Señales y sistemas I

Capítulo III: Señales y sistemas de tiempo discreto

Profesor: Néstor Becerra Yoma

Agradecimientos:
Profesor Manuel Duarte Mermoud (Transformada Z)
Enrique Guerrero Merino
Adio Stefoni Escudero

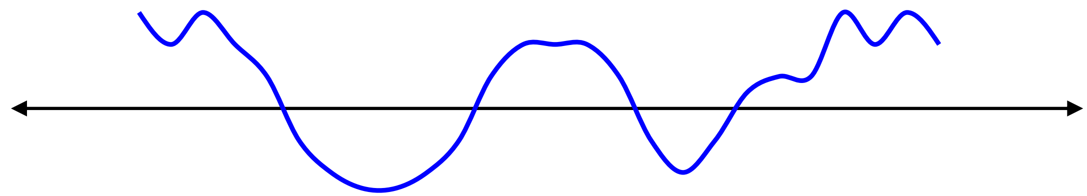
3.1 Características de las señales digitales

- Señales digitales, idea: señales que toman valores en tiempos discretos (mediante muestreo) y en amplitudes discretas, codificada en bits
- Para producirlas es necesario:
  - Tomar muestras de la señal
  - Cuantizar la amplitud de la señal
  - Codificar la señal

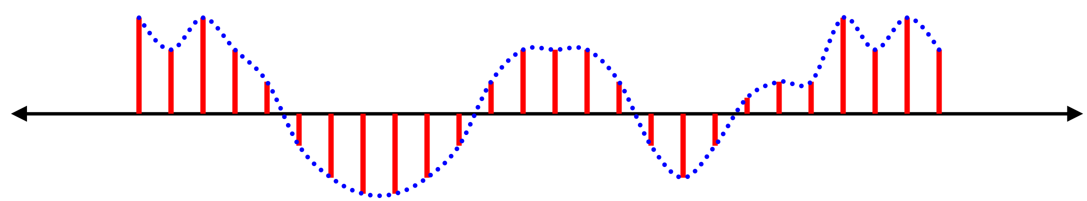
$$f(t) \rightarrow \boxed{\text{Muestrear}} \xrightarrow{f(nT)} \boxed{\text{Cuantizar}} \xrightarrow{f_q(nT)} \boxed{\text{Codificar}} \xrightarrow{\text{bits}}$$

3.1 Características de las señales digitales

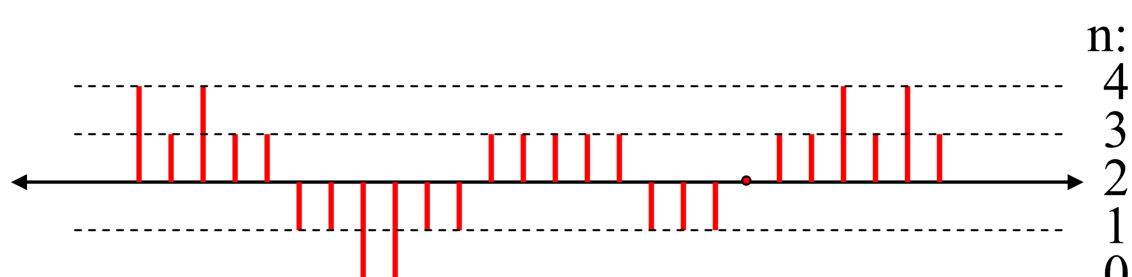
Señal continua



Señal de tiempo discreto



Señal cuantizada con 5 posibles valores



3.1 Características de las señales digitales

- Al transformar una señal de tiempo discreto en una señal digital se comete un error de aproximación => error de cuantización
- Las señales continuas que se transforman a tiempo discreto pueden ser reconstruidas sin error si se cumple la condición para el teorema del muestreo:

$$f_{MAXseñal} \leq \frac{f_{MUESTREO}}{2}$$

- Las señales continuas que se transforman en digitales no pueden ser reconstruidas sin error

3.1 Características de las señales digitales

- Ventajas de la transmisión digital:
  - Robustez al ruido (el ruido debe ser grande para que se cometa un error)
  - Procesamiento digital y multicanalización
  - Sencillas de medir y evaluar
  - Facilidad para medir rendimiento y tasas de error
  - Pueden ser almacenadas con facilidad

3.1 Características de las señales digitales

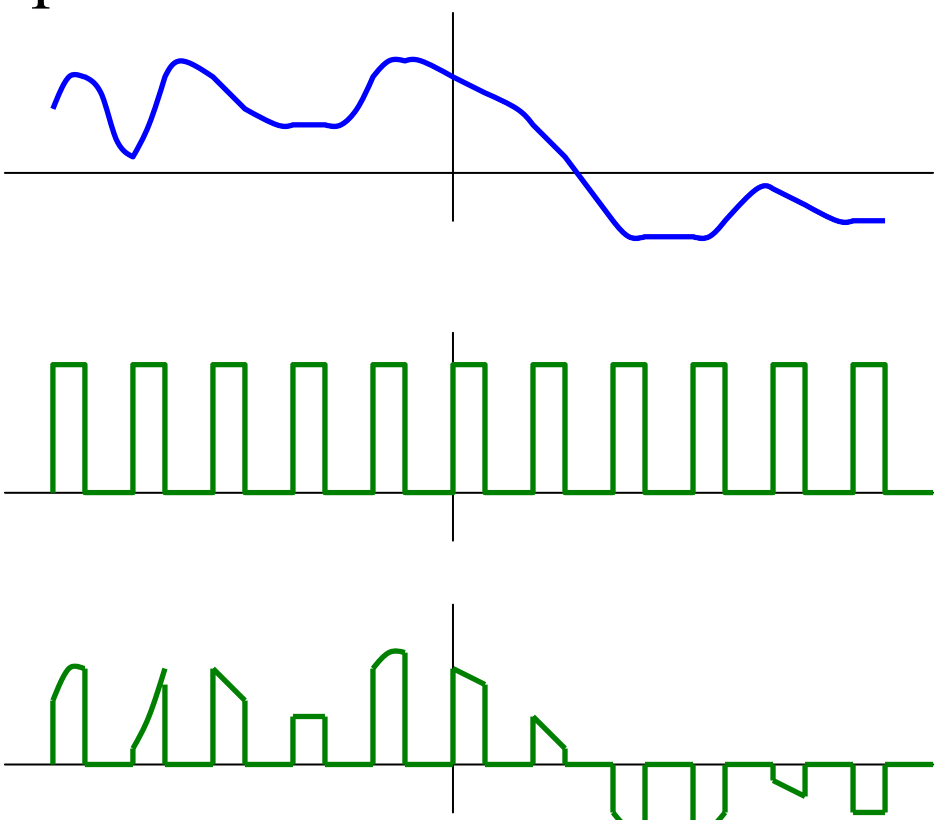
- Desventajas
  - Conversión digital/análoga, conversión análoga/digital y procesamiento digital conllevan retrasos
  - La conversión A/D introduce un ruido de cuantización
  - Se necesita sincronización
  - Incompatibilidad con sistemas analógicos

3.2 Teorema del Muestreo

- Gran desarrollo de la computación => digitalización de señales mediante muestreo, posterior reconstrucción de la señal
- Condición necesaria en el proceso para no perder información: teorema del muestreo:
  - Una señal de ancho de banda B [Hz] puede ser muestreada sin pérdida de información si se toman valores con una separación menor o igual a 1/(2B) segundos.

3.2 Teorema del Muestreo

- Primera aproximación: muestreo usando tren de pulsos



$f(t)$

$P_T(t)$: pulsos, periodo T

$f_S(t) = f(t)P_T(t)$

3.2 Teorema del Muestreo

$$f_S(t) = f(t) P_T(t)$$ $P_T(\cdot)$ es periódica

$$f_S(t) = f(t) \sum_{n=-\infty}^{\infty} P_n e^{jn\omega_0 t}$$

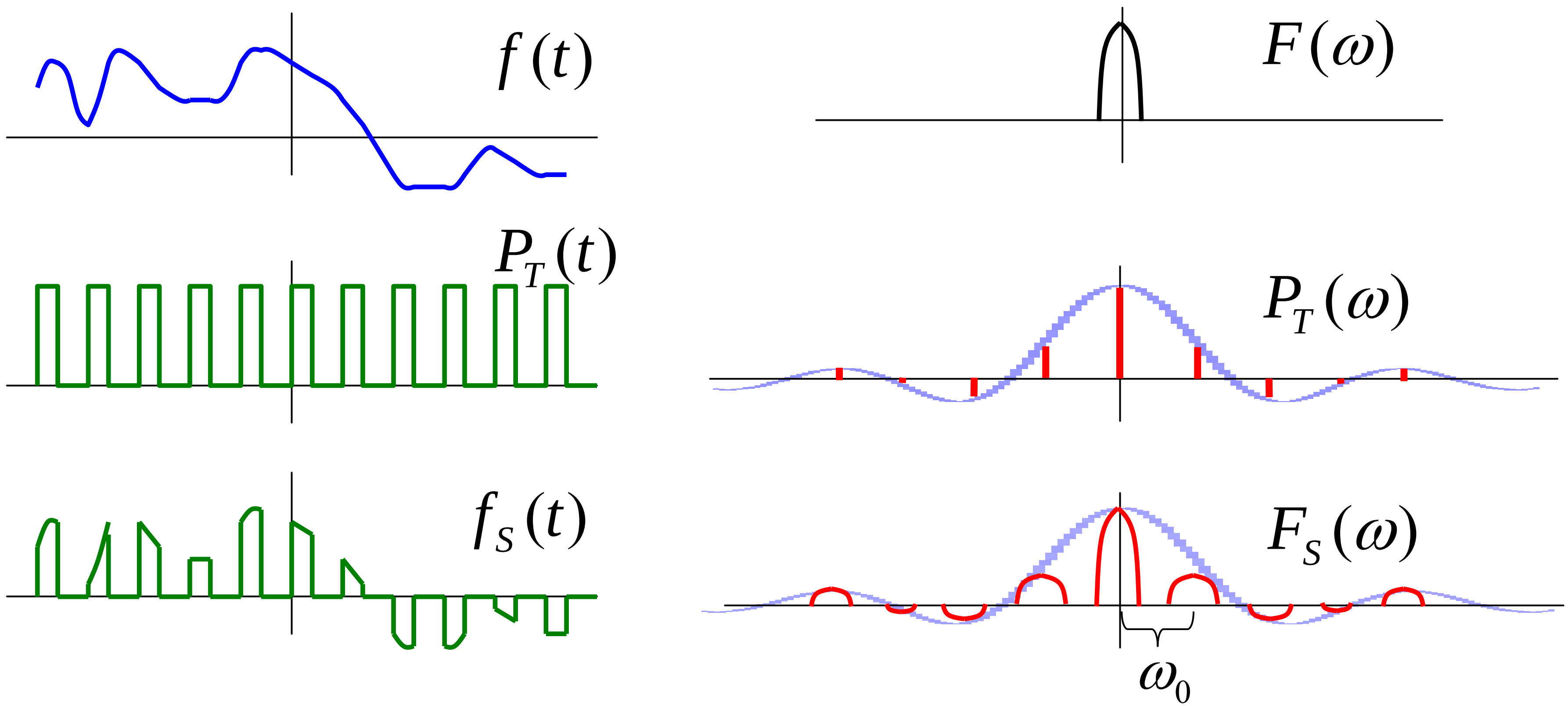
$$\Im\{f_S(t)\} = \Im\left\{ f(t) \sum_{n=-\infty}^{\infty} P_n e^{jn\omega_0 t} \right\}$$

$$\Im\{f_S(t)\} = \sum_{n=-\infty}^{\infty} P_n \Im\left\{ f(t) e^{jn\omega_0 t} \right\}$$

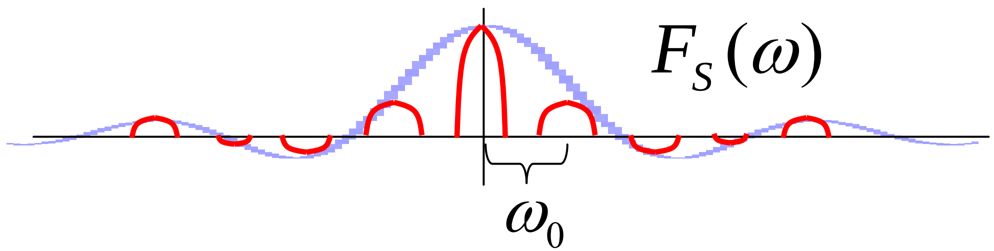
$$\Im\{f_S(t)\} = \sum_{n=-\infty}^{\infty} P_n F(\omega - n\omega_0)$$ El espectro $F(\omega)$ se repite

3.2 Teorema del Muestreo

$$\Im\{ f_S(t)\} = P_0 F(\omega) + \sum_{\substack{n=-\infty \\ n\neq 0}}^{\infty} P_n F(\omega - n\omega_0), \quad \omega_0 = \frac{2\pi}{T}$$



3.2 Teorema del Muestreo



$$\omega_0 = \frac{2\pi}{T}$$

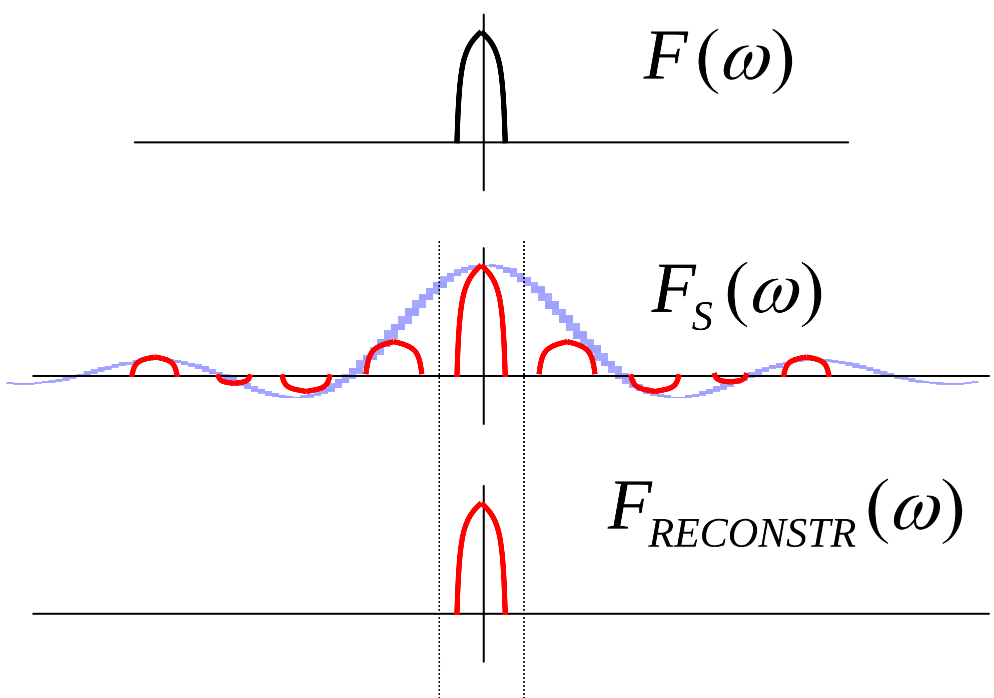
- Si el T de muestreo aumenta => ω_0 disminuye => se puede producir traslape de espectros para T muy grande. El traslape se alcanza cuando:

$$\omega_0 = \frac{2\pi}{T} = 2W \Rightarrow T = \frac{1}{2B} \quad \text{(frecuencia de Nyquist)}$$

- Para evitar traslape: $T < \frac{1}{2B}$

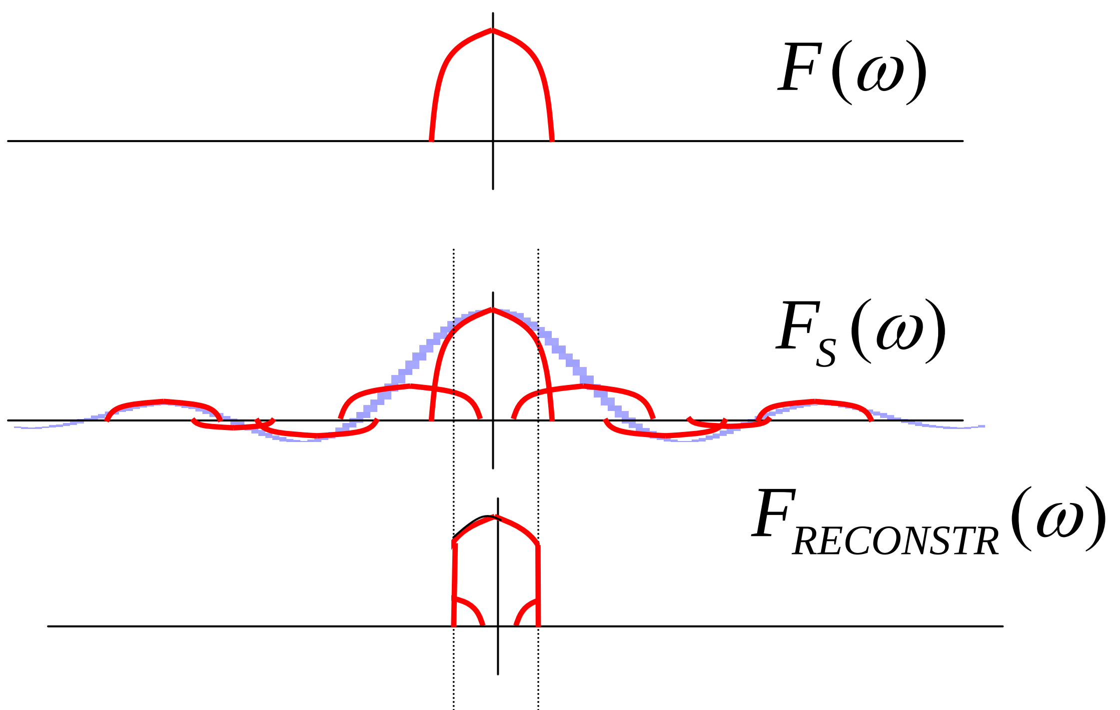
3.2 Teorema del Muestreo

- Reconstrucción de la señal => efecto práctico: pasa la banda entre $-\omega_0/2$ y $\omega_0/2$



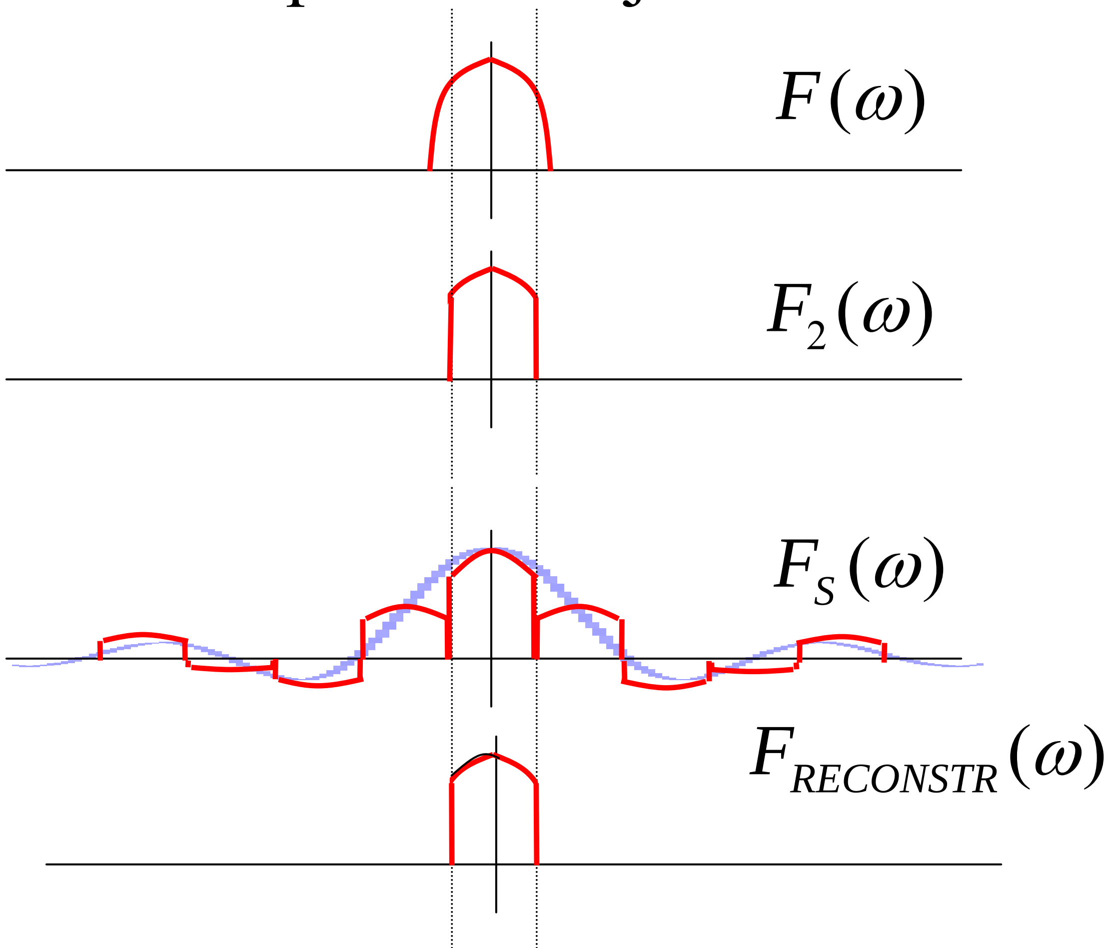
3.3 Efecto alias

- Traslape espectral: efecto alias, T muy grande
- Resultado: distorsión de la señal al reconstruirla.



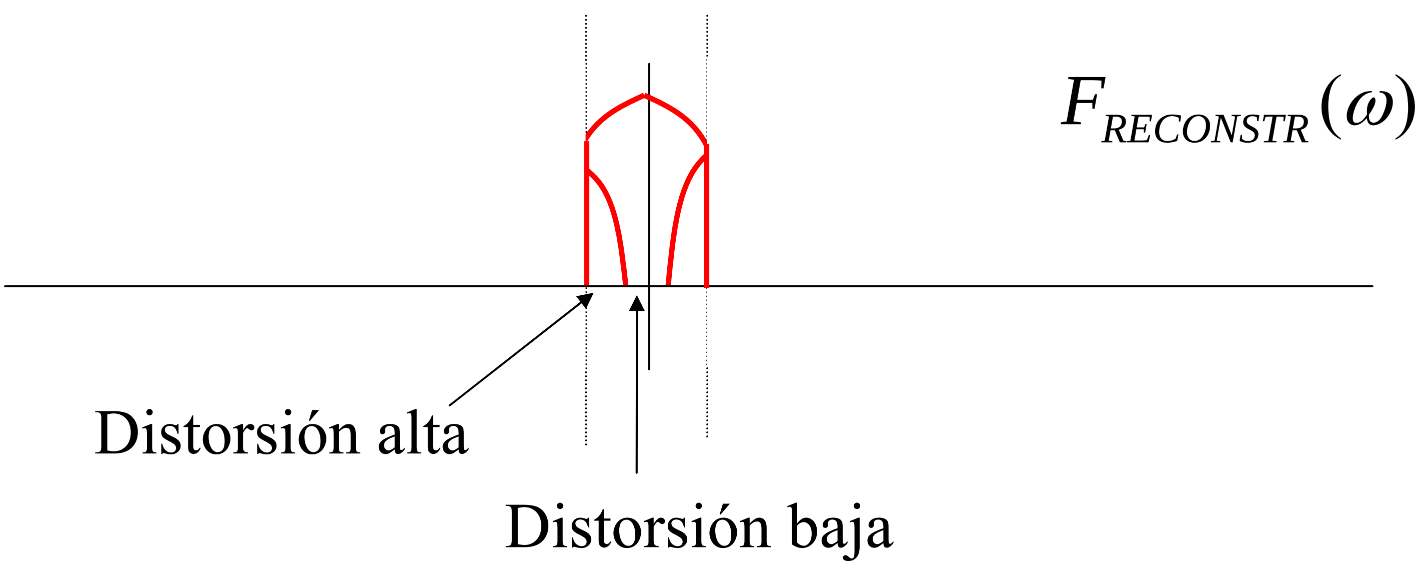
3.3 Efecto alias

- Solución: filtro prealias: baja la distorsión



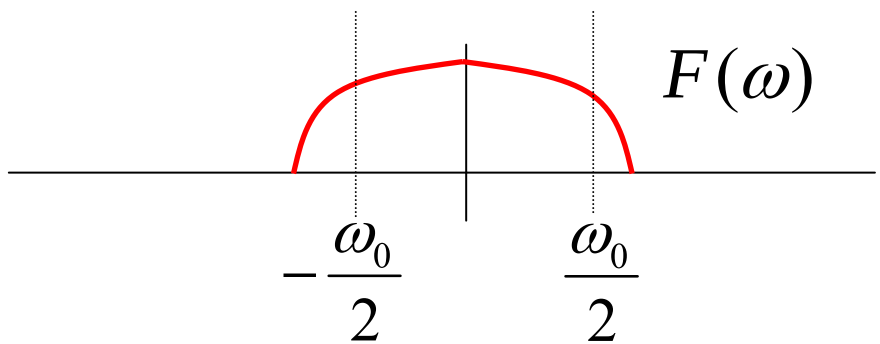
3.3 Efecto alias

- En general, el espectro de una señal se hace menor a medida que aumenta la frecuencia
- La principal distorsión se produce cerca de $\omega_0/2$



3.3 Efecto alias

- Estimación de la cantidad de efecto alias (comparativo):

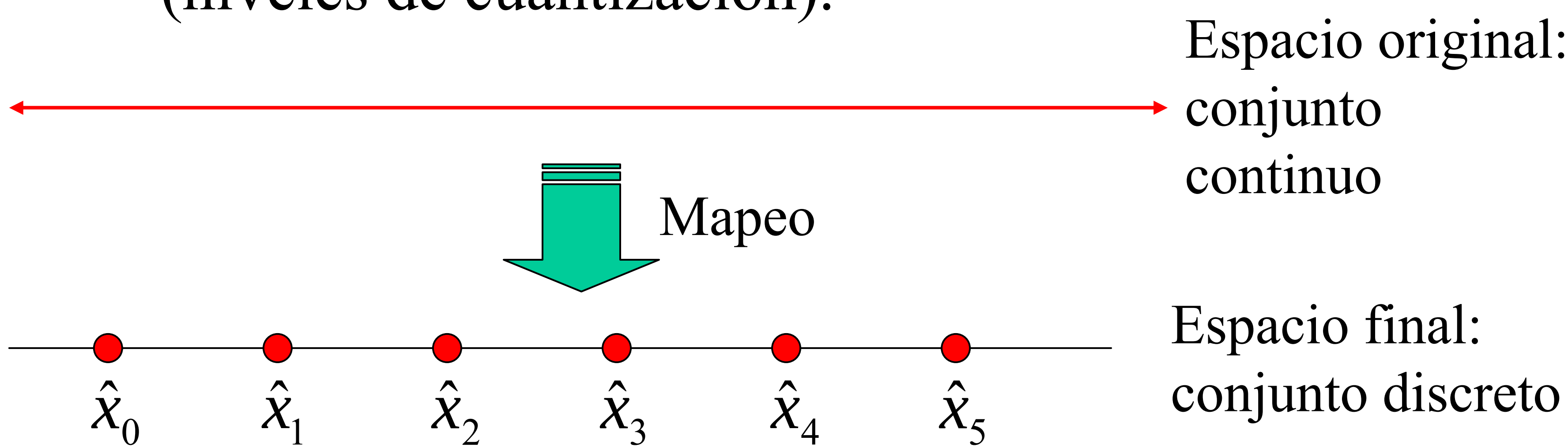


$$\%alias = \frac{\int_{\omega=\omega_0/2}^{\infty} \left(\frac{\omega_0-\omega}{\omega_0}\right)^2 \left|F(\omega)\right|^2 d\omega}{\int_{\omega=0}^{\omega_0/2} \left|F(\omega)\right|^2 d\omega}$$

- El término cuadrático penaliza el alias para frecuencias cercanas a $\omega_0/2$

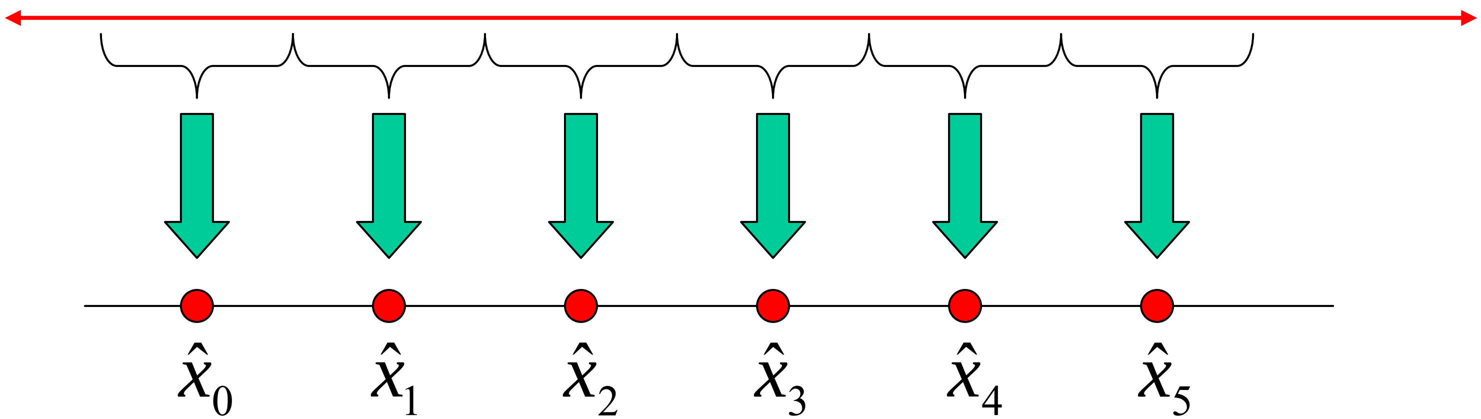
3.4 Cuantización

- Para digitalizar una señal, una vez que la señal ha sido muestreada, se debe aplicar el siguiente paso: la cuantización
- La cuantización es una transformación no lineal que consiste en mapear todas las amplitudes posibles de la señal a un conjunto finito de valores (niveles de cuantización).



