REPRESENTACIÓN MATEMÁTICA

Qué nos dice un instrumento

Ya hemos visto que un instrumento es una interfase con el mundo físico. Como entrada se tiene un fenómeno físico y a la salida se obtiene una información parcial de ese fenómeno. En general esa información la obtenemos mediante una magnitud que varía en función de la entrada (sistema analógico) o bien mediante números (sistema digital). En todo caso el tratamiento de cómo funciona un instrumento y las relaciones que existen entre su entrada y su salida se realiza mediante un *modelo matemático* del sistema.

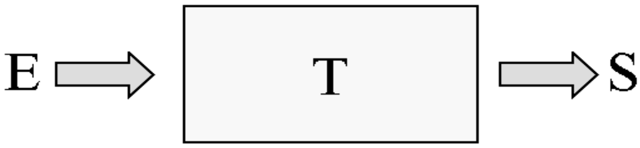


Figura 2-1. El análisis de un instrumento se realiza considerando una función de transferencia que relaciona la salida con la entrada.

La aproximación más sencilla es suponer un instrumento ideal en el cual la señal de entrada E(t) varía con el tiempo y a su salida tenemos otra señal S(t) que también varía con el tiempo. La *función de transferencia* de un sistema se define como el cociente de la *transformada de Laplace* de la señal de salida con respecto a la *transformada de Laplace* de la señal de entrada:

$$L(s)=\frac{L(S(\tau))}{L(E(\tau))}$$

recordemos que la transformada de Laplace se define como:

$$L(F(\tau))=\int_0^\infty e^{-s\tau}F(\tau)d\tau=f(s)$$

La función F(τ) se puede obtener a partir de f(s) mediante la transformada inversa de Laplace. Esto nos permite obtener la señal de entrada en función de la señal presente en la salida. La función de transferencia de un sistema complejo se puede obtener en función de las funciones de transferencia de cada uno de los elementos que lo componen. La función de transferencia de un conjunto de elementos en serie (cada uno de ellos tiene como señal de entrada la señal de salida del elemento anterior) se obtiene multiplicando las funciones de transferencia correspondientes a cada uno de los elementos.

$$T(s)=T_1(s)\cdot T_2(s)\cdot T_3(s)\cdot T_4(s)$$

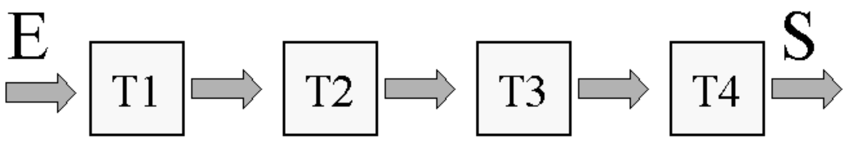


Figura 2-2. La función de transferencia de una serie de elementos es el producto de las funciones de transferencia

Sin embargo, cada uno de los elementos que constituyen un instrumento incorpora a la señal de salida otra señal generada en el propio elemento o procedente de interferencias con otros fenómenos y que se conoce como *ruido*. El ruido generado en un instrumento se refiere a la entrada como si se tratara de una señal (*nivel de ruido equivalente a la entrada*). Es importante tener presente que toda señal de entrada menor que este nivel NO puede ser detectada.

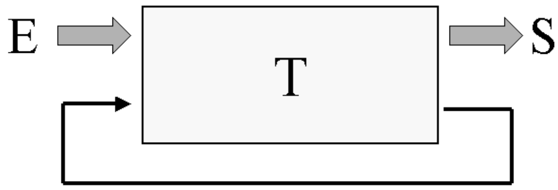


Figura 2-3. La realimentación es una técnica habitual para estabilizar un sistema.

Un efecto muy frecuente es que una parte de la señal de salida se transfiere otra vez a la entrada. Este efecto se conoce como *realimentación*. Bajo determinadas condiciones, la realimentación estabiliza el sistema y puede mejorar sus características globales, en otros casos conduce a inestabilizar el sistema, provocando oscilaciones o llevando al sistema fuera de sus límites de trabajo (saturación).

Medidas multiparamétricas

En el mundo físico los fenómenos son muy complejos y frecuentemente aparecen interferencias entre distintas magnitudes, tanto a nivel del sensor como directamente en forma de parámetros que condicionan el fenómeno. Esto es tanto más acusado cuanto mayor es la sensibilidad de los instrumentos empleados

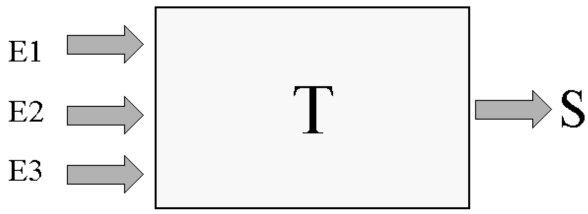


Figura 2-4. Un instrumento diseñado para estudiar un determinado proceso (E1) puede presentar interferencias con otros fenómenos (E2 y E3).

La solución cada vez más adoptada es ampliar el número de instrumentos, de modo que puedan medirse cada vez más parámetros y, basándose en un largo proceso de búsqueda de correlaciones entre los distintos datos, definir las funciones de transferencia que caracterizan la influencia de cada uno de los parámetros en el desarrollo del fenómeno. Esta es la aproximación que se sigue actualmente para tratar de definir los posibles precursores de un terremoto o de una erupción.

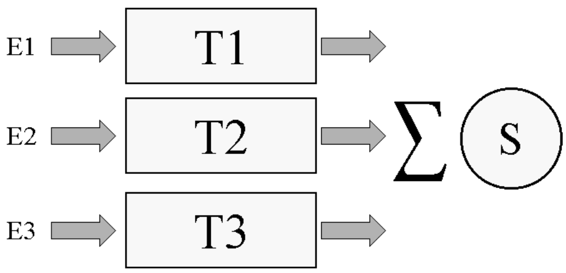


Figura 2-5. En muchos casos es necesario disponer de múltiples instrumentos para poder determinar las influencias mutuas entre los distintos fenómenos.

El tratamiento analítico de cada uno de los instrumentos debe hacerse de forma que en él se consideren todas las señales que puedan influir en la señal de salida. El conjunto de instrumentos debe poder definir cada una de ellas y además se debe disponer de suficiente número de instrumentos redundantes a fin de poder diferenciar los efectos de carácter local de los regionales.

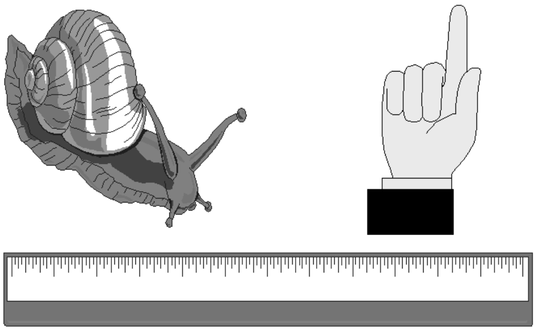


Figura 2-6. Ejemplo de sistema analógico, digital y mixto.

Sistemas analógicos y digitales

Los sistemas digitales están en la misma base de nuestra cultura, si la ciencia existe es porque tenemos números y tenemos números porque tenemos dedos. En cambio, el término analógico requiere una más compleja introducción, analógico procede de análogo y data de aquellos tiempos, no tan lejanos, en los que muchos problemas se tenían que resolver experimentalmente, buscando un sistema análogo que respondiera a las mismas ecuaciones y que pudiéramos reproducir en el laboratorio. Todavía hoy hay muchos problemas de la mecánica de los fluidos que sólo pueden resolverse analógicamente.

