

## 5.- REALIZACIÓN DE SISTEMAS EN TIEMPO DISCRETO.

### 5.1.- INTRODUCCIÓN.

Los sistemas digitales que hemos analizado responden a una función de transferencia dada por:

$$H(z) = \frac{\sum_{k=0}^M b_k \cdot z^{-k}}{1 + \sum_{k=1}^N a_k \cdot z^{-k}}$$

Dado que la suma de convolución nos permite calcular la salida de un sistema LTI, éste sería un procedimiento adecuado para calcular la respuesta de un sistema ante una determinada entrada. Esta implementación es válida para sistemas FIR, ya que la suma de convolución es finita, pero no es utilizable para sistemas IIR, en los que se utiliza una implementación recursiva.

$$\text{IMPLEMENTACIÓN FIR. } y(n) = \sum_{k=0}^M h(k)x(n-k) \quad (1)$$

$$\text{IMPLEMENTACIÓN IIR: } y(n) = \sum_{k=0}^M b_k \cdot x(n-k) - \sum_{k=1}^N a_k \cdot y(n-k) \quad (2)$$

Observamos que, en ambos casos, se trata de una suma de productos entre los coeficientes del filtro y señales de entrada y salida retardadas.

La implementación puede ser hardware o software, según la aplicación. En ambos casos las señales y los coeficientes se representan con precisión finita.

La implementación DIRECTA de un filtro (sin manipular la función de transferencia, tal como indican las ecuaciones 1 y 2), basada en la suma de convolución o la ecuación en diferencias puede no proporcionar resultados satisfactorios debido a trabajar con aritmética finita, es por tanto útil analizar implementaciones alternativas que mejoren dichos resultados. Las representaciones alternativas realizan una reorganización del diagrama de bloques.

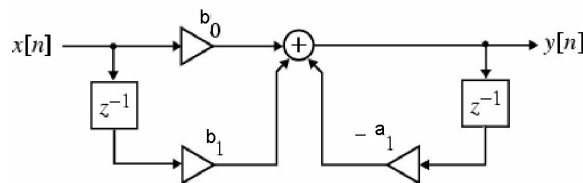
Los parámetros que van a determinar las prestaciones de una estructura son los siguientes:

- Efectos de precisión finita.
- Complejidad computacional. (Número de operaciones suma y producto necesarias)
- Requisitos de memoria. (Relacionado con el número de retardos presentes)
- Posibilidad de procesamiento en paralelo.

La representación de un sistema mediante bloques básicos conectados entre sí se denomina REALIZACIÓN O ESTRUCTURA. Ésta proporciona la relación entre la entrada y salida y algunas variables intermedias necesarias para su implementación.

Cuando se trabaja con precisión infinita, todas las estructuras proporcionan idénticos resultados, sin embargo estos cambian cuando la representación es finita.

Por ejemplo, dado el siguiente diagrama de bloques:



Extraído de: Digital Signal Processing. A computer-based approach. S. K. Mitra

La ecuación en diferencias es:  $y(n) = -a_1 y(n-1) + b_0 x(n) + b_1 x(n-1)$ , es decir podemos calcular  $y(n)$  para  $n \geq -1$  conociendo la condición inicial  $y(-1)$  y la entrada  $x(n)$  para  $n \geq -1$ . El algoritmo de computación es el siguiente:

$$y[0] = -a_1 y[-1] + b_0 x[0] + b_1 x[-1]$$

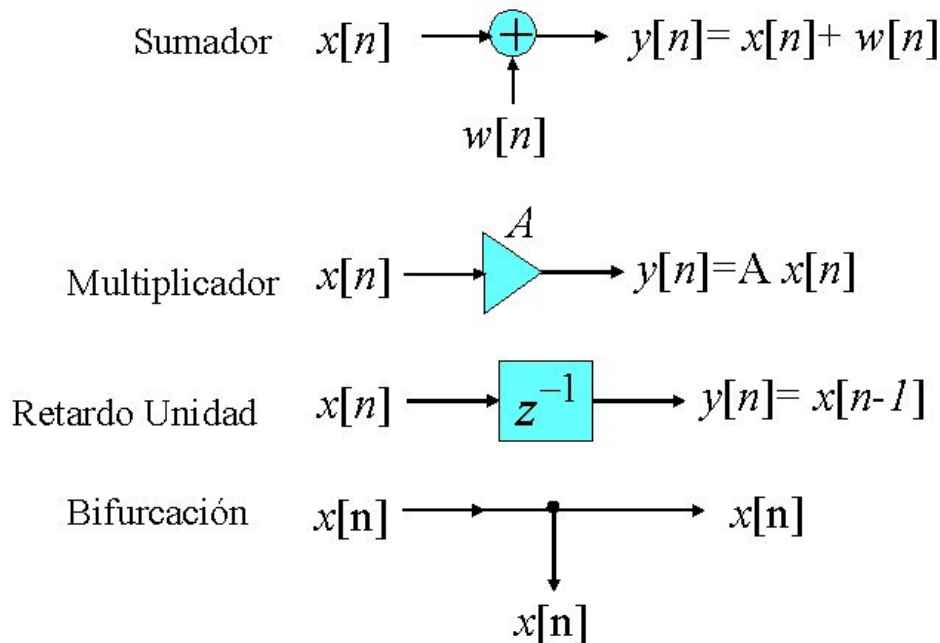
$$y[1] = -a_1 y[0] + b_0 x[1] + b_1 x[0]$$

$$y[2] = -a_1 y[1] + b_0 x[2] + b_1 x[1]$$

...

## 5.2.- Diagramas de bloques.

La representación de diagramas de bloques se realiza mediante los siguientes bloques básicos:



La representación en diagrama de bloques permite:

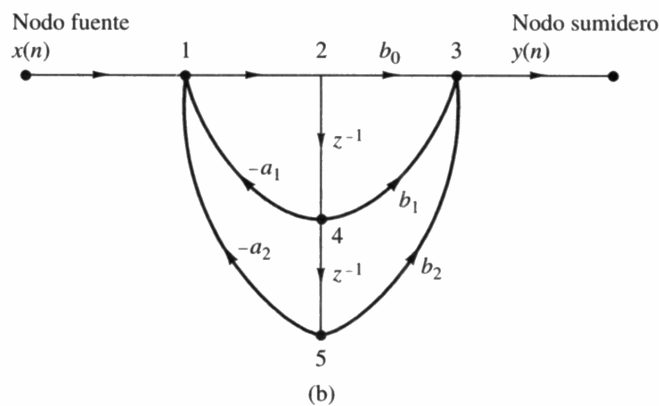
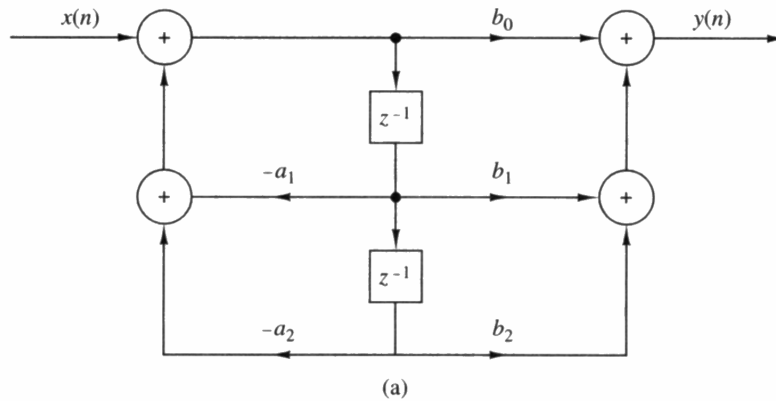
1. Escribir el algoritmo de computación.
2. Determinar la relación entre la entrada y la salida.
3. Manipular dicho diagrama para obtener otros equivalentes (igual relación entrada-salida), con distinto algoritmo de computación.
4. Determinar los requerimientos *hardware* (Memoria, carga computacional)

### Representación de diagramas de bloques mediante grafos.

Una representación alternativa a los diagramas de bloques son los GRAFOS. Un gráfico está formado por un conjunto de **nodos** conectados por **ramas orientadas**, y caracterizadas por una **transmitancia de rama**. La salida de un nodo es igual a la suma de las señales a su entrada. La siguiente figura muestra un sistema representado mediante diagrama de bloques y con su grafo correspondiente. Observamos dos nodos especiales:

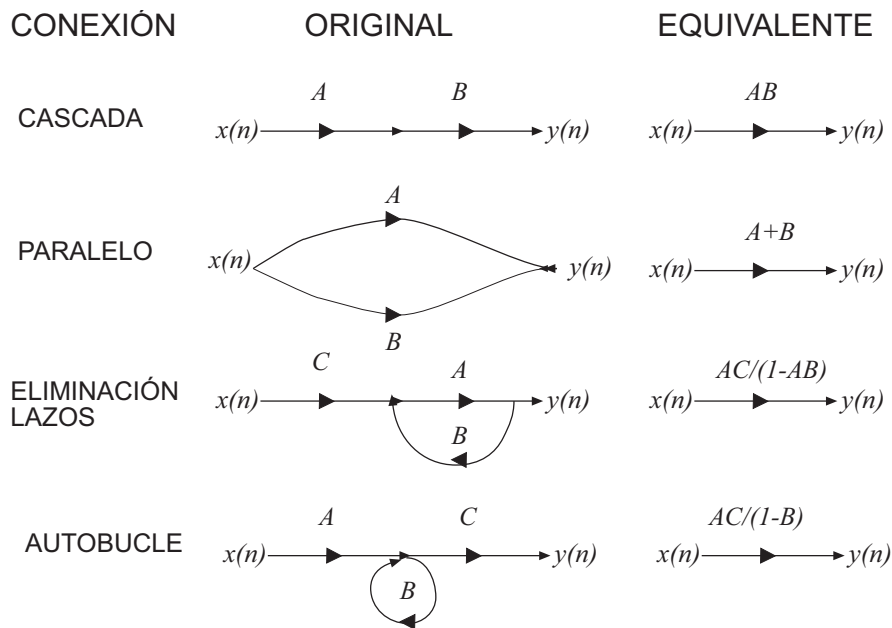
**Nodo fuente:** es aquel que no tiene ramas de entrada.

**Nodo sumidero:** es aquel que no tiene ramas de salida.



Extraído de: Tratamiento Digital de Señales. J.G. Proakis

Algunas propiedades de interés:



Para que un diagrama de bloques sea realizable físicamente no debe tener lazos (bucles) sin retardos. Si esto ocurre podemos sustituir dicho lazo sin retardo por su sistema equivalente aplicando las propiedades anteriores.

#### Estructuras canónicas y no canónicas.

Una estructura se dice que es CANÓNICA si el número de retardos presentes en el diagrama de bloques coincide con el orden del sistema, en otro caso se dice que es NO canónica. Un ejemplo de estructura canónica es el diagrama de bloques anterior.

#### Estructuras equivalentes.

Dos diagramas de bloques son EQUIVALENTES si representan a la misma función de transferencia.

Si bien la función de transferencia es la misma, las estructuras equivalentes difieren en:

- Carga computacional
- Robustez ante cálculos con registros de longitud finita.
- Sensibilidad frente a la cuantificación de los coeficientes.

#### Obtención de estructuras equivalentes: la TRASPOSICIÓN.

Un sencillo procedimiento para la obtención de estructuras equivalentes es la trasposición. Se basa en un teorema de la teoría de Grafos que nos dice lo siguiente:

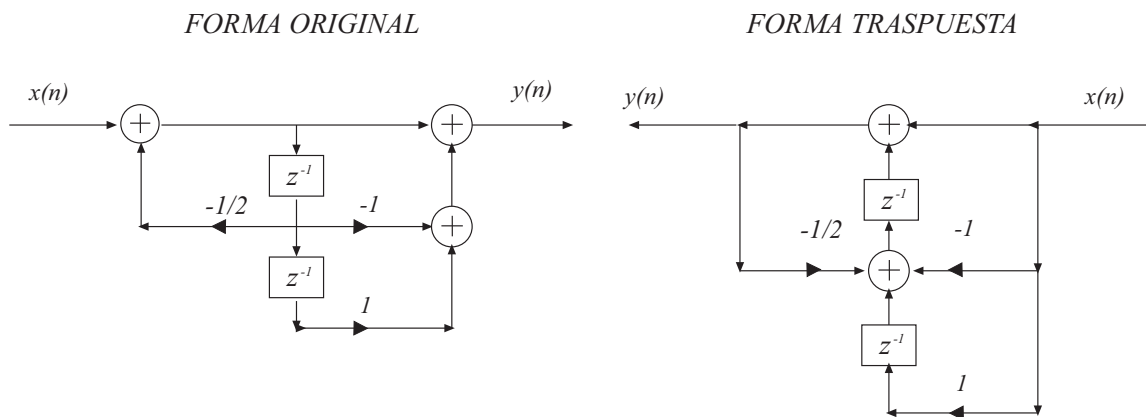
*Si en un grafo se intercambian la entrada y la salida, y la dirección de todas las ramas, el grafo obtenido es equivalente al original.*

Como consecuencia del cambio de sentido de todas las ramas los nodos distribuidores (bifurcación) pasarán a ser nodos suma y viceversa.

A lo largo de este capítulo veremos otros procedimientos para la obtención de estructuras equivalentes y en el siguiente, analizaremos el comportamiento de las estructuras más usuales frente a la cuantificación.

### TRASPOSICIÓN DE UNA ESTRUCTURA

ENTRADA <---> SALIDA  
 NODO FUENTE <---> NODO SUMIDERO  
 NODO SUMADOR <---> NODO DISTRIBUIDOR



### 5.3.- Estructuras Básicas para sistemas FIR.

#### 5.3.1.- Forma directa I

Se dice que una estructura es DIRECTA cuando los coeficientes de los multiplicadores coinciden con los de la ecuación en diferencias.

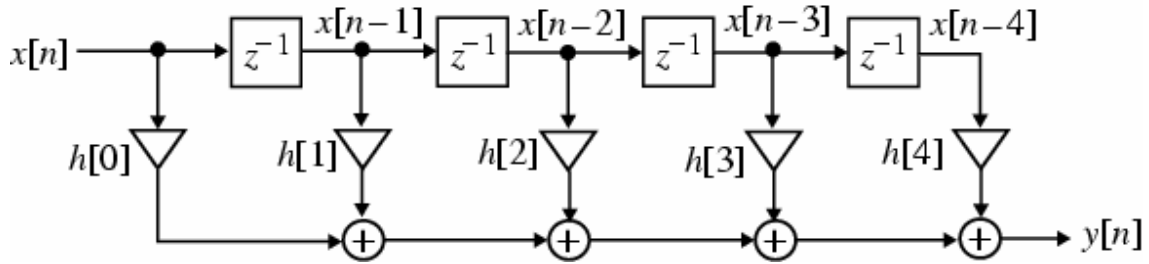
Dada la función de transferencia de un sistema FIR<sup>1</sup>  $H(z) = \sum_{k=0}^M h(k) \cdot z^{-k}$ , su ecuación en

diferencias viene dada por  $y(n) = \sum_{k=0}^M h(k) \cdot x(n-k)$ . Si representamos esta ecuación mediante

diagrama de bloques obtenemos la estructura siguiente:

<sup>1</sup> Nota téngase en cuenta que cuando estudiamos los filtros FIR su función de transferencia se definió como

$H(z) = \sum_{k=0}^{N-1} h(k) \cdot z^{-k}$ , siendo  $N$  el número de términos y  $N-1$  el orden. Ahora empleamos



Extraído de: Digital Signal Processing. A computer-based approach. S. K, Mitra.

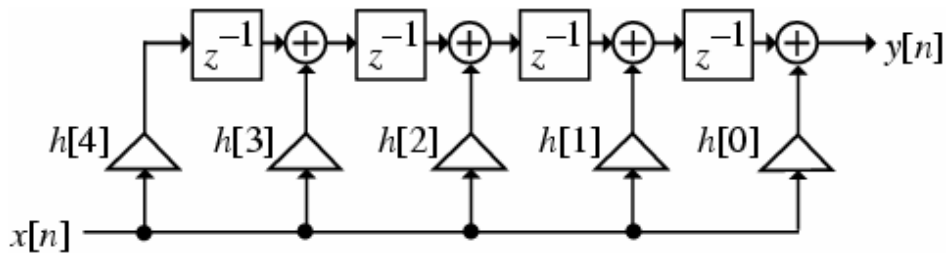
El sistema tiene M+1 coeficientes luego la carga computacional es:

- Productos: M+1
- Sumas: M

La estructura es canónica.

La forma directa I para los filtros FIR también se denomina LINEA DE RETARDO o FORMA TRANSVERSAL.

La forma traspuesta de esta estructura es:



Extraído de: Digital Signal Processing. A computer-based approach. S. K, Mitra.

Tanto la forma original como la traspuesta son canónicas.