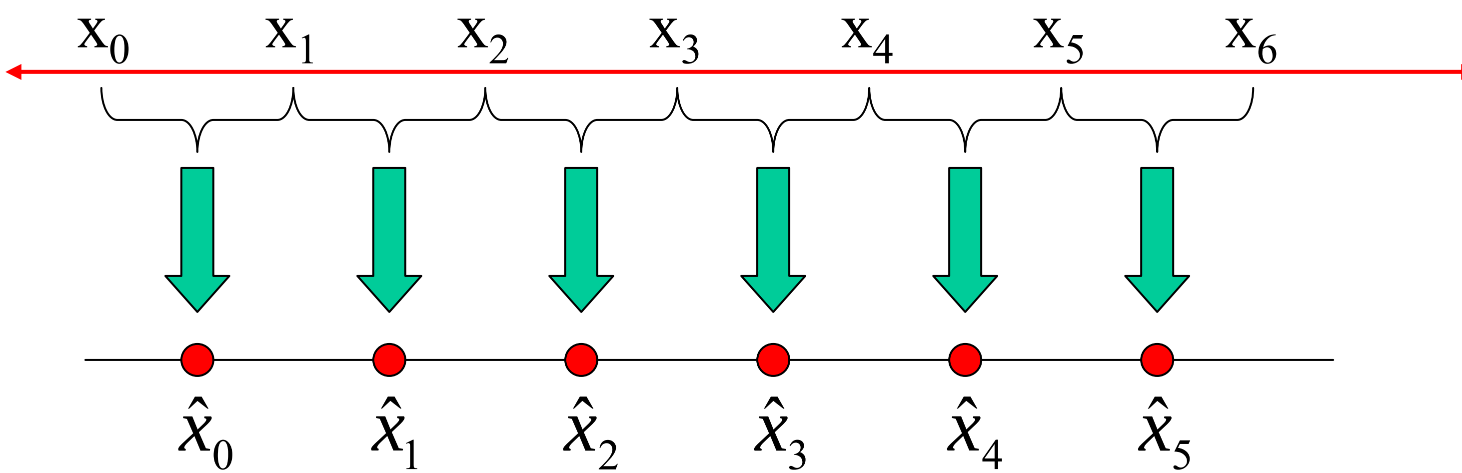
3.4 Cuantización

- Cada nivel de cuantización se transforma en un representante de un cierto sector (intervalo) del espacio original.



- Los límites entre las zonas indicadas se llaman niveles de decisión y se denominan x_0, x_1, etc.

3.4 Cuantización



- Luego, los intervalos están dados por:

$$I_k = \{x \mid x_k < x < x_{k+1}\}$$

- Cualquier elemento x del conjunto original que se encuentre dentro de I_k será aproximado por $\hat{x}_k$

3.4 Cuantización

- El cuantizador más común es el *cuantizador lineal o uniforme*. Se caracteriza porque todos los intervalos son iguales. Está definido por:

$$\hat{x}_{k+1} - \hat{x}_k = \Delta$$

$$\mathrm{x}_{k+1} - \mathrm{x}_k = \Delta$$

- Δ corresponde al ancho de los intervalos (paso del cuantizador).
- El error de cuantización está limitado por:

$$-\frac{\Delta}{2} < e_q \leq \frac{\Delta}{2}$$

3.4 Cuantización

- La relación anterior no es válida si el rango dinámico de la señal es mayor que el rango del cuantizador (en este caso, el error puede ser mayor)
- El codificador asigna un número binario único a cada nivel de cuantización. Luego, se requieren a lo menos tantos números binarios distintos como niveles de cuantización hayan
- Con b bits se pueden representar 2^b niveles.

3.4 Cuantización

- Luego, si se tienen L niveles de cuantización, se debe elegir el mínimo b tal que $2^b \geq L$ o, equivalentemente, $b \geq \log_2 L$
- Si se usan b bits, la resolución o paso del cuantizador está dado por:

$$\Delta = \frac{R}{2^b}$$

- R: rango del cuantizador

3.4 Cuantización

- Ejemplo: Se tiene la siguiente señal muestreada:
  - f(t)={ -2 , -1 , 0 , 1.2 , 1.8 , 2}
- Se tiene un cuantizador lineal de 4 bits con rango [-2, 2].
- Encontrar la salida del cuantizador
  - Solución: El número de niveles N está dado por:

$$N = 2^{n^{\circ} de\, bits} = 2^4 = 16$$

  - Luego, los niveles posibles son n=0,1,...,15.

3.4 Cuantización

- El rango del cuantizador es $f \in [-2,2]$
- Los niveles son n=0, 1, ..., 15
- El nivel 0 debe corresponder a f = –2, y el nivel 15 debe corresponder a f = 2

$n = a * f + b$ Ecuación "de la recta", se fija con 2 puntos

$$(n = 0,\ f = -2):\ 0 = a(-2) + b \Rightarrow b = 2a$$

$$(n = 15,\ f = 2): 15 = a(2) + b \Rightarrow a = \frac{15}{4} \Rightarrow b = \frac{15}{2}$$

$$n = \frac{15}{4} f + \frac{15}{2} \quad \Rightarrow \quad n = 3.75 f + 7.5$$

3.4 Cuantización

$$n = 3.75 f + 7.5$$ f(t)={ -2 , -1 , 0 , 1.2 , 1.8 , 2 }

- Luego, para cada f (t) se debe calcular su n

Valor de f	Valor de n	Aproximar a
$f = -2$	$n = 3.75(-2) + 7.5 = 0$	$n = 0$
$f = -1$	$n = 3.75(-1) + 7.5 = 3.75$	$n = 4$
$f = 0$	$n = 3.75(0) + 7.5 = 7.5$	$n = 8$
$f = 1.2$	$n = 3.75(1.2) + 7.5 = 12$	$n = 12$
$f = 1.8$	$n = 3.75(1.8) + 7.5 = 14.25$	$n = 14$
$f = 2$	$n = 3.75(2) + 7.5 = 15$	$n = 15$

3.5 Error de Cuantización

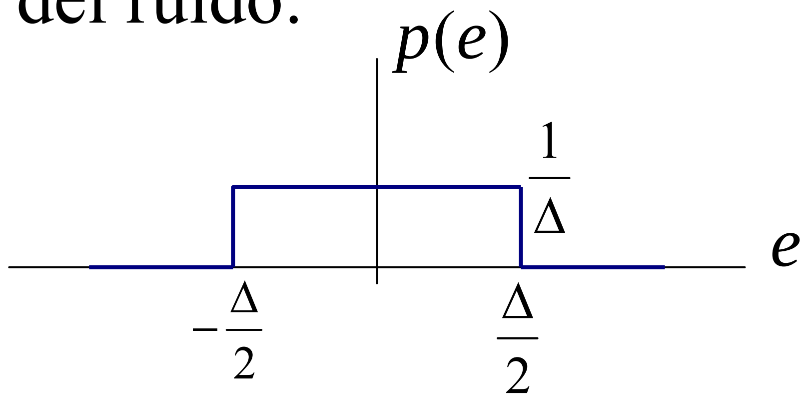
- Error de cuantización depende de la señal => es difícil obtener una expresión analítica salvo en casos sencillos:
- Se asumirá que el error de cuantización es aleatorio, se modela como un ruido sumado a la señal original.
- Δ corresponde al ancho de los intervalos (paso del cuantizador).
- Se usan n bits para codificar y el codificador es lineal.

3.5 Error de Cuantización

- Los supuestos son:
  - Error uniformemente distribuido: $-\frac{\Delta}{2} < e_q(n) < \frac{\Delta}{2}$
  - La secuencia de error $\{e_q(n)\}$ es ruido blanco estacionario, es decir, $e_q(n)$ y $e_q(m)$ no están correlacionados para n distinto de m
  - La secuencia de error $\{e_q(n)\}$ no está correlacionada con la secuencia de señal $\{f(n)\}$
  - La secuencia de la señal $\{f(n)\}$ es de media cero y estacionaria

3.5 Error de Cuantización

- Bajo los supuestos anteriores, es posible calcular la potencia del ruido:



$$P_n = \sigma_e^{\ 2} = \int_{-\Delta/2}^{\Delta/2} e^2 p(e) de = \int_{-\Delta/2}^{\Delta/2} e^2 \frac{1}{\Delta} de =$$

$$= \frac{1}{\Delta} \int_{-\Delta/2}^{\Delta/2} e^2 de = \frac{\Delta^2}{12}$$

3.5 Error de Cuantización

- Tenemos las ecuaciones:

$$\left(\frac{S}{N}\right)_{dB} = 10\log\left(\frac{P_f}{P_n}\right)[dB] \qquad \Delta = \frac{R}{2^b} \qquad P_n = \frac{\Delta^2}{12} = \frac{R^2}{12\times 2^{2b}}$$

- Al mezclarlas, se obtiene:

$$\left(\frac{S}{N}\right)_{dB} = 10\log P_f - 10\log P_n =$$

$$= 10\log P_f - 10\log R^2 + 10\log 12 + 20b\log 2$$

$$\left(\frac{S}{N}\right)_{dB} = 10\log P_f - 20\log R + 10.79 + 6.02b[dB]$$

3.5 Error de Cuantización

$$\left(\frac{S}{N}\right)_{dB} = \underbrace{10\log P_f}_{\text{Término que depende de la señal de entrada}} - \underbrace{20\log R}_{\text{Término que depende del rango del cuantizador}} + 10.79 + \underbrace{6.02b[dB]}_{\text{Término que depende del nº de bits usados}}$$

- Al agregar un bit extra al cuantizador se logra que la razón señal a ruido aumente en 6,02[dB]

$$+6{,}02[dB] = \times 2(potencia) = \times 4(amplitud\ RMS)$$

3.5 Error de Cuantización

- Casos particulares:
  - La señal de entrada tiene la máxima potencia posible:

$$P_f = \left(\frac{R}{2}\right)^2 = \frac{R^2}{4} \Rightarrow \left(\frac{S}{N}\right)_{dB} = 4.8 + 6.02b[dB]$$

Amplitud máxima posible

  - Señal de entrada con distribución gaussiana (media cero y varianza $\sigma^2 = P_f$) , rango del cuantizador $R = 3\sigma$.

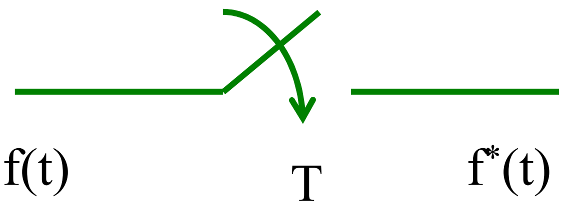
$$R = 3\sigma, P_f = \sigma^2 \Rightarrow \left(\frac{S}{N}\right)_{dB} = 1.25 + 6.02b[dB]$$

3.5 Error de Cuantización

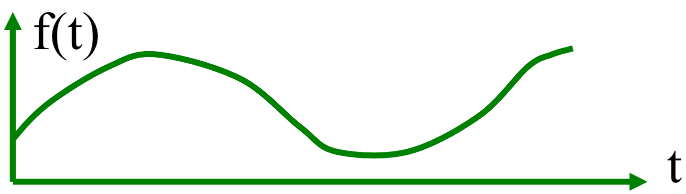
- Casos particulares (cont.):
  - La señal de entrada es uniformemente distribuida entre –R/2 y R/2:

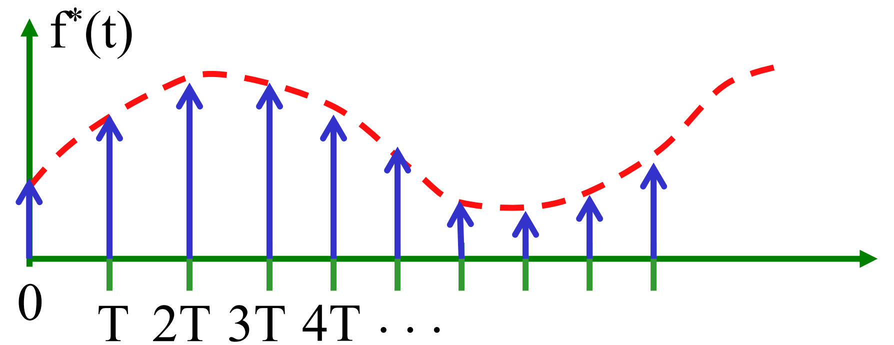
$$SNR = \left(\frac{S}{N}\right)_{dB} = 6.02b\,[\,dB\,]$$

3.6 Transformada Z



Muestrador ideal





$$f^*(t) = \sum_{k=0}^{\infty} f(kT)\delta(t - kT)$$

$$L\{f^*(t)\} = F^*(s) = f(0)L\{\delta(t)\} + f(T)L\{\delta(t-T)\} + \ldots$$

$$= \sum_{k=0}^{\infty} f(kT)\underbrace{L\{\delta(t-kT)\}}_{e^{-kTs}}$$

3.6 Transformada Z

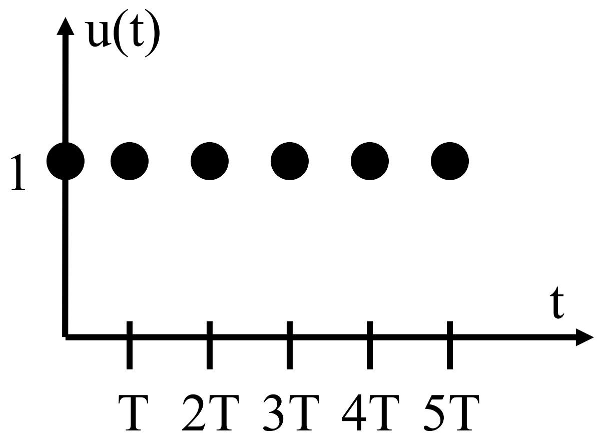
$$F^*(s) = \sum_{k=0}^{\infty} f(kT) e^{-kTs}$$

- Sea el cambio de variables: $z = e^{sT}$
- Se define la transformada Z de la función f*(t), de la siguiente manera:

$$Z\left\{ f^*(t) \right\} = F(z) = F^*(s)\Big|_{s=\frac{1}{T}Ln\,z} = \sum_{k=0}^{\infty} f(kT) z^{-k}$$

3.6 Transformada Z

- Ejemplos:
  - Escalón



$$u(t) = 1 \quad \forall t \geq 0$$

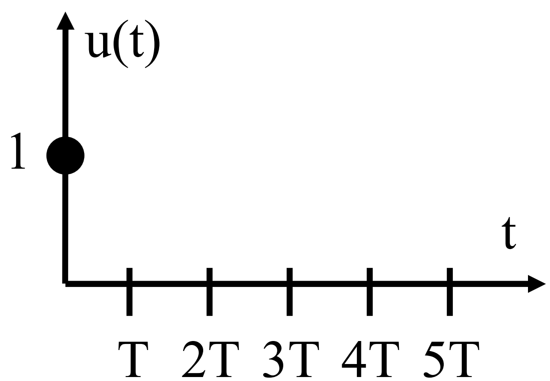
$$U(z) = \sum_{k=0}^{\infty} u(kT) z^{-k}$$

$$= \sum_{k=0}^{\infty} z^{-k} = \frac{1}{1 - z^{-1}}$$

$$U(z) = \frac{z}{z-1} \qquad \left| z^{-1} \right| < 1 \quad |z| > 1$$

3.6 Transformada Z

- Ejemplos:
  - Pulso unitario



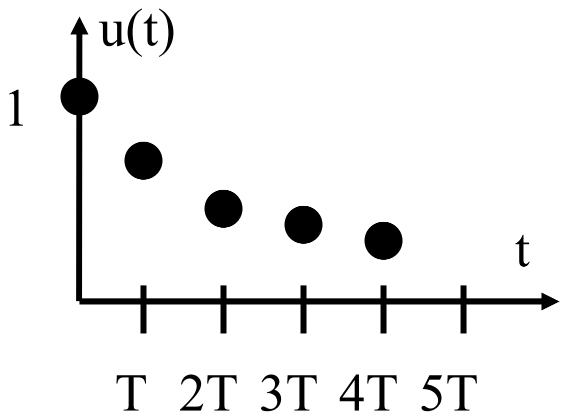
$$u(kT)=\begin{cases}1 & si \quad k=0\\ 0 & si \quad k\neq 0\end{cases}$$

$$U(z)=u(0)z^0=1$$

3.6 Transformada Z

- Ejemplos:
  - Exponencial

$$u(t) = e^{-kt}$$



$$U(z) = \sum_{0}^{\infty} e^{-kT} z^{-k}$$

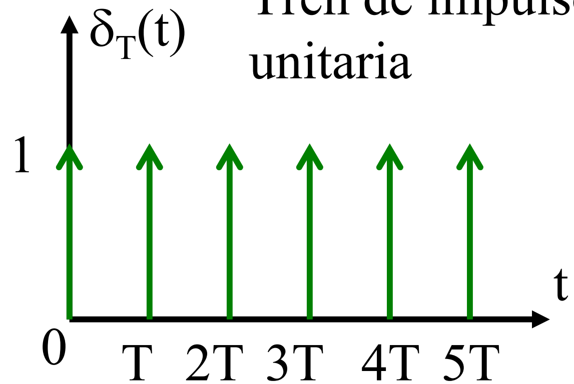
$$= \frac{1}{1 - e^{-T} z^{-1}}$$

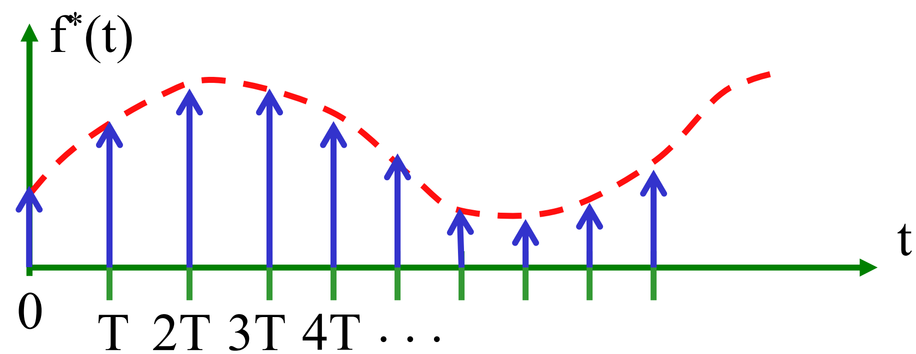
$$U(z) = \frac{z}{z - e^{-T}}$$

3.6 Transformada Z

- Convolución compleja:

Tren de impulsos de área unitaria





$$\Delta_T(s) = L\{\delta_T(t)\} = \sum_{k=0}^{\infty} e^{-kTs} = \frac{1}{1-e^{-Ts}}$$

$$f^*(t) = f(t)\,\delta_T(t)$$

3.6 Transformada Z

- Convolución compleja:

$$F^*(s) = L\{\, f(t)\delta_T(t)\,\} = \frac{1}{2\pi j}\oint_C F(\lambda)\Delta_T(s-\lambda)\,d\lambda$$

$$F(z) = F^*(s)\Big|_{s=\frac{1}{T}Ln\,z}$$

S_k: polos de F(λ) dentro de C

$$F^*(s) = \frac{1}{2\pi j}\oint_C \frac{F(\lambda)}{1-e^{-T(s-\lambda)}}\,d\lambda$$

$$\therefore\ F(z) = \sum_k res\left\{\frac{F(\lambda)}{1-z^{-1}e^{T\lambda}}\right\}\Bigg|_{\lambda=s_k} \qquad (*)$$

3.6 Transformada Z

- Convolución compleja (ejemplo):

$$f(t) = \operatorname{sen}\omega t,\ t = kT,\ k \in \mathrm{N} \cup \{0\}$$

$$F(s) = \frac{\omega}{s^2 + \omega^2};\ \text{polos en } s = \pm j\omega$$

$$F(s) = \frac{j/2}{s + j\omega} - \frac{j/2}{s - j\omega}$$

Obteniendo residuos de la expresión (*)

$$F(z) = \frac{j}{2}\left[\frac{1}{1 - e^{-j\omega T} z^{-1}} - \frac{1}{1 - e^{j\omega T} z^{-1}}\right]$$

$$\therefore \quad F(z) = \frac{z \operatorname{sen}\omega T}{z^2 - 2z\cos\omega T + 1}$$

3.6 Transformada Z

- Propiedades de la transformada Z:

Dominio Secuencia	Dominio Z
$f(k)$	$F(z)$
$af(k)$	$aF(z)$
$f_1(k)+f_2(k)$	$F_1(z)+F_2(z)$
$f(k+1)$	$zF(z)-zf(0)$
$f(k+2)$	$z^2F(z)-z^2f(0)-zf(1)$
$f(k+N)$	$z^NF(z)-z^Nf(0)-\cdots-zf(N-1)$
$a^kf(k)$	$F(z/a)$
$f(0)$	$\lim_{z\to\infty} F(z) \quad (*)$
$f(\infty)$	$\lim_{z\to 1} (z-1)F(z) \quad (**)$
$\sum_{n=0}^{k} f(n)g(k-n)$	$F(z)G(z)$
$f(k-N)$	$z^{-N}F(z)$

* Si el límite existe

** Si $(1-z^{-1})F(z)$ tiene todos su polos en la región $|z|<1$

3.6 Transformada Z

- Tabla de transformadas Z (I):

T	S	Z
$u(t)$	$\frac{1}{s}$	$\frac{z}{z-1}$
t	$\frac{1}{s^2}$	$\frac{Tz}{(z-1)^2}$
$t^2/2$	$\frac{1}{s^3}$	$\frac{T^2 z(z+1)}{2(z-1)^3}$
e^{-at}	$\frac{1}{s+a}$	$\frac{z}{z-e^{-aT}}$
te^{-at}	$\frac{1}{(s+a)^2}$	$\frac{Tze^{-aT}}{(z-e^{-aT})^2}$
$e^{-at}\operatorname{sen}\omega t$	$\frac{\omega}{(s+a)^2+\omega^2}$	$\frac{ze^{-aT}\operatorname{sen}\omega T}{z^2-2ze^{-aT}\cos\omega T+e^{-2aT}}$
$e^{-at}\cos\omega t$	$\frac{s+a}{(s+a)^2+\omega^2}$	$\frac{z^2-ze^{-aT}\cos\omega T}{z^2-2ze^{-aT}\cos\omega T+e^{-2aT}}$

3.6 Transformada Z

- Tabla de transformadas Z (II):

T	S	Z *
$e^{-at}\cos(\omega t-\theta)$	$\dfrac{\cos\theta\,(s+a)+\omega\,\mathrm{sen}\,\theta}{(s+a)^2+\omega^2}$	$\dfrac{z\cos\theta\,(z-\alpha)-z\beta\,\mathrm{sen}\,\theta}{(z-\alpha)^2+\beta^2}$

T	Z
a^{k}	$\dfrac{z}{z-a}$
$k\,a^{k-1}$	$\dfrac{z}{(z-a)^2}$
$\frac{1}{2}k(k-1)a^{k-2}$	$\dfrac{z}{(z-a)^3}$
$\dfrac{1}{(M-1)!}\left[\prod_{i=0}^{M-2}(k-i)\right]a^{k-M+1}$	$\dfrac{z}{(z-a)^M}$

* Donde:

$$\alpha = e^{-aT}$$

$$\beta = e^{-aT}\,\mathrm{sen}\,\omega T$$

3.7 Transformada Z inversa

$$F(z) = \sum_{k=0}^{\infty} f(kT) z^{-k}$$

- Descomposición en fracciones parciales sobre z^{-1}

$$F(z) = \frac{A}{1-\alpha z^{-1}} + \frac{B}{1-\beta z^{-1}} + \cdots$$

$$A = F(z)(1-\alpha z^{-1})\,|_{z=\alpha}$$

$$B = F(z)(1-\beta z^{-1})\,|_{z=\beta}$$

Si $\alpha = e^{-aT}$ y $\beta = e^{-bT}$,...

$$f(kT) = Ae^{-akT} + Be^{-bkT} + \cdots$$

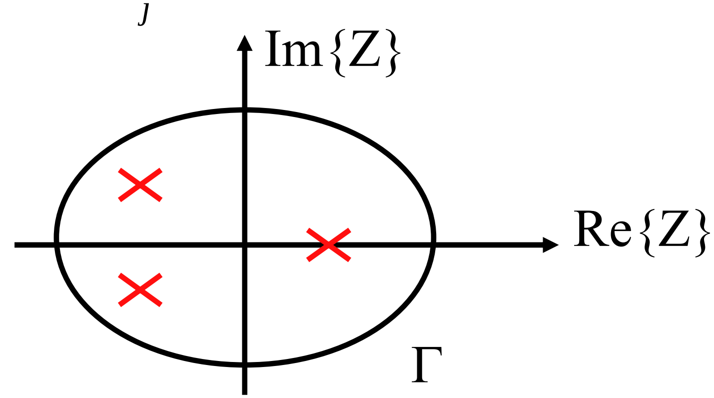
3.7 Transformada Z inversa

$$F(z) = \sum_{k=0}^{\infty} f(kT)\, z^{-k}$$

- O bien, serie de Laurent en torno a z=0:

$$E(z) = F(z)\, z^{n-1} \Rightarrow f(nT) = \frac{1}{2\pi j} \oint_{\Gamma} E(z)\,dz = \frac{1}{2\pi j} \oint_{\Gamma} z^{n-1} F(z)\,dz$$

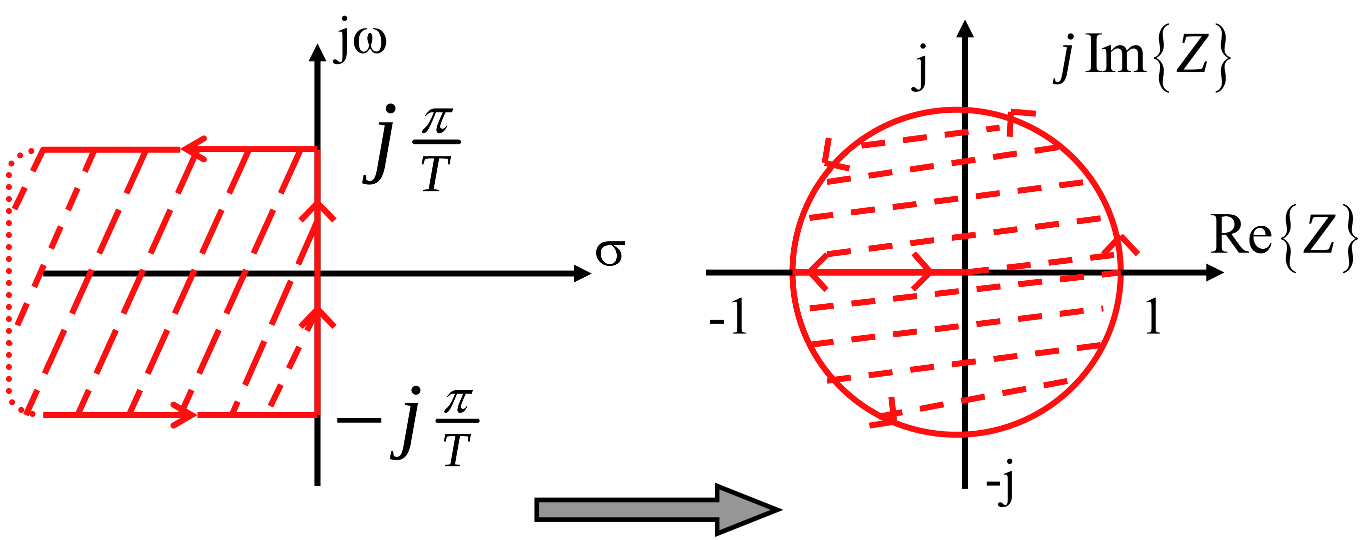
$$\therefore\ f(nT) = \sum_{j} res(z^{n-1} F(z), p_j)$$



3.8 Espacio-Z

- Cambio en la topología del espacio:

Plano S Plano Z



$$z = e^{sT}$$

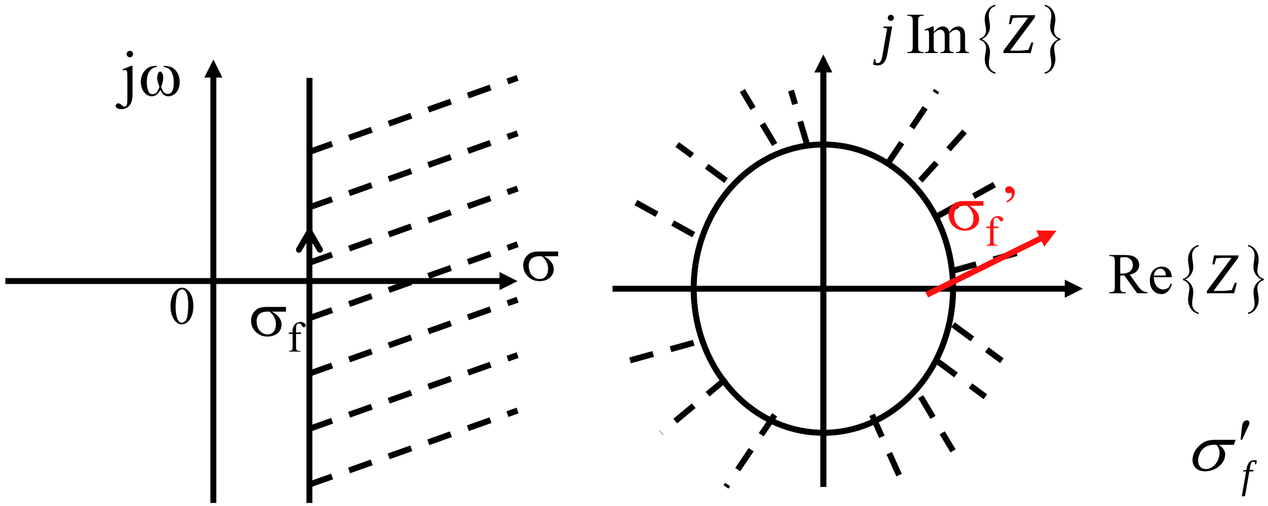
$$|z| = e^{\sigma T}$$

$$\angle z = \omega T$$

3.8 Espacio-Z

- Cambio en la topología del espacio:

Convergencia de la Transformada Z



$$\sigma'_f = e^{\sigma_f T}$$

Región de convergencia de la Trasformada de Laplace.

Región de convergencia de la Trasformada Z

3.9 Laplace v/s Z

Primer orden continuo

$$\dot{x}(t)+ax(t)=bu(t) \qquad / L$$

$$x(0)=x_0$$

$$X(s)=\frac{1}{s+a}x_0+\frac{1}{s+a}bU(s)$$

$$x(t)=e^{-at}x_0+\int_0^t e^{-a(t-\tau)}bu(\tau)d\tau$$

$$H(s)=\frac{b}{s+a};\ h(t)=be^{-at};$$

Primer orden discreto

$$x(k+1)+ax(k)=bu(k) \qquad / Z$$

$$x(0)=x_0$$

$$X(z)=\frac{z}{z+a}x_0+\frac{1}{z+a}bU(z)$$

$$x(k)=(-a)^k x_0+\sum_{n=0}^{k-1}(-a)^n bu(k-n-1)$$

$$H(z)=\frac{b}{z+a};\ h(k)=b(\text{-}a)^{k-1}$$

3.10 Función de transferencia discreta

- Para representar funciones de transferencia de sistemas discretos se puede usar la transformada Z
- La función de transferencia H(z) es la transformada Z de la respuesta al impulso.

$$H(z)=\sum_{k=0}^{\infty} h(k) z^{-k}$$

- También se puede calcular H(z) aplicando la transformada Z a la ecuación de diferencias.
- Recuerdo: Z{x(t+1)} = z*(Z{x(t)} - x(0))

3. 10 Función de transferencia discreta

- Filtros digitales: Se tienen dos tipos: FIR e IIR

FIR (Finite Impulse Response):

•La salida es función únicamente de la entrada .

•El polinomio que describe este filtro:

$$Y(n) = a_0 x(n) + a_1 x(n-1) + a_2 x(n-2) + \dots\dots\dots + a_k x(n-k)$$

•Un filtro FIR también se puede definir por:

$$A = \begin{bmatrix} a_0 & a_1 & a_2 & & & a_k \end{bmatrix}$$

¿Porqué Respuesta Impulsiva Finita?

3.10 Función de transferencia discreta

IIR (Infinite Impulse Response):

•La salida en n depende de la entrada y de la salida hasta n-1.