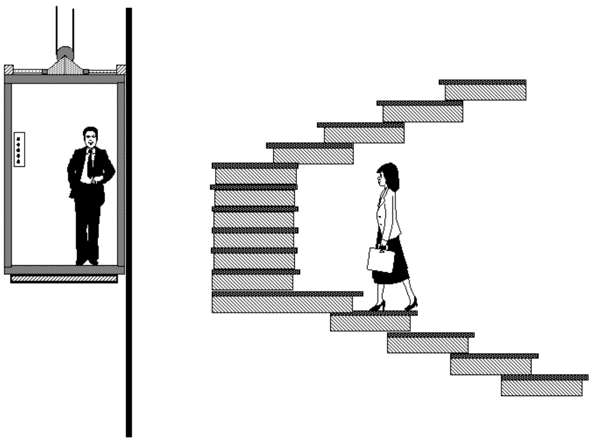


Figura 2-7. No siempre los sistemas digitales son más eficaces que los analógicos.

Además, se piensa erróneamente que todo sistema digital es mejor que un sistema analógico y, en muchas ocasiones, la solución analógica es más rápida, más sencilla y mucho más barata. De hecho, los coches para avanzar llevan ruedas que son un sistema analógico y no patas que sería el sistema digital. El conocimiento de sí un sistema, en su esencia, responde de un modo analógico o digital no sólo simplifica el proceso de medida, sino que siempre mejora la precisión de la misma. Muchos procesos muy complejos, como puede ser la gestión de las grandes catástrofes, aceptan una digitalización sencilla del tipo *todo* o *nada*. Estos dos únicos niveles (0/1) permiten después construir complejos sistemas de decisión.

Muestreo y retención

En el mundo de la instrumentación nos encontramos que muchos de los aparatos que consideramos analógicos, en realidad son digitales, y es que no sólo se trata de medir un objeto, sino que debemos dar un número y este número representa a ese objeto en el momento en el que se hizo la medida.

Figura 2-8. Cada cierto tiempo se realiza un muestreo del bicho.

Este proceso nos introduce dos conceptos fundamentales en el mundo digital: muestreo y digitalización. Supongamos que estamos siguiendo el peso de un bicho, cada cierto tiempo se le pesa y obtenemos un número que representa el peso del bicho en ese momento.

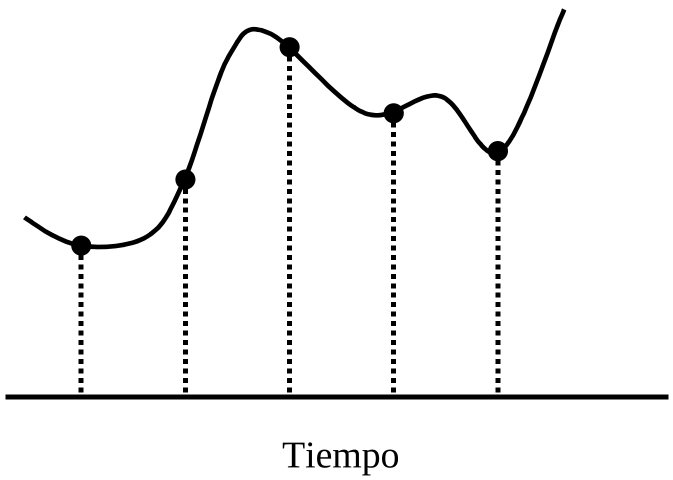
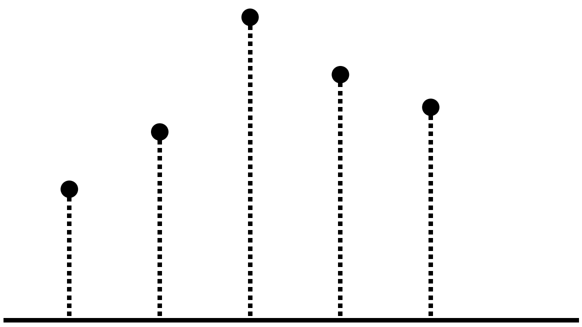


Figura 2-9. Muestreo: cada cierto tiempo se realiza una medida.

Pero esta información es parcial ya que sólo conocemos el peso en los momentos en que se ha pesado y desconocemos totalmente que ha pasado entre dos medidas.

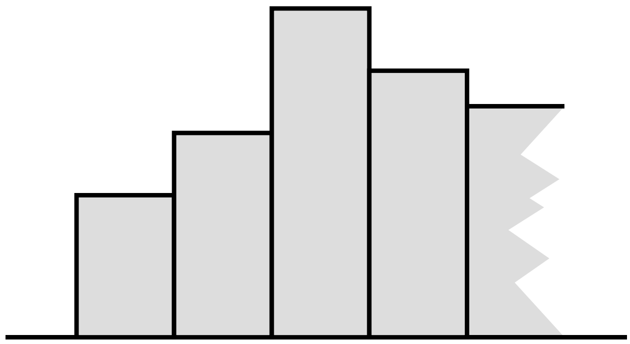
Figura 2-10. Sólo tenemos la información correspondiente al instante en el que se ha realizado la medida.



El seguimiento correcto supondría estar realizando continuamente pesadas, determinar cual es el muestreo óptimo exige un conocimiento exhaustivo del fenómeno a medir, especialmente de su evolución temporal. Cuando estamos midiendo un proceso que presente fuertes fluctuaciones y además muy rápidas, el problema de la definición del período de muestreo se agrava. En general, se procede a medir a intervalos iguales, aunque en algunos casos el muestreo con intervalos variable pueda aportar una mayor información del fenómeno. Cuando el muestreo se realiza regularmente, el *intervalo de muestreo* es uno de los parámetros más importantes del proceso de la digitalización, y se suele expresar como *número de muestras por unidad de tiempo* o simplemente expresando la *frecuencia* de muestreo en Hz.

Supongamos que se mide una señal con una frecuencia de muestreo fm. La señal, a nuestros efectos, ha desaparecido y sólo tenemos los valores que corresponden a cada uno de los instantes del muestreo.

Figura 2-11. Retención: entre medida y medida asignamos a la señal el valor de la última medida.

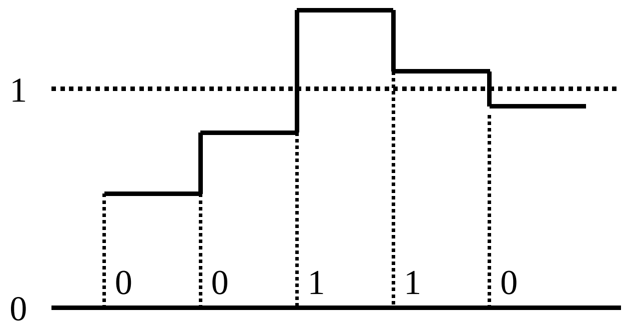


Entre medida y medida, el valor que tenemos para la señal es el correspondiente a la última medida realizada. A esta función se la denomina *retención*. En general el muestreo y la retención se ejecutan consecutivamente y por el mismo dispositivo, por lo que habitualmente nos encontramos con dispositivos de *muestreo y retención (Sample&Hold).*

Digitalización

La siguiente etapa consiste en la cuantificación (*digitalización*) de la medida, para ello se compara cada muestra retenida con un nivel, según supere o no dicho nivel se le asigna un 0 ó un 1.

Figura 2-12. Digitalización: se compara la señal con un nivel de referencia. Se asigna 1 si es mayor y 0 si es menor.



La señal está ahora representada por la secuencia de números 0,0,1,0. Sin embargo la representación que obtenemos si pretendemos restaurar la señal original es muy pobre.

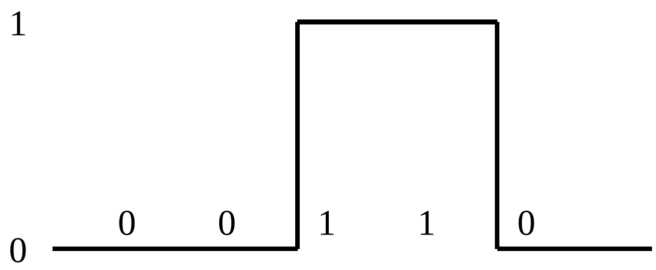


Figura 2-13. La reconstrucción de la señal con dos niveles es muy deficiente.