### 5.3.2.- Formas en Cascada.

Dada la función de transferencia de un filtro FIR, puede ser descompuesta en secciones de segundo orden de acuerdo con la expresión:

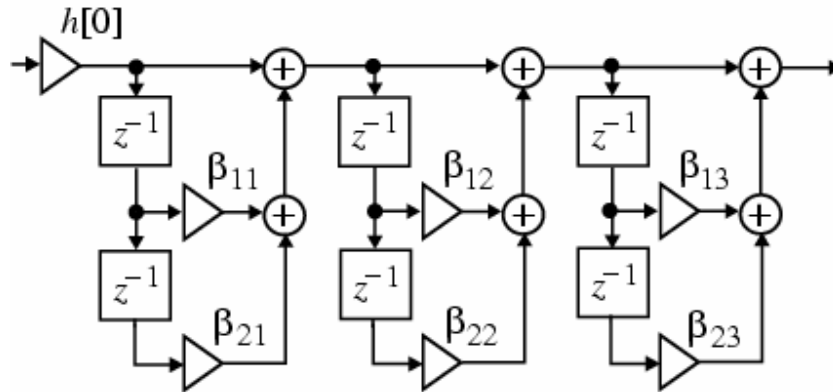
$$H(z) = h[0] \prod_{k=1}^K (1 + \beta_{1k} z^{-1} + \beta_{2k} z^{-2}) \text{ siendo } K = \frac{M}{2} \text{ para } M \text{ par y } K = \frac{M+1}{2} \text{ para } M \text{ impar.}$$

En este caso, en la última de las secciones  $\beta_{2k} = 0$ .

$$H(z) = \sum_{k=0}^M h(k) \cdot z^{-k} \text{ con lo que el número de términos es } M+1 \text{ y el orden } M, \text{ salvo en la estructura del}$$

muestreo en frecuencia, para poder reutilizar las expresiones ya calculadas.

Por ejemplo para  $M=6$ , una estructura en cascada formada por bloques representados mediante la forma directa I vendría dada por:



Extraído de: Digital Signal Processing. A computer-based approach. S. K. Mitra.

### 5.3.3.- Estructuras Polifase para sistemas FIR.

Una estructura polifase es un tipo de descomposición en paralelo. Estas descomposiciones realizan una reordenación de los coeficientes del filtro. Veámoslo mediante un ejemplo. Consideremos un filtro FIR con  $M=8$

$$H(z) = h[0] + h[1]z^{-1} + h[2]z^{-2} + h[3]z^{-3} + h[4]z^{-4} + h[5]z^{-5} + h[6]z^{-6} + h[7]z^{-7} + h[8]z^{-8}$$

Hagamos una reordenación de términos pares e impares

$$H(z) = (h[0] + h[2]z^{-2} + h[4]z^{-4} + h[6]z^{-6} + h[8]z^{-8}) + (h[1]z^{-1} + h[3]z^{-3} + h[5]z^{-5} + h[7]z^{-7})$$

sacando factor común  $z^{-1}$  en los términos impares

$$H(z) = (h[0] + h[2]z^{-2} + h[4]z^{-4} + h[6]z^{-6} + h[8]z^{-8}) + z^{-1}(h[1] + h[3]z^{-2} + h[5]z^{-4} + h[7]z^{-6})$$

y utilizando la notación siguiente:

$$E_0(z) = h[0] + h[2]z^{-1} + h[4]z^{-2} + h[6]z^{-3} + h[8]z^{-4}$$

$$E_1(z) = h[1] + h[3]z^{-1} + h[5]z^{-2} + h[7]z^{-3}$$

Podemos expresar.

$$H(z) = E_0(z^2) + z^{-1}E_1(z^2)$$

Los filtros  $E_0(z)$  y  $E_1(z)$  se denominan FILTROS SUBFASE.

Otra descomposición posible sería:



$$H(z) = E_0(z^3) + z^{-1}E_1(z^3) + z^{-2}E_2(z^3)$$

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

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

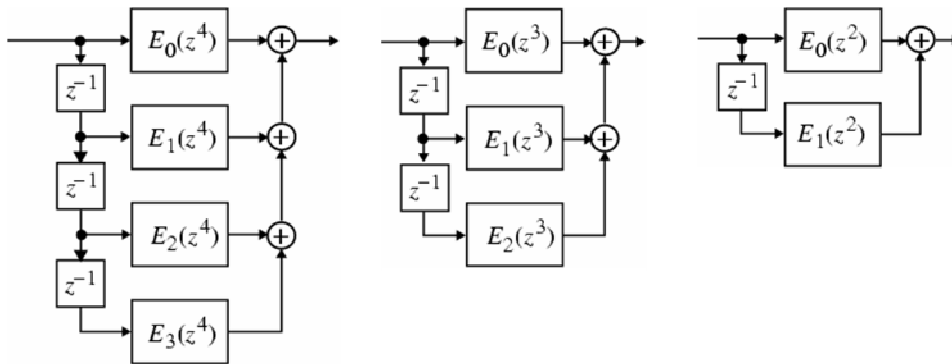
$$E_2(z) = h[2] + h[5]z^{-1} + h[8]z^{-2}$$

En general una descomposición polifase de L ramas para un filtro FIR de orden M, vendría definida por:

$$H(z) = \sum_{m=0}^{L-1} z^{-m} E_m(z^L)$$

$$E_m(z) = \sum_{n=0}^{\lfloor M/L \rfloor} h[Ln + m]z^{-n}$$

La siguiente figura muestra la estructura en paralelo obtenida para descomposiciones de 4,3 y 2 ramas.



Extraído de: Digital Signal Processing. A computer-based approach. S. K, Mitra.

Cada uno de los filtros subfase  $E_m(z^L)$  puede ser implementado con alguna de las estructura vistas anteriormente, si bien, para que la estructura resultante sea canónica, es necesario que estos filtros compartan los retardos.

Téngase en cuenta que dado un filtro FIR con función de transferencia  $H(z)$  el filtro  $H(z^L)$ , tiene una respuesta obtenida a partir de la original pero insertando L-1 ceros entre muestra consecutivas; es decir, se ha realizado una interpolación por un factor L.

Este tipo de descomposición se utiliza en el procesado de señales de tasa múltiple; es decir, procesado de señales en el que intervienen etapas de diezmado e interpolación, que veremos en el tema 7.

### 5.3.4.- Estructuras FIR de fase Lineal.

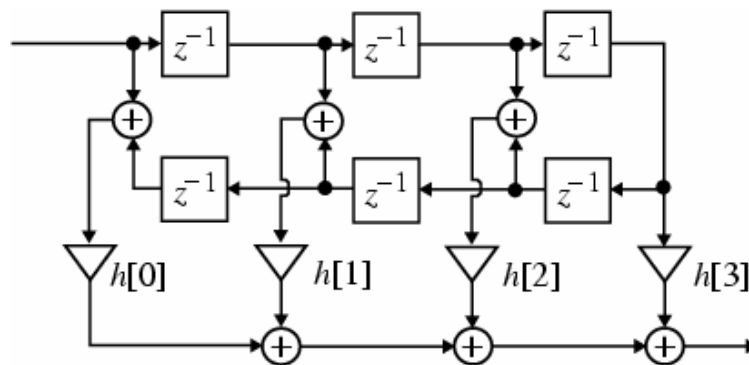
Si un filtro FIR es de fase lineal, la estructura directa puede ser simplificada ya que debido a la simetría (o antisimetría) de los coeficientes podemos reducir el número de productos. Por ejemplo, consideremos el siguiente filtro FIR que nos indican que es de tipo I:

$$H(z) = h[0] + h[1]z^{-1} + h[2]z^{-2} + h[3]z^{-3} + h[2]z^{-4} + h[1]z^{-5} + h[0]z^{-6}$$

Podemos agrupar los términos de la forma

$$H(z) = h[0](1 + z^{-6}) + h[1](z^{-1} + z^{-5}) + h[2](z^{-2} + z^{-4}) + h[3]z^{-3}$$

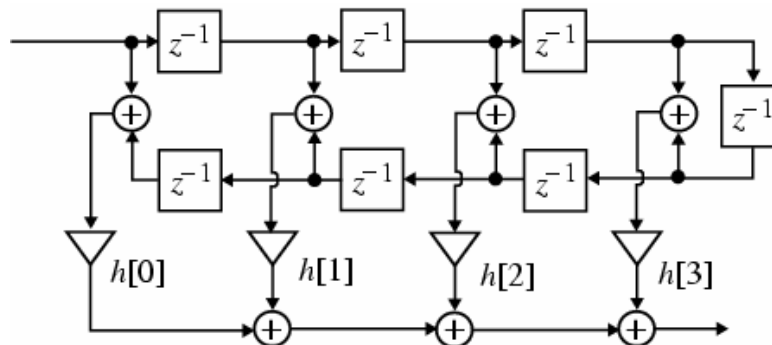
Con lo que la forma directa I sería:



Extraído de: Digital Signal Processing. A computer-based approach. S. K, Mitra.

La forma directa requiere 7 productos pero la estructura de fase lineal tiene sólo 4. (En general  $\frac{M+1}{2}$ ). Para un filtro de tipo II, podríamos realizar una descomposición análoga. Por ejemplo

un filtro con 8 términos tendría una estructura:



Extraído de: Digital Signal Processing. A computer-based approach. S. K, Mitra.

En este caso pasamos de 8 productos a 4 (en general  $\frac{M}{2}$ ). El número de sumas no se modifica.

Para los tipos III, y IV obtenemos expresiones similares, si bien habrá un cambio de signo en algunos coeficientes que se tendrá en consideración al realizar las sumas.

