conversion for Multi-Standard Software Defined Radios," Proc. of the *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2007*, April 2007, vol. 2, pp. II–329 –II–332.
- [12] D. Babic and M. Renfors, "Power Efficient Structure for Conversion Between Arbitrary Sampling Rates," *IEEE Signal Processing Letters*, January. 2005, vol. 12, no. 1, pp. 1–4.
- [13] S. Charanjit, S. Manjeet, and S. Sanjay, "Efficient Implementation of Sample Rate Conversion," Proc. of the *International Journal of Advanced Computer Science and Applications, IJACSA 2010*, December 2010, pp. 35–41.
- [14] F. de Aquino, C. da Rocha, and L. Resende, "Design of CIC Filters for Software Radio System," Proc. of the *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2006*, May 2006, vol. 3, p. III.
- [15] W. Abu-Al-Saud and G. Stuber, "Modified CIC Filter for Sample Rate Conversion in Software Radio Systems," *IEEE Signal Processing Letters*, May 2003, vol. 10, no. 5, pp. 152–154.
- [16] J. TecpanecatI-Xihuitl, R. Aguilar-Ponce, A. Kumar, and M. Bayoumi, "Design of a power-efficient interleaved CIC architecture for software defined radio receivers," Proc. of the *48th Midwest Symposium on Circuits and Systems*, August 2005, vol. 2, pp. 1831–1834 .
- [17] K. Yeung and S. Chan, "The Design and Multiplier-less Realization of Software Radio Receivers with Reduced System Delay," *IEEE Transactions on Circuits and Systems I: Regular Papers*, December 2004, vol. 51, no. 12, pp. 2444–2459.
- [18] D. Babic, J. Vesma, and M. Renfors, "Decimation by Irrational Factor Using CIC Filter and Linear Interpolation," Proc. of the *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2001*, May 2001, vol. 6, pp. 3677–3680.
- [19] N. Michael and A. Vinod, "Reconfigurable Architecture for Arbitrary Sample Rate Conversion in Software Defined Radios," Proc. of the *IEEE 19th International*

Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC 2008, September 2008, pp. 1–6.

- [20] Z. Jiang and J. Willson, A.N., “Efficient Digital Filtering Architectures Using Pipelining/Interleaving,” *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, February 1997, vol. 44, no. 2, pp. 110–119.