La situación mejora si en vez de comparar con sólo 2 niveles lo hacemos con cuatro aunque todavía es muy difícil reconocer en ella la señal original.

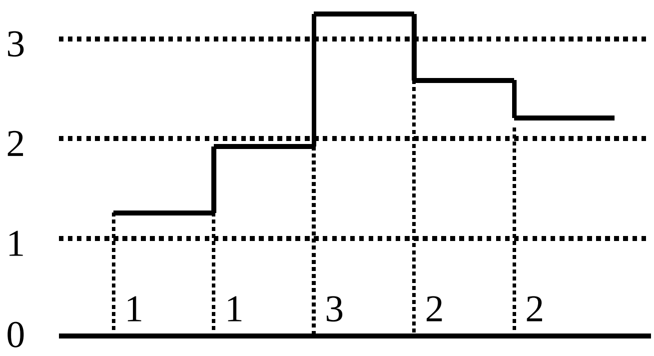


Figura 2-14. Digitalización con cuatro niveles.

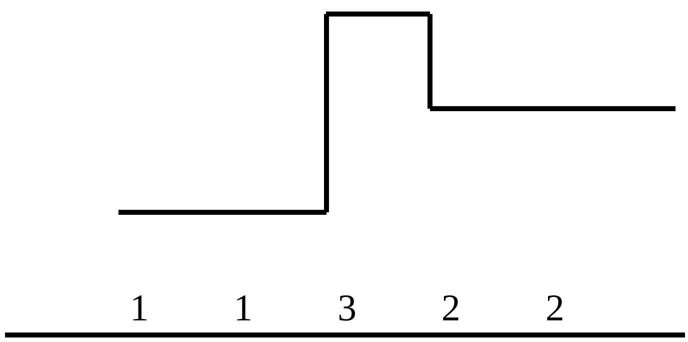


Figura 2-15. Reconstrucción de una señal digitalizada con cuatro niveles.

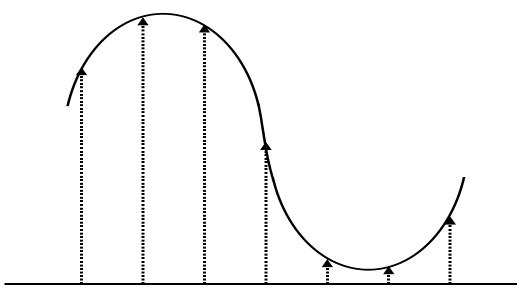


Figura 2-16. Muestreo de una señal sinusoidal.

Frecuencia de muestreo

El conocimiento de la señal será tanto mejor cuanto más elevada sea la resolución del cuantificador y más elevada la frecuencia de muestreo. Para que todavía podamos reconocer que es una función sinusoidal no podemos muestrear menos de dos veces por ciclo.

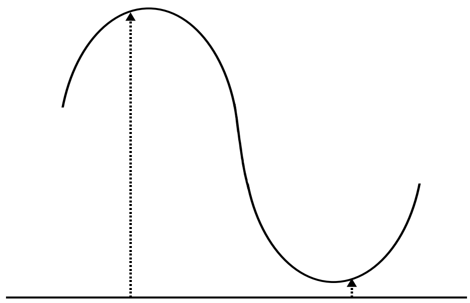


Figura 2-17. El muestreo debe ser al menos de dos muestras por ciclo.

Si la señal a muestrear posee una frecuencia mayor que la mitad de la frecuencia de muestreo se nos presenta el fenómeno del *aliasing*

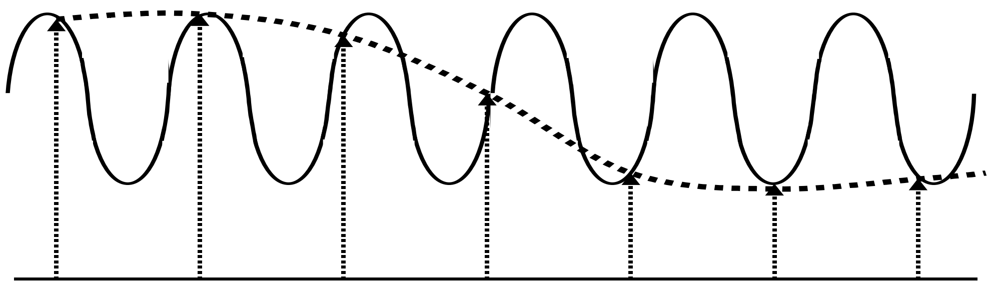


Figura 2-18. Si el muestreo es inferior a dos muestras por ciclo aparece el fenómeno del aliasing. La reconstrucción de la señal (línea de trazos) corresponde a una señal de frecuencia menor.

Representación matemática

Una señal se representa mediante una función continua del tipo:

$$S(t) \qquad -\infty < t < \infty$$

Una señal se denomina periódica de periodo T cuando se cumple:

$$S(t) = S(t + T) \qquad -\infty < t < \infty$$

Una señal es de energía limitada cuando se cumple:

$$E > \int_{-\infty}^{\infty} S^2(t)\, dt$$

Una señal se denomina de duración limitada cuando se anula fuera de un intervalo de tiempo especificado:

$$S(t)=0 \qquad t<t_1$$

$$S(t)=0 \qquad t>t_2$$

Una aplicación no es más que una regla por la cual elementos de un conjunto S se asignan a elementos de otro conjunto.

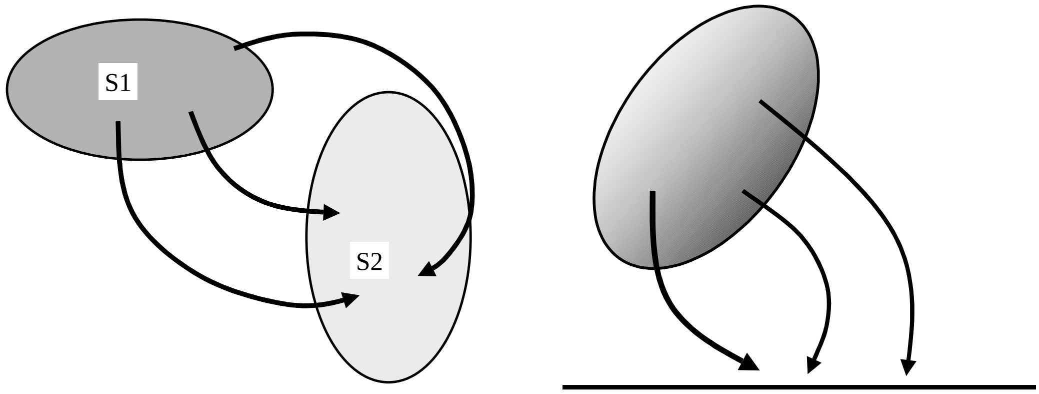


Figura 2-19. Una aplicación es una regla por la cual los elementos de un conjunto S1 se asignan a elementos de otro conjunto S2 . Un caso particular, de gran interés en el proceso de señales es la aplicación del conjunto de las señales en el eje real.

En proceso de señales la transformación más frecuente es la transformada de Fourier.
Si S_1 es el conjunto de las señales de energía acotada:

$$S_1 = \left\{ x, \int_{-\infty}^{\infty} x^2(t)\, dt \right\}$$

La transformada de Fourier es una aplicación en otro conjunto S_2 de funciones de cuadrado integrable:

$$S_2 = \left\{ X, \int_{-\infty}^{\infty} X^2(f)\, df \right\}$$

Esta aplicación está descrita por:

$$S_1 \Rightarrow S_2 \qquad X(f) = \int_{-\infty}^{\infty} x(t)\, e^{-2\pi f\, it} dt$$

donde $i=\sqrt{-1}$. La transformación inversa es

$$S_2 \Rightarrow S_1 \qquad x(t) = \int_{-\infty}^{\infty} X(f)\, e^{2\pi f\, it} df$$

Al conjunto S_1 se le denomina espacio *tiempo* y al conjunto S_2 espacio *frecuencia*. El resultado X(f) de aplicar una transformada de Fourier a un señal x(t) se denomina *espectro* de la señal x(t). Siempre debe tenerse presente que ambos espacios son distintos. Sin embargo cuando se tiene la solución de un problema en uno de los espacios, simultáneamente se tiene la solución en el otro, la cual podrá tener interés práctico o no.