### 5.3.5.- Estructuras FIR de fase Lineal en cascada.

Para filtro FIR con ceros complejos, la fase lineal implica la existencia de ceros en

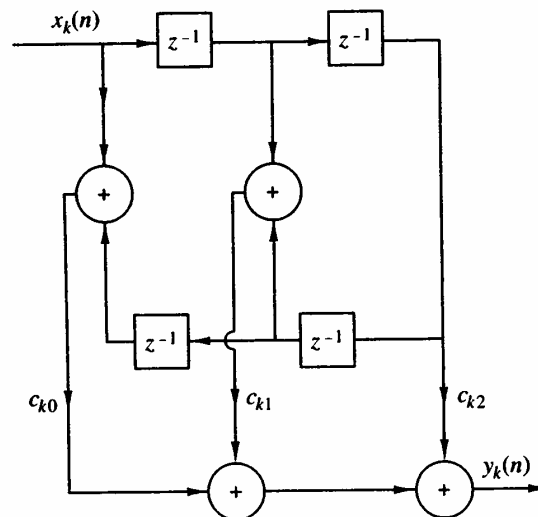
$z_k, z_k^*, \frac{1}{z_k}$  y  $\frac{1}{z_k^*}$  por lo que en la descomposición en cascada podemos agrupar términos de 4º

orden disminuyendo el numero de productos:

2 Secciones de 2º orden → 6 productos y 4 sumas

1 Sección de 4º orden → 3 productos y 4 sumas

$$H(z) = C_{k0} (1 - z_k z^{-1}) (1 - z_k^* z^{-1}) \left(1 - \frac{1}{z_k} z^{-1}\right) \left(1 - \frac{1}{z_k^*} z^{-1}\right) = C_{k0} + C_{k1} z^{-1} + C_{k2} z^{-2} + C_{k1} z^{-3} + C_{k0} z^{-4}$$



Extraído de: Tratamiento Digital de Señales. J.G. Proakis

### 5.3.6.- Estructuras de Muestreo en Frecuencia.

En el tema 3 vimos que para los filtros FIR de fase lineal, diseñados por el método del muestreo en frecuencia, su función de transferencia puede escribirse como:

$$H(z) = \frac{1 - z^{-M}}{M} \cdot \sum_{k=0}^{M-1} H(k) \cdot \frac{1}{1 - e^{j \frac{2\pi}{M} k} \cdot z^{-1}}$$

Siendo  $M$  el orden del filtro y  $H(k) = H\left(\frac{2\pi k}{M}\right)$   $0 \leq k \leq M-1$  las muestras de la respuesta en frecuencia consideradas.

La expresión de  $H(z)$  nos indica que ésta se puede expresar como una descomposición en cascada de dos filtros. A su vez el segundo filtro está expresado como una descomposición en paralelo que tendrá tantos términos como valores no nulos contenga  $H(k)$ . Para evitar tener coeficientes complejos podemos agrupar términos teniendo en cuenta las propiedades de simetría de la respuesta impulsional. Así, por ejemplo, para un filtro de tipo I tenemos  $H(M-k) = H^*(k)$  por lo que agrupando términos del tipo:

$$\frac{H(k)}{1 - e^{j\frac{2\pi}{M}k} \cdot z^{-1}} + \frac{H(M-k)}{1 - e^{j\frac{2\pi}{M}(M-k)} \cdot z^{-1}} = \frac{A(k) + B(k)z^{-1}}{1 - 2\cos\left(\frac{2\pi k}{M}\right) \cdot z^{-1} + z^{-2}} \quad \begin{aligned} A(k) &= H(k) + H(M-k) \\ B(k) &= H(k)e^{-j\frac{2\pi}{M}k} + H(M-k)e^{j\frac{2\pi}{M}k} \end{aligned}$$

Luego la expresión de  $H(z)$  queda como:

$$H(z) = \frac{1 - z^{-M}}{M} \cdot \left[ \frac{H(0)}{1 - z^{-1}} + \sum_{k=1}^{\frac{M-1}{2}} \frac{A(k) + B(k)z^{-1}}{1 - 2\cos\left(\frac{2\pi k}{M}\right) \cdot z^{-1} + z^{-2}} \right] \text{ M impar}$$

Si el número de términos fuese par obtendríamos expresiones similares:

$$H(z) = \frac{1 - z^{-M}}{M} \cdot \left[ \frac{H(0)}{1 - z^{-1}} + \underbrace{\frac{H(M/2)}{1 + z^{-1}}}_{\text{vale 0}} + \sum_{k=1}^{\frac{M-1}{2}} \frac{A(k) + B(k)z^{-1}}{1 - 2\cos\left(\frac{2\pi k}{M}\right) \cdot z^{-1} + z^{-2}} \right] \text{ M par}$$

Si analizamos la carga computacional observamos que cada uno de los bloques de segundo orden del banco de filtros, requiere 3 operaciones producto, es decir en total  $3 \cdot \frac{M-1}{2}$  productos. Aunque a priori parece mayor que una estructura directa FIR de fase lineal, en el caso de filtros en los que la banda de paso sea estrecha, el número de muestras de la respuesta en frecuencia no nulas es muy reducido, por lo que considerar una estructura de muestreo en frecuencia puede ser más adecuado.

**Ejemplo:**

Dibuja el diagrama de bloques en forma directa y muestreo en frecuencia de un sistema FIR de fase lineal simétrico número de términos  $M=32$ , sabiendo que las muestras de la respuesta en frecuencia vienen definidas como

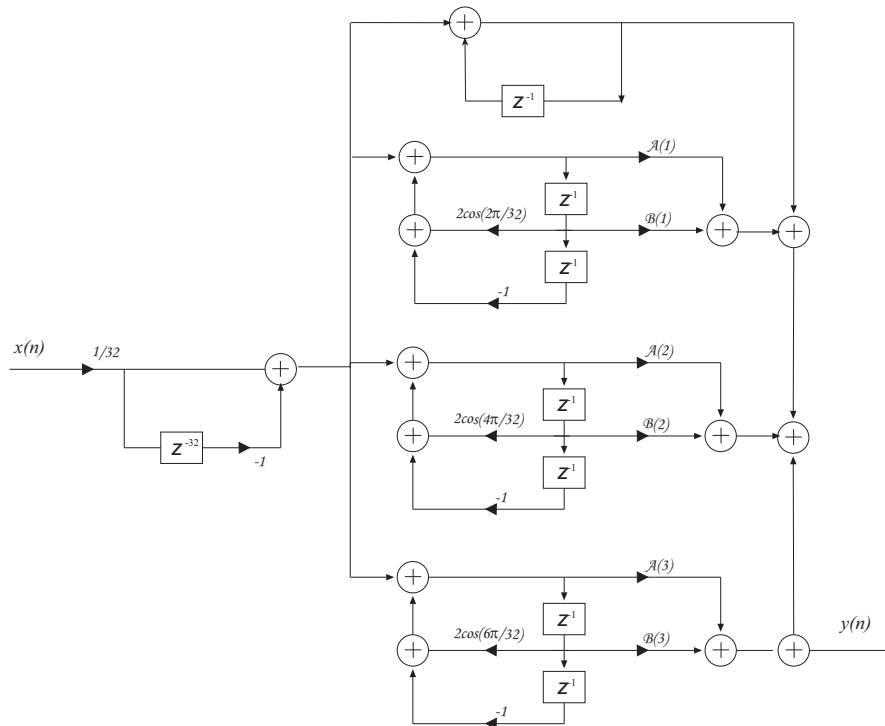
$$H\left(\frac{2\pi k}{M}\right) = A\left(\frac{2\pi k}{M}\right) e^{j\Phi\left(\frac{2\pi k}{M}\right)}, \text{ con}$$

$$A\left(\frac{2\pi k}{M}\right) = \begin{cases} 1 & k = 0,1,2 \\ 1/2 & k = 3 \\ 0 & k = 4,5,\dots,15 \end{cases} \quad \text{y } \Phi\left(\frac{2\pi k}{M}\right) \text{ la correspondiente a los filtros FIR de tipo II}$$

Analiza la complejidad computacional de cada una de las estructuras.

Se trata de un filtro FIR de tipo II, por lo que para una implementación directa necesitamos  $\frac{M}{2} = 16$  productos, donde hemos tenido en cuenta la simetría de los coeficientes.

Por el método del muestreo en frecuencia observamos que únicamente vamos a tener un término correspondiente a la muestra 0, y 3 bloques de segundo orden donde agrupamos el resto de muestras no nulas. Además la muestra en cero es la unidad por lo que no requiere ningún producto. Como cada bloque de segundo orden requiere 3 productos tendremos en total 9 productos. Las estructura obtenida se muestra en la figura siguiente. Observamos como en este caso se ha reducido la carga computacional.