•El polinomio que describe este filtro:

$$Y(n) = a_0 x(n) + a_1 x(n-1) + a_2 x(n-2) + \ldots\ldots\ldots + a_k x(n-k)$$
$$+ b_1 y(n-1) + b_2 y(n-2) + \ldots\ldots\ldots + b_L y(n-L)$$

•Un filtro IIR también se puede definir por:

$A = [a_0 \quad a_1 \quad a_2 \qquad\qquad a_k]$ y $B = [b_1 \quad b_2 \quad b_3 \qquad\qquad b_L]$

¿Porqué Respuesta Impulsiva Infinita?

¿Estabilidad?

3.10 Función de transferencia discreta

La transformada Z se emplea para analizar sistemas discretos. (De forma similar como opera Laplace para sistemas continuos).

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

Mediante las propiedades de la transformada Z se puede concluir:

$$x(n) \leftrightarrow X(z)$$

$$x(n-k) \leftrightarrow X(z) z^{-k}$$

3.10 Función de transferencia discreta

Ejemplo:

Considere el filtro FIR

$$Z\{y(n)\} = Z\{a_0 x(n) + a_1 x(n-1) + a_2 x(n-2) + \ldots.. + a_k x(n-k)\}$$

$$Z\{y(n)\} = Z\{a_0 x(n)\} + Z\{a_1 x(n-1)\} + Z\{a_2 x(n-2)\} + \ldots.. + Z\{a_k x(n-k)\}$$

$$Z\{y(n)\} = a_0 Z\{x(n)\} + a_1 Z\{x(n-1)\} + a_2 Z\{x(n-2)\} + \ldots.. + a_k Z\{x(n-k)\}$$

$$Y(z) = a_0 X(z) + a_1 X(z) z^{-1} + a_2 X(z) z^{-2} + \ldots.. + a_k X(z) z^{-k}$$

$$\frac{Y(z)}{X(z)} = a_0 + a_1 z^{-1} + a_2 z^{-2} + \ldots.. + a_k z^{-k} = H(z)$$

3.10 Función de transferencia discreta

Ejercicio:

¿Cual es la función de transferencia H(z) de un filtro IIR?

3.10 Función de transferencia discreta

- La respuesta en frecuencia de un sistema discreto se puede estimar reemplazando Z por:

$$z = e^{+jw} = cos(w) + jsen(w)$$

$$H(z) = 1 + az^{-1}$$

Ejemplo: Sea

$$H(e^{+jw}) = 1 + ae^{-jw} = 1 + a\left[cos(w) - jsen(w)\right]$$

$$H(e^{+jw}) = 1 + a \cdot cos(w) - j \cdot a \cdot sen(w)$$

$$\left|H(e^{+jw})\right| = \sqrt{1 + 2a\,cos(w) + a^2}$$

$$\angle H(e^{+jw}) = arctan\left(\frac{-asen(w)}{1 + a\,cos(w)}\right)$$

3.10 Función de transferencia discreta

Ejercicio:

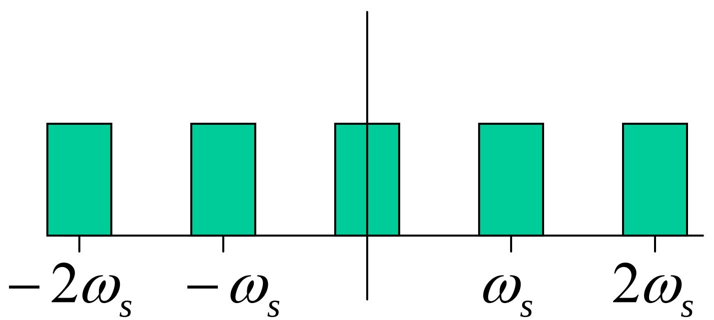
Grafique

$$\left|H(e^{+jw})\right| \quad \text{en} \quad H(z) = 1 + az^{-1}$$

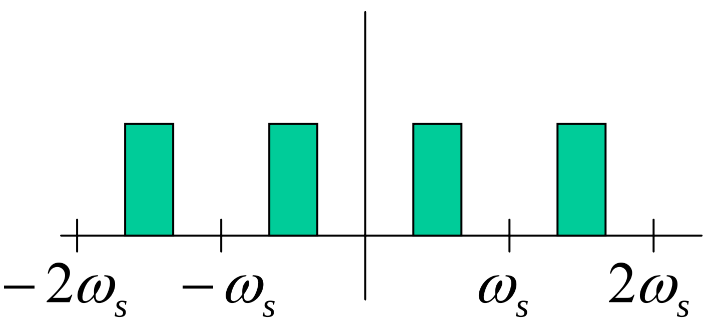
Para a=0.9 y a= -0.9. Determine para cada caso el tipo de filtro: pasa-bajas, pasa-altas, pasa-banda, o rechaza-banda.

3.10 Función de transferencia discreta

- La función de transferencia permite calcular la respuesta en frecuencia del filtro discreto al hacer $z = e^{j\omega T_s}$, donde T_s es el período de muestreo
- Se debe notar que la respuesta en frecuencia $H(e^{j\omega T_s})$ es periódica, ya que $e^{j\omega T_s}$ también lo es.
- Para ver si un filtro es pasabajos o pasaaltos, conviene fijarse sólo en $\omega \in [-\omega_s/2\,,\ \omega_s/2]$



Filtro pasa bajos

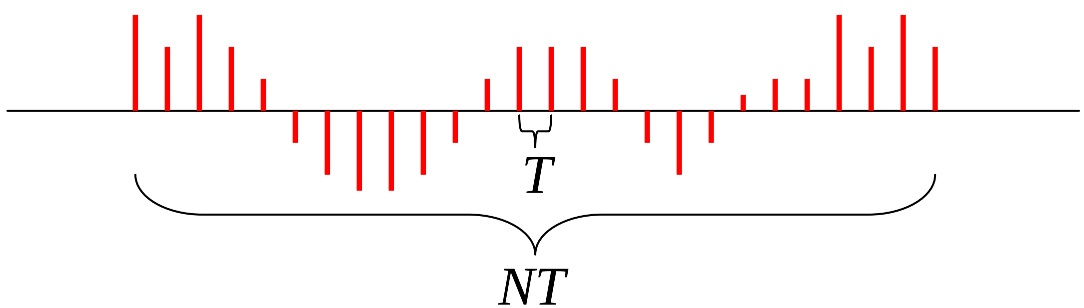


Filtro pasa altos

3.11 Transformada de Fourier discreta

- Se tiene N muestras uniformemente distribuidas

$$f(kT) = f(0), f(T), f(2T), \ldots, f\big((N-1)T\big)$$



- La transformada de Fourier discreta (DFT) se define como:

$$F_d(n\Omega) = \sum_{k=0}^{N-1} f(kT)e^{-j\,n\Omega\,kT} \qquad \Omega = \frac{2\pi}{NT}$$

3.11 Transformada de Fourier discreta

$$F_d(n\Omega) = \sum_{k=0}^{N-1} f(kT) e^{-j\,n\Omega\,kT} \qquad \Omega = \frac{2\pi}{NT}$$

$$\Rightarrow F_d(n\Omega) = \sum_{k=0}^{N-1} f(kT) e^{-j\,n\frac{2\pi}{N}k}$$

- Ω y T no aparecen de forma explícita en la DFT
- Se puede obtener a partir de la transformada de Fourier, haciendo aproximaciones

3.11 Transformada de Fourier discreta

- Sea:

$$\tilde{f}(t) = \begin{cases} f(t) & 0 \leq t < NT \\ 0 & otro\ caso \end{cases}$$

- Su transformada es:

$$\tilde{F}(\omega) = \int_{t=0}^{NT} f(t) e^{-j\omega t} dt \qquad /\omega \to n\Omega,\ t \to nT$$

$$\tilde{F}(n\Omega) \approx \sum_{k=0}^{N-1} f(kT) e^{-jn\Omega kT} T$$

- Luego:

$$TF_D(n\Omega) = \tilde{F}(\omega)\,|_{\omega = n\Omega}$$

3.11 Transformada de Fourier discreta

- El espectro obtenido es periódico, de periodo NΩ:

$$F_D(n\Omega + N\Omega) = \sum_{k=0}^{N-1} f(kT)e^{-j(n\Omega+N\Omega)kT}$$

$$= \sum_{k=0}^{N-1} f(kT)e^{-jn\Omega kT}e^{-jN\Omega kT} \qquad / \Omega = \frac{2\pi}{NT}$$

$$= \sum_{k=0}^{N-1} f(kT)e^{-jn\Omega kT}e^{-j2\pi k} = F_D(n\Omega)$$

3.11 Transformada de Fourier discreta

- La exactitud en el cálculo de la transformada también es afectada por el efecto alias.
- La transformada es periódica => la frecuencia más alta que se puede determinar corresponde a $n=N/2$, es decir, $(N/2)\Omega=1/(2T)$ => de acuerdo al teorema del muestreo

3.11 Transformada de Fourier discreta

- También hay una transformada inversa de Fourier discreta:

$$f(kT) = \frac{1}{N}\sum_{n=0}^{N-1} F_D(n\Omega)e^{jn\Omega k\mathrm{T}}$$

- La transformada inversa es exacta respecto a la transformada directa, a menos que se produzca efecto alias.
- La transformada inversa es periódica, periodo NT
- Propiedades semejantes a la transformada de Fourier continua.

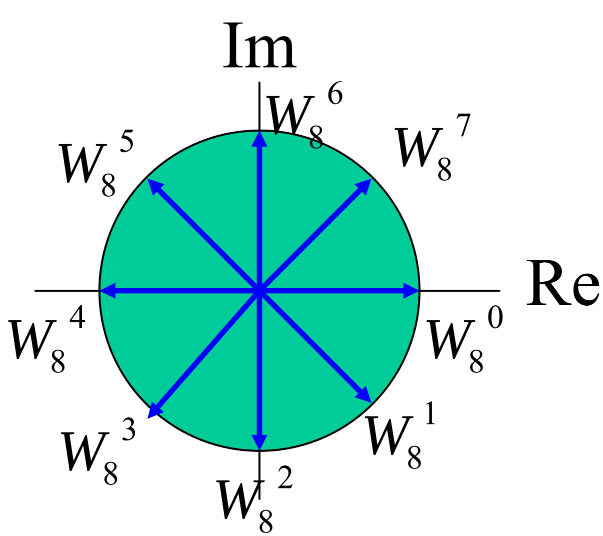
3.12 Transformada de Fourier rápida

- El cálculo de la DFT requiere N^2 multiplicaciones.
- Tiempo de cálculo excesivo para N grande.
- Algoritmo FFT (fast Fourier transform): permite calcular la DFT de forma rápida: $N\log_2 N$ multiplicaciones.
- Dos formas de visualizarla: a partir de sumas por bits o a partir de paridad/imparidad.

3.12 Transformada de Fourier rápida

- La notación W_N para la exponencial de la DFT:
  - Dado N, se define W_N como el círculo unitario dividido en N partes, con ángulos negativos

$$W_N = e^{-j\Omega T} = e^{-j\frac{2\pi}{N}} \quad \Rightarrow \quad W_N = 1\angle\frac{-360^{\circ}}{N}$$



$$W_N^{N\,x} = e^{-j2\pi\frac{N}{N}x} = 1$$

$$W_N^{2} = e^{-j2\pi\frac{2}{N}} = e^{-j\frac{2\pi}{N/2}} = W_{N/2}$$

$$W_N^{x}{}^{*} = W_N^{-x} = W_N^{(N-x)}$$

3.12 Transformada de Fourier rápida

- Forma 1 (ejemplo para 4 puntos):

$$F_d(n\Omega) = \sum_{k=0}^{N-1} f(kT) e^{-jn\Omega kT}$$

$$F_d(n\Omega) = \sum_{k=0}^{3} f(kT) W_4^{nk}, \quad \boxed{W_N = e^{-j\Omega T} = e^{\frac{2\pi}{N}}}$$

- Se pueden escribir k y n como números binarios:

$$k = (k_1, k_0) = \{00, 01, 10, 11\}, \ k = 2k_1 + k_0$$

$$n = (n_1, n_0) = \{00, 01, 10, 11\}, \ n = 2n_1 + n_0$$

3.12 Transformada de Fourier rápida

- Luego:

$$F_D(n_1, n_0) = \sum_{k_0=0}^{1} \sum_{k_1=0}^{1} f(k_1, k_0) W_4^{(2n_1+n_0)(2k_1+k_0)}$$

$$W_4^{(2n_1+n_0)(2k_1+k_0)} = W_4^{2n_0k_1} W_4^{(2n_1+n_0)k_0}$$

$$W_4^{4n_1k_1} = 1$$

$$\underbrace{F_D(n_1, n_0) = \sum_{k_0=0}^{1} \left(\underbrace{\sum_{k_1=0}^{1} f(k_1, k_0) W_4^{2n_0k_1}}_{f_1(n_0, k_0)} \right) W_4^{(2n_1+n_0)k_0}}_{f_2(n_0, n_1)}$$

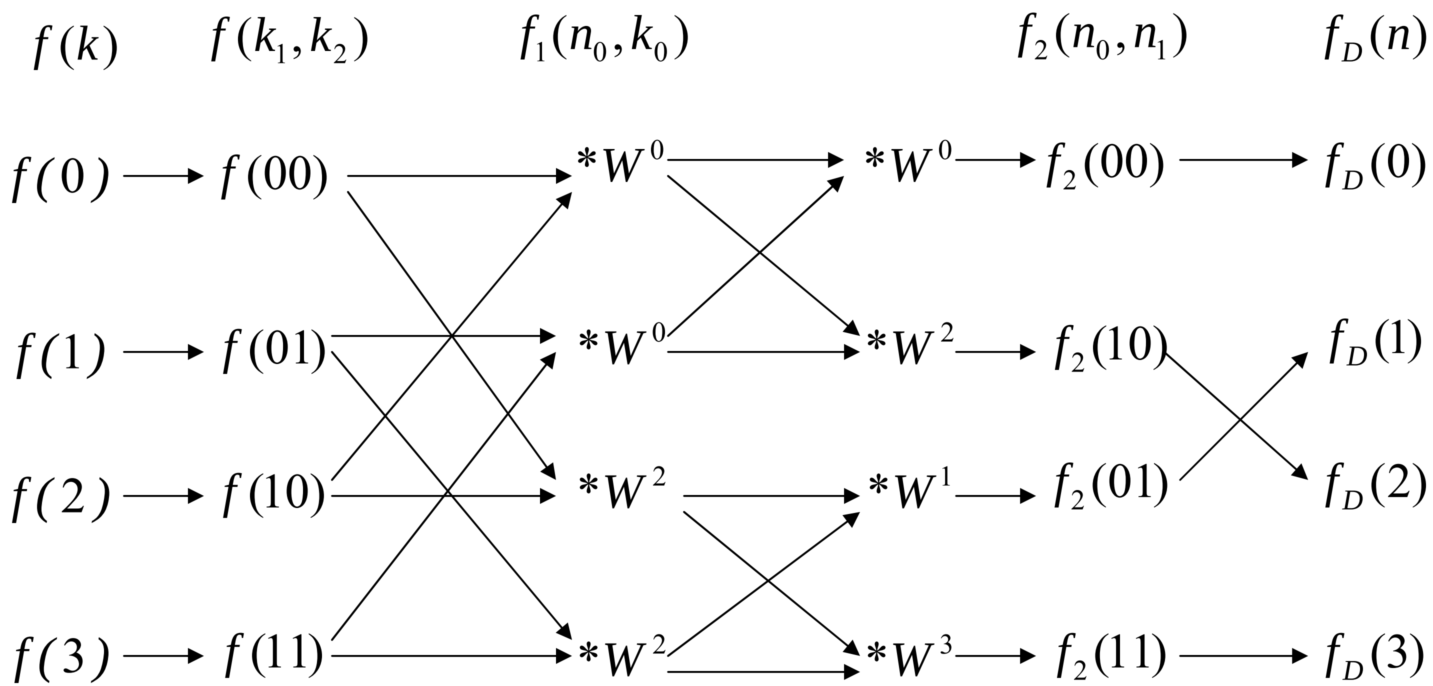
3.12 Transformada de Fourier rápida

$$f_1(n_0,k_0)=\sum_{k_1=0}^{1} f(k_1,k_0)W_4^{2n_0k_1}$$

$$\left\{\begin{array}{l} f_1(n_0,k_0)=\displaystyle\sum_{k_1=0}^{1} f(k_1,k_0)W_2^{n_0k_1} \quad \Big\} \text{ Una para pares, otra para impares} \\ f_2(n_0,n_1)=\displaystyle\sum_{k_0=0}^{1} f_1(n_0,k_0)W_4^{(2n_1+n_0)k_0} \quad \Big\} \text{ Suma ponderada de las anteriores} \\ F_D(n_1,n_0)=f_2(n_0,n_1) \end{array}\right.$$

3.12 Transformada de Fourier rápida

- Ejemplo: “mariposa” para 4 puntos:



3.12 Transformada de Fourier rápida

- DFT: lento (N^2), requiere poca memoria
- FFT: rápido ($N \log_2 N$), requiere bastante memoria (recursivo), requiere que el número de puntos sea potencia de 2.

N	N^2	$N \log_2 N$
2	4	2
32	1024	160
256	65536	2048
1024	1048576	10240

3. 12 Transformada de Fourier rápida

- Resumen FFT:
  - 1. Se elige el nº de muestras tal que $N=2^r$ con r entero. Si es necesario, se pueden incluir ceros aumentados.
  - 2. Para N muestras en el tiempo existen N frecuencias distintas.
  - 3. Como resultado de la extensión periódica, los puntos de muestra 0 y N son iguales, tanto en el tiempo como en la frecuencia ($f_0=f_N$; $F_{D0}=F_{DN}$)
  - 4. Las componentes de frecuencia positiva están en (0,N/2), las componentes de frecuencia negativa están en (N/2,N). Ocurre lo mismo en el tiempo (tiempos positivos y negativos)

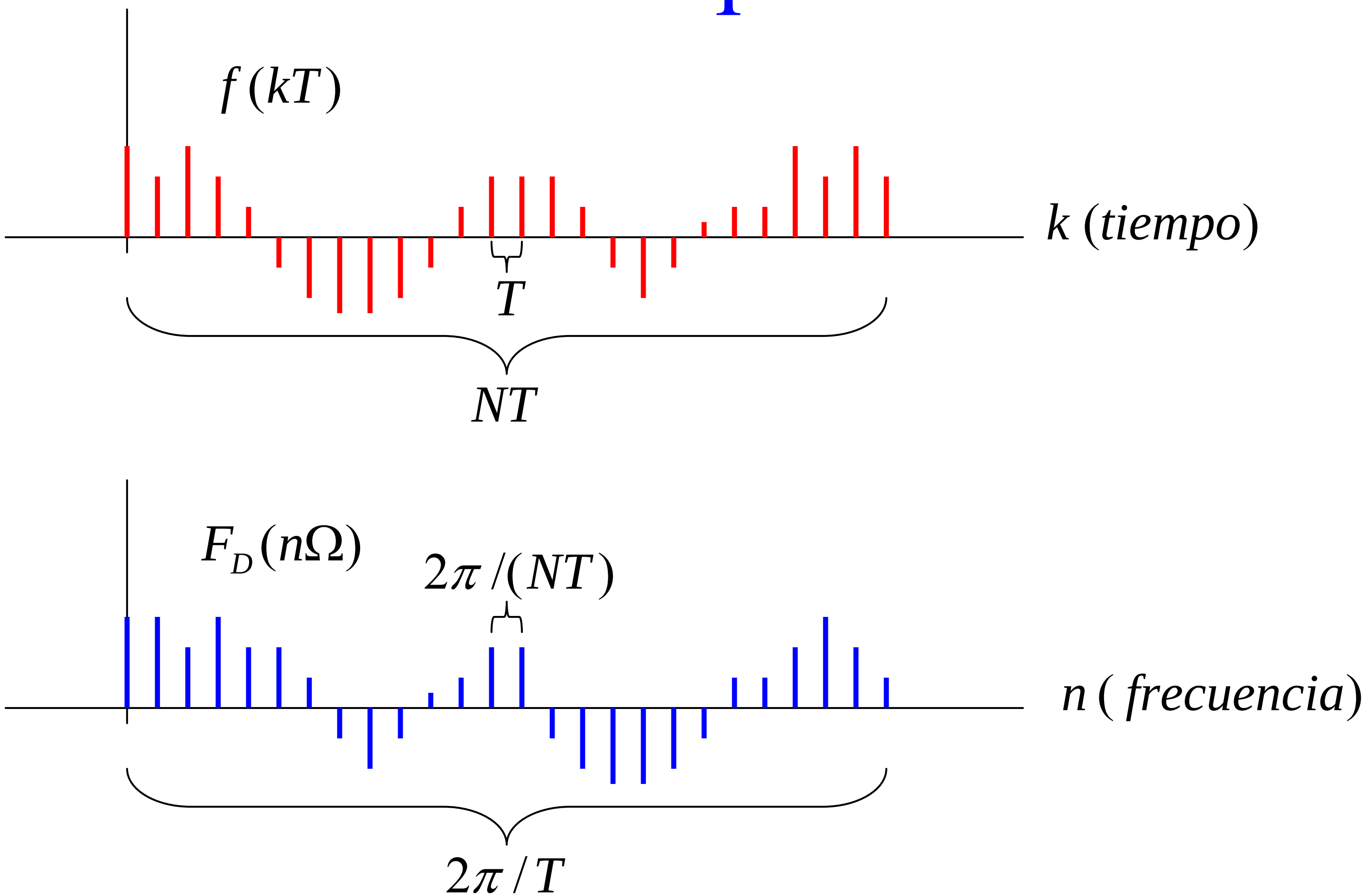
3.12 Transformada de Fourier rápida

- 5. Para funciones de valor real, las componentes de frecuencia positiva son complejas conjugadas de las componentes de frecuencia negativa. Los puntos n=0 y n=N/2 son comunes a ambos, por lo que tienen valor real.
- 6. La componente de frecuencia más alta (es decir, n=N/2) corresponde a 1/(2T) [Hz]. La frecuencia máxima visible se puede aumentar disminuyendo el espaciamiento entre frecuencias en el tiempo

3. 12 Transformada de Fourier rápida

– 7. El espaciamiento entre componentes de frecuencia es 1/(NT) [Hz], se puede disminuir agregando ceros aumentados a la secuencia de muestras

3. 12 Transformada de Fourier rápida



3.13 Transformada de Fourier de corto plazo (STFT)

- En general la DFT tiene aplicaciones en señales compuestas por sinusoides cuya fase, amplitud y frecuencia no dependen del tiempo ni del largo de la señal.
- Por ejemplo, la señal $x(n) = \cos(\omega_0 n^2)$ posee frecuencia instantánea $2n\omega_0$, variable en el tiempo. Utilizar DFT en este caso puede resultar engañoso => se utiliza STFT

3.13 Transformada de Fourier de corto plazo (STFT)

- Se define la transformada de tiempo corto de Fourier como

$$X_{STFT}(e^{j\omega},n)=\sum_{m=-\infty}^{\infty}x(n-m)w(m)e^{j\omega m}$$

- Donde w(m) es una <u>función ventana:</u> pondera un número finito de componentes de la sumatoria y anula otras.
- Si w(m)=1, se recupera la DFT. La estrategia radica en subdividir la señal en pedazos pequeños (w(m) se encarga de seleccionar los pedazos) y aplicar DFT a cada uno de ellos.
- Se obtiene una función bivariada: dependencia con ω y n . Se grafica con las variables en sendos ejes, denotando la magnitud de la transformada con tonos de grises: <u>espectrograma</u>

3.13 Transformada de Fourier de corto plazo (STFT)

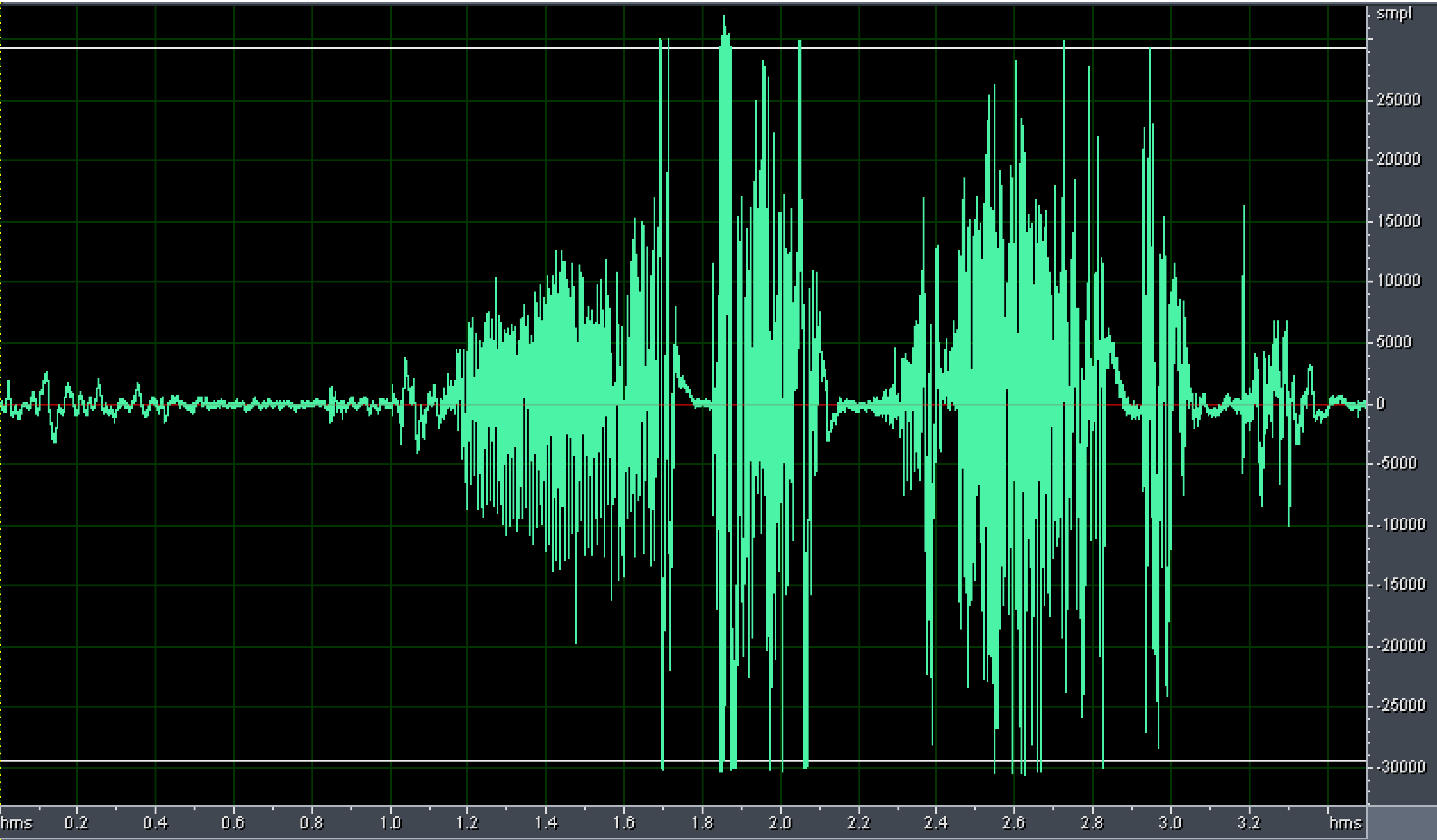
- Puede recuperarse la porción de señal que se encuentra en cada ventana, siempre que el largo de ésta sea menor que el número de muestras en el espacio de frecuencia:
- Definiendo la ventana w(m) para $0 \leq m \leq R-1$ y tomando N muestras de frecuencia, con $N \leq R$, $X_{STFT}(k,n) = X_{STFT}(e^{j2\pi k/N}, n)$

$$= \sum_{m=0}^{R-1} x(n-m)w(m)e^{j2\pi km/N}, \;\; 0 \leq k \leq N-1$$

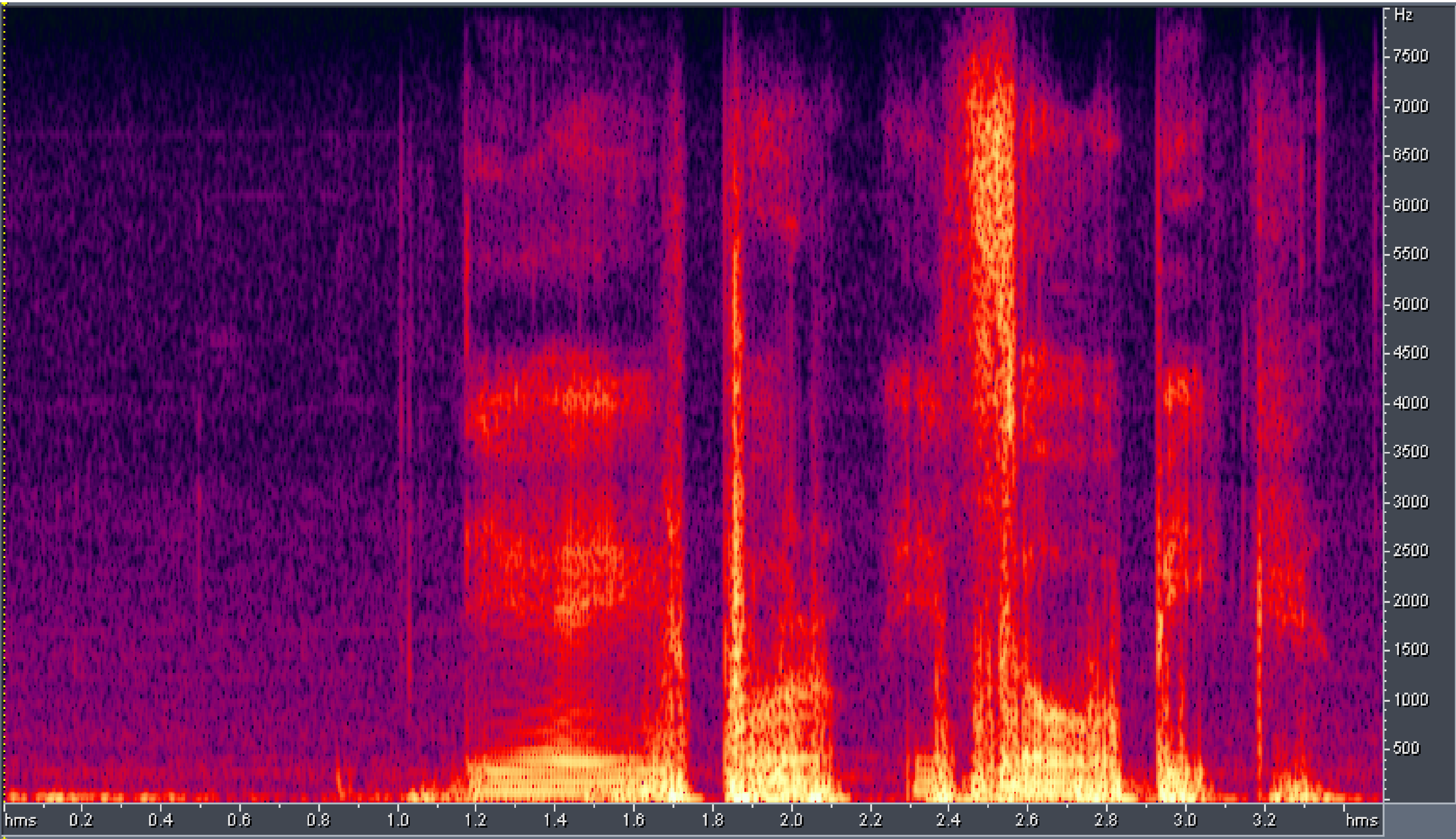
lo cual es la DFT de x(n-m)w(m). Aplicando IDFT,

- $$x(n-m) = \frac{1}{Nw(m)} \sum_{k=0}^{N-1} X_{STFT}(k,n)e^{j2\pi km/N}, 0 \leq m \leq R-1$$
- Fijando distintos n, se puede recuperar la señal completa a partir de sus transformadas cortas.