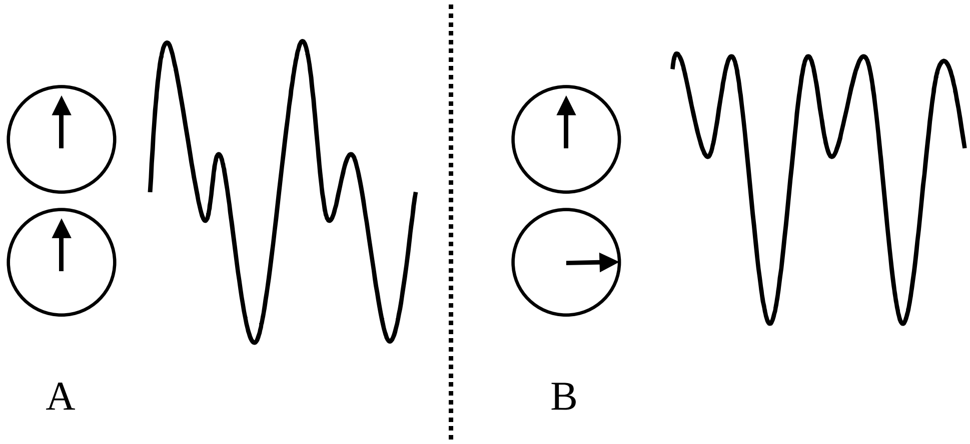


Figura 2-20. La figura muestra dos señales obtenidas superponiendo dos componentes sinusoidales de distinta frecuencia, sin diferencia de fase (A) y con una diferencia de fase de $\pi/2$ (B).

Es importante destacar que a una señal definida en el tiempo, le corresponde un espectro de frecuencias, pero no hay que olvidar que a cada frecuencia le corresponde una fase. El espectro nos dice cuales son las componentes armónicas de la señal pero también cual es la relación de fases inicial existente entre ellas. Las mismas amplitudes en el espectro no tienen porque corresponder a las mismas señales. Cuando nos quedamos sólo con el espectro de amplitudes estamos perdiendo información sobre la señal. El problema radica en que es mucho más fácil trabajar con amplitudes que con fases, además la calibración de la respuesta en amplitud de un instrumento se puede realizar con equipos modestos, mientras que la calibración en fase exige técnicas mucho más sofisticadas. Las características espectrales permiten establecer también una clasificación de los distintos tipos de señales, de modo similar a como se ha hecho anteriormente en el espacio tiempo: Una señal es *monocromática* cuando su espectro sólo es distinto de cero en un sólo punto. Una señal se dice que tiene un espectro *blanco* cuando todas las frecuencias presentan la misma amplitud. Cuando eso sólo ocurre dentro de un ancho de banda entre f_1 y f_2 se dice que la señal presenta un espectro *rosa*. Una función de banda limitada es aquella cuyo espectro se anula fuera de un determinado margen de frecuencias F. Una función de energía limitada que tiene un especial interés es la función *delta* de Dirac o función *impulso*, definida de la siguiente forma:

$$\delta(x) = 0 \qquad |x| > 0$$

$$\int_{-\infty}^{\infty} \delta(x)\, dx = 0$$

Obsérvese que con esta definición de la función $\delta(x)$ se cumple:

$$s(t)\,\delta(t-\tau) = s(\tau)$$

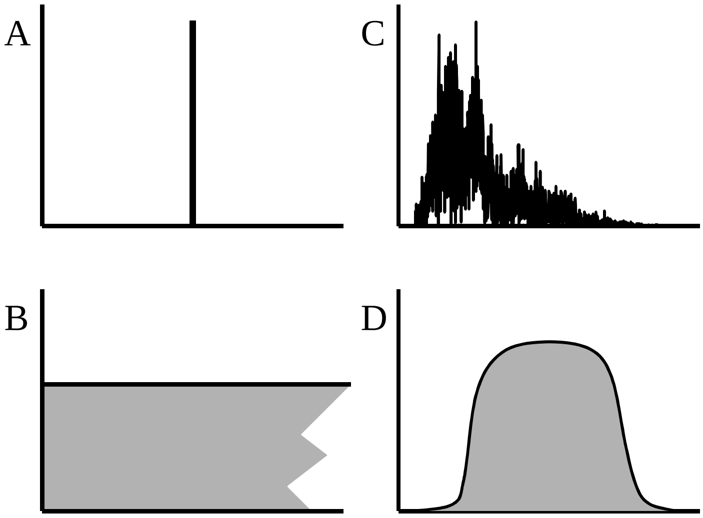


Figura 2-21. Ejemplos del espectro de distinto tipo de señales: (A) señal monocromática, (B) ruido blanco, (C) ruido sísmico registrado con un geófono de corto periodo, (D) ruido rosa.

De acuerdo con esto, la operación del muestreo de una señal s(t) se puede representar mediante:

$$
\begin{aligned}
s_m &= s(t)\delta(t-\infty) + \ldots + s(t)\delta(t-3\tau) + s(t)\delta(t-2\tau) + \\
&+ s(t)\delta(t-\tau) + s(t)\delta(t) + s(t)\delta(t+\tau) + s(t)\delta(t+2\tau) + \\
&+ s(t)\delta(t+3\tau) + \ldots + s(t)\delta(t+\infty)
\end{aligned}
$$

o lo que es lo mismo:

$$s_m = \sum_{i=-\infty}^{\infty} s(t)\ \delta(t-i)$$

Es decir, el muestreo de una señal se reduce a multiplicar la señal por una serie de funciones delta espaciadas el intervalo de muestreo.

Es importante tener presente que así como el espectro de una señal es una función continua, el espectro de una señal muestreada es discontinuo y además se repite cada intervalo de frecuencias igual a la frecuencia de muestreo (espectro imagen). El problema surge cuando la señal no es de ancho de banda limitado a menos de la mitad de la frecuencia de muestreo, ya que en tal caso, hay una fracción del espectro que se superpone con los espectros imagen, apareciendo el fenómeno del aliasing. Para evitarlo, la única solución es limitar el espectro de la señal por debajo de la mitad de la frecuencia de muestreo o frecuencia de *Nyquist.*

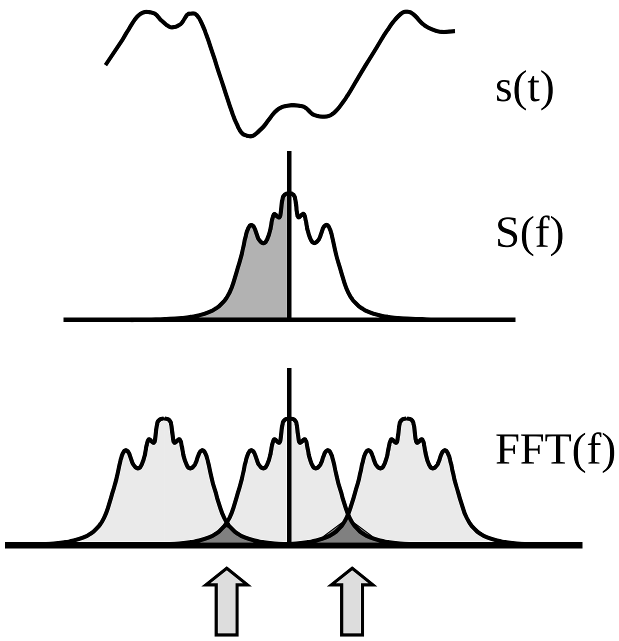


Figura 2-22. La figura muestra el espectro S(f) de una señal continua s(t) y el espectro FFT(f) de la misma señal muestreada. Obsérvese la repetición del espectro con periodicidad la frecuencia de muestreo y el solapamiento de los espectros.