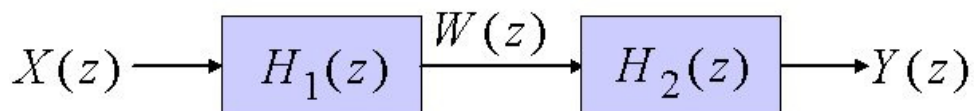


#### 5.4.- Estructuras Básicas para sistemas IIR.

A partir de la ecuación en diferencias de un filtro IIR lo primero que observamos es que ésta está definida de forma recursiva ( $y(n) = \sum_{k=0}^M b_k \cdot x(n-k) - \sum_{k=1}^N a_k \cdot y(n-k)$ ). Dicha recursividad se reflejará en el diagrama de bloques como una realimentación de la salida del sistema hacia elementos previos.

##### 5.4.1.- Formas directas

Dada la función de transferencia de un filtro IIR, consideremos expresado como una descomposición en cascada de un sistema todo ceros y otro todo polos.

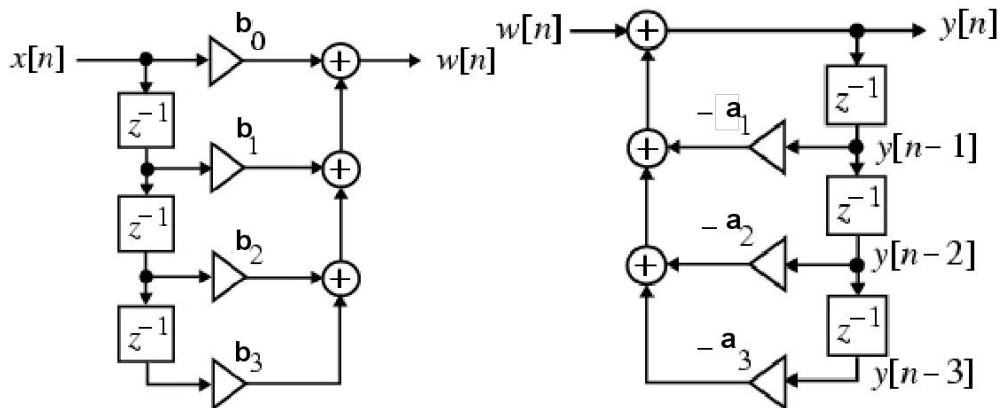


$$H_1(z) = \frac{W(z)}{X(z)} = P(z) = \sum_{k=0}^M b_k \cdot z^{-k}$$

$$H_2(z) = \frac{Y(z)}{W(z)} = \frac{1}{D(z)} = \frac{1}{1 + \sum_{k=1}^M a_k \cdot z^{-k}}$$

Las siguientes figuras muestran la implementación de  $H_1(z)$  y  $H_2(z)$  para  $N=M=3$ ,

**Forma directa I**



Extraído de: Digital Signal Processing. A computer-based approach. S. K. Mitra

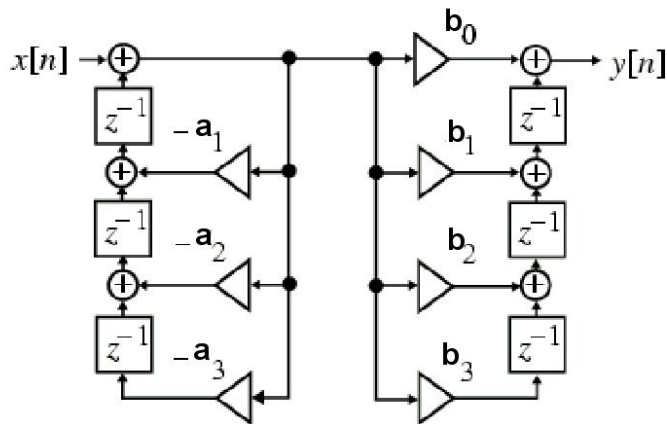
Las ecuaciones en diferencias son:

$$w[n] = b_0x[n] + b_1x[n-1] + b_2x[n-2] + b_3x[n-3]$$

$$y[n] = w[n] - a_1y[n-1] - a_2y[n-2] - a_3y[n-3]$$

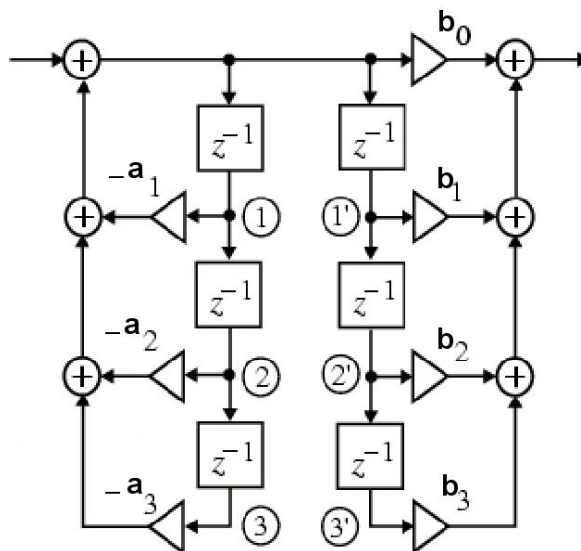
La conexión en cascada de ambos diagramas, como se muestra en la figura siguiente se denomina FORMA DIRECTA I (versión traspuesta). Esta estructura no es canónica ya que el sistema es de orden= $\max(N,M)=3$  y el número de retardos es 6.

## FD.I Traspuesta



Extraído de: Digital Signal Processing. A computer-based approach. S. K. Mitra

Por otra parte, teniendo en cuenta que la conexión en cascada es conmutativa podemos intercambiar la conexión dando lugar a una estructura equivalente que tiene la particularidad que podemos eliminar una de las ramas centrales ya que en ambas tenemos las mismas señales en los nodos 1--> 1' 2--> 2' y 3--> 3' tal como se indica en la figura.

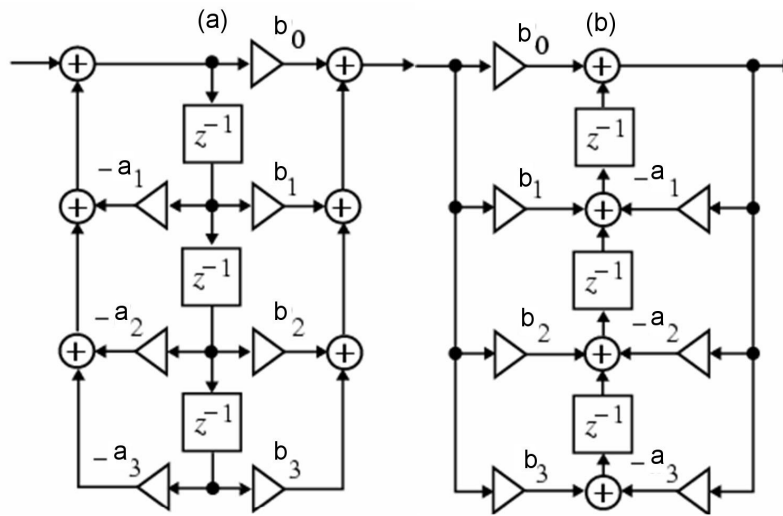


Extraído de: Digital Signal Processing. A computer-based approach. S. K. Mitra

En la siguiente figura mostramos la estructura resultante, llamada FORMA DIRECTA II (a), y su versión traspuesta (b).

Ambas estructuras son CANÓNICAS. El número de productos es el mismo en ambas, y si consideramos sumas de 2 sumandos, el número de sumas también coincide (Téngase en cuenta que en la rama central hay 2 sumas de 3 sumando cada una.)





Extraído de: Digital Signal Processing. A computer-based approach. S. K, Mitra

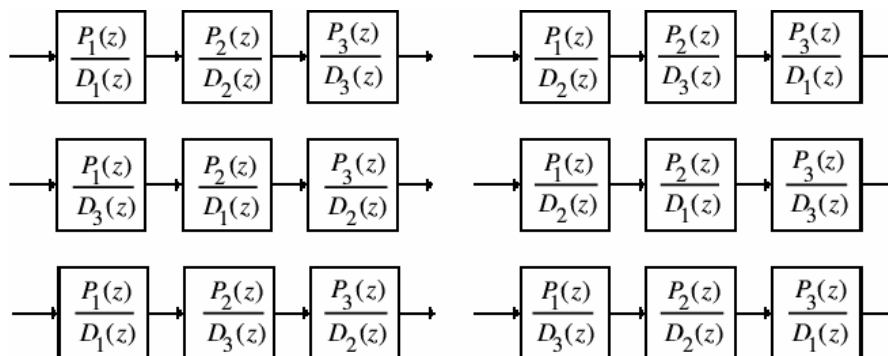
### 5.4.2.- Descomposiciones en cascada IIR.

Expresando el numerador y denominador de la función de transferencia como un producto de polinomios de menor orden (en general de orden 1 para raíces reales, y orden 2 para raíces complejas) un filtro digital puede expresarse como una cascada de secciones de menor orden.