3.13 Transformada de Fourier de corto plazo (STFT)



3.13 Transformada de Fourier de corto plazo (STFT)



3.13 Recorte de señales por funciones ventana

- Para obtener ciertas características en ciertos filtros (como un espectro en frecuencia acotado) se requieren considerar secuencias infinitas (en este caso, una señal infinitamente larga en el tiempo)
- Es imposible considerar funciones infinitas en la práctica => se utilizan sólo algunos términos
- Simplemente dejar de considerar algunos de ellos (lo cual equivale a multiplicar los términos por una función rect(·)) induce distorsiones (en el dual, se hace convolución, en este ejemplo, con un Sa(·))
- Se utilizan las funciones ventana para ponderar las secuencias, de manera de inducir menos lóbulos laterales

3.13 Recorte de señales por funciones ventana

- Funciones ventana
  - Ventana rectangular

$$w(t) = \begin{cases} 1 & |t| < \tau/2 \\ 0 & otro \end{cases}$$

$$W(f) = \tau Sa(\frac{2\pi f \tau}{2})$$

  - Ventana triangular

$$w(t) = \begin{cases} 1 - \dfrac{2|t|}{\tau} & |t| \leq \tau/2 \\ 0 & otro \end{cases}$$

$$W(f) = \frac{\tau}{2}[Sa(\frac{\pi f \tau}{2})]^2$$

3.13 Recorte de señales por funciones ventana

- Funciones ventana
  - Ventana Hanning

$$w(t)=\begin{cases}\cos^2\left(\dfrac{\pi t}{\tau}\right)=\dfrac{1+cos\left(\dfrac{2\pi t}{\tau}\right)}{2} & |t|\le \tau/2\\ 0 & otro\end{cases}$$

$$W(f)=\frac{\tau}{2}Sa(\pi f\tau)\left[\frac{1}{1-(f\tau)^2}\right]$$

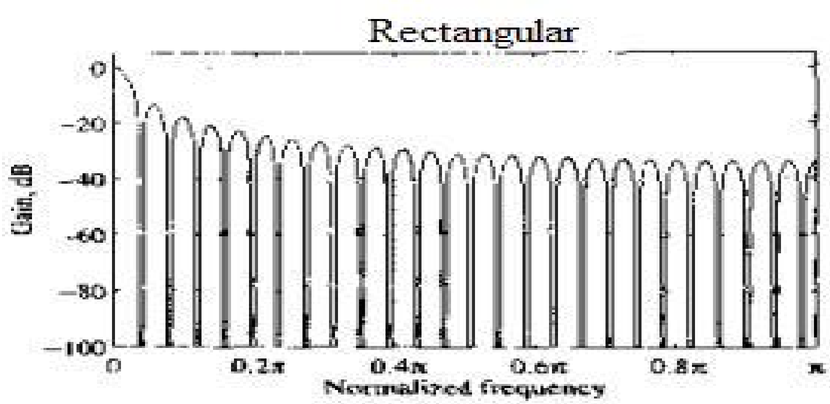
  - Ventana Hamming

$$w(t)=\begin{cases}0,54+0,46\cos\left(\dfrac{2\pi t}{\tau}\right) & |t|\le \tau/2\\ 0 & otro\end{cases}$$

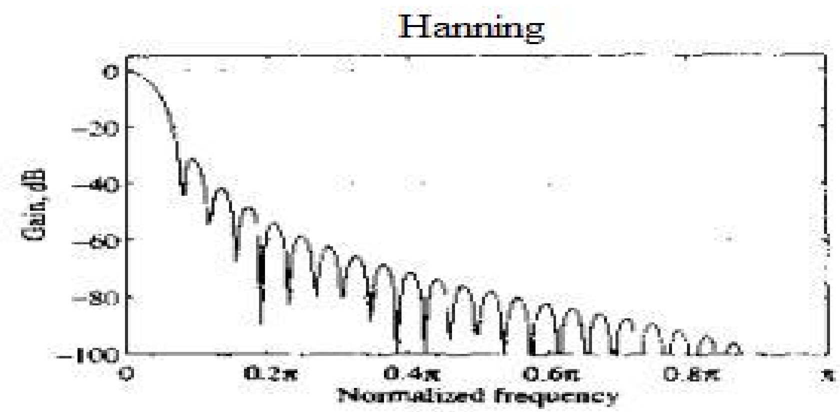
$$W(f)=\frac{\tau sen(\pi f\tau)}{\pi f\tau}\left[\frac{0,54-0,08(f\tau)^2}{1-(f\tau)^2}\right]$$

3.13 Recorte de señales por funciones ventana

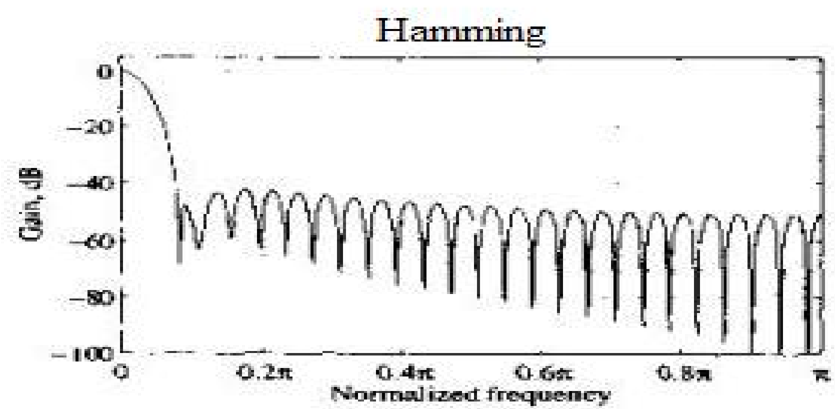
- Funciones ventana: ganancias en dB



(a)



(b)



(c)

3.13 Recorte de señales por funciones ventana

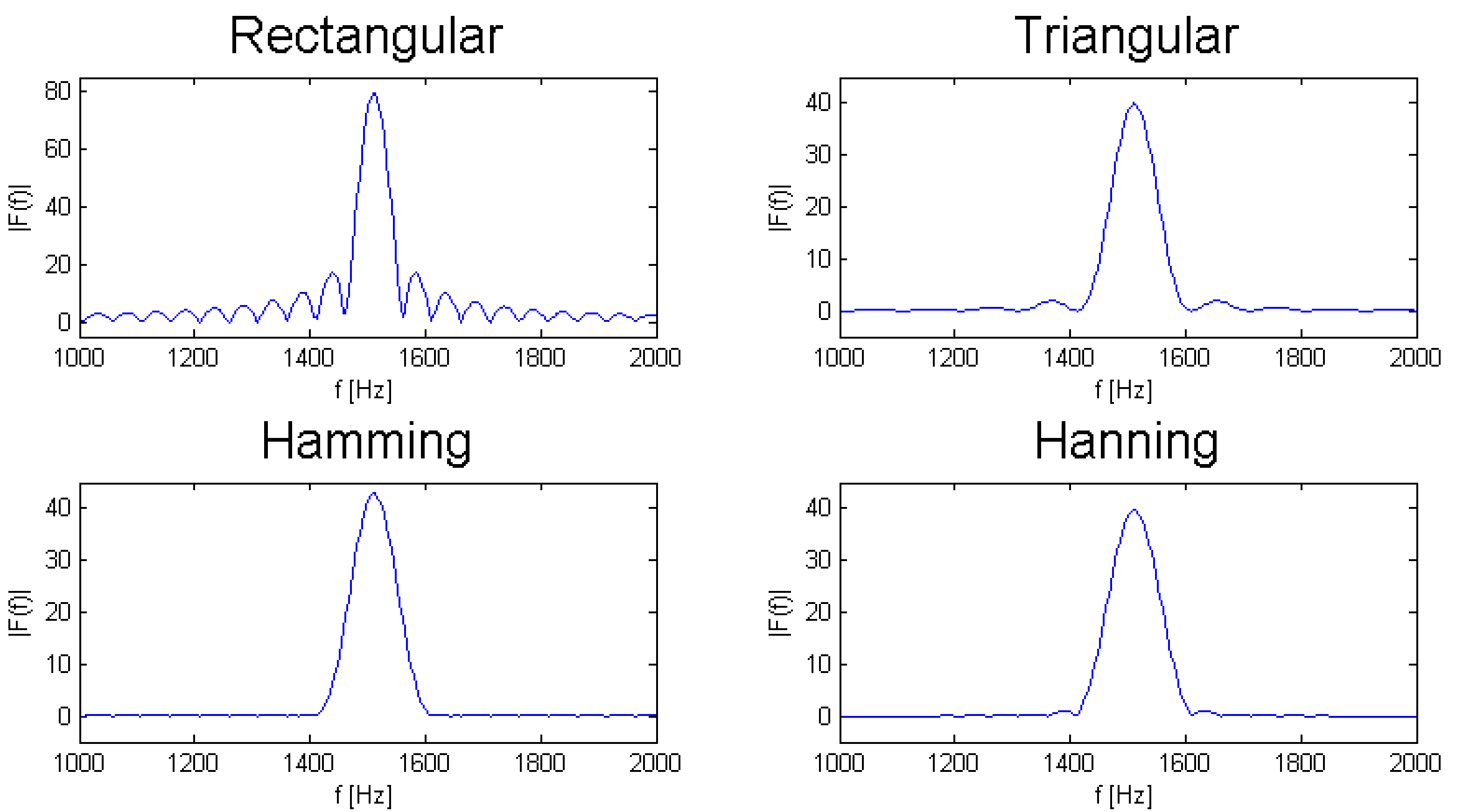
- Funciones ventana:
  - Mientras más rápido decaigan los lóbulos laterales, mejor la ventana (induce menos distorsión a la señal que se recorta), sopesando el uso de recursos y tiempo de cálculo, tan vital en aplicaciones digitales. Mientras más compleja la ventana, mayor tiempo de procesamiento puede requerir

3.13 Recorte de señales por funciones ventana

- Ejemplo:
  - Señal senoidal $f(t)=\sin(2\pi*1500t)$ muestreada a $Fs=8000$ Hz
  - Se aplica una ventana de largo 160 muestras (20 ms)

3.13 Recorte de señales por funciones ventana

- Ejemplo:



3.14 Sistemas de tiempo discreto

- Un sistema de tiempo discreto toma como entrada una secuencia de valores x(t) y entrega como salida otra secuencia y(t)
- Los sistemas de tiempo discreto se pueden expresar mediante ecuaciones de diferencias, por ejemplo:

$$y(t) = x(t) + x(t-1)$$

$$y(t) = y(t-1) + 2x(t)$$

$$y(t) = y^2(t-2) - y(t-3) + x(t-1)$$

3.14 Sistemas de tiempo discreto

En un sistema causal, la salida actual no puede depender de las entradas futuras. Por ejemplo, el siguiente sistema no es causal:

$$y(t) = x(t+1) + 2x(t) + x(t-1)$$

- En un sistema no-causal, la salida puede aparecer antes de que se aplique la entrada => no pueden implementarse en "tiempo real"

3.14 Sistemas de tiempo discreto

- Los sistemas de tiempo discreto lineales se pueden caracterizar mediante su respuesta al impulso discreto, el que se define como:

$$\delta(t) = \begin{cases} 1 & t = 0 \\ 0 & t \neq 0 \end{cases}$$

- En base al tipo de respuesta al impulso que presenten, se pueden clasificar como FIR o IIR

3.14 Sistemas de tiempo discreto

- Filtros FIR: presentan una respuesta al impulso de duración temporal finita, ej:

$$y(t) = x(t) + 0.6x(t-1) - 0.2x(t-2)$$

- En este tipo de filtros, la salida se calcula a partir de las entradas actual y anteriores, pero no a partir de las salidas anteriores

3.14 Sistemas de tiempo discreto

- Filtros IIR: presentan una respuesta al impulso de duración temporal infinita, ej:

$$y(t) = x(t) + 0.5y(t-1)$$

- En este tipo de filtros, la salida actual se calcula a partir de las salidas anteriores y de las entradas actual y anteriores.

3.14 Sistemas de tiempo discreto

- Ejemplo: Calcular la respuesta al impulso del sistema:

$$y(t) = x(t) - 0.4x(t-1) + 0.2x(t-2)$$

- Solución:

$$t = 0 \qquad y(0) = \delta(0) - 0.4\delta(-1) + 0.2\delta(-2) = 1$$

$$t = 1 \qquad y(1) = \delta(1) - 0.4\delta(0) + 0.2\delta(-1) = -0.4$$

$$t = 2 \qquad y(2) = \delta(2) - 0.4\delta(1) + 0.2\delta(0) = 0.2$$

$$t = 3 \qquad y(3) = \delta(3) - 0.4\delta(2) + 0.2\delta(1) = 0$$

3.14 Sistemas de tiempo discreto

$$\Rightarrow h(0) = 1\,,\; h(1) = -0.4\,,\; h(2) = 0.2$$

$$h = \{1\,, -0.4\,, 2\}$$

- En este caso, como el filtro es FIR, la respuesta al impulso vale cero para todo t, excepto t=0, t=1 y t=2.
- Si se conoce la respuesta al impulso de un filtro, es posible calcular su salida ante cualquier entrada que se desee.

3.14 Sistemas de tiempo discreto

- La salida del filtro ante una secuencia de entrada x(n) puede calcularse mediante una convolución:

$$y(n) = \sum_{k=-\infty}^{\infty} x(n-k)h(k) = x * h$$

- En el caso de filtros FIR, sólo es necesario hacer la convolución en los n para los cuales $h(n) \neq 0$
- La convolución tiene algunas propiedades:

$$f * g = g * f \qquad f*(g+h) = f*g + f*h$$

$$(f*g)*h = f*(g*h) = f*g*h$$

3.14 Sistemas de tiempo discreto

- Ejemplo: Calcular la salida del sistema FIR anterior si la entrada es la secuencia {1,2,1,0,0,...}

Sistema: $h(0)=1,\ h(1)=-0.4,\ h(2)=0.2$

Entrada: $x(0)=1,\ x(1)=2,\ x(2)=1$

$$y(0)=x(0-0)h(0)+x(0-1)h(1)+x(0-2)h(2)=0$$

$$=1*(1)+0*(-0.4)+0*(0.2)=1$$

$$y(1)=x(1-0)h(0)+x(1-1)h(1)+x(1-2)h(2)=$$

$$=2*(1)+1*(-0.4)+0*(0.2)=1.6$$

3.14 Sistemas de tiempo discreto

Sistema: $h(0)=1,\ h(1)=-0.4,\ h(2)=0.2$

Entrada: $x(0)=1,\ x(1)=2,\ x(2)=1$

$$y(2)=x(2-0)h(0)+x(2-1)h(1)+x(2-2)h(2)=$$
$$=1*(1)+2*(-0.4)+1*(0.2)=0.4$$
$$y(3)=x(3-0)h(0)+x(3-1)h(1)+x(3-2)h(2)=$$
$$=0*(1)+1*(-0.4)+2*(0.2)=0$$
$$y(4)=x(4-0)h(0)+x(4-1)h(1)+x(4-2)h(2)=$$
$$=0*(1)+0*(-0.4)+1*(0.2)=0.2$$

3.14 Sistemas de tiempo discreto

Sistema: $h(0)=1,\ h(1)=-0.4,\ h(2)=0.2$

Entrada: $x(0)=1,\ x(1)=2,\ x(2)=1$

Salida: $y=\{1,1.6,0.4,0,0.2,0,0,0,\ldots\}$

- Como el sistema es FIR, si la entrada es de duración temporal finita, la salida también lo es.

3.15 Generalización a procesamiento de señales 2D

- Los análisis anteriores son extensibles a señales 2D, teniendo en cuenta que una imagen digital es una imagen que se encuentra muestreada. Cada muestra se llama PIXEL (picture element). Para tonos de grises, cada pixel corresponde a 8 bits (que representan números del 0 al 255, donde 0 significa negro y 255, blanco). Para color, existen tres canales (R de Red, G de Green y B de Blue) que indican las proporciones de rojo, verde y azul que se deben utilizar para crear determinados colores

3.15 Generalización a procesamiento de señales 2D

- Aun se sigue teniendo aliasing…

Con Aliasing

Sin Aliasing

3.15 Generalización a procesamiento de señales 2D

- Y existen filtros específicos para detección de bordes…

-1	0	1
-2	0	2
-1	0	1

Horizontal

1	2	1
0	0	0
-1	-2	-1

Vertical

Filtro Sobel

3.15 Generalización a procesamiento de señales 2D

- Y existen filtros específicos para detección de bordes…

-1	0	1
-1	0	1
-1	0	1

Horizontal

1	1	1
0	0	0
-1	-1	-1

Vertical

Filtro Prewitt

3.15 Generalización a procesamiento de señales 2D

- Y existen filtros específicos para detección de bordes…

0	1
-1	0

Horizontal

1	0
0	-1

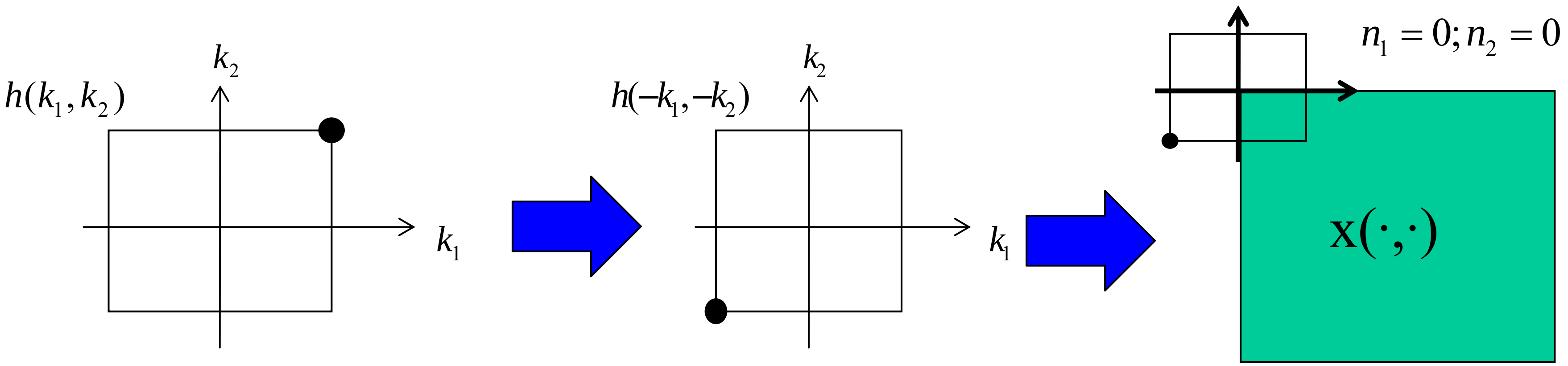
Vertical

Filtro Roberts

3.15 Generalización a procesamiento de señales 2D

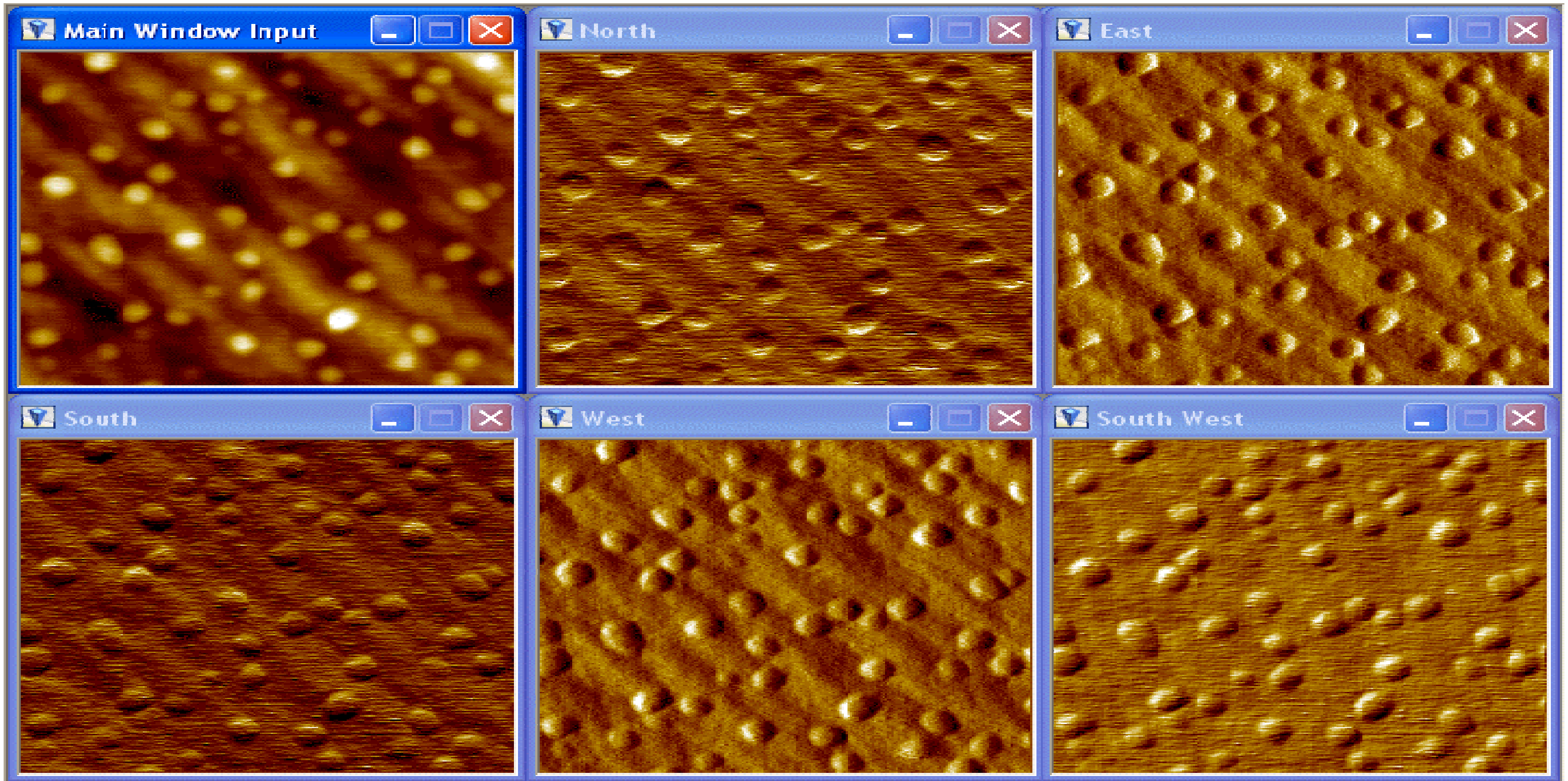
- Convolución discreta 2D:

$$y(n_1,n_2)=\sum_{k_1=0}^{M-1}\sum_{k_2=0}^{N-1}x(k_1,k_2)h(n_1-k_1,n_2-k_2)$$



3.15 Generalización a procesamiento de señales 2D

- Convolución discreta 2D:
  - Acentuación de ejes con filtro Roberts (pasaaltos)



3.15 Generalización a procesamiento de señales 2D

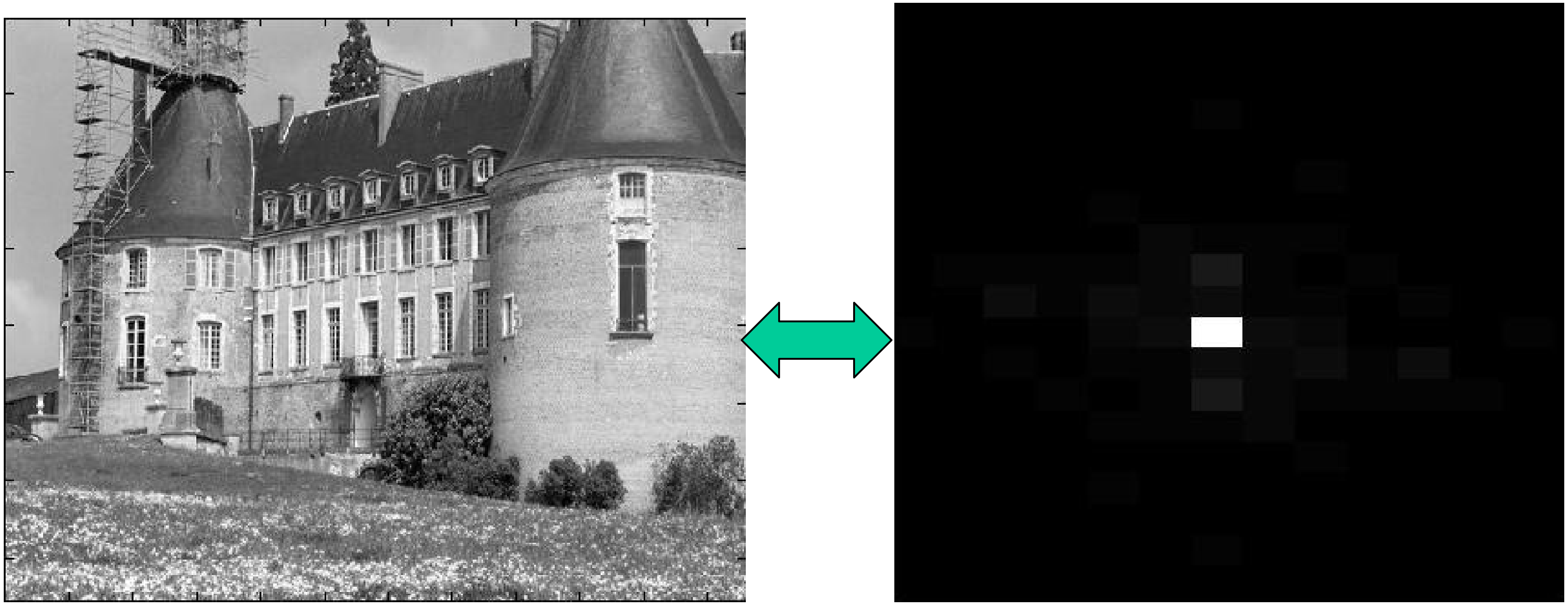
- DTFT 2D:

$$X(\omega_1,\omega_2)=\sum_{n_1=-\infty}^{\infty}\sum_{n_2=-\infty}^{\infty}x(n_1,n_2)e^{-j\omega_1 n_1}e^{-j\omega_2 n_2}$$

$$x(n_1,n_2)=\frac{1}{(2\pi)^2}\int_{\omega_1=-\pi}^{\pi}\int_{\omega_2=-\pi}^{\pi}X(\omega_1,\omega_2)e^{j\omega_1 n_1}e^{j\omega_2 n_2}d\omega_1 d\omega_2$$

3.15 Generalización a procesamiento de señales 2D

- DTFT 2D:



  - La mayor cantidad de componentes de frecuencia de las imágenes visuales están en las frecuencias bajas

3.16 Realización de filtros digitales

- Los filtros digitales se pueden caracterizar por su ecuación de diferencias:

$$y(n)=\sum_{k=0}^{M} a_k x(n-k)-\sum_{k=1}^{N} b_k y(n-k)$$

- La ecuación de diferencias puede ser implementada en hardware o software (utilizando computadores): se requieren solamente las operaciones de multiplicación, suma y retardo:

3.16 Realización de filtros digitales

- Realización: ordenamiento de los elementos del filtro, algoritmo de implementación
- Se puede representar una realización a través de un diagrama de bloques
- Dos realizaciones son equivalentes si tienen la misma función de transferencia discreta, H(z)

3.16 Realización de filtros digitales

- No se pueden implementar los coeficientes del filtro con precisión infinita=> distintas realizaciones equivalentes, en la práctica, tienen distinta eficacia en rigor: se busca la mejor realización.
- Para analizar un diagrama de bloques, se escriben las expresiones para las señales de salida, en términos de señales internas, que luego se eliminan, obteniendo la ecuación pedida.

3.16 Realización de filtros digitales

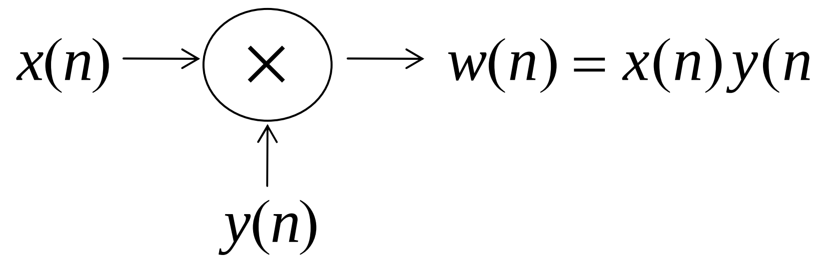
- La estructura de un filtro digital es canónica si el número de retardos en el diagrama de bloques es igual al orden de la ecuación de diferencias

3.16 Realización de filtros digitales

- Retardo unitario: $x(n) \rightarrow \boxed{z^{-1}} \rightarrow y(n) = x(n-1)$
- Suma/Resta: $x(n) \rightarrow \underset{\uparrow \pm \, y(n)}{\boxed{+}} \rightarrow w(n) = x(n) \pm y(n)$
- Multiplicador o ganancia: $x(n) \rightarrow \overset{A}{\triangleright} \rightarrow y(n) = Ax(n)$
- Ramificación: $x(n) \longrightarrow y_1(n) = x(n)$, $\quad y_2(n) = x(n)$

3.16 Realización de filtros digitales

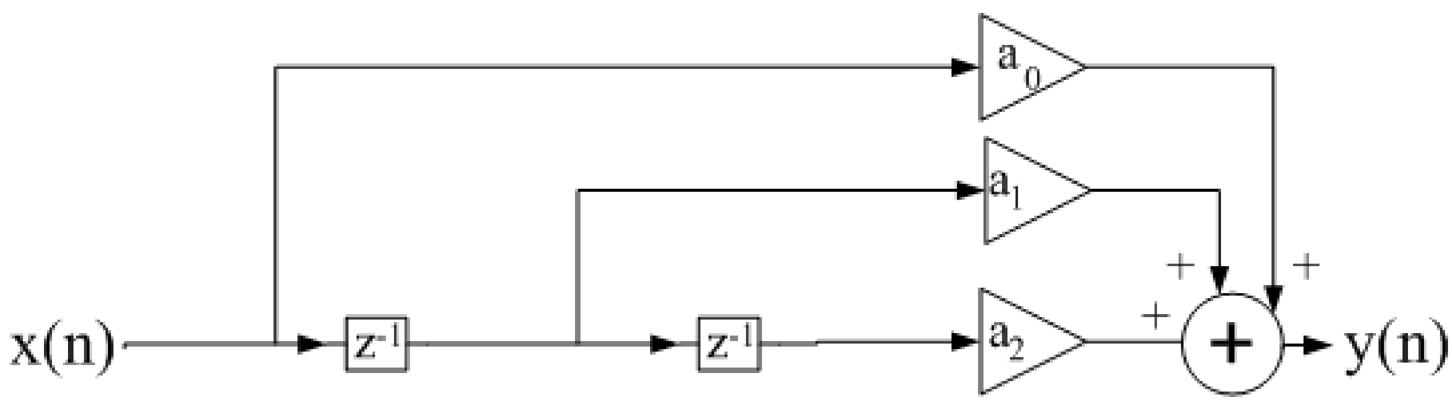
- Multiplicador de señal:



- Realizaciones directas: los coeficientes de multiplicación son los coeficientes de la función de transferencia discreta.
- Un filtro FIR de orden N requiere N+1 coeficientes, N+1 multiplicadores y N sumadores para su implementación

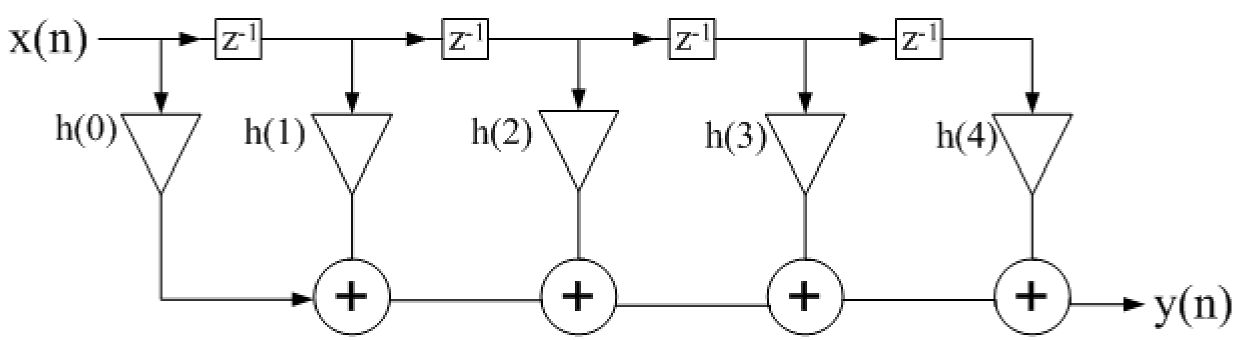
3.16 Realización de filtros digitales (FIR)

- Para filtros FIR: $H(z) = \sum_{k=0}^{N} a_k z^{-k}$
- Realización no recursiva: la salida depende solo de la entrada retardada.