Filtros

Un filtro es un dispositivo que permite modificar el espectro de una señal. Estos dispositivos se pueden implementar antes del muestreo (filtros analógicos) o después de la digitalización (filtros digitales). Sin embargo, si queremos evitar que se nos produzca el fenómeno del aliasing, deberemos situar el filtro antes del dispositivo de muestreo. Para no perder información el filtro *antialias* ideal seria un filtro rectangular con corte a la frecuencia de Nyquist, de forma que no dejara pasar ninguna frecuencia superior a la frecuencia de Nyquist y no afectara a las frecuencias inferiores.

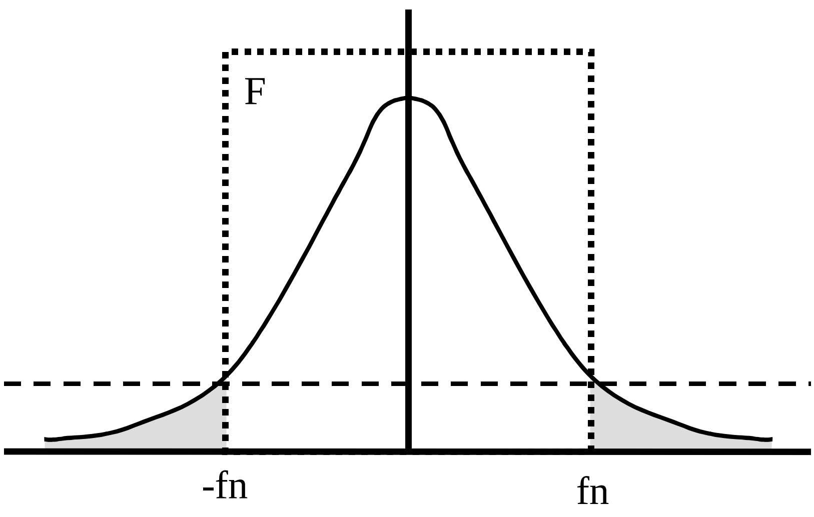


Figura 2-23. Para evitar el fenómeno del aliasing es necesario filtrar la señal de forma que su espectro quede limitado por debajo de la frecuencia (fn) de Nyquist (mitad de la frecuencia de muestreo).

Un filtro *pasa bajos* deja pasar todas las frecuencias inferiores a la frecuencia de *corte*, el *filtro pasa altos* las frecuencias superiores a la de *corte*. Un filtro pasa banda deja pasar un *ancho de banda* de frecuencias centrado sobre la frecuencia *central* del filtro. El problema surge cuando en la práctica no es posible construir un filtro rectangular, sino que el filtro deja pasar todas las frecuencias, sólo que a partir de una determinada zona del espectro atenúa unas frecuencias más que otras.

La frecuencia de corte ahora no queda perfectamente determinada, por lo que por convenio se ha definido la frecuencia de corte como aquella frecuencia en la que la energía de la señal cae a la mitad (-3dB). Otro elemento a tener en cuenta es la pendiente del filtro, que se suele expresar como la atenuación que sufre la señal en dB por octava. Para un filtro pasa banda el ancho de banda se define entre los dos puntos de corte a -3dB y la frecuencia central como la frecuencia media de los dos cortes.

Conversión analógico digital

La conversión de una señal analógica a digital requiere la combinación de todos estos elementos. Primero hay que limitar el espectro de la señal por debajo de la frecuencia de Nyquist, para lo cual se incluirá un filtro pasa bajos (F), después será necesario muestrear (S) y retener la información (H) y seguidamente se procederá a cuantificarla mediante el conversor analógico digital propiamente dicho. Si se desea poder trabajar con distintas velocidades de muestreo deberemos diseñar el filtro pasa bajos de forma que sea posible ajustarlo a las distintas frecuencias. Todos estos elementos habrá que concebirlos de forma que no se pierda información, pero tampoco diseñar el sistema por encima de nuestras necesidades, ya que ello supone un coste mucho más elevado así como una mayor demanda de energía y dificultad de calibración y mantenimiento.

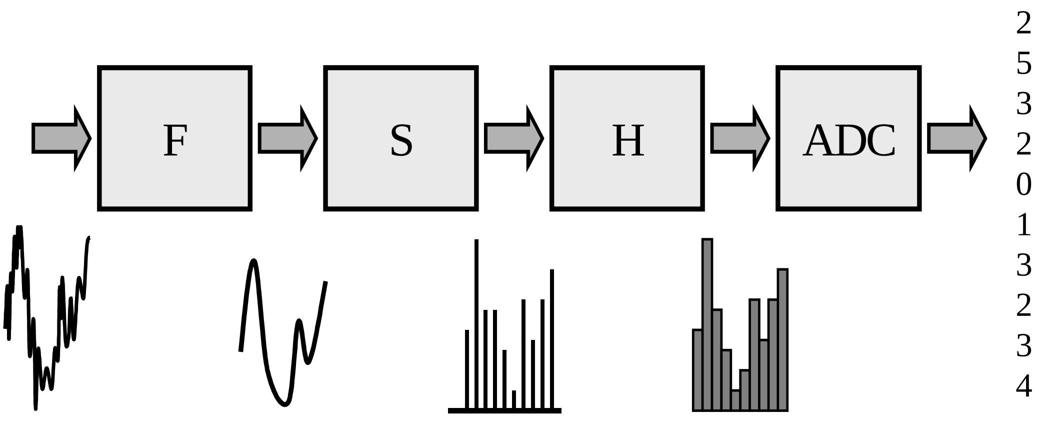


Figura 2-24. La figura muestra el conjunto de funciones necesarias para digitalizar una señal. (F) filtro de entrada, (S) muestra, (H) retención, (ADC) conversor analógico digital.

Determinación de la respuesta en frecuencia

En un dispositivo analógico la respuesta en frecuencia se puede obtener fácilmente a partir del conjunto de funciones de transferencia correspondientes a cada uno de los módulos. De hecho hay muchos programas de diseño electrónico que construyen la función de transferencia del circuito de modo automático. Otra solución es aplicar un impulso a la entrada y analizar la respuesta en frecuencia de la salida. Este método es muy difícil de realizar directamente con métodos analógicos, pero es muy fácil en sistemas digitales, ya que podemos aplicar un algoritmo para el cálculo de la transformada de Fourier directamente a la serie de datos digitales obtenida a la salida del sistema. Como ejemplo se muestra la determinación de la respuesta en frecuencia de un filtro ampliamente utilizado en sistemas de adquisición de datos por su sencillez. Este filtro está definido por la relación de recurrencia: $Y_{i+1} = Y_i + k\,(X_i - Y_i)$ con $k = 1/32$.