Por ejemplo consideremos un sistema factorizado de la forma:

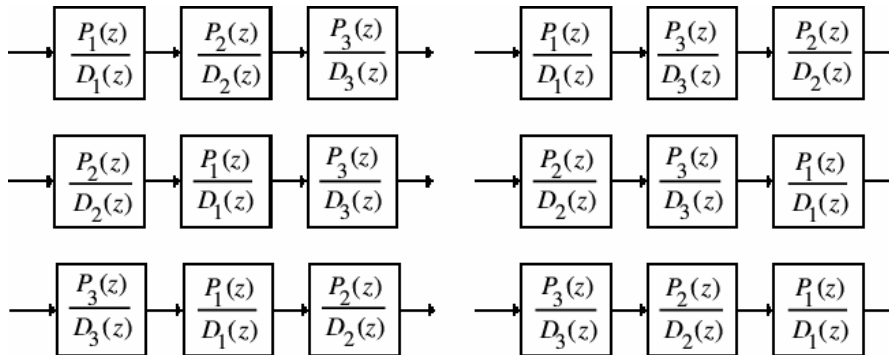
$$H(z) = \frac{P(z)}{D(z)} = \frac{P_1(z)P_2(z)P_3(z)}{D_1(z)D_2(z)D_3(z)}$$

A partir de esta descomposición tenemos, por una parte, 6 formas distintas de agrupar los ceros y polos tal como se indica:



Extraído de: Digital Signal Processing. A computer-based approach. S. K, Mitra

y además para cada agrupación tenemos 6 formas distintas de ordenar las secciones:



Extraído de: Digital Signal Processing. A computer-based approach. S. K. Mitra

Tenemos un total de 36 combinaciones posibles de conexión en cascada. Cada una de ellas tiene un comportamiento distinto cuando se trabaja con aritmética de coma fija.

En general  $H(z)$  es factorizada en etapas de primer y segundo orden de la forma siguiente:

$$H(z) = b_0 \prod_k \left( \frac{1 + \beta_{1k} z^{-1} + \beta_{2k} z^{-2}}{1 + \alpha_{1k} z^{-1} + \alpha_{2k} z^{-2}} \right) \text{ con } \alpha_{2k} = \beta_{2k} = 0 \text{ para etapas de primer orden.}$$

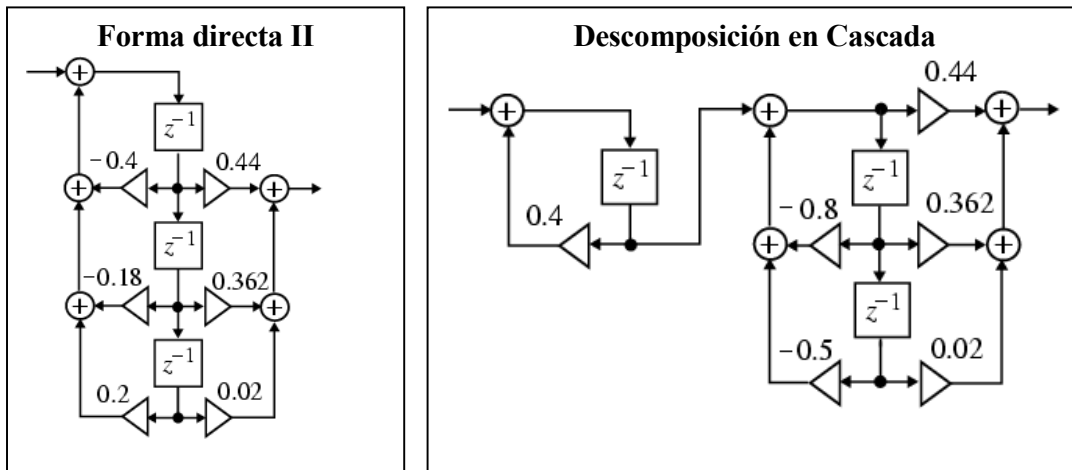
### Ejemplo:

Representa la forma directa II y una descomposición es cascada para el sistema

$$H(z) = \frac{0.44z^{-1} + 0.362z^{-2} + 0.02z^{-3}}{1 + 0.4z^{-1} + 0.18z^{-2} - 0.2z^{-3}}$$

Si factorizamos la expresión anterior obtenemos:  $H(z) = \left( \frac{0.44 + 0.362z^{-1} + 0.02z^{-2}}{1 + 0.8z^{-1} + 0.5z^{-2}} \right) \left( \frac{z^{-1}}{1 - 0.4z^{-1}} \right)$

Luego los diagramas de bloques obtenidos serán:



Extraído de: Digital Signal Processing. A computer-based approach. S. K, Mitra

Si analizamos las cargas computacionales, cada bloque de segundo orden requiere 4 Sumas y 5 Productos.

Veremos más adelante que las descomposiciones en cascada presentan un buen comportamiento cuando se trabaja con registros finitos, además la utilización de bloques repetidos permite reutilizar el código. Cada etapa de segundo orden se suele implementar mediante la forma directa II para disminuir el número de retardos, dando así lugar a formas canónicas.

### 5.4.3.- Descomposicion en paralelo IIR.

Si la función de transferencia de un sistema IIR causal se descompone en fracciones simples y agrupamos en términos de primer y segundo orden, para tener coeficientes reales, llegamos a una expresión del tipo:

$$H(z) = \gamma_0 + \sum_k \left( \frac{\gamma_{0k} + \gamma_{1k}z^{-1}}{1 + \alpha_{1k}z^{-1} + \alpha_{2k}z^{-2}} \right) \text{ con } \alpha_{2k} = \gamma_{1k} = 0 \text{ para polos reales.}$$

Posteriormente cada una de los términos suma de 1<sup>er</sup> y 2<sup>o</sup> orden se implementarán con alguna de las estructuras vistas, en general la forma directa II.

**Ejemplo:**

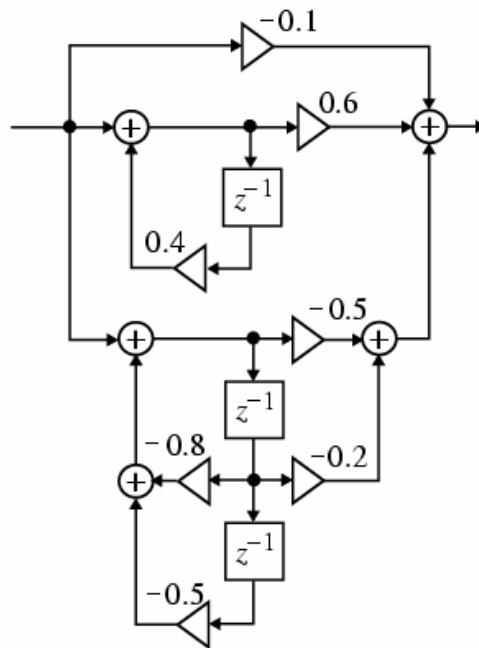
Realizar una descomposición en paralelo del sistema definido por la función de

transferencia siguiente: 
$$H(z) = \frac{0.44z^{-1} + 0.362z^{-2} + 0.02z^{-3}}{1 + 0.4z^{-1} + 0.18z^{-2} - 0.2z^{-3}}$$

Si realizamos una descomposición en potencias de  $z^{-1}$  obtenemos:

$$H(z) = -0.1 + \frac{0.6}{1 - 0.4z^{-1}} + \frac{-0.5 - 0.2z^{-1}}{1 + 0.8z^{-1} + 0.5z^{-2}}$$

Cuyo diagrama de bloques, utilizando la forma directa II, es el siguiente:



Extraído de: Digital Signal Processing. A computer-based approach. S. K. Mitra

**5.5.- Estructuras en CELOSÍA (lattice).**

La estructura en celosía (*lattice*), ampliamente utilizada en el procesado de voz, se caracteriza por su robustez numérica y modularidad para su implementación, lo que la hace muy adecuada para la implementación de filtros. Vamos a analizar 3 casos: sistema todo ceros (MA), sistema todo polos (AR), y sistema con ceros y polos (ARMA).