3.16 Realización de filtros digitales (FIR)

- Formas directas:
  - Estructuras de filtros para las cuales los coeficientes involucrados coinciden con aquellos de la ecuación de diferencias.
  - Ejemplo: $$y(n) = \sum_{k=0}^{4} h(k)x(n-k)$$

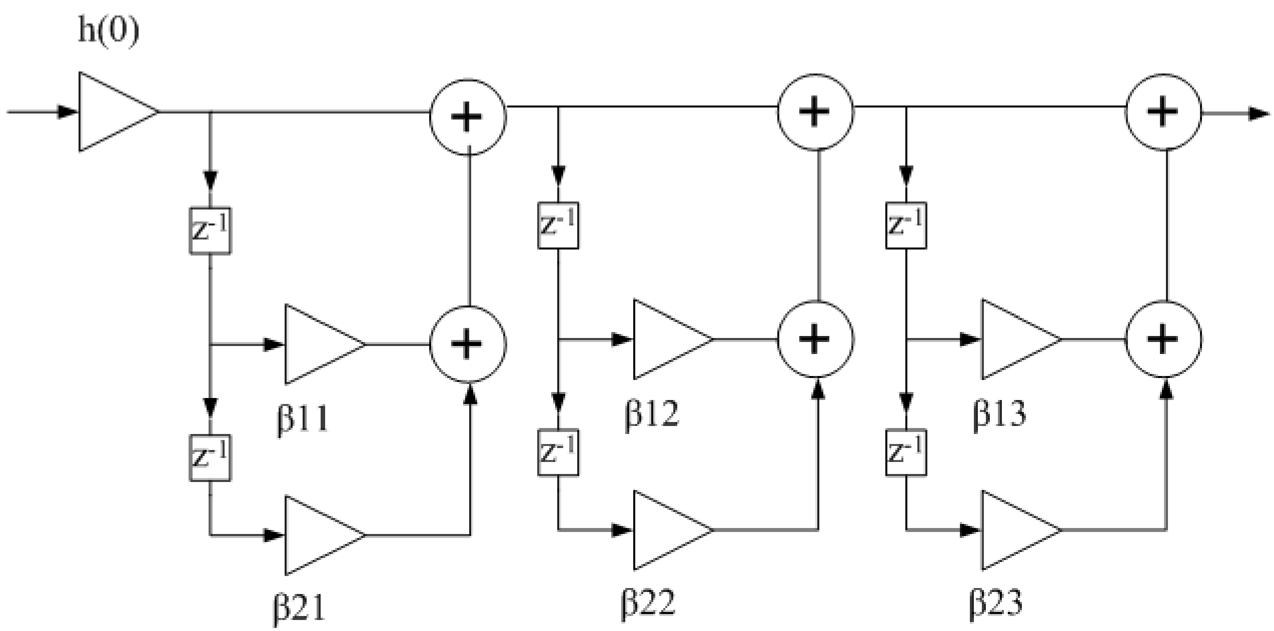


3.16 Realización de filtros digitales (FIR)

- Formas en cascada:
  - Para filtros FIR de alto orden, se pueden realizar implementaciones de varios filtros de orden 1 ó dos en cascada. $$H(z) = h(0)\prod_{k=1}^{K}(1+\beta_{1k}z^{-1}+\beta_{2k}z^{-2})$$
  - Si N es par, K=N/2. De otro modo, si la cascada es de filtros de orden 1 y N impar, K=(N+1)/2

3.16 Realización de filtros digitales (FIR)

- Formas en cascada:



3.16 Realización de filtros digitales (FIR)

- Forma polifásica: se agrupan los términos pares e impares y se factoriza un retardo, de manera de obtener dos filtros FIR en z^2, y así sucesivamente

$$H(z) = (h(0) + h(2)z^{-2} + h(4)z^{-4} + \ldots)$$

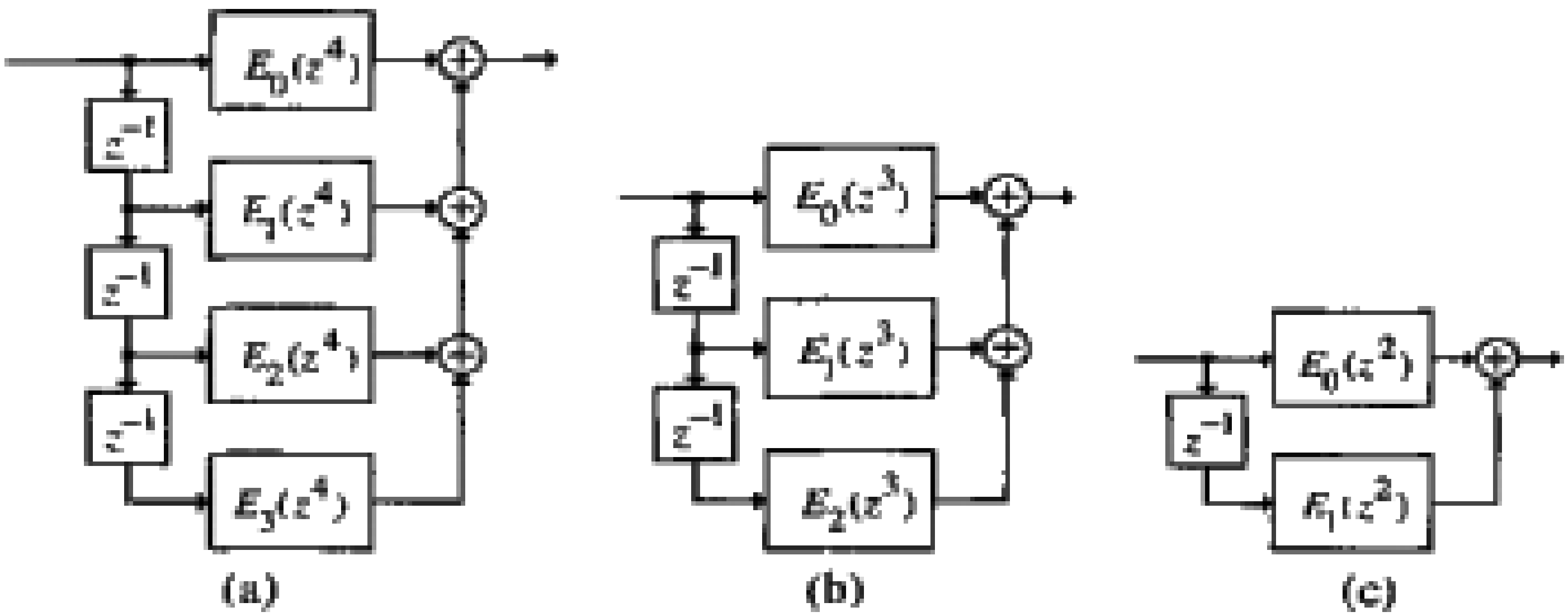
$$+ z^{-1}(h(0) + h(3)z^{-2} + h(5)z^{-4} + \ldots) = E_0(z^2) + z^{-1}E_1(z^2)$$

...

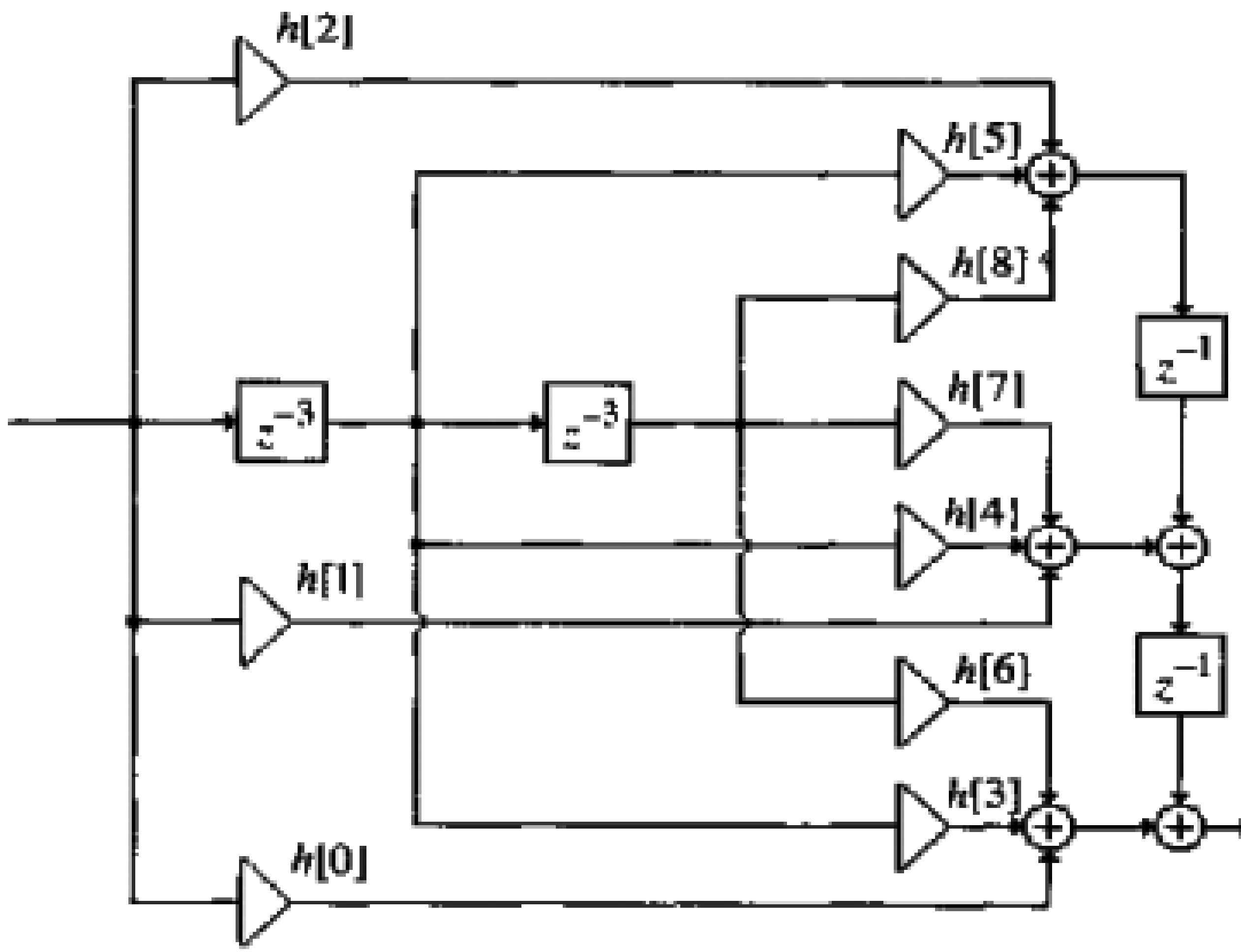
$$H(z) = \sum_{m=0}^{L-1} z^{-m} E_m(z^L)$$

$$E_m(z) = \sum_{n=0}^{\lfloor (N+1)/L \rfloor} h(Ln+m)z^{-n}, 0 \le m \le L-1, \; h(n) = 0 \text{ si } n > N$$

3.16 Realización de filtros digitales (FIR)



3.16 Realización de filtros digitales (FIR)



3.16 Realización de filtros digitales (FIR)

- Formas de fase lineal: se puede explotar la simetría o antisimetría de la función de transferencia para reducir hasta a la mitad el número de coeficientes necesarios para la implementación.

3.16 Realización de filtros digitales (FIR)

$$H(z)=\sum_{k=0}^{6} h(k) z^{-k}$$

h(n) es simétrica

- $h(n)=h(N-n)$
- $H(z)=h(0)(1+z^{-6})$

$+h(1)(z^{-1}+z^{-5})$

$+h(2)(z^{-2}+z^{-4})$

$+h(3)z^{-3}$

h(n) es antisimétrica

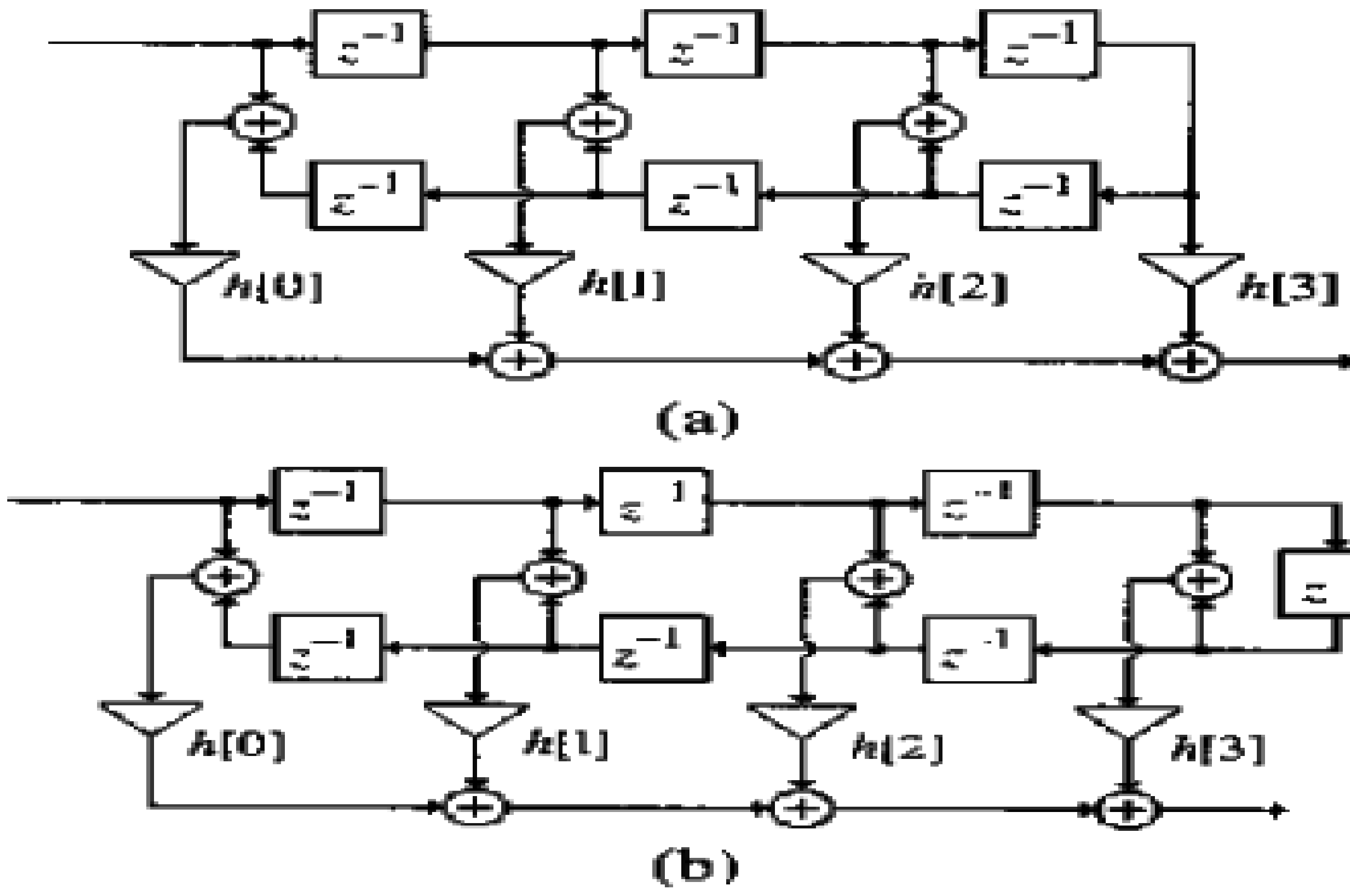
- $h(n)=-h(N-n)$
- $H(z)=h(0)(1-z^{-6})$

$+h(1)(z^{-1}-z^{-5})$

$+h(2)(z^{-2}-z^{-4})$

$+h(3)z^{-3}$

3.16 Realización de filtros digitales (FIR)



3.16 Realización de filtros digitales (IIR)

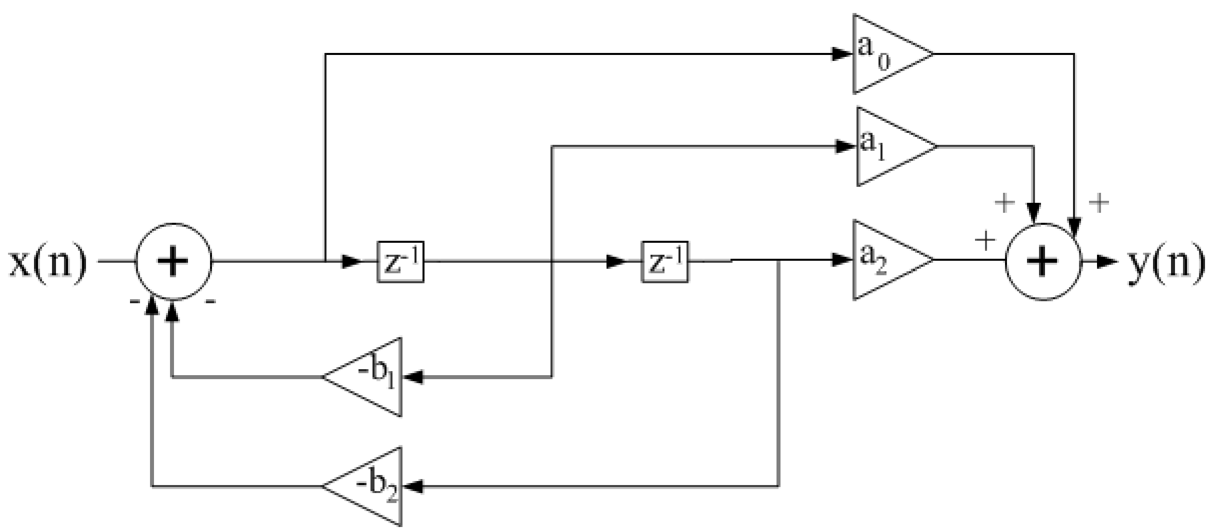
- En general para un filtro IIR, como ya se ha visto, se requiere configurar el filtro de manera de producir una función racional de transferencia. Es importante aquí contar con valores anteriores de la salida para conformar el siguiente paso.

3.16 Realización de filtros digitales

- Para filtros IIR:

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

- <u>Realización recursiva</u>: la salida depende de sí misma retardada.



3.16 Realización de filtros digitales (IIR)

- Formas directas: En este caso se requieren, para un filtro IIR de orden N, 2N+1 coeficientes y multiplicadores, y 2N sumadores para la implementación.

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

3.16 Realización de filtros digitales (IIR)

- El filtro de este modo puede verse como la cascada entre un filtro FIR

$$N(z) = \sum_{k=0}^{N} a_k z^{-k}$$

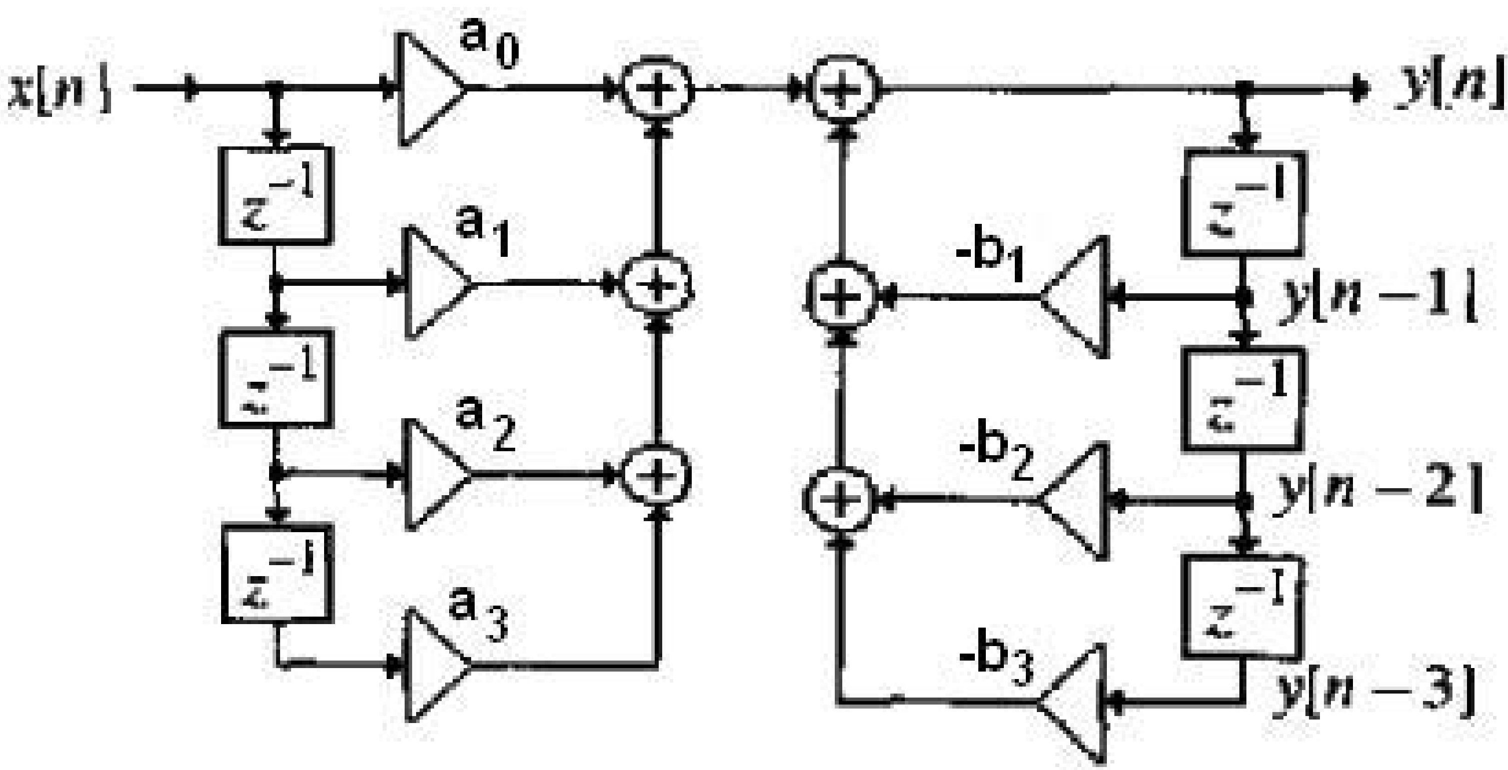
…Y un filtro tal que

$$H_2(z) = \frac{1}{D(z)} = \frac{1}{1 + \sum_{k=1}^{N} b_k z^{-k}}$$

$$y(n) = w(n) - b_1 y(n-1) - \ldots - b_N y(n-3)$$

3.16 Realización de filtros digitales (IIR)

- La realización directa I es entonces:

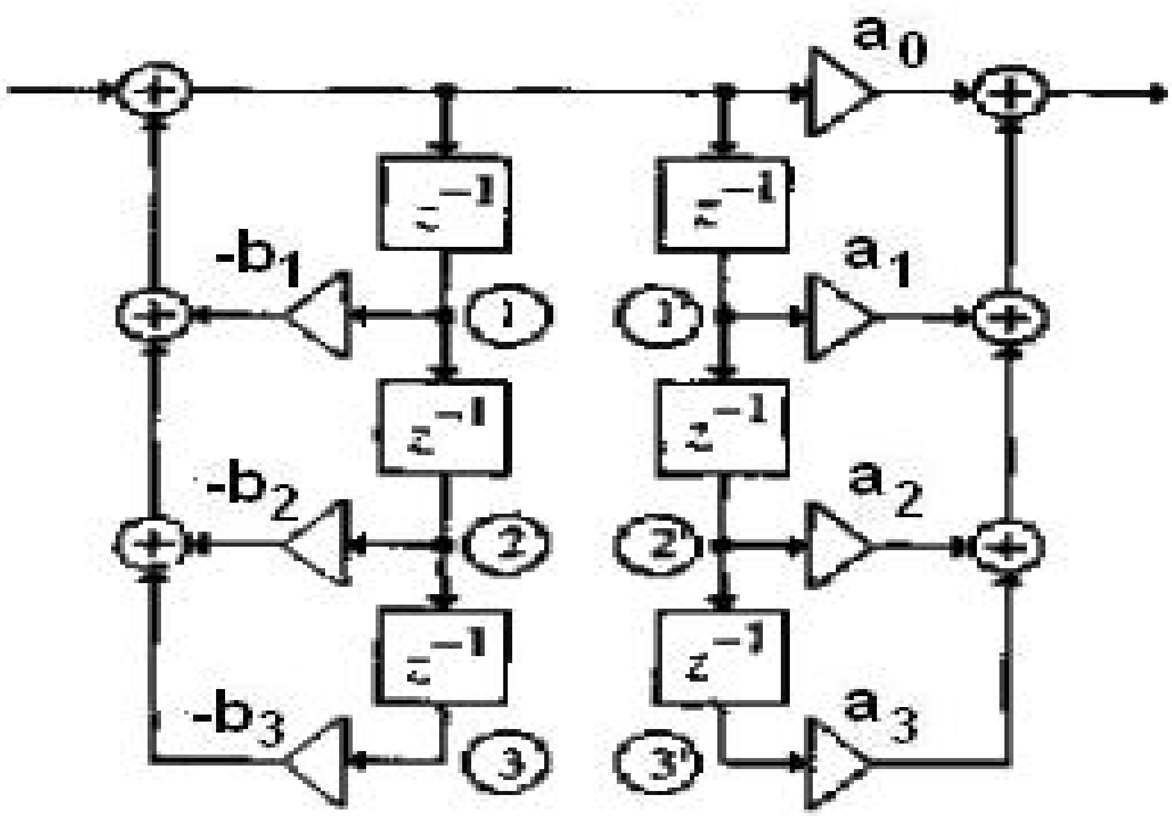


3.16 Realización de filtros digitales (IIR)

- Se llama realización directa I porque existe una segunda: esta realización no es canónica (2N retardos para un filtro de orden N)
- Se pueden compartir los 3 retardos (se tiene la misma señal a lo largo de sus rutas)

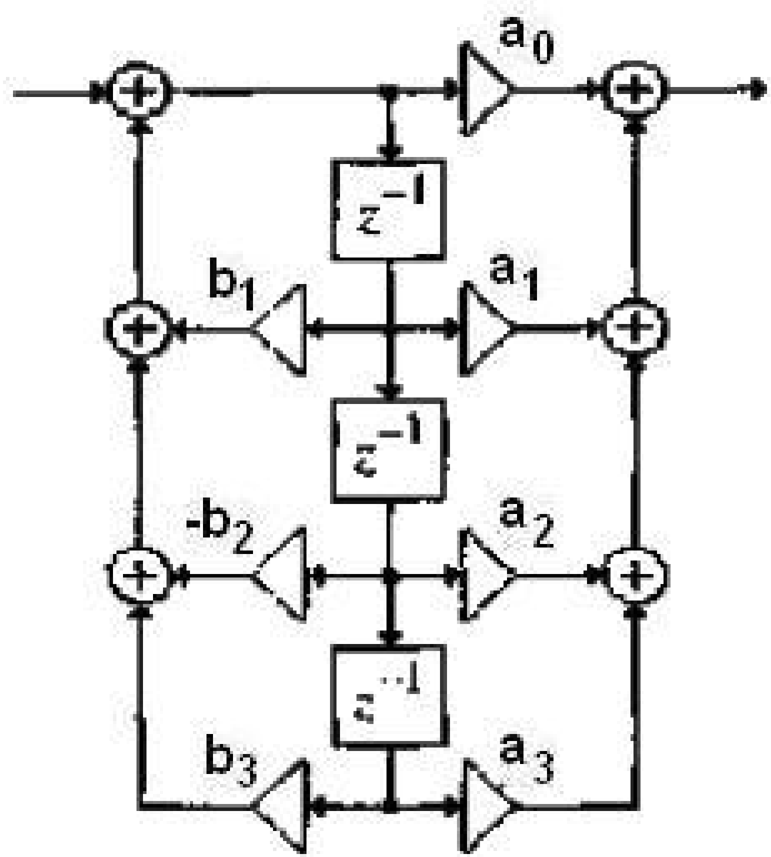
3.16 Realización de filtros digitales (IIR)

- Transponiendo ambos sub-filtros en la cascada se obtiene:



3.16 Realización de filtros digitales (IIR)

- Compartiendo los nodos se llega a la forma directa II:



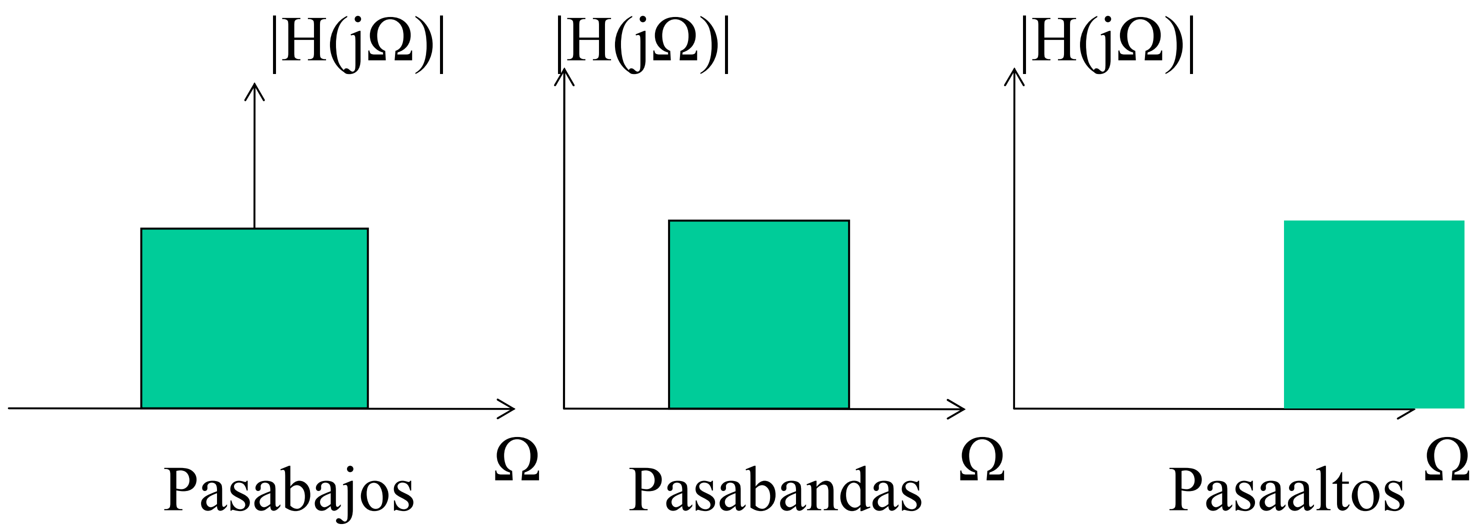
3.16 Realización de filtros digitales (IIR)

- Formas en cascada: encontrando la factorización de los polinomios de la función de transferencia se puede realizar el filtro como una cascada de sub-filtros Iir menores. Usualmente éstos son de orden 2 ó 1.

$$H(z) = a_0 \prod_k \left(\frac{1 + \beta_{1k} z^{-1} + \beta_{2k} z^{-2}}{1 + \alpha_{1k} z^{-1} + \alpha_{2k} z^{-2}} \right)$$

3.17 Filtros análogos

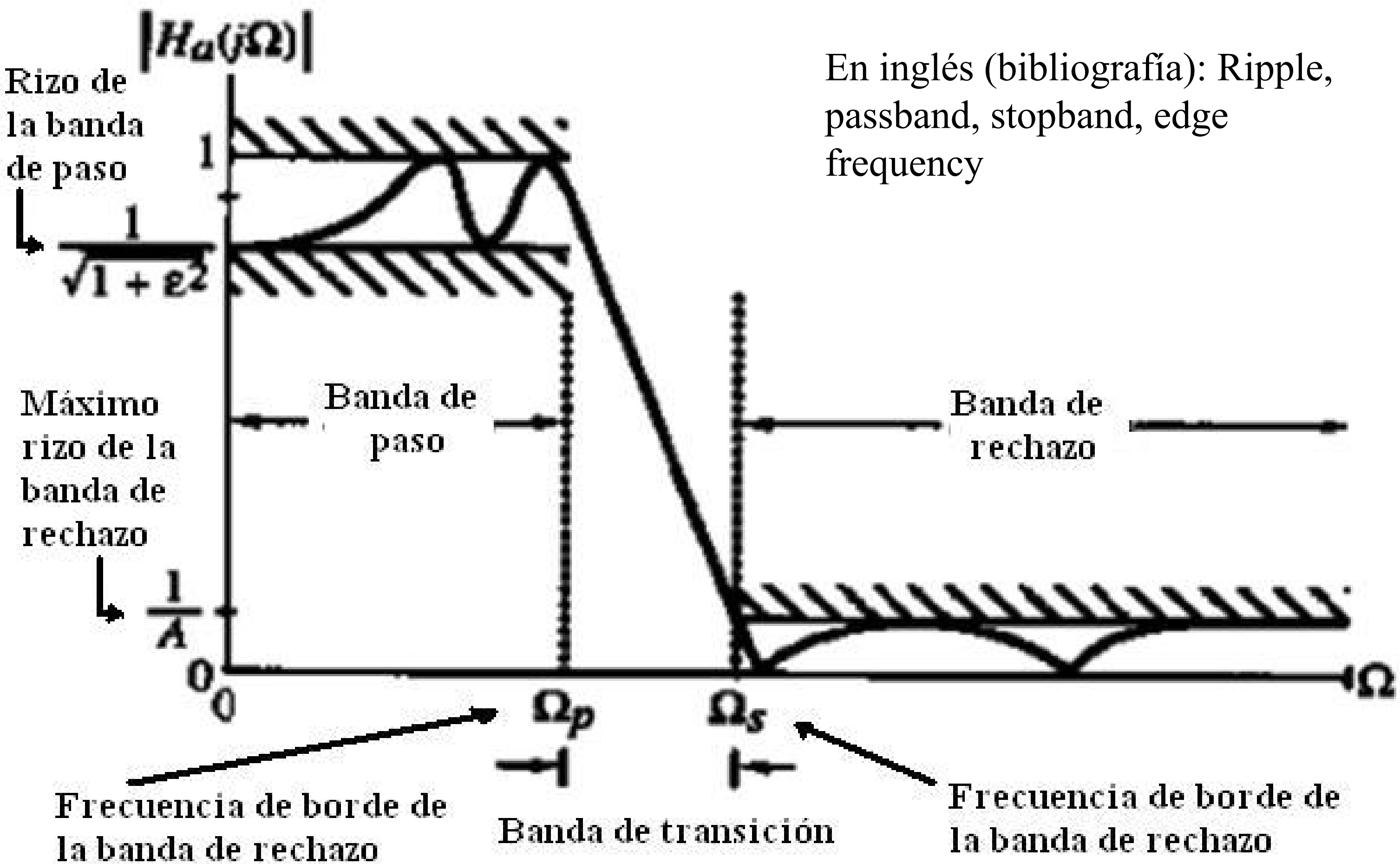
- Un filtro análogo es aquél que tiene una respuesta continua. Poseen una función de transferencia que se denotará H(jΩ).
- Idealmente:



3.17 Filtros análogos

- En la práctica es imposible alcanzar una función como las ideales para los filtros=> cabe definir ciertos parámetros

En inglés (bibliografía): Ripple, passband, stopband, edge frequency



3.17 Filtros análogos

- En la figura anterior, A y ε son parámetros arbirarios que describen cuán alejados estamos de la situación ideal. Lo mismo ocurre con las frecuencias de borde.
- Se definen el pico del rizo de la banda de paso (peak passband ripple) y la mínima atenuación de la banda de rechazo (minimum stopband attenuation) como, respectivamente:

$$\alpha_p = -20log_{10}\left(\frac{1}{\sqrt{1+\epsilon^2}}\right)$$

$$\alpha_s = -20log_{10}\left(\frac{1}{A}\right)$$

- Se define la razón de transición o parámetro de selectividad, como

$$k = \frac{\Omega_p}{\Omega_s}$$

3.17 Filtros análogos

- El parámetro de discriminación se define como:

$$k_1 = \frac{\epsilon}{\sqrt{A^2 - 1}}$$

- A continuación se revisan los tres tipos de filtros más importantes para categoría pasabajos. El resto de los filtros se podrá obtener a partir de las transformaciones espectrales pertinentes (se explica más adelante)

3.17 Filtros análogos

- Aproximación de Butterworth de orden N: éstos tienen una magnitud en frecuencia tal que

$$|H_a(j\Omega)|^2 = \frac{1}{1+(\Omega/\Omega_c)^{2N}}$$

  - Sus primeras 2N-1 derivadas en 0 son nulas: magnitud máximamente plana en Ω=0.
  - Ganancia:

$$G(\Omega) = 10log_{10}(|H_a(j\Omega)|^2)dB$$

$$G(0) = 0;\ G(\Omega_c) = 10log_{10}(1/2) \simeq -3dB$$

3.17 Filtros análogos

- Aproximación de Butterworth de orden N: el problema de diseño se resuelve con las ecuaciones

$$|H_a(j\Omega_p)|^2 = \frac{1}{1+(\Omega_p/\Omega_c)^{2N}} = \frac{1}{1+\epsilon^2}$$

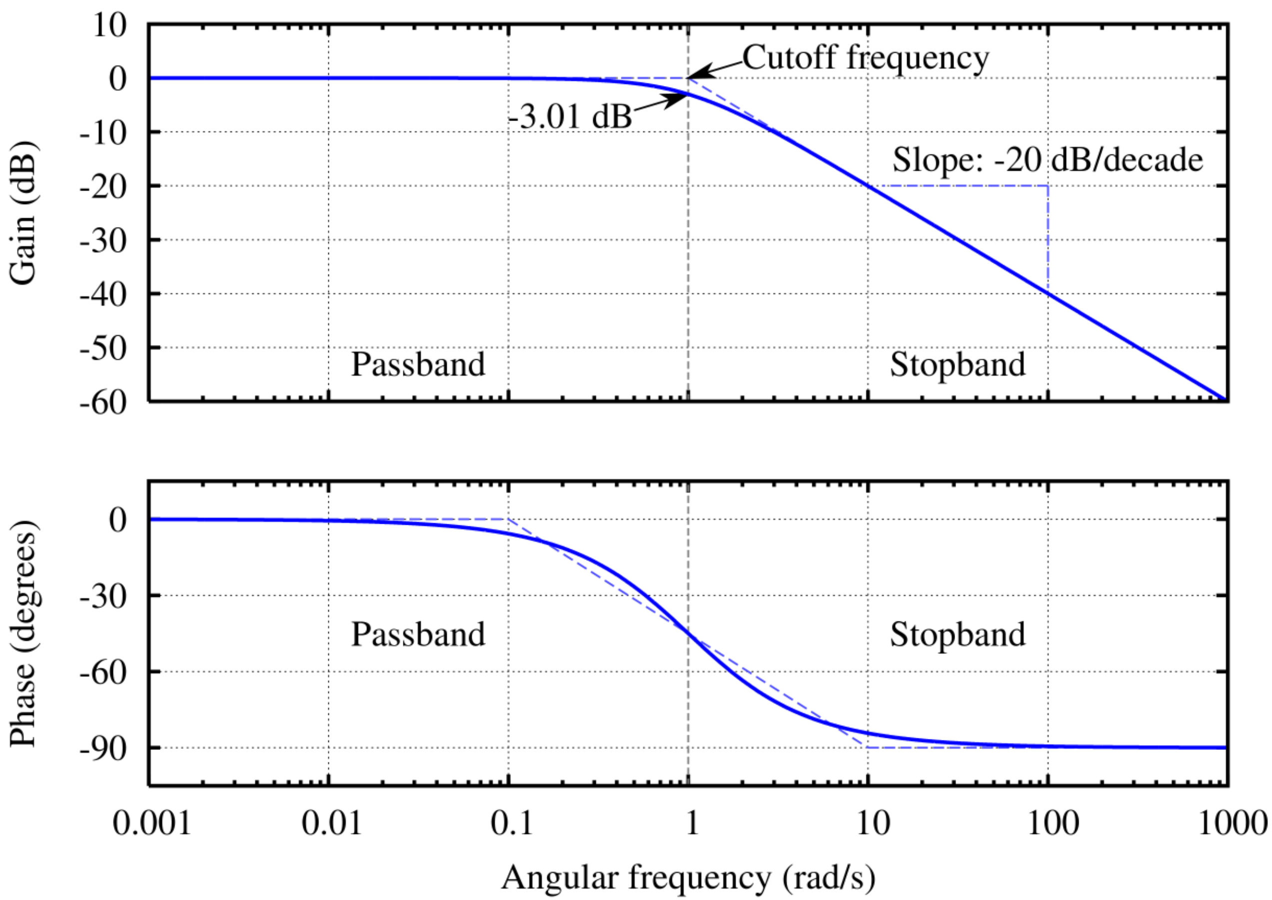
$$|H_a(j\Omega_s)|^2 = \frac{1}{1+(\Omega_s/\Omega_c)^{2N}} = \frac{1}{A^2}$$

- La función de transferencia del filtro es:

$$H_a(s) = \frac{\Omega_c^{\,N}}{\prod_{l=1}^{N}(s-p_l)}; \quad p_l = \Omega_c e^{j[\pi(N+2l-1)/2N]}, \; l = 1,\ldots,N$$

3.17 Filtros análogos

- Aproximación de Butterworth de orden N:



3.17 Filtros análogos

- Aproximación de Chebyshev tipo I orden N: éstos tienen una magnitud en frecuencia tal que

$$|H_a(j\Omega)|^2 = \frac{1}{1+\epsilon^2 T_N^2(\Omega/\Omega_p)}$$

T_N : Son los polinomios de Chebyshev de orden N

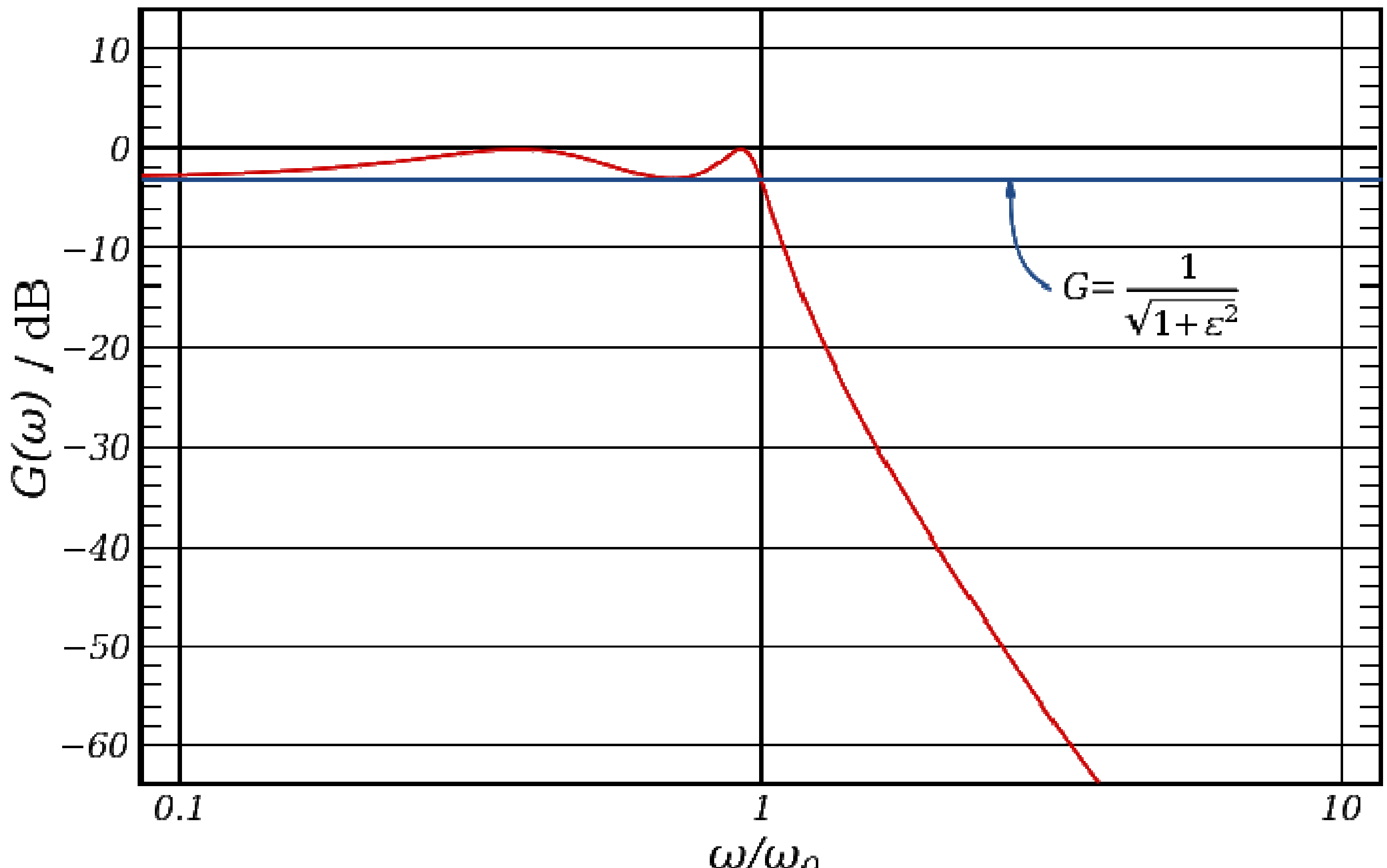
  – Su función de transferencia es

$$H_a(s) = \frac{\Omega_c{}^N}{\prod_{l=1}^{N}(s-p_l)}; \;\; p_l = \sigma_l + j\Omega_l, \; l = 1,\ldots,N$$

  – σ_l y Ω_l tienen una definición complicada y no se abordarán en esta ocasión

3.17 Filtros análogos

- Aproximación de Chebyshev tipo I orden N:



3.17 Filtros análogos

- Aproximación de Chebyshev tipo II orden N: éstos tienen una magnitud en frecuencia tal que

$$|H_a(j\Omega)|^2 = \frac{1}{1-\epsilon^2\left[\dfrac{T_N(\Omega_s/\Omega_p)}{T_N(\Omega_s/\Omega)}\right]^2}$$

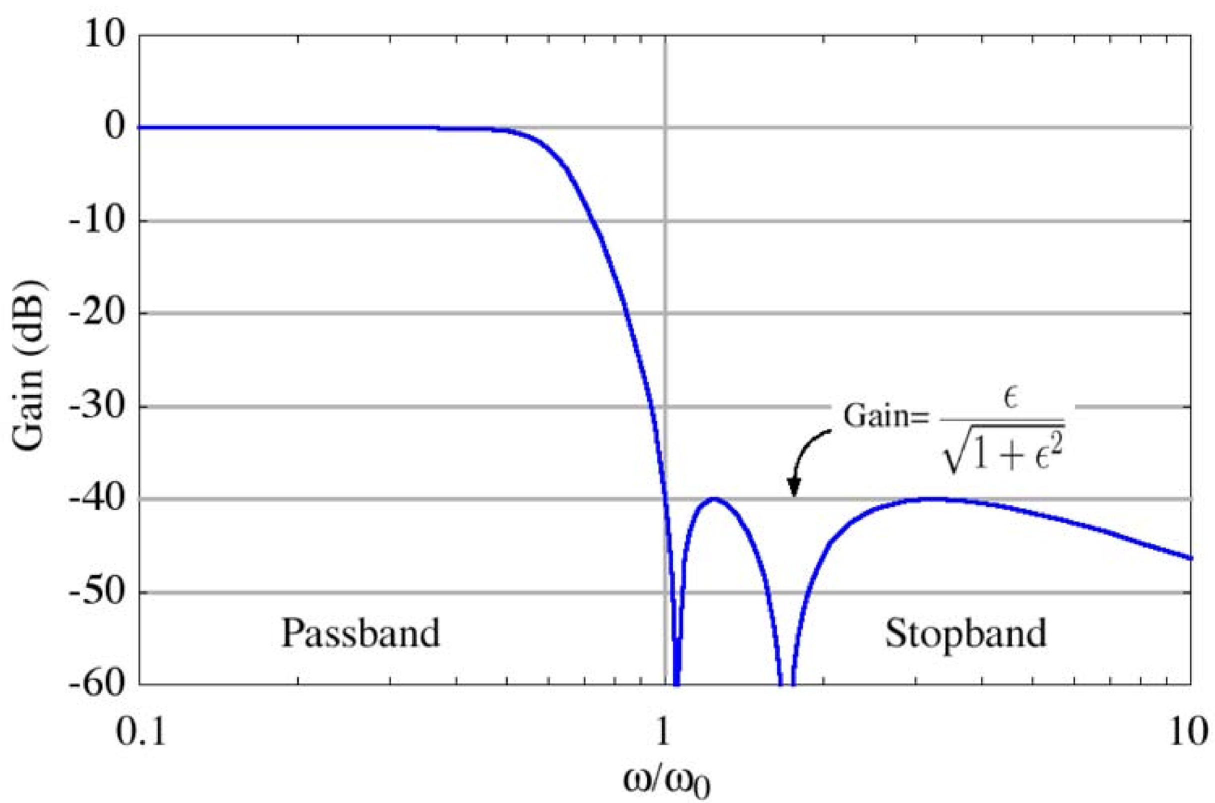
  - Su función de transferencia es

$$H(s) = C_0 \, \prod_{l=1}^{N}(s-z_l) \Big/ \prod_{l=1}^{N}(s-p_l)$$

  - z_l y p_l tienen una definición complicada y no se abordarán en esta ocasión

3.17 Filtros análogos

- <u>Aproximación de Chebyshev tipo II orden N</u>:



3.17 Filtros análogos

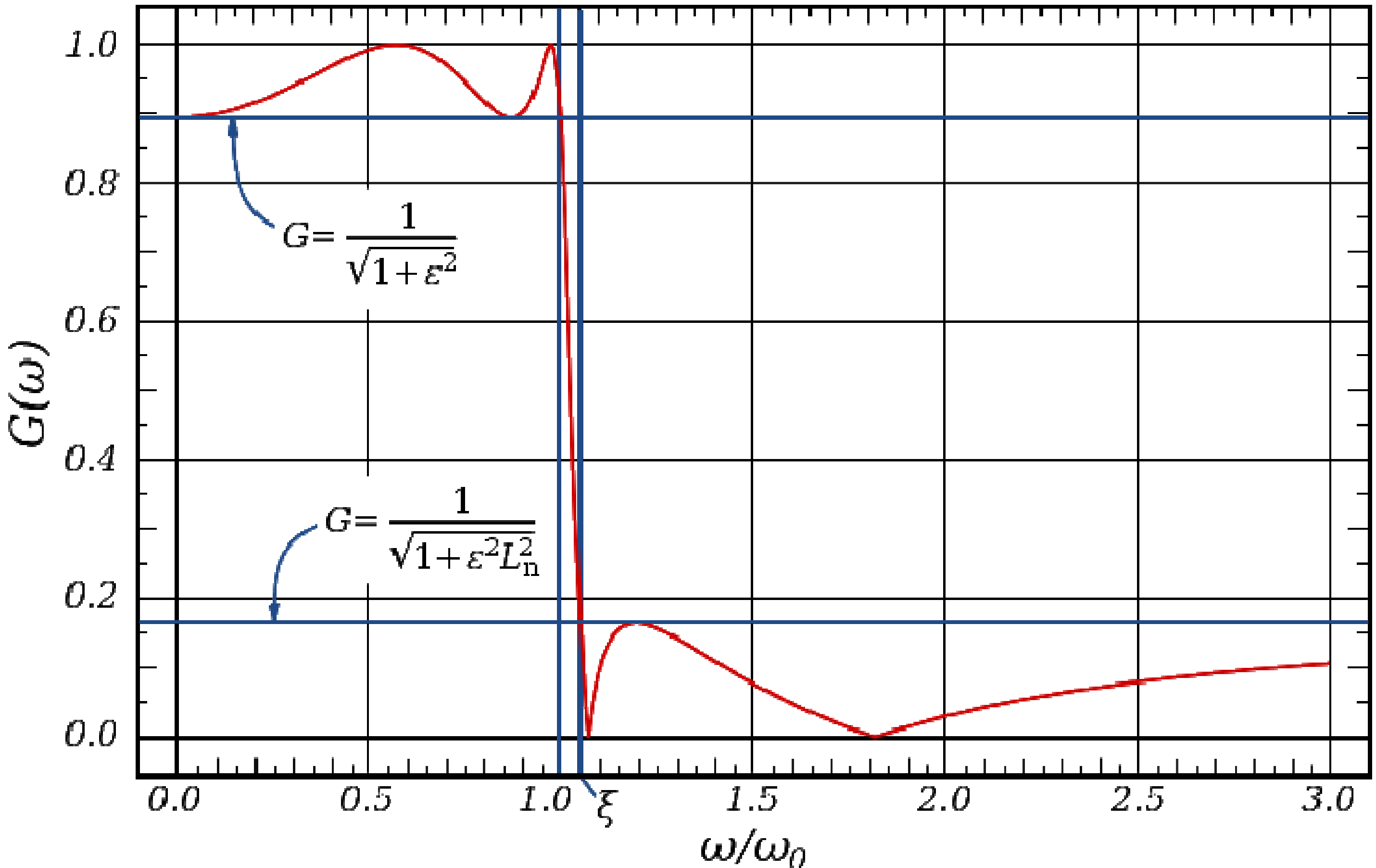
- Aproximación elíptica (Cauer) de orden N: éstos tienen una magnitud en frecuencia tal que

$$|H_a(j\Omega)|^2 = \frac{1}{1+\epsilon^2 R_N^2(\Omega/\Omega_p)}$$

  - R_N es una función racional que satisface $R_N(1/\Omega) = 1/R_N(\Omega)$, con las raíces del numerador en el intervalo (0,1) y las del denominador en el intervalo $(1, \infty)$

3.17 Filtros análogos

- Aproximación elíptica (Cauer) de orden N :



3.17 Filtros análogos

- Para pasar de los pasabajos recién vistos a los otros filtros, se aplican las siguientes transformaciones (s será el espacio del pasabajos original y $\hat{s}$ el del filtro deseado) :
  - Para pasa altos con frecuencia de borde $\hat{\Omega}_p$: $s = \dfrac{\Omega_p \hat{\Omega}_p}{\hat{s}}$
  - Para pasa bandas de frecuencia de borde menor $\hat{\Omega}_{p1}$ y mayor $\hat{\Omega}_{p2}$:

$$s = \Omega_p \frac{\hat{s}^2 + \hat{\Omega}_{p1}\hat{\Omega}_{p2}}{\hat{s}(\hat{\Omega}_{p2} - \hat{\Omega}_{p1})}$$

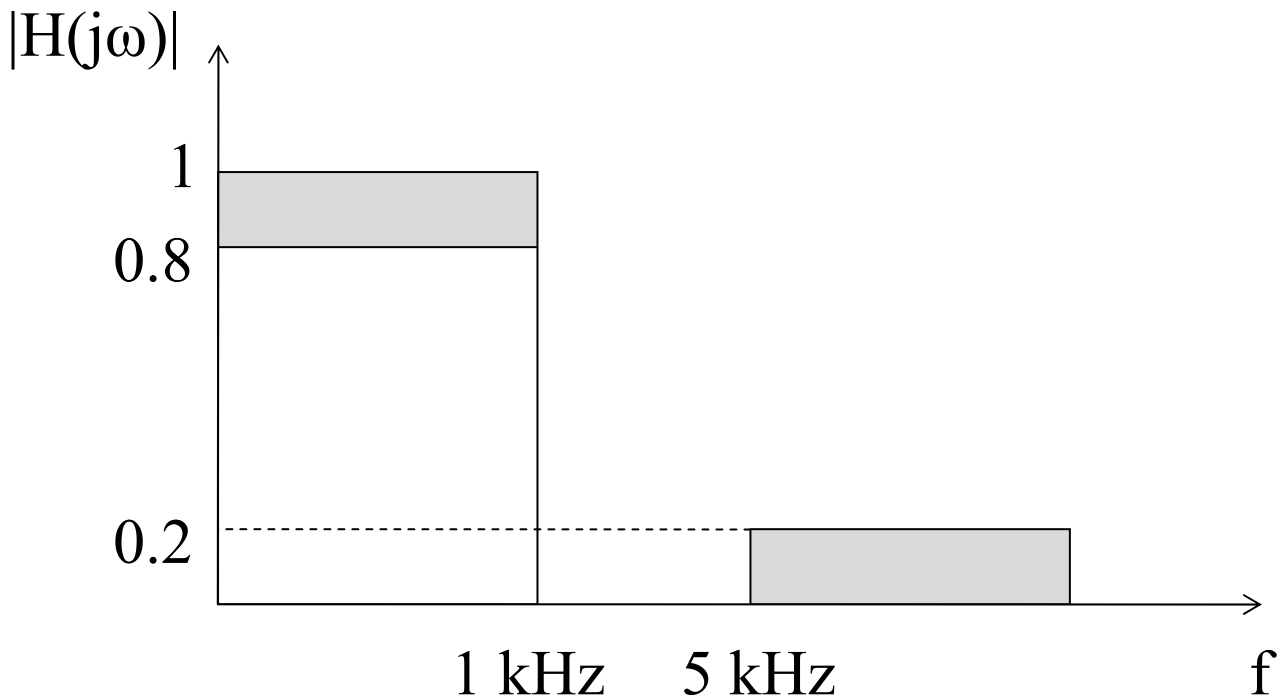
3.17 Filtros análogos

– Para rechaza bandas de banda de paso inferior $\hat{\Omega}_{s1}$ y superior $\hat{\Omega}_{s2}$:

$$s = \Omega_s \frac{\hat{s}(\hat{\Omega}_{s2} - \hat{\Omega}_{s1})}{\hat{s}^2 + \hat{\Omega}_{s1}\hat{\Omega}_{s2}}$$

3.17 Filtros análogos

- Ejemplo: obtenga la función de transferencia del filtro de las siguientes especificaciones:



3.17 Filtros análogos

- Sol: vamos a usar filtro Butterworth:

$$\frac{1}{1+\epsilon^2}=0,64=\frac{1}{1+(\Omega_p/\Omega_c)^{2N}}\Rightarrow(\Omega_p/\Omega_c)^{2N}=0,5625$$

$$\frac{1}{A^2}=0,04=\frac{1}{1+(\Omega_s/\Omega_c)^{2N}}\Rightarrow(\Omega_s/\Omega_c)^{2N}=24$$

$$(\Omega_s/\Omega_p)^{2N}=(5)^{2N}=42,\overline{6}\Rightarrow N=1,1\overline{6}\approx 2$$

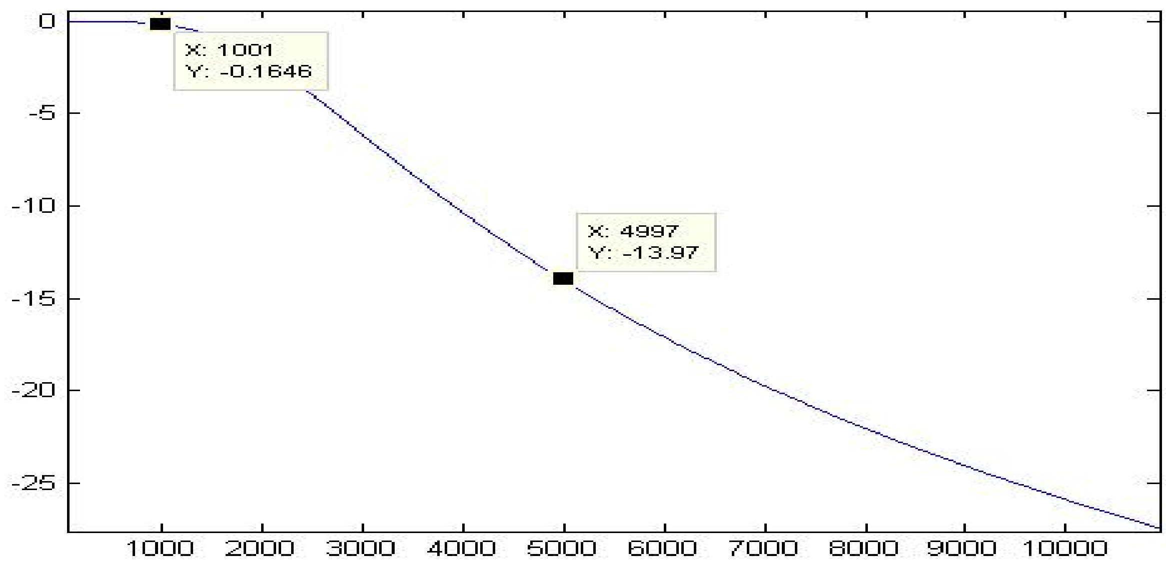
$$\Rightarrow\Omega_c=\Omega_s/2.213=14.193,7\left[{}^{rad}\!/_{s}\right]$$