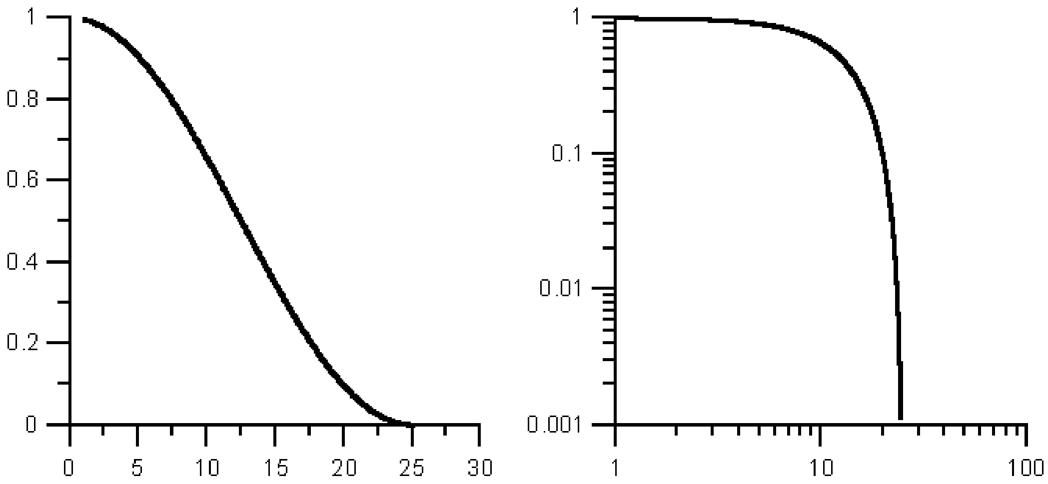


Figura 2-25. Respuesta temporal a una función impulso de un filtro digital pasa-bajos definido de acuerdo con las siguientes relaciones de recurrencia: $Y_{i+1} = Y_i + k\,(X_i - Y_i)$ con $k = 1/16$. y la respuesta en frecuencia del filtro

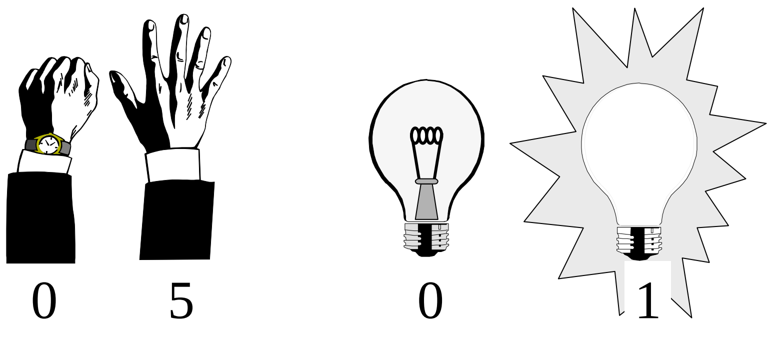


Figura 2-27. El sistema de numeración decimal parte de tener diez dedos y tenemos un diez caracteres para representarlo (0,1,2,3,4,5,6,7,8,9). El sistema binario sólo contempla dos estados (apagado y encendido) y tenemos dos caracteres para representarlo (0,1).

Aritmética binaria

Estamos habituados a utilizar el sistema de numeración decimal, pero ello sólo es consecuencia natural de tener diez dedos, si el hombre hubiera desarrollado trece dedos utilizaríamos un sistema en base 13. Por ello, no debe extrañarnos que al ir desarrollando determinados autómatas se hayan ido introduciendo paralelamente otros sistemas de numeración. En particular, para un sistema eléctrico (electrónico) lo más simple es utilizar dos estados, encendido y apagado, conectado o desconectado, sí o no, 0 ó 1.Formalmente, al conjunto de elementos distintos que se utilizan para establecer un sistema de numeración se le denomina base del sistema de numeración. Así tenemos un sistema de numeración de base 10 (decimal) y un sistema de numeración de base 2 (binario). Todos los computadores trabajan con un sistema de numeración binario, traduciendo los valores al sistema decimal sólo para la entrada y salida de la información. A esta unidad mínima de información se la denomina bit. Así un bit es algo que puede adoptar dos estados 0 o 1, sí o no, etc.

En el sistema decimal, cuando se agotan los caracteres pasamos a la posición siguiente a la izquierda, así el número veintiséis lo representamos como:

$$26$$

y cada posición a la izquierda representa una potencia de 10.

$$10^5\ 10^4\ 10^3\ 10^2\ 10^1\ 10^0$$

Igualmente se hace en un sistema binario, sólo que aquí cada posición a la izquierda representa una potencia de 2.

$$2^7\ 2^6\ 2^5\ 2^4\ 2^3\ 2^2\ 2^1\ 2^0$$

Así el número dos se representa:

$$10$$

Los dieciséis primeros números se pueden representar con cuatro bits:

Decimal	Binario	Hexadecimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

Para facilitar la escritura de números en binario se utiliza habitualmente el sistema hexadecimal que tiene base 16 y equivale a 4 bits. La traducción entre ambos sistemas es inmediata. Los sistemas de cómputo operan con grupos de bits (palabras) que empezaron siendo múltiplos de 4 y actualmente son múltiplos de 8. Los sistemas más pequeños trabajan con palabras de 8 bits, que se pueden unir para representar números cada vez mayores. Una palabra de 8 bits se conoce con el nombre de Byte. Un Byte puede almacenar un número desde 2^0 a $2^7 + 2^6 + 2^5 + 2^4 + 2^3 + 2^2 + 2^1 + 2^0$ es decir de 0 a 255. El desplazar un dígito a la izquierda en el sistema decimal equivale a multiplicar por 10 y desplazarlo a la derecha a dividir por diez. Por ejemplo:

610 si se desplaza a la izquierda 6100
610 si se desplaza a la derecha 61

En el sistema binario es idéntico, sólo que equivale a multiplicar por 2 o a dividir por 2. Por ejemplo el número seis (0110):

0110 si se desplaza a la izquierda 1100 que es doce
0110 si se desplaza a la derecha 0011 que es tres

Esta propiedad es muy importante ya que permite realizar operaciones de multiplicar y dividir por 2 de forma muy rápida con sólo desplazar los bits.

Suma de números binarios

Para sumar dos números binarios se utiliza la siguiente definición:

+	0	1
0	0	1
1	1	0*

Donde * indica que se ha producido el acarreo de 1 bit. Por ejemplo:

00000101 00000110	5 6
00001011	11

00001000 00000110	10 6
00010000	11

Números positivos y negativos

En el sistema de numeración decimal un número positivo y un número negativo se diferencian por un símbolo extra (-), así tenemos 5, -6, etc. En un sistema binario no es posible introducir un nuevo símbolo, pues sólo tenemos 0, 1. La solución es considerar el bit situado más a la izquierda como el bit del signo. De esta forma podríamos representar:

0000011 (3)

1000011 (-3)

pero esto nos lleva a complicar la operación de suma, pues la suma de un número y el opuesto debe ser 0.

00000011 10000011	3 -3
10001111	-15 ¿?

Evidentemente, esta forma de representar los números negativos no es operativa. Para solucionar este problema se introduce el siguiente procedimiento para cambiar el signo a un número: se cambian los 0 en 1 y viceversa y se suma 1 (complemento a 2):

00000011	3
11111100	0 →1
00000001	+1
11111101	-3

Con esta definición la suma de un número y su opuesto es 0

00000101 11111011	5 -5
00000000	0

Para restar dos números basta con cambiar el signo (cambiar 0 y 1, sumar 1) y hacer una suma. De acuerdo con esto una palabra de 8 bits considerando el bit 7 como signo sería:

01111111	127
----	---
00000001	1
00000000	0
11111111	-1
----	---
10000001	-127
10000000	-128

Hay que tener esto presente cuando se trabaje con conversores analógico digitales, ya que normalmente proporcionan los datos de salida sin signo, incluso cuando operan en modo bipolar.

Multiplicación

Para multiplicar dos números binarios podemos partir de la definición de producto y sumar el multiplicando tantas veces como indique le multiplicador. Otra posibilidad, mucho más eficiente es utilizar el desplazamiento a la izquierda como producto por dos, de esta forma una multiplicación se puede realizar sólo con desplazar y sumar.