### 5.5.1.- Celosía FIR.

Dado un filtro FIR con función de transferencia  $H(z) = \sum_{k=0}^M h(k)z^{-k}$  vamos a definir un

$$\text{conjunto de filtros } A_m(z) = \sum_{k=0}^m \alpha_m(k)z^{-k} \quad m \geq 1 \quad H(z) = A_M(z) \\ \alpha_m(0) = 1$$

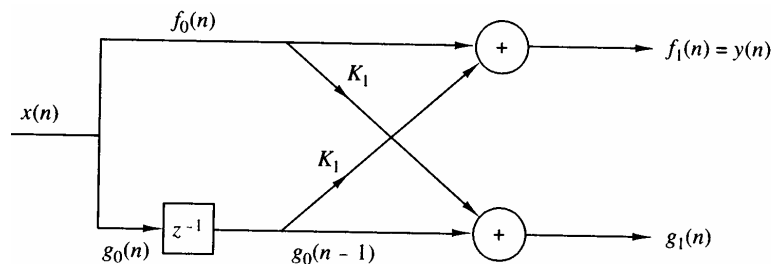
La respuesta impulsional es:  $h_m(0) = 1, \dots, h_m(k) = \alpha_m(k)$

Para este conjunto de filtros la ecuación en el dominio temporal será:

$$A_m(z) = \frac{Y(z)}{X(z)} \Rightarrow y(n) = x(n) + \underbrace{\sum_{k=1}^m \alpha_m(k)x(n-k)}_{\text{Predictor Lineal}}$$

Para un filtro de orden 1:  $m = 1 \quad y(n) = x(n) + \alpha_1(1)x(n-1)$

En lugar de las estructuras anteriores consideremos la siguiente:



Extraído de: Tratamiento Digital de Señales. J.G. Proakis

Las ecuaciones son:

$$\left. \begin{aligned} f_1(n) = y(n) &= f_0(n) + k_1 g_0(n-1) \\ g_1(n) &= f_0(n)k_1 + g_0(n-1) \end{aligned} \right\} \Rightarrow y(n) = x(n) + k_1 x(n-1)$$

Si  $k_1 = \alpha_1(1)$  esta estructura representa al filtro de orden 1.

Si consideramos 2 etapas en cascada tendremos

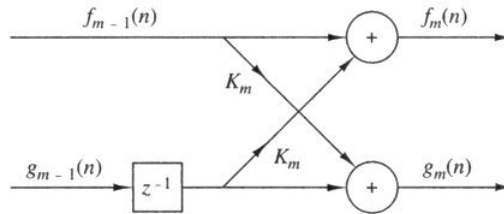
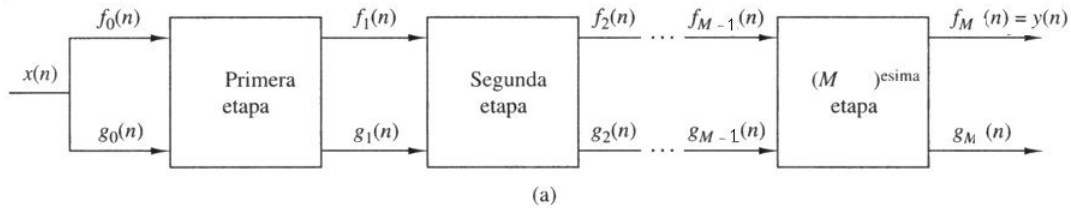
$$\left. \begin{aligned} f_2(n) = y(n) &= f_1(n) + k_2 g_1(n-1) \\ g_2(n) &= f_1(n)k_2 + g_1(n-1) \end{aligned} \right\} \Rightarrow \begin{aligned} y(n) &= (f_0(n) + k_1 g_0(n-1)) + k_2 (f_0(n-1)k_1 + g_0(n-2)) \\ y(n) &= x(n) + k_1(1 + k_2)x(n-1) + k_2 x(n-2) \end{aligned}$$

Luego :  $\alpha_2(2) = k_2, \quad \alpha_2(1) = k_1(1 + k_2)$

Si calculamos  $g_2(n) = k_2x(n) + k_1(1 + k_2)x(n-1) + x(n-2)$  observamos que la expresión es como  $f_2(n)$  pero reflejando los coeficientes.

Los valores  $k_i$  se denominan COEFICIENTES DE REFLEXIÓN.

En general para un sistema de M bloques tendremos:



Extraído de: Tratamiento Digital de Señales. J.G. Proakis

Como  $f_M(n)$  es la salida del sistema la podemos expresar como:

$$y(n) = f_M(n)$$

$$f_m(n) = \sum_{k=0}^m \alpha_m(k)x(n-k)$$

$$F_m(z) = A_m(z)X(z)$$

$$A_m(z) = \sum_{k=0}^m \alpha_m(k)z^{-k}$$

Análogamente para  $g_m(n)$

$$g_m(n) = \sum_{k=0}^m \beta_m(k)x(n-k)$$

$$G_m(z) = B_m(z)X(z)$$

$$\beta_m(k) = \alpha_m(m-k)$$

$$B_m(z) = \sum_{k=0}^m \beta_m(k)z^{-k}$$

Se puede obtener fácilmente que  $B_m(z) = z^{-m}A_m(z^{-1})$

Si consideramos la expresión general de la celosía y tomamos transformadas Z

$$\left. \begin{aligned} f_0(n) &= x(n) \\ g_0(n) &= x(n) \\ f_m(n) &= f_{m-1}(n) + k_m g_{m-1}(n-1) \\ g_m(n) &= k_m f_{m-1}(n) + g_{m-1}(n-1) \end{aligned} \right\} \xrightarrow{TZ} \left. \begin{aligned} F_0(z) &= X(z) \\ G_0(z) &= X(z) \\ F_m(z) &= F_{m-1}(z) + k_m z^{-1} G_{m-1}(z) \\ G_m(z) &= k_m F_{m-1}(z) + z^{-1} G_{m-1}(z) \end{aligned} \right\}$$

Dado que  $F_m(z) = A_m(z)X(z)$  y  $G_m(z) = B_m(z)X(z)$  podemos obtener la relación  
**(Directa → Celosía)**

$$A_{m-1}(z) = \frac{A_m(z) - k_m B_m(z)}{1 - k_m^2}$$

$$\text{Como } B_m(z) = z^{-m} A_m(z^{-1}), \alpha_m(m) = k_m, \alpha_m(0) = 1$$

Que nos permite obtener los coeficientes de reflexión a partir de  $H(z)$ .

Para obtener la expresión de  $H(z)$  conocidos los coeficientes de reflexión utilizaremos las expresiones **(Celosía → Directa)**

$$A_m(z) = A_{m-1}(z) + k_m z^{-1} B_{m-1}(z)$$

$$B_m(z) = z^{-m} A_m(z^{-1})$$

$$B_0(z) = A_0(z) = 1$$

### 5.5.2.- Celosía IIR todo polos.

Dada función de transferencia de un sistema todo polos  $H(z) = \frac{1}{1 + \sum_{k=1}^N \alpha_N(k) z^{-k}} = \frac{1}{A_N(z)}$

La ecuación en diferencias será:  $y(n) = -\sum_{k=1}^N \alpha_N(k) y(n-k) + x(n)$ , si en este sistema

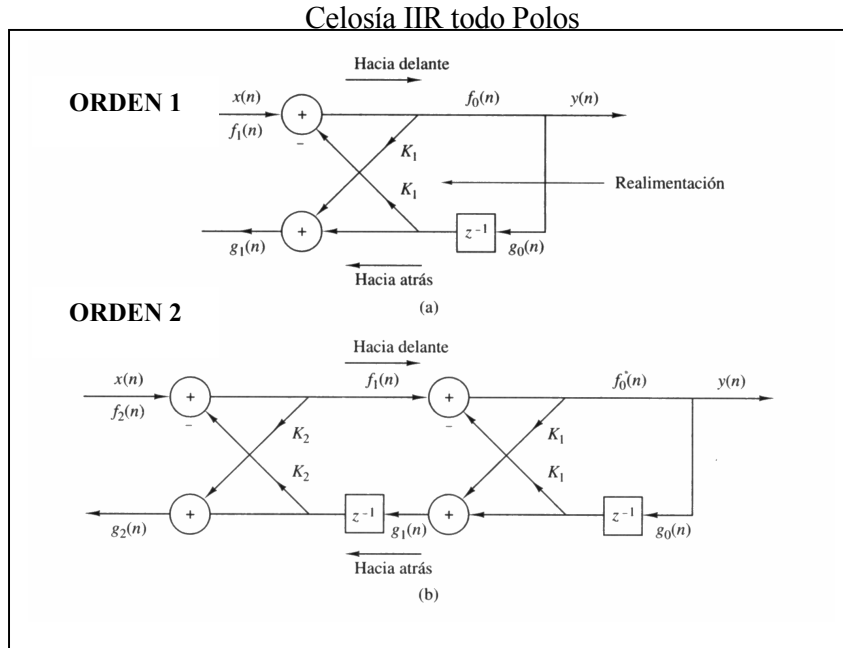
intercambiamos la salida y la entrada tenemos:  $y(n) = x(n) + \sum_{k=1}^N \alpha_N(k) x(n-k)$ , que es un

sistema FIR del que ya conocemos la relación entre la función de transferencia y los coeficientes de reflexión.