3.17 Filtros análogos

– La solución entonces es:

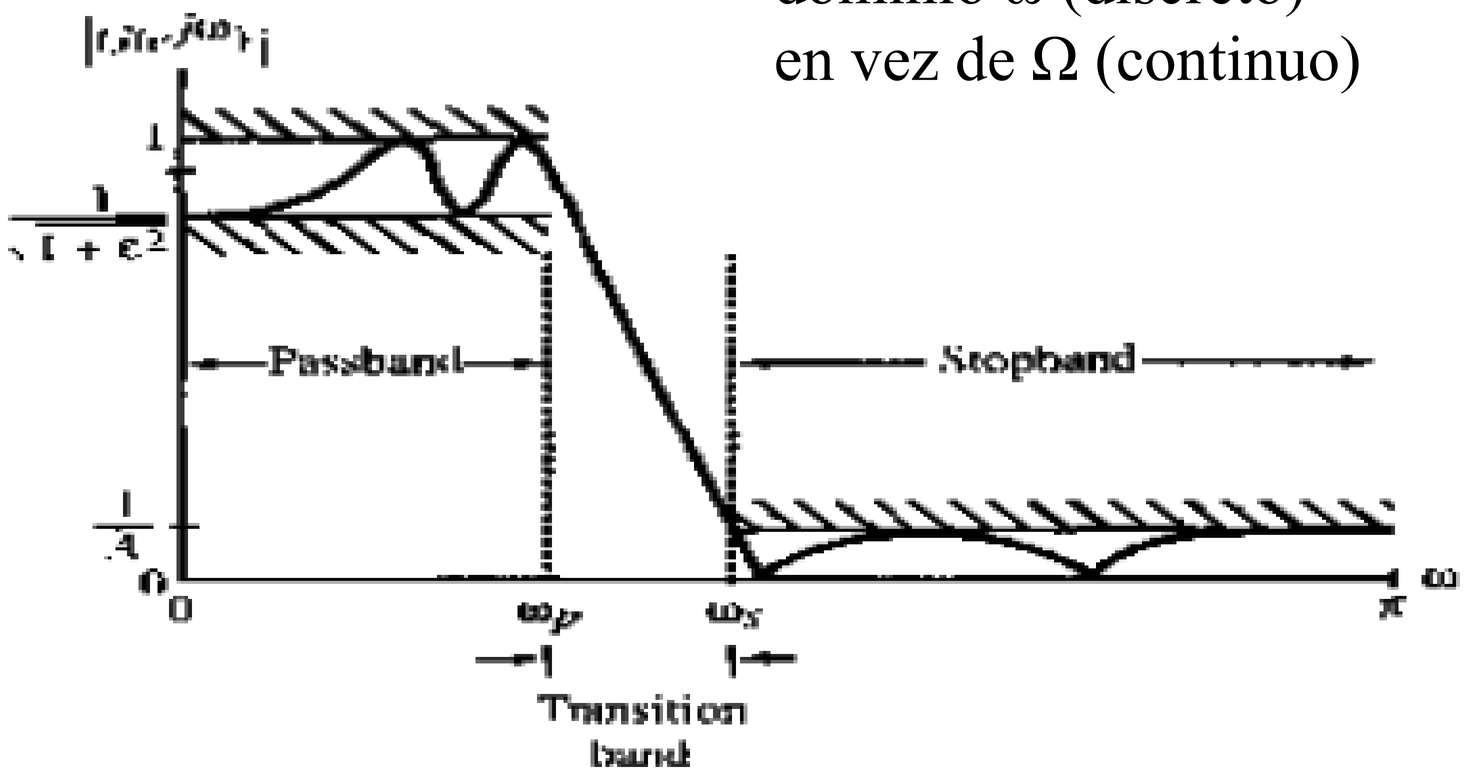
$$\therefore H_a(s) = \frac{201.4624}{(s - 14.193,7e^{j3\pi/4})(s - 14.193,7e^{j5\pi/4})}$$

– Y su diagrama de bode:



3.18 Diseño de filtros digitales a partir de diseño de filtro análogo

– Para los filtros digitales se tienen las mismas definiciones que para filtros análogos, usando el dominio ω (discreto) en vez de Ω (continuo)



3.18 Diseño de filtros digitales a partir de diseño de filtro análogo

- En general, los filtros se desarrollan en términos de las frecuencias angulares normalizadas, ωp y ωs

$$\omega_p = \frac{\Omega_p}{F_T} = 2\pi F_p T$$

$$\omega_s = \frac{\Omega_s}{F_T} = 2\pi F_s T$$

Donde Fp y Fs son las frecuencias en Hz de borde en Hz del filtro digital, FT es la frecuencia de muestreo y T el periodo de muestreo.

3.18 Diseño de filtros digitales a partir de diseño de filtro análogo

Selección del tipo de filtro: ¿FIR o IIR?

FIR

- Polinomio en z^{-1}
- Siempre es estable
- Fase lineal implica menor distorsión, además de reducción de términos, por simetría:

$$h(n) = \pm h(N-n)$$

- Requiere muchos órdenes para una respuesta pedida

IIR

- Función racional en z^{-1}
- Se debe ser cuidadoso con la estabilidad
- En general no tiene fase lineal
- Requiere pocos órdenes para una respuesta pedida (del orden de 10 o más veces menos que el FIR=> computacionalmente más eficiente, aunque el resto del sistema puede entorpecer la eficiencia)

3.18 Diseño de filtros digitales a partir de diseño de filtro análogo

- Enfoques básicos al diseño de filtros digitales:
  - Para los filtros IIR, ya sabemos la forma de la respuesta:

$$G(z) = \sum_{k=0}^{M} a_k z^{-k} \Big/ \sum_{k=0}^{N} b_k z^{-k}$$

  - Podemos aprovechar lo que sabemos de filtros análogos para diseñar los filtros digitales…

3.18 Diseño de filtros digitales a partir de diseño de filtro análogo

- Enfoques básicos al diseño de filtros digitales
  - …pues en filtros análogos $H(s) = N(s) / D(s)$
  - Es interesante este enfoque pues:
    - Las aproximaciones análogas son altamente avanzadas
    - Existen tablas y herramientas para su diseño

3.18 Diseño de filtros digitales a partir de diseño de filtro análogo

- Enfoques básicos al diseño de filtros digitales:
  - Para filtros FIR, la orientación es diferente, justamente por la forma de su función de transferencia:
  $$G(z)=\sum_{n=0}^{N} h(n)z^{-k}$$
  - En este caso es claramente mejor el enfoque dado por la serie de Fourier con funciones ventana

3.18 Diseño de filtros digitales a partir de diseño de filtro análogo

- Estimación del orden del filtro:
  - Para los filtros de tipo IIR, como comenzaremos a partir de filtros análogos, obtendremos dependiendo del filtro análogo que queramos utilizar y las especificaciones de éste el orden N del filtro. Desarrollando se logra obtener:
    - Para filtro Butterworth

$$N = \frac{1}{2}\frac{log_{10}[(A^2-1)/\epsilon^2]}{log_{10}(\Omega_s/\Omega_p)} = \frac{log_{10}(1/k_1)}{log_{10}(1/k)}$$

3.18 Diseño de filtros digitales a partir de diseño de filtro análogo

- Estimación del orden del filtro:
  - Para filtros Chebyshev tipos I y II

$$N = \frac{\cosh^{-1}(\sqrt{A^2 - 1}/\epsilon)}{\cosh^{-1}(\Omega_s / \Omega_p)} = \frac{\cosh^{-1}(1/k_1)}{\cosh^{-1}(1/k)}$$

  - Para filtro elíptico o Cauer se hace una estimación

$$N \simeq \frac{2\log_{10}(4/k_1)}{\log_{10}(1/\rho)} \qquad \rho_0 = \frac{1-\sqrt{k'}}{2(1+\sqrt{k'})}$$

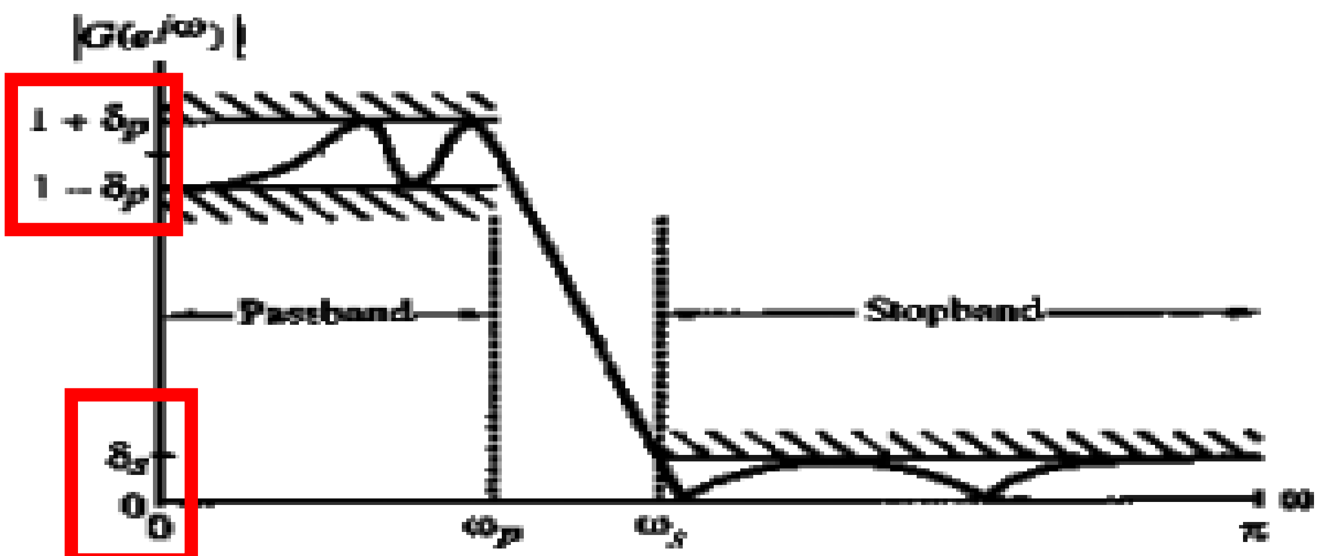
$$\rho = \rho_0 + 2\rho_0^5 + 15\rho_0^9 + 150\rho_0^{13} \qquad k' = \sqrt{1-k^2}$$

3.18 Diseño de filtros digitales a partir de diseño de filtro análogo

- Estimación del orden del filtro:
  - Para filtros FIR el tratamiento es diferente. No existe una forma estándar de calcular el mínimo orden para satisfacer las condiciones del filtro, pero sí hay aproximaciones que varios autores han descrito. Los detalles se pueden buscar en bibliografía.
  - Para no complicar las cosas, vamos a definir un par de parámetros extra:

3.18 Diseño de filtros digitales a partir de diseño de filtro análogo

- Estimación del orden del filtro:
  - Rizo pico de la banda de paso: δ_p
  - Rizo pico de la banda de rechazo: δ_s
  - De este modo generalizamos el análisis anterior a cuando la ganancia máxima en la banda de paso es mayor que 1



3.18 Diseño de filtros digitales a partir de diseño de filtro análogo

- Estimación del orden del filtro:
  - Una primera aproximación es, entonces (fórmulas de Kaiser)

$$N \approx \frac{-20log_{10}(\sqrt{\delta_p\delta_s}) - 13}{14.6(\omega_s - \omega_p)/2\pi}$$

  - Observaciones:
    - N es inversamente proporcional al ancho de la banda de transición (más abrupto=> más términos) y no depende de la posición de ésta
    - Se puede sacrificar δ_p por δ_s manteniendo el número de términos (puede ser interesante en alguna aplicación)
    - La fórmula no funciona muy bien con filtros pasabanda demasiado anchos o demasiado angostos

3.18 Diseño de filtros digitales a partir de diseño de filtro análogo

- Estimación del orden del filtro:
  - Aproximación para pasabanda angosto

$$N \approx \frac{-20log_{10}(\delta_s)+0.22}{(\omega_s-\omega_p)/2\pi}$$

  - Aproximación para pasabanda ancho

$$N \approx \frac{-20log_{10}(\delta_s)+5.94}{27(\omega_s-\omega_p)/2\pi}$$

3.18 Diseño de filtros digitales a partir de diseño de filtro análogo

– Fórmula de Hermann et al

$$N \approx \frac{D_\infty(\delta_p,\delta_s) - F(\delta_p,\delta_s)[(\omega_s-\omega_p)/2\pi]^2}{(\omega_s-\omega_p)/2\pi} \qquad \delta_p \geq \delta_s$$

$$D_\infty(\delta_p,\delta_s) = [a_1(\log_{10}\delta_p)^2 + a_2(\log_{10}\delta_p) + a_3]\log_{10}\delta_s - [a_4(\log_{10}\delta_p)^2 + a_5(\log_{10}\delta_p) + a_6]$$

$$F(\delta_p,\delta_s) = b_1 + b_2[\log_{10}\delta_p - \log_{10}\delta_s]$$

$a_1 = 0.005309$ $a_2 = 0.07114$ $a_3 = -0.4761$

$a_4 = 000266$ $a_5 = 05921$ $a_6 = 0.4278$

$b_1 = 1101217$ $b_2 = 051244$

3.18 Diseño de filtros digitales a partir de diseño de filtro análogo

- Escalamiento de la función de transferencia digital
  - Debe escalarse la función de transferencia discreta antes de implementarse (se quiere un filtro, no un amplificador).
  - Para esto se multiplica la función de transferencia por $K = 1/|G_{max}(e^{j\omega})|$, en que $|G_{max}(e^{j\omega})|$ es el máximo valor de la magnitud de la respuesta de frecuencia del filtro en el rango $0 \leq \omega \leq \pi$.

3.19 Diseño de filtro digital IIR en base a diseño de filtro analógico

- Objetivo: dado un filtro analógico con función de transferencia H(s), se quiere obtener un filtro digital de similares características.
- Estrategia: cambiar el espacio s al espacio z de manera de simplemente evaluar el filtro análogo en s(z) y obtener así H(z)

3.19 Diseño de filtro digital IIR en base a diseño de filtro analógico

- Aproximación bilineal:
  - Considérese la función de transferencia

$$G(s)=\frac{1}{s}$$

$$\Rightarrow Y(s)=\frac{1}{s}X(s)\Rightarrow y(t)=y(t_0)+\int_{t_0}^{t}x(\tau)d\tau$$

$$\therefore t_0=kT, t=kT+T\Rightarrow \mathrm{y(kT+T)=y(kT)+}\int_{kT}^{kT+T}x(\tau)d\tau$$

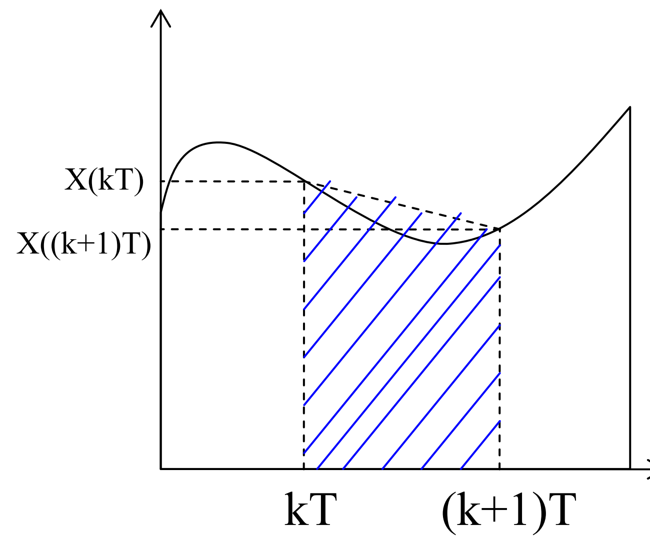
3.19 Diseño de filtro digital IIR en base a diseño de filtro analógico

- Aproximación bilineal:
  - Se aproxima el integrando por una recta

$$\hat{y}((k+1)T) = \hat{y}(kT) + \frac{T}{2}\{x(kT) - x((k+1)T)\}$$

$$(z-1)\hat{Y}(z) = \frac{T}{2}(z+1)X(z)$$

$$\therefore\ H(z) = \frac{T}{2}\left(\frac{z+1}{z-1}\right) \approx G(s) \Rightarrow s \approx \frac{2}{T}\left(\frac{z-1}{z+1}\right)$$



$$\therefore\ s = \frac{2}{T}\left(\frac{z-1}{z+1}\right)$$ Es la aproximación bilineal

3.19 Diseño de filtro digital IIR en base a diseño de filtro analógico

- Con la aproximación bilineal se logra:
  - Trazar el eje $j\Omega$ del dominio s en el círculo unitario del dominio z
  - Convertir una función de transferencia estable análoga en una función de transferencia estable discreta.

3.19 Diseño de filtro digital IIR en base a diseño de filtro analógico

- Ejemplo: convierta el filtro análogo con función de transferencia $G(s)=\frac{s+0.1}{(s+0.1)^2+16}$ a un filtro IIR mediante la transformación bilineal. El filtro digital tiene una resonancia en $\omega_r=\frac{\pi}{2}$
- Solución: la transformación bilineal es $s=\frac{2}{T}\left(\frac{z-1}{z+1}\right)=\sigma+j\Omega$

Al considerar la respuesta en frecuencia, s=jΩ, $z=e^{j\omega}$

$$\Rightarrow \Omega=\frac{2}{T}\tan\frac{\omega}{2}$$

3.19 Diseño de filtro digital IIR en base a diseño de filtro analógico

- Aquí, la frecuencia de resonancia del filtro análogo es Ω=4. Como se quiere $\omega_r = \frac{\pi}{2}$, imponemos la condición en la fórmula para Ω, de manera de obtener T:

$$\Rightarrow T = \frac{2}{\Omega} \tan \frac{\pi}{4} = \frac{1}{2}$$

$$\therefore H(z) = G(s)\Big|_{s=4\left(\frac{z-1}{z+1}\right)} = \frac{0.128 + 0.006 z^{-1} - 0.122 z^{-2}}{1 + 0.0006 z^{-1} + 0.975 z^{-2}}$$

3.20 Diseño de un filtro digital pasabajo IIR

- De las especificaciones de un filtro digital y usando la transformada inversa se obtienen las especificaciones para un filtro analógico pasa bajo.
- Con las especificaciones del filtro analógico se determina la función de transferencia analógica $H_a(s)$.
- Usando la transformación bilineal, la función de transferencia analógica $H_a(s)$ se transforma en una función digital $G(z)$.

3.20 Diseño de un filtro digital pasabajo IIR

- Considere el diseño de un filtro pasabada IIR digital, con las siguientes características:
  - Frecuencia de borde del pasabanda $\omega_p = 0.25\pi$.
  - Rizado de pasa banda no mayor a 0.5 dB.
  - La minima atenuación rechaza banda a la frecuencia de borde del rechaza banda ω_s de 0.25π es 15 dB.

- Si $\left|G(e^{j0\pi})\right| = 1$, se requiere que:

$$20\log_{10}\left|G(e^{j0.25\pi})\right| \geq -0.5dB \quad \text{y} \quad 20\log_{10}\left|G(e^{j0.55\pi})\right| \leq -15dB$$

3.20 Diseño de un filtro digital pasabajo IIR

- De las frecuencias digitales de borde de la banda se obtienen las frecuencias analógicas, las que están dadas por:

$$\Omega_p = tg\left(\frac{\omega_p}{2}\right) = 0.4142 \qquad \Omega_s = tg\left(\frac{\omega_s}{2}\right) = 1.1708$$

- La razón de transición inversa nos dice que:

$$\frac{1}{k} = \frac{\Omega_s}{\Omega_p} = 2.8266$$

3.20 Diseño de un filtro digital pasabajo IIR

- Del rizado de 0.5dB se obtiene ε^2 de la ecuación:

$$\alpha_p = -20\log\left(\frac{1}{\sqrt{1+\varepsilon^2}}\right) \qquad \Rightarrow \qquad \varepsilon^2 = 0.1220$$

- De la atenuación minima de 15 dB se obtiene A:

$$\alpha_s = -20\log\left(\frac{1}{A}\right) \qquad \Rightarrow \qquad A^2 = 31.6227$$

3.20 Diseño de un filtro digital pasabajo IIR

- Luego la razón de discriminación inversa queda dada por:

$$\frac{1}{k_1} = \frac{\sqrt{A^2 - 1}}{\varepsilon} = 15.8419$$

- Con esta información el orden del filtro es:

$$N = \frac{\log_{10}\left({}^{1}\!/_{k_1} \right)}{\log_{10}\left({}^{1}\!/_{k} \right)} = 2.6586 \approx 3$$

3.20 Diseño de un filtro digital pasabajo IIR

- Con dicho orden y la aproximación de Butterworth se determina la frecuencia de corte a 3 dB.
- Para obtener el menor rizado se usa la frecuencia de borde del pasa-banda y la magnitud minima del pasa-banda, obteniéndose así la siguiente expresión:

$$\left|H_a(j\Omega_p)\right|^2 = \frac{1}{1+\left(\frac{\Omega_p}{\Omega_c}\right)^{2N}} = \frac{1}{1+\varepsilon^2}$$

- Reemplazando los valores, se tiene $\Omega_c = 1.4199 \cdot \Omega_p = 0.5881$

3.20 Diseño de un filtro digital pasabajo IIR

- Ahora se reemplazan los valores calculados en la ecuación

$$H_a(s) = \frac{\Omega_c{}^N}{\prod_{l=1}^{N}(s - p_l)} \qquad p_l = \Omega_c e^{j[\pi(N+2l-1)/2N]} \quad l = 1, \ldots, N$$

- La función es:

$$H_a(s) = \frac{0.2034}{(s+0.5881)(s^2+0.5881s+0.3459)}$$

3.20 Diseño de un filtro digital pasabajo IIR

- Finalmente se aplica la transformación bilineal a la ecuación anterior, obteniéndose de esta forma la función deseada:

$$G(z) = H_a(s)\Big|_{s=\frac{1-z^{-1}}{1+z^{-1}}}$$

$$G(z) = \frac{0.0662(1+z^{-1})^3}{(1-0.2593\cdot z^{-1})(0.3917\cdot z^{-2} - 0.6762\cdot z^{-1} + 1)}$$

- Este procedimiento se puede generalizar para otros tipos de filtro en 5 pasos mostrados a continuación.

3.20 Diseño de un filtro digital IIR

- 1.- Con la ecuación $\Omega = tg\left(\frac{\omega}{2}\right)$ se obtiene la frecuencia de un filtro analógico del mismo tipo al filtro digital deseado.
- 2.- Convertir las especificaciones de frecuencia del filtro digital en una de un filtro pasa bajo analógico.
- 3.- Diseñar el filtro pasa-bajo analógico usando el método estudiado en la sección 3.17.
- 4.- Usando el inverso de la transformación de frecuencia, convertir la función del filtro pasa-bajo analógico en la función $H_D(s)$.
- 5.- Usar la transformación bi-lineal $s = \frac{2}{T}\left(\frac{1-z^{-1}}{1+z^{-1}}\right)$ en $H_D(s)$ para obtener la función de transferencia $G_D(z)$

3.20 Diseño de un filtro digital IIR

- Ejemplo: Diseñar un filtro digital Pasaalto Butterworth IIR

- Las especificaciones del filtro deseado son:
  - Frecuencia de borde del pasa-banda $F_p = 700Hz$
  - Frecuencia de borde del rechaza-banda $F_s = 500Hz$
  - Rizado del pasa-banda 1dB.
  - Atenuación minima del rechaza-banda 32dB.
  - Frecuencia de muestreo $F_T = 2000Hz$

3.20 Diseño de un filtro digital IIR

- 1.- Las frecuencias de borde normalizadas son:

$$\omega_p = \frac{2\pi F_p}{F_T} = 0.7\pi \qquad \omega_s = \frac{2\pi F_s}{F_T} = 0.5\pi$$

Las frecuencias del filtro análogo son:

$$\hat{\Omega}_p = tg\left(\frac{\omega_p}{2}\right) = 1.9626 \qquad \hat{\Omega}_s = tg\left(\frac{\omega_s}{2}\right) = 1$$

- 2.- Con la transformación $\Omega = -\frac{\Omega_p \hat{\Omega}_p}{\hat{\Omega}}$ se obtienen las frecuencias para un filtro pasa-bajo a partir de las frecuencias obtenidas para el filtro pasa alto.

$$\Omega_s = 1.9626 \qquad \Omega_p = 1$$

3.20 Diseño de un filtro digital IIR

- 3.- Diseño del filtro pasa-bajo:
  - Rizado 1 dB => $\alpha_p = -20\log\left(1 \big/ \sqrt{1+\varepsilon^2} \right)$ => $\varepsilon^2 = 0.2589$
  - Atenuación minima 32dB => $\alpha_s = -20\log\left(1 \big/ A \right)$ => $A^2 = 1584.8932$
  - Razón de transición inversa => $\dfrac{1}{k} = 1.9626$
  - Razón de discriminación => $\dfrac{1}{k_1} = 78.2162$
  - Orden del filtro => $N = 6.4650 \approx 7$
  - Frecuencia de corte => $\left| H_a(j\Omega_p) \right| = \dfrac{1}{1+\left(\dfrac{\Omega_p}{\Omega_c} \right)^{2N}} = \dfrac{1}{1+\varepsilon^2}$ => $\Omega_c = 1.1013$

3.20 Diseño de un filtro digital IIR

- Con los datos obtenidos los polos quedan:

$$p_1 = -0.2451 + j1.0737 \qquad p_5 = -0.9922 - j0.4778$$
$$p_2 = -0.6866 + j0.8610 \qquad p_6 = -0.6866 - j0.8610$$
$$p_3 = -0.9922 + j0.4778 \qquad p_7 = -0.2451 - j1.0737$$
$$p_4 = -1.1013$$

- La función de transferencia queda:

$$H_a(s) = \frac{1.9649}{(s+1.1013)(s^6 + 3.8479s^5 + 8.0096s^4 + 10.6697s^3 + 9.7145s^2 + 5.6605s + 1.7842)}$$

3.20 Diseño de un filtro digital IIR

- 4.- Convertir la función del filtro pasa-bajo analógico en la función digital, usando la transformación espectral.

$$s = \frac{1.9626}{\hat{s}}$$

Luego la función $H_D(s)$ queda:

$$H_D(s) = \frac{s^7}{(s+1.7821)(s^6+6.2265s^5+20.9725s^4+45.2077s^3+666045s^2+62.7984s+32.0300)}$$

3.20 Diseño de un filtro digital IIR

- 5.- Obtener la función de transferencia final con una frecuencia de muestreo $F_T = 2000Hz$
  La transformada bilineal queda, $s = 4000\left(\dfrac{1-z^{-1}}{1+z^{-1}}\right)$

  Finalmente:

$$H_D(s) = \frac{0.9980\left(z-1\right)^7}{\left(z-0.9991\right)\left(z^6-5.9969z^5+14.9844z^4-19.9689z^3+14.9689^2-5.9845z+9969\right)}$$

3.20 Diseño de un filtro digital IIR

- A su vez se puede tener un segundo método de diseño; este método solo difiere en los dos últimos pasos del primer método de diseño, los cuales son:
- 4.- Usar la transformación bilineal $s = \frac{2}{T}\left(1 - z^{-1} \big/ 1 + z^{-1}\right)$

  para convertir la función del filtro pasa-bajo analógico en la función $G_{LP}(z)$ de un filtro digital IIR.
- 5.- Transformar $G_{LP}(z)$ en la función de transferencia digital $G_D(z)$ usando la transformada espectral apropiada para obtener la función de transferencia.