5x10 = 50				
00000101	Multiplicador	Resultado con 16 bits		5
00001010	Multiplicando			10
0000010**1**	Sumar (b0 = 1)	00000000 00001010	0000000000001010	10
000001**0**1	Desplazar y pasar (b1 = 0)	00000000 00010100	- - - - - - -	
00000**1**01	Desplazar y sumar (b2 = 0)	00000000 00101000	0000000000101000	30
0000**0**101	Desplazar y pasar (b3 = 0)	00000000 01010000	- - - - - - -	
000**0**0101	Desplazar y pasar (b4 = 0)	00000000 10100000	- - - - - - -	
00**0**00101	Desplazar y pasar (b5 = 0)	00000001 01000000	- - - - - - -	
0**0**000101	Desplazar y pasar (b6 = 0)	00000010 10000000	- - - - - - -	
00000101	Desplazar y pasar (b7 = 0)	00000101 00000000	- - - - - - -	
Final			0000000000110010	50

Obsérvese que el producto de dos palabras de 8 bits es una palabra de 16 bits. El producto de dos palabras de 8 bits supone realizar ocho operaciones de desplazamiento, ocho comparaciones y hasta ocho sumas y todo ello con doble precisión (dos palabras). El incluir muchas operaciones aritméticas supone mucho tiempo de cálculo para el procesador y si se utilizan sistemas mínimos enseguida se supera el tiempo disponible para realizarlo.

Tipos de datos

En los sistemas de microproceso muy pequeños, como son los microcontroladores, sólo disponemos de palabras de 8 bits consideradas como enteros sin signo (0 a 255) con las que se puede realizar una serie de operaciones elementales:

Suma	B1 + B2
Resta	B1 – B2
Desplazamiento a la izquierda	B1 <<
Desplazamiento a la derecha	B1 >>
AND lógico	B1∩B2
OR lógico	B1∪B2
Incrementar	B1 + 1
Decrementar	B1 - 1

Además se puede controlar si estas operaciones provocan *arrastre, cero, negativo*. Cuando hay que repetir un conjunto de instrucciones un determinado número de veces es preferible cargar un contador y decrementar cada vez, pues cuando llegue a cero el propio procesador nos avisa y de esta forma nos ahorramos una comparación y eso supone ahorrar tiempo. En procesadores más complejos se dispone de un repertorio mayor de operaciones, pero en realidad no es cierto, sino que la máquina descompone las nuevas operaciones como combinación de las operaciones elementales. En cuanto los tipos de números que podemos utilizar en procesadores como los que ocupan los computadores personales tenemos:

Tipos enteros			
Tipo	Definición	bits	Valores
Carácter	char	8	-128 a 127
Carácter sin signo	unsigned char	8	0 a 255
Entero corto	short integer	16	-32768 a 32767
Entero corto sin signo	unsigned short integer	16	0 a 65535
Entero	integer	32	0 a 4294967295
Entero sin signo	unsigned integer	32	-214748,648 a 2147483647

Para cada uno de estos tipos están definidas las correspondientes operaciones. Es posible combinar en una operación variables de distinto tipo, pero hay que adoptar precauciones y en general habrá que realizar antes una conversión de tipos cuidando la precisión y tipo del resultado que se desea obtener.

Tipos en coma flotante				
Tipo	Definición	Precisión	bits	Valores
Flotante	float	7 dígitos	32	$1.18x10^{-38}$ a $3.40x10^{38}$
Flotante doble	double	15 dígitos	64	$2.23x10^{-308}$ a $1.79x10^{308}$
Flotante largo	long double	18 dígitos	80	$3.37x10^{-4932}$ a $1.18x10^{4932}$

La representación de un número en coma flotante en binario es algo artificiosa y se debe ir con mucho cuidado ya que no todos los procesadores adoptan exactamente la misma definición. Podemos definir los números más pequeños que 1 en función de potencias negativas de 2

$$... 2^8\ 2^7\ 2^6\ 2^5\ 2^4\ 2^3\ 2^2\ 2^1\ 2^0\ 2^{-1}\ 2^{-2}\ 2^{-3}\ 2^{-4}\ 2^{-5}\ 2^{-6}\ 2^{-7}\ 2^{-8}\ ...$$

un número de 8 bits con 3 fraccionarios se representará en la forma

$$b_4\ b_3\ b_2\ b_1\ b_0.b_{-1}\ b_{-2}\ b_{-3}$$

La definición más extendida (*IEEE 754 Floating Point Standard*) introduce dos tipos de 32 y 64 bits:

32 bits		
1 bit	8 bits	23 bits
signo	Exponente	Mantisa

64 bits		
1 bit	11 bits	52 bits
signo	Exponente	mantisa

La mantisa se supone normalizada, es decir desplazada a la izquierda hasta que el bit más significativo sea 1, y como siempre es 1 se suprime, por lo que habrá que incorporarlo cada vez que queramos reconstruir un número en coma flotante a formato decimal. El exponente se almacena desplazado para que siempre sea positivo (+127 en el formato de 32 bits y +1023 en el de 64 bits). Un exponente con todos los bits a 1 o a 0 indica un número no válido. Este permite controlar el desbordamiento por arriba o por debajo en las distintas operaciones. Por ejemplo se trata de representar –3/16 en formato binario en coma flotante (float)

Normalización	-3/16	$-(2^{-3} + 2^{-4})$
Desplazamiento del exponente	–3 + 127	01111100
Signo y magnitud	–3	1 11000000000000000000000
Eliminar el primer bit de la mantisa		1000000000000000000000
Representación final	1 01111100 1000000000000000000000	

En general, en los sistemas muy pequeños no hay necesidad de utilizar números en coma flotante, pues difícilmente podremos implementar las librerías matemáticas para operar con ellos, pero si es frecuente tener que convertir un formato en otro. Hemos visto que la representación de un número en binario requiere utilizar varias palabras, el orden que se utiliza para almacenar estas palabras en el computador depende del tipo de procesador que se utilice, así Intel y Motorola utilizan criterios distintos, por lo que frecuentemente es necesario realizar funciones de conversión.

Filtros digitales

Son muchos los algoritmos que se han desarrollado para implementar funciones de filtrado digital. Aquí sólo vamos a tratar los más sencillos y que son los más aptos para operar en sistemas de microproceso en tiempo real, donde las posibilidades de cálculo aritmético son muy reducidas. En la bibliografía se presentan referencias que tratan el tema exhaustivamente.

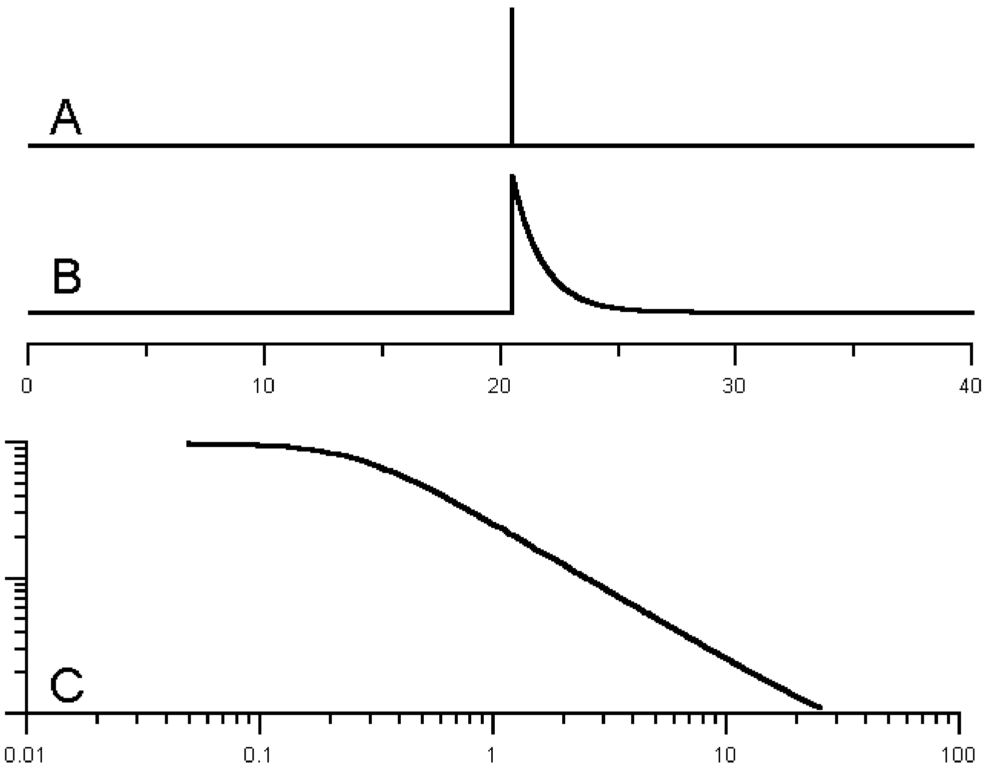


Figura 2-28. Proceso seguido para obtener la respuesta en frecuencia de un filtro digita. A) se construye una función impulso (valor cero en todos los puntos menos en uno de ellos que se selecciona hacia la mitad de la secuencia). B) se obtiene la función filtrada. C) se calcula la transformada de Fourier discreta (FFT) que nos da la respuesta del filtro.