Si utilizamos las ecuaciones de la celosía FIR e intercambiamos entrada y salida tenemos las ecuaciones siguientes para la celosía IIR todo polos:

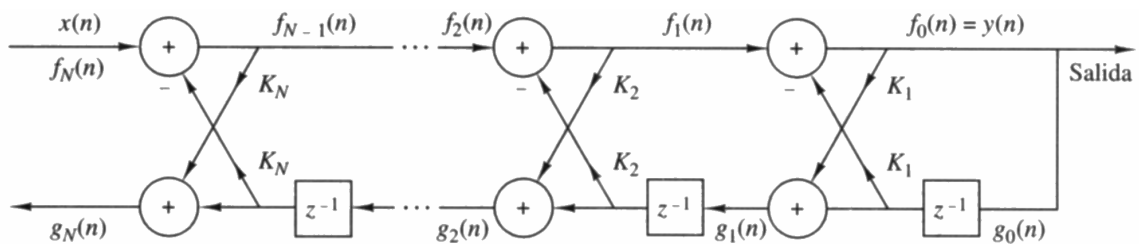
$$\begin{aligned}
 x(n) &= f_N(n) \\
 f_{m-1}(n) &= f_m(n) - k_m g_{m-1}(n-1) \\
 g_m(n) &= k_m f_{m-1}(n) + g_{m-1}(n-1) \\
 y(n) &= f_0(n) = g_0(n)
 \end{aligned}$$

Si tenemos en cuenta estos cambios en la estructura, obtenemos los diagramas de bloques que a continuación se muestran.



Extraído de: Tratamiento Digital de Señales. J.G. Proakis

En general:



Extraído de: Tratamiento Digital de Señales. J.G. Proakis

En el diagrama observamos claramente la realimentación del sistema a través de las señales  $g_i(n)$  propia de los sistemas recursivos.

Los coeficientes de reflexión son idénticos a los obtenidos para el filtro FIR, si bien en el diagrama se ordenan en orden inverso.



La estabilidad del filtro IIR solo polos está garantizada si  $|k_m| < 1 \quad \forall m$ . (Test de estabilidad de Schur-Cöhn)

Este tipo de filtros se ha utilizado para modelizar el tracto vocal, en este sentido,  $k_m$  representa la reflexión del sonido en cada una de las diferentes cavidades que lo forman.

### 5.5.3.- Celosía escalonada (*Lattice-Ladder*).

La estructura en celosía escalona, celosía en escalera o *lattice-ladder* nos proporciona una estructura para la representación de sistemas que tienen ceros y polos. Consideremos un sistema general ARMA.

$$H(z) = \frac{\sum_{k=0}^M c_M(k) \cdot z^{-k}}{1 + \sum_{k=1}^N a_N(k) \cdot z^{-k}} = \frac{C_M(z)}{A_N(z)} \quad \text{con } M < N$$

Si utilizamos una variable intermedia

$$H(z) = \frac{Y(z)}{W(z)} = \frac{C_M(z)}{A_N(z)} \quad \rightarrow \quad \begin{aligned} \frac{1}{A_N(z)} &= \frac{W(z)}{X(z)} \\ C_M(z) &= \frac{Y(z)}{W(z)} \end{aligned}$$

Las ecuaciones en diferencias serán:

$$w(n) = -\sum_{k=1}^N a_N(k)w(n-k) + x(n) \quad \leftarrow \text{IIR todo polos}$$

$$y(n) = \sum_{k=0}^M c_M(k)w(n-k) \quad \leftarrow \text{Combinación lineal de las salidas del sistema todo polos}$$

En un filtro IIR todo polos hemos visto que  $g_m(n)$  es una combinación lineal de las salidas actual y anteriores, además

$$\frac{G_m(z)}{Y(z)} = B_m(z)$$

Cualquier otra combinación de  $g_m(n)$  seguirá siendo un sistema todo ceros. Consideremos

$$y(n) = \sum_{m=0}^M y_m g_m(n)$$

Como

$$\frac{Y(z)}{X(z)} = \frac{1}{X(z)} \sum_{m=0}^M v_m G_m(z) \stackrel{\substack{\uparrow \\ F_N(z)=X(z) \\ F_0(z)=G_0(z)=Y(z)}}}{=} \sum_{m=0}^M v_m \underbrace{\frac{G_m(z)}{G_0(z)}}_{B_m(z)} \underbrace{\frac{F_0(z)}{F_N(z)}}_{\frac{1}{A_N(z)}} = \frac{\sum_{m=0}^M v_m B_m(z)}{A_N(z)}$$

Comparando la expresión anterior con la general de un sistema ARMA obtenemos que

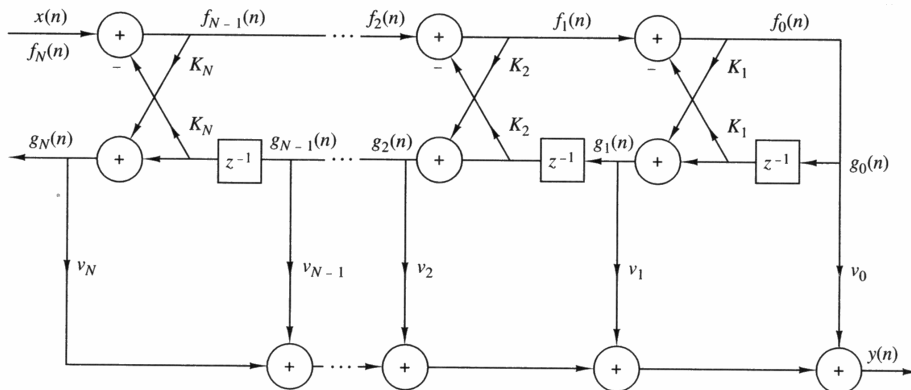
$$C_M(z) = \sum_{m=0}^M v_m B_m(z) \quad \text{que podemos expresar de forma recursiva como}$$

$$C_{m-1}(z) = C_m(z) - v_m B_m(z) \text{ siendo } v_m = c_m(m)$$

Resumen:

Para obtener la estructura en celosía ARMA, calcularemos los coeficientes de reflexión como en los casos anteriores, considerando un sistema todo polos, y posteriormente calcularemos los coeficientes  $v_m$  con la expresión  $C_{m-1}(z) = C_m(z) - v_m B_m(z)$ ,  $v_m = c_m(m)$ .

La estructura resultante es la siguiente:



Extraído de: Tratamiento Digital de Señales. J.G. Proakis

**Ejemplo:**

Determine la estructura en celosía para el sistema  $H(z) = \frac{1 - 0.8z^{-1} + 0.15z^{-2}}{1 + 0.1z^{-1} - 0.72z^{-2}}$

Solución:

En primer lugar consideramos el sistema todo polos, para el cálculo de los coeficientes de reflexión:

$$A_2(z) = 1 + 0.1z^{-1} - 0.72z^{-2} \quad \Rightarrow \alpha_2(2) = k_2 = -0.72$$

Utilizando la recursión:

$$A_1(z) = \frac{A_2(z) - k_2 B_2(z)}{1 - k_2^2} = \frac{1 + 0.1z^{-1} - 0.72z^{-2} + 0.72(-0.72 + 0.1z^{-1} + z^{-2})}{1 - 0.72^2} = 1 + 0.3571z^{-1}$$

$$\Rightarrow \alpha_1(1) = k_1 = 0.3571$$

Ahora calculamos  $v_m$ :

$$C_2(z) = 1 - 0.8z^{-1} + 0.15z^{-2} \quad \Rightarrow v_2 = C_2(2) = 0.15$$

Utilizando la recursión:

$$C_1(z) = C_2(z) - 0.15B_2(z) = 1 - 0.8z^{-1} + 0.15z^{-2} - 0.15(-0.72 + 0.1z^{-1} + z^{-2}) = 1.108 - 0.815z^{-1}$$

$$v_1 = C_1(1) = -0.815$$

$$C_0(z) = C_1(z) + 0.815B_1(z) = 1.108 - 0.815z^{-1} + 0.815(0.3571 + z^{-1}) = 1.3991$$

$$v_0 = C_0(0) = 1.3991$$

#### 5.5.4.- Funciones de Matlab relacionadas.

Descomposición en paralelo (Fracciones simples):  $[R,P,K] = \text{residuez}(B,A)$   
 $[B,A] = \text{residuez}(R,P,K)$

Descomposición en Cascada:  $[SOS,G] = \text{zp2sos}(Z,P,K)$   
 $[SOS,G] = \text{tf2sos}(B,A)$   
 $[Z,P,K] = \text{sos2zp}(SOS,G)$   
 $[B,A] = \text{sos2tf}(SOS,G)$

Estructuras en Celosía  $[K,V] = \text{tf2latc}(B,A)$   
 $[B,A] = \text{latc2tf}(K,V)$

#### Funciones de filtrado:

Forma Directa II traspuesta:  $Y = \text{filter}(B,A,x);$

Celosía escalonada  $[F,G] = \text{latcfilt}(K,V,X);$

Filtrado Bidireccional

$$Y = \text{filtfilt}(B, A, X);$$

**Funciones adicionales:**

Raíces de un polinomio:

$$\text{roots}(C)$$

Cálculo de un polinomio a partir de sus raíces:

$$\text{poly}(V)$$