3.21 Recorte de señales por funciones ventana

- Para obtener ciertas características en ciertos filtros (como un espectro en frecuencia acotado) se requieren considerar secuencias infinitas (en este caso, una señal infinitamente larga en el tiempo)
- Es imposible considerar funciones infinitas en la práctica => se utilizan sólo algunos términos
- Simplemente dejar de considerar algunos de ellos (lo cual equivale a multiplicar los términos por una función rect(·)) induce distorsiones (en el dual, se hace convolución, en este ejemplo, con un Sa(·))
- Se utilizan las funciones ventana para ponderar las secuencias, de manera de inducir menos lóbulos laterales

3.21 Recorte de señales por funciones ventana

- Funciones ventana
  - Ventana rectangular

$$w(t) = \begin{cases} 1 & |t| < \tau/2 \\ 0 & otro \end{cases}$$

$$W(f) = \tau Sa(\frac{2\pi f \tau}{2})$$

  - Ventana triangular

$$w(t) = \begin{cases} 1 - \dfrac{2|t|}{\tau} & |t| \leq \tau/2 \\ 0 & otro \end{cases}$$

$$W(f) = \frac{\tau}{2}[Sa(\frac{\pi f \tau}{2})]^2$$

3.21 Recorte de señales por funciones ventana

- Funciones ventana
  - Ventana Hanning

$$w(t)=\begin{cases}\cos^2\left(\dfrac{\pi t}{\tau}\right)=\dfrac{1+cos\left(\dfrac{2\pi t}{\tau}\right)}{2} & |t|\leq \tau/2\\ 0 & otro\end{cases}$$

$$W(f)=\frac{\tau}{2}Sa(\pi f\tau)\left[\frac{1}{1-(f\tau)^2}\right]$$

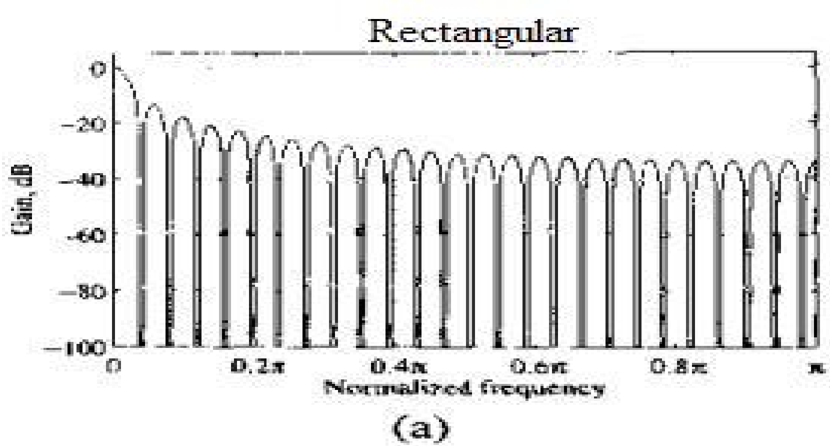
  - Ventana Hamming

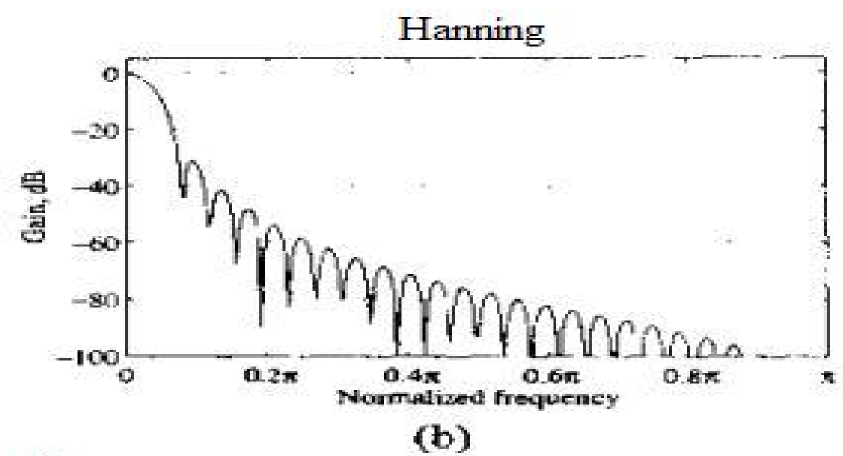
$$w(t)=\begin{cases}0,54+0,46\cos\left(\dfrac{2\pi t}{\tau}\right) & |t|\leq \tau/2\\ 0 & otro\end{cases}$$

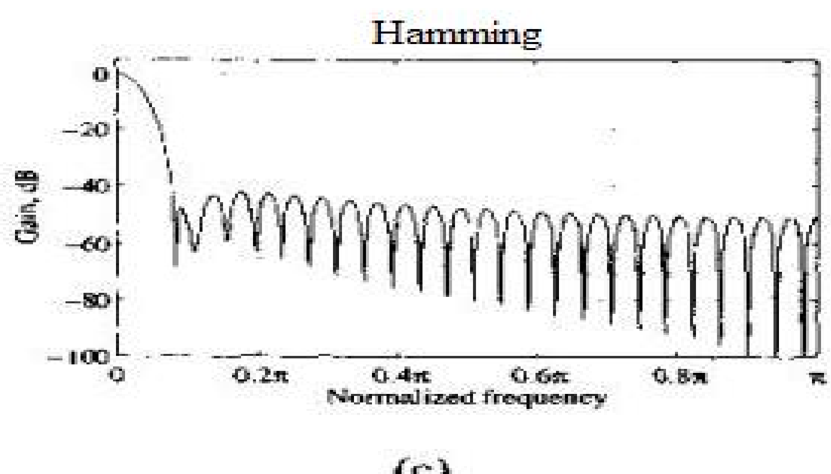
$$W(f)=\frac{\tau sen(\pi f\tau)}{\pi f\tau}\left[\frac{0,54-0,08(f\tau)^2}{1-(f\tau)^2}\right]$$

3.21 Recorte de señales por funciones ventana

- Funciones ventana: ganancias en dB





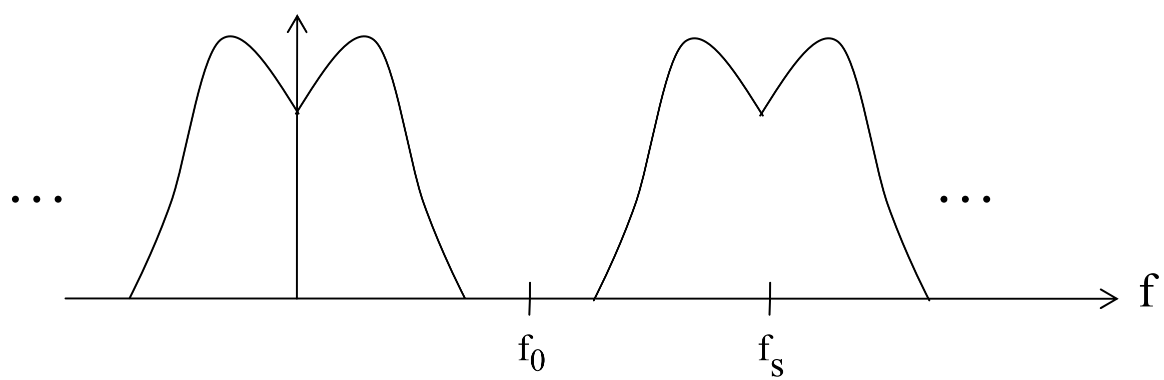


3.21 Recorte de señales por funciones ventana

- Funciones ventana:
  - Mientras más rápido decaigan los lóbulos laterales, mejor la ventana (induce menos distorsión a la señal que se recorta), sopesando el uso de recursos y tiempo de cálculo, tan vital en aplicaciones digitales. Mientras más compleja la ventana, mayor tiempo de procesamiento puede requerir

3.22 Método de la serie de Fourier para el diseño de filtros FIR

- Sea la respuesta de amplitud de un filtro que es periódica en frecuencia. Se la puede representar utilizando la serie de Fourier:



$$A(f) = \sum_{m=-M}^{M} C_m e^{j2\pi mTf} \quad \text{con } C_m = C_{-m} = \frac{2}{f_s} \int_0^{\frac{f_s}{2}} A_d(f) \cos(2\pi mTf)\, df$$

3.22 Método de la serie de Fourier para el diseño de filtros FIR

- $A(f)$ resulta de aproximar la serie de Fourier de $A_d(f)$ (la respuesta del filtro deseada) utilizando solo algunos términos : se ha utilizado la ventana rect(·) para hacer el recorte.
- Considerando $z = e^{j\omega T}$, $H_1(Z) = \sum_{m=-M}^{M} C_m z^m$, el cual es un filtro no causal. Se retarda:

$$H(z) = z^{-m} H_1(z) = \sum_{m=-M}^{M} C_m z^{m-M}$$

- Haciendo i=M-m

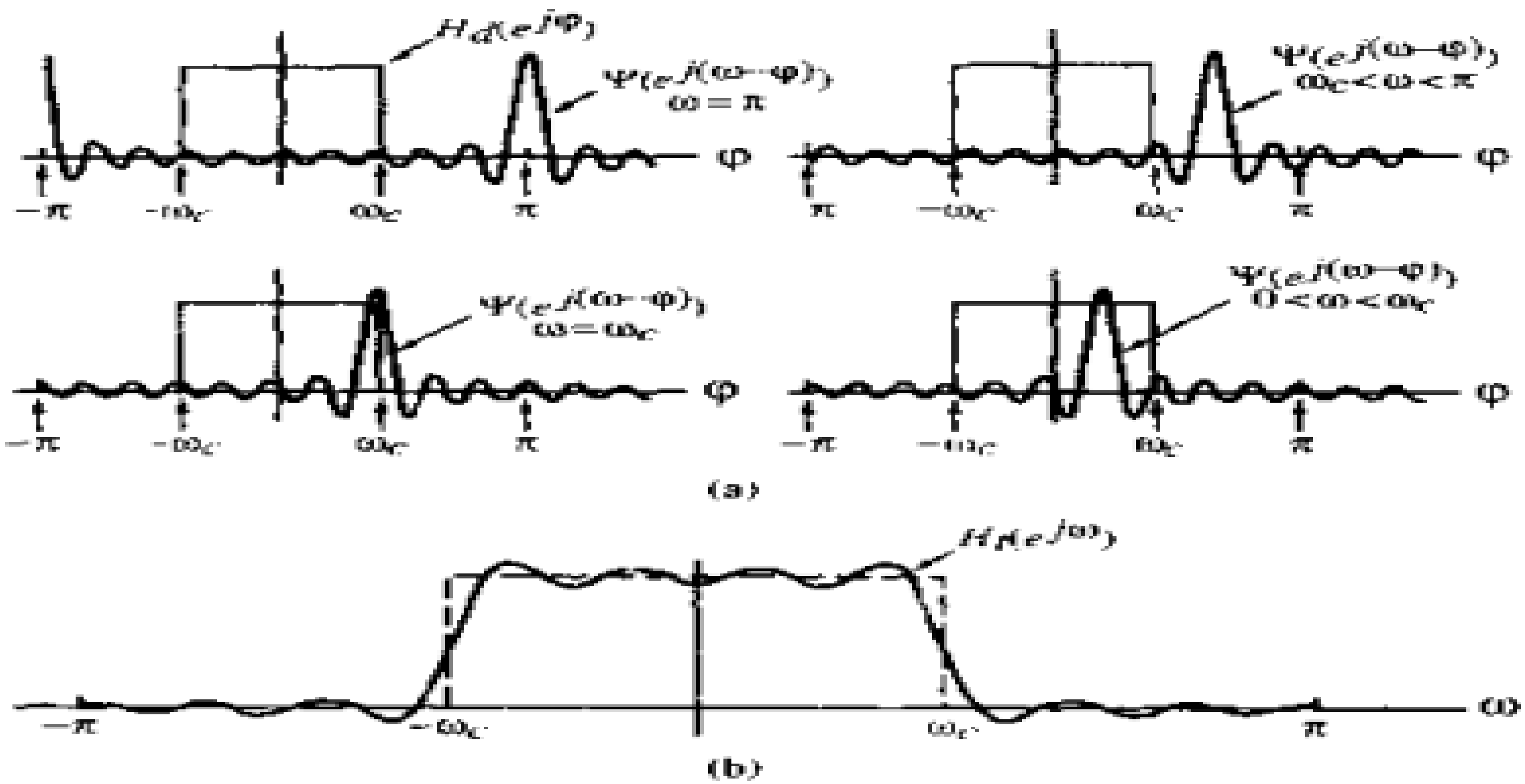
$$H(z) = \sum_{i=0}^{2M} a_i z^{-i}, \text{ con } a_i = C_{M-i}$$

3.22 Método de la serie de Fourier para el diseño de filtros FIR

- Para utilizar otras ventanas, basta utilizar, en vez del coeficiente C_m anterior, el coeficiente $C_{m'} = W_m C_m$, con W_m el peso asociado a la función ventana respectiva.
- Para realizar análisis comparativos puede normalizarse el cálculo realizado con respecto a $f_o = f_s / 2$. Se define $v = f / f_0$ Luego

$$A(v) = \sum_{m=-M}^{M} C_m e^{jm\pi v} \qquad C_m = \int_0^1 A_d(v) cos(m\pi v) dv$$

3.22 Método de la serie de Fourier para el diseño de filtros FIR



- Efecto de truncar los términos de un filtro ideal utilizando una función ventana rectangular: convolución con un sampling en el espacio de la frecuencia.

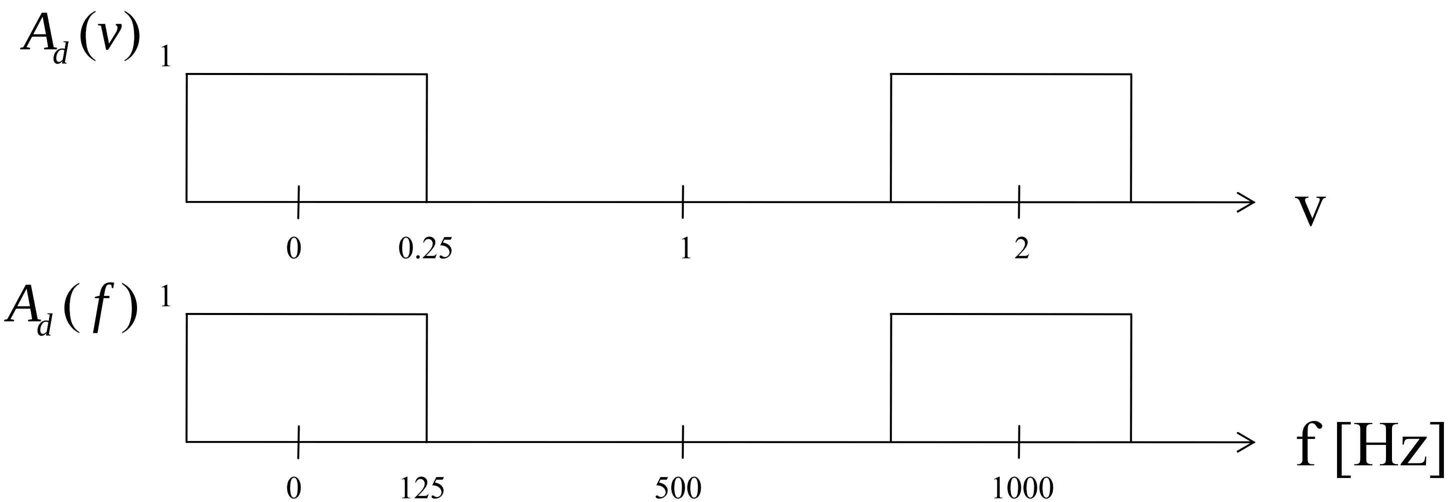
3.22 Método de la serie de Fourier para el diseño de filtros FIR

- Ejemplo: diseño de un filtro pasabajos FIR utilizando la serie de Fourier:

$$A_d = \begin{cases} 1 & 0 \le f \le 125Hz \\ 0 & otro \end{cases}$$

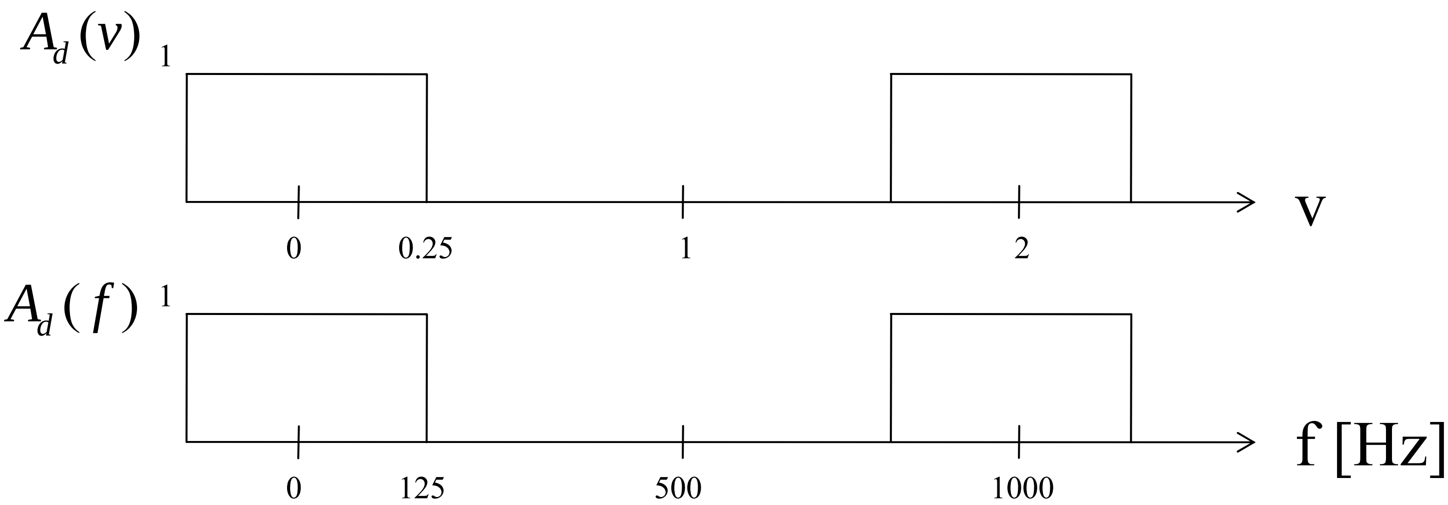
- fs=1kHz y la respuesta al impulso debe estar limitada a 20 retardos

3.22 Método de la serie de Fourier para el diseño de filtros FIR



$$C_m = \int_0^{0.25} \cos(m\pi v)dv = \left.\frac{sen(m\pi v)}{m\pi}\right|_0^{0.25} = \frac{sen(0.25m\pi)}{m\pi}$$

3.22 Método de la serie de Fourier para el diseño de filtros FIR



$$C_0=0.25; C_1=0.23; C_2=0.16;$$

$$C_3=0.08; C_4=0; C_5=-0.05;$$

$$C_6=-0.05; C_7=-0.03; C_8=0;$$

$$C_9=0.03; C_{10}=0.03;$$

3.23 Modulación digital

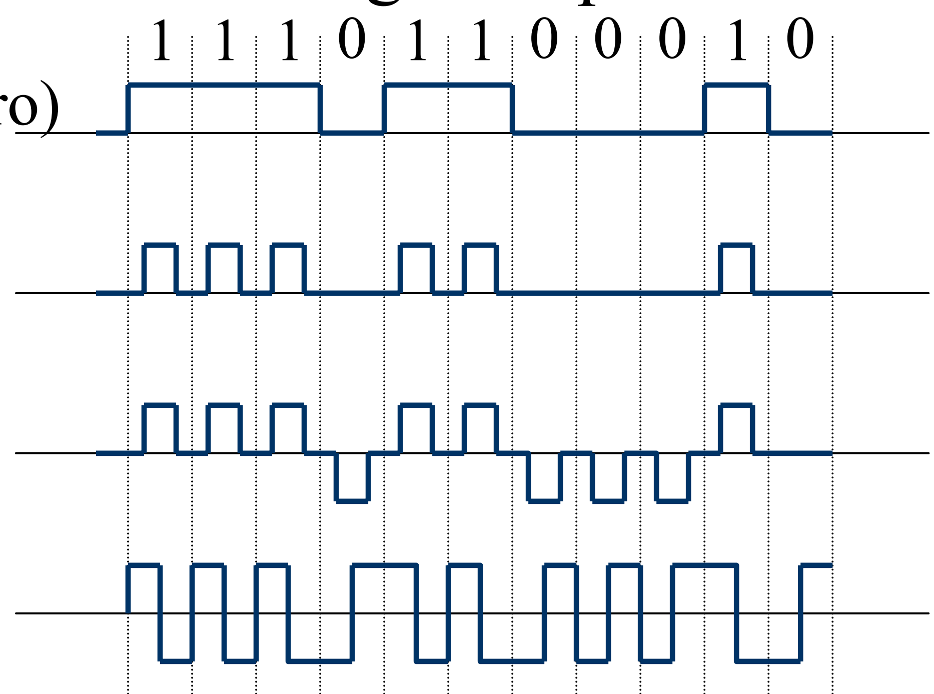
- Modulación PCM (pulse code modulation): Consiste en digitalizar la señal y luego codificar la salida como un tren de pulsos.
- Existen varios posibles códigos de pulsos a usar:

NRZ (non-return to zero)

RZ (return to zero)

RB (return to bias)

Manchester



3.23 Modulación digital

- DPCM: PCM diferencial, se envía por el canal la diferencia entre muestras sucesivas.

$$f_{DPCM}(t) = f_{PCM}(t) - f_{PCM}(t-1)$$

- Ya no es necesario que la señal f(t) esté limitada a un rango fijo, pero su pendiente debe estar limitada a un rango R
- Si existe un error en la transmisión de un dato en un sistema DPCM, toda la señal después del error se verá afectada.

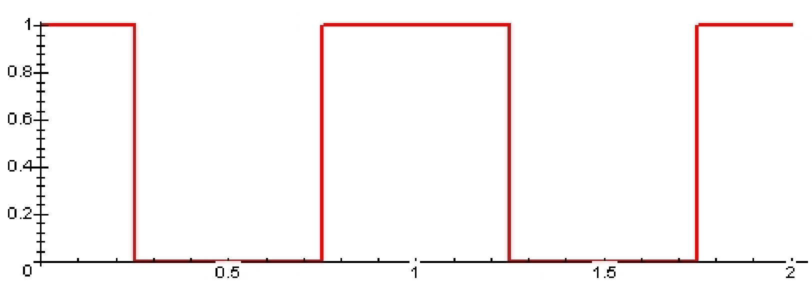
3.23 Modulación Delta

- Modulación delta (DM): es una variante de DPCM en la que sólo se usa un bit : 0 (la señal baja) o 1 (la señal sube)
- No se pueden transmitir valores constantes de forma exacta, la señal de salida "oscila" en torno a la constante. Si la oscilación es mayor que la máxima frecuencia de f(t), entonces puede eliminarse con un pasabajos.

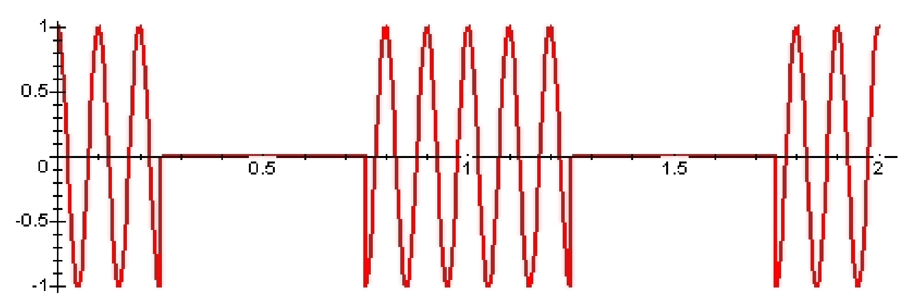
3.23 Modulación digital

- Las señales PCM pueden, además, ser moduladas usando AM, FM , etc. De este modo, se generan los siguientes modos de modulación:
  - Modulación ASK: La amplitud de la portadora depende del código PCM : $f_{ASK}(t) = f_{PCM}(t)\cos(\omega_c t)$
  - QASK o QAM: Se envían 2 señales en cuadratura

$$f_{QASK}(t) = f_{1PCM}(t)\cos(\omega_c t) + f_{2PCM}(t)\mathrm{sen}(\omega_c t)$$



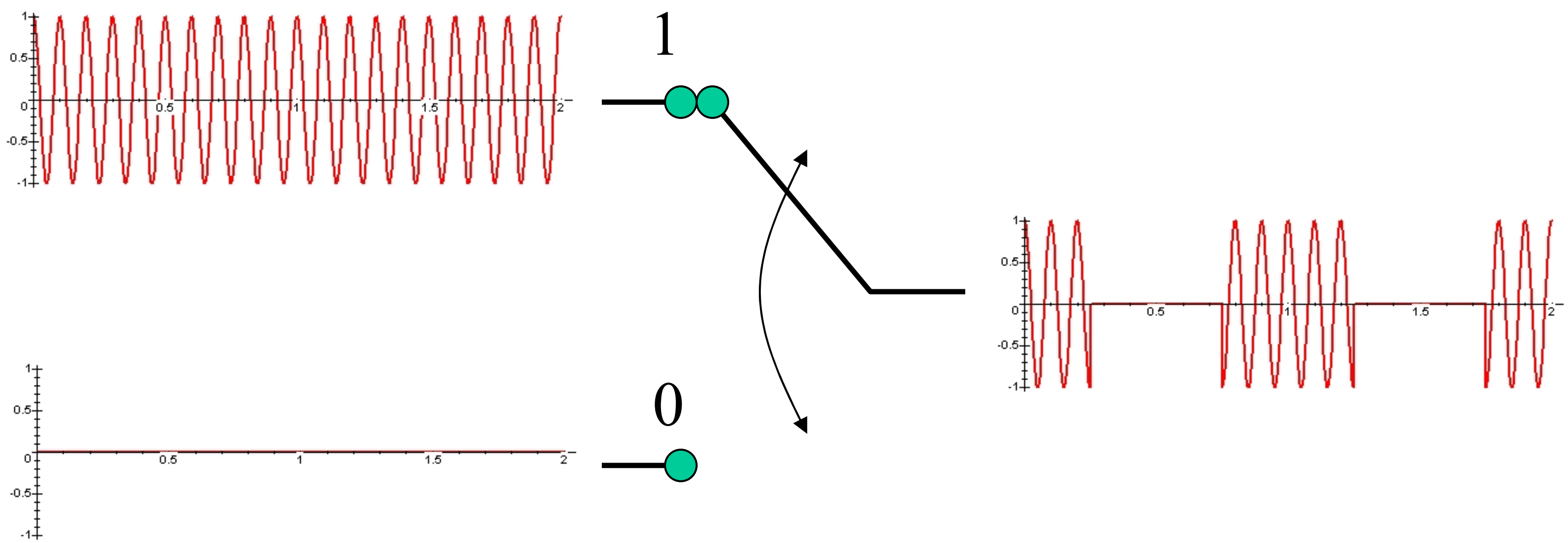
$f_{PCM}(t)$



$f_{ASK}(t)$

3.23 Modulación digital

– La modulación ASK se puede generar conmutando entre 2 fuentes.

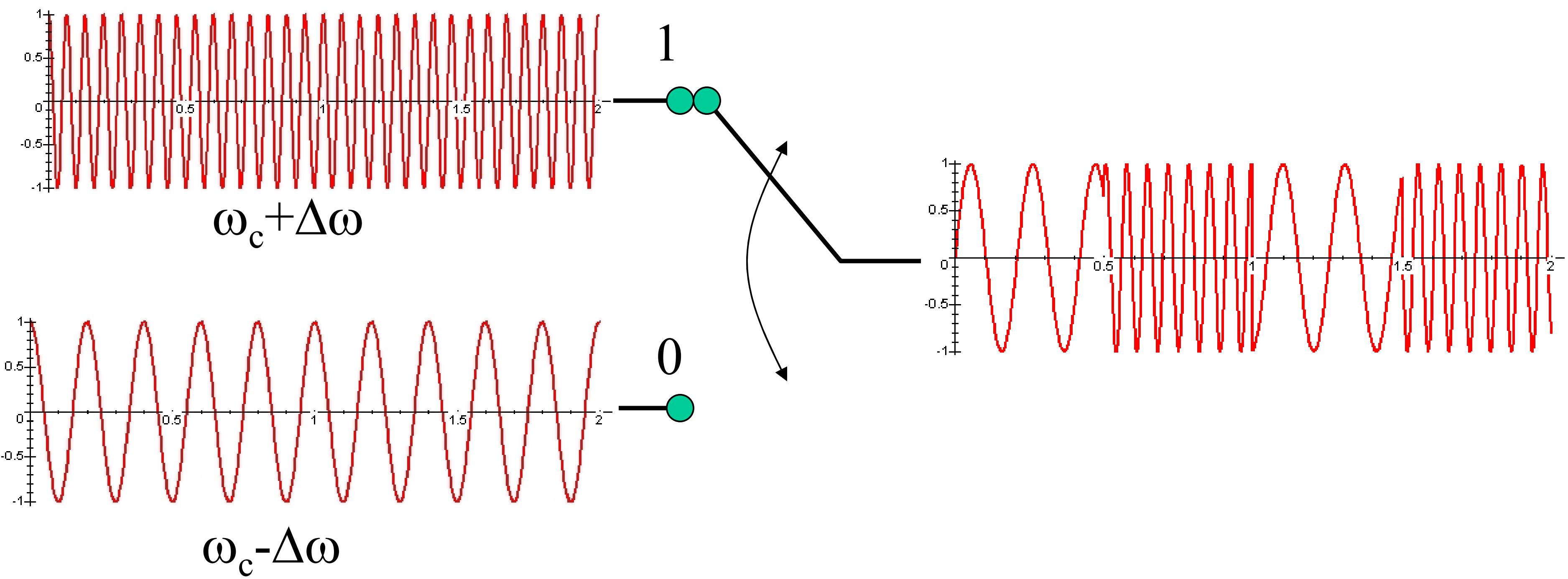


– La modulación QAM se puede usar para enviar 2 señales distintas o bien para enviar 2 bits de la misma señal a la vez (se conmuta entre 4 fuentes)

3.23 Modulación digital

– Modulación FSK: Es el “equivalente” a FM

$$f_{FSK}(t) = K\cos\left((\omega_c \pm \Delta\omega)t\right)$$



3.23 Modulación digital

– El hecho de conmutar entre 2 fuentes hace que la señal FSK tenga puntos de discontinuidad => se produce "desparramamiento" espectral (se contamina todo el espectro), por lo que puede usarse sólo en ambientes cerrados.

– Para evitar esto, existe también CPFSK (FSK continua), que se genera igual que una señal FM

$$f_{FSK}(t) = K\cos\left(\omega_0 t + \beta\int_{\tau=0}^{t} f_{PCM}(\tau)\,d\tau\right)$$

3.23 Modulación digital

– Modulación PSK: Similar a PM, se puede generar también conmutando entre 2 señales con la misma frecuencia y distinta fase (normalmente separadas en 180º).
– Modulación QPSK: Se conmuta entre 4 señales con la misma frecuencia, pero con fases 0º, 90º, 180º y 270º => se pueden enviar bits de a pares (2 bits a la vez)
– La probabilidad de error de ambos sistemas es la misma (a igual amplitud y frecuencia).

3.23 Modulación digital

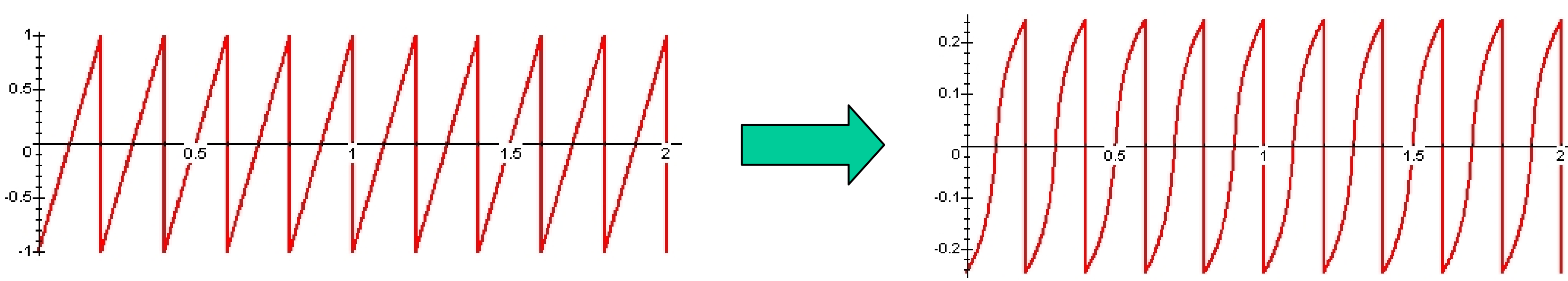
- CDMA: Cada usuario tiene asignado un código único (pseudoruido, PN) que corresponde a un tren de pulsos largo y único. Los distintos códigos PN forman (aproximadamente) una base ortogonal. Además, el código PN debe tener una frecuencia de muestreo mucho mayor que la señal a enviar.
- Para enviar la señal $f_{PCM}(t)$, se multiplica por el código PN y luego por una sinusoide portadora.

$$f_{CDMA}(t) = K \times q_{PN}(t)\, f_{PCM}(t) \cos(\omega_c t)$$

- Para demodular la señal CDMA, se debe multiplicar por la sinusoide y luego por <u>el mismo</u> código PN que se usó al enviarla.

3.24 Companción

- Compresión/expansión: Se refiere al proceso de comprimir una señal (en el lado del transmisor) y expandirla (en el lado del receptor)
- La idea es que el error cometido en cada muestra sea proporcional a la amplitud de la muestra => compresión logarítmica de la señal de entrada
- 2 estándares para comprimir: ley-μ y ley-A



3.24 Compansión

- Ley-m: Se usa en Estados Unidos y Japón.

$$V_{salida}(t) = \frac{V_{máximo} \times \ln\left(1 + \mu \dfrac{V(t)}{V_{máximo}}\right)}{\ln(1+\mu)}, \quad V(t) > 0$$

- Ley-A: Usada en Europa.

$$V_{salida}(t) = V_{máximo} \frac{A \times V(t)/V_{máximo}}{1 + \ln A}, \quad 0 \leq \frac{V(t)}{V_{máximo}} \leq \frac{1}{A}$$

$$V_{salida}(t) = V_{máximo} \frac{1 + \ln\left(A \times V(t)/V_{máximo}\right)}{1 + \ln A}, \quad \frac{1}{A} < \frac{V(t)}{V_{máximo}} \leq \frac{1}{A}$$

3.24 Compansión

- Las compansiones mostradas se pueden ver:
  - 1) Como una compresión/descompresión usada antes de la cuantización
  - 2) Como una cuantización no lineal
- Ambas formas de interpretarla son igualmente válidas

3.25 Multiplexión temporal (TDM)

- Pueden transmitirse varias señales digitales a través del mismo canal utilizando este principio
- Se le asignan intervalos de tiempo de cierta duración a cada canal, un bit de identificación u otros algoritmos