Estos filtros tienen muchas aplicaciones, como es mejorar la relación señal ruido, separar la señal en múltiples ventanas frecuenciales, eliminar la deriva de cero, etc. Sin embargo no sirven como filtro antialias, es decir la señal de entrada en el conversor tienen que tener su espectro limitado a la mitad de la frecuencia de muestreo. En caso contrario el espectro se contamina y ya es imposible recuperar la señal original. Otro tema es reducir la frecuencia de muestreo de una señal: supongamos una señal de frecuencia f_1 que se desea transformar en otra de frecuencia f_2 con $f_1 > f_2$. Para ello filtramos f_1 con un filtro pasa bajos de frecuencia de corte menor que $f_2/2$ y seguidamente se reduce el muestreo tomando un punto de cada f_1/ f_2. En la práctica es muy frecuente que se reduzcan series de datos temporales alegremente, simplemente eliminando los puntos sobrantes y sin tener presente que esto puede dar origen a la aparición de frecuencias falsas debidas a fenómenos de aliasing. También es frecuente que se midan sistemas cada cierto tiempo (una hora, un día) y no se tengan en cuenta la posibilidad de fenómenos de aliasing. Como ejemplo podemos citar la medida de inclinómetros en volcanes activos, donde la presencia del temblor volcánico (tremor) perturba considerablemente la medida. La única solución es medir a una frecuencia muy alta, por encima de la frecuencia de respuesta del instrumento y después reducir mediante un filtrado digital adecuado la serie de medidas a la frecuencia de muestreo deseada.

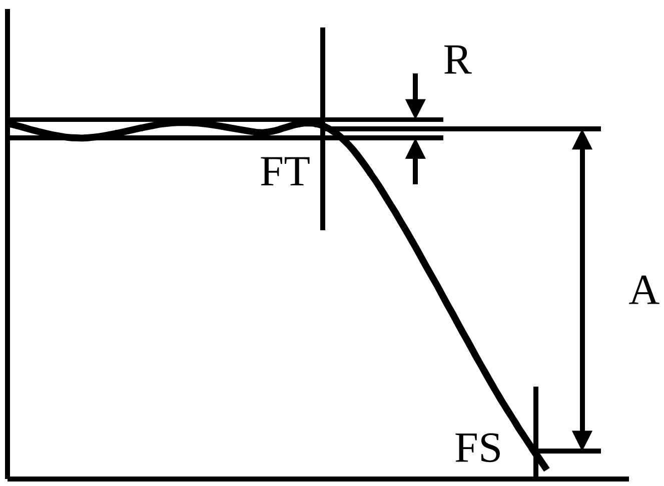


Figura 2-29. Parámetros necesarios para definir un filtro pasa bajos: FT frecuencia de corte (banda transmitida). FS frecuencia de supresión (banda suprimida). R rizado. A atenuación.

Filtros FIR

Un sistema FIR es un sistema lineal invariante en el tiempo que tiene una respuesta impulsional de duración finita (FIR *finite-duration impulse response*) en contraposición a los sistemas IIR que presentan una respuesta infinita (IIR *infinite-duration impulse response*). Un filtro FIR está definido en la forma:

$$Y_i = \sum_k b_k \; X_{i-k}$$

La implementación de este tipo de filtros precisa la ejecución de M-1 posiciones de memoria para guardar los datos de entrada, M multiplicaciones y M-1 sumas, por lo que sólo es posible implementarlos eficazmente en sistemas dotados de procesador matemático o especialmente diseñados para el tratamiento de señales como son los DSP (*Digital Signal Processor*). Como este tipo de filtro es simétrico (o antisimétrico) es posible reducir a la mitad el número de multiplicaciones utilizando más memorias para guardar los productos intermedios. La respuesta impulsional del filtro es idéntica a los coeficientes del filtro. Esto permite partir de la respuesta del filtro y mediante FFT inversa calcular los coeficientes, debiéndose realizar sucesivas aproximaciones hasta obtener un resultado aceptable. Es conveniente empezar con un número muy grande de coeficientes e ir reduciéndolo hasta alcanzar un compromiso entre respuesta y número de coeficientes. Actualmente existen muchas utilidades que permiten calcular en forma sencilla estos coeficientes y la mayoría de los programas de cálculo matemático, junto con las utilidades de análisis espectral (FFT) incluyen métodos para determinar los coeficientes de este tipo de filtros. Como ejemplo de filtros FIR se muestran dos filtros pasa bajos y un filtro pasa banda

Filtro pasa-bajos	
Frecuencia de muestreo	50 Hz.
Frecuencia de corte	16 Hz.
Frecuencia de supresión	25 Hz.
Rizado en banda transmitida	1 db
Atenuación	200 db
N	Coeficiente FIR
0	0.045189931861
1	-0.149673814469
2	0.613367234529
3	0.613367234529
4	-0.149673814469
5	0.045189931861

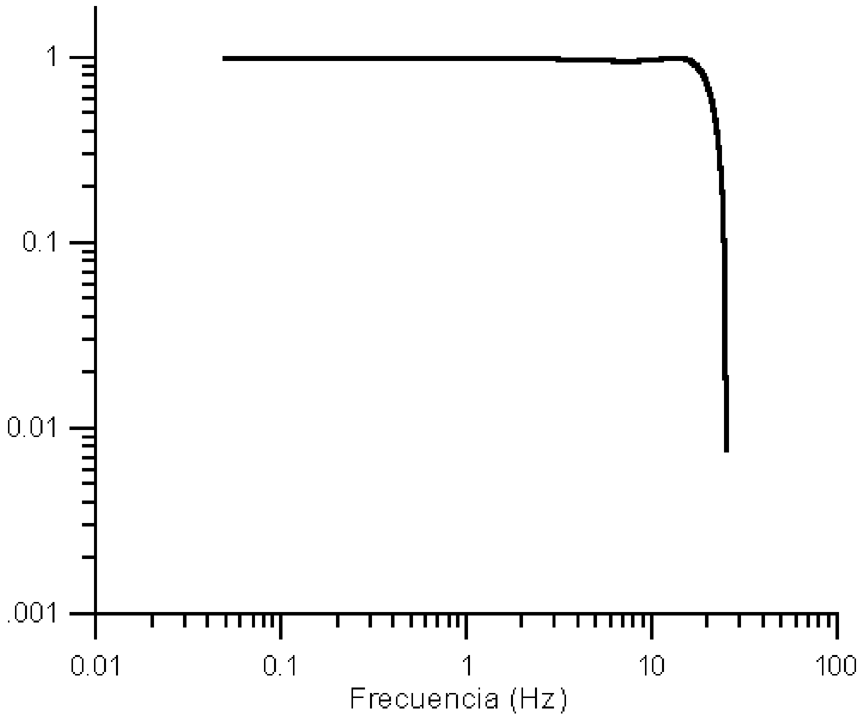


Figura 2-30. Filtro FIR pasa bajos de 16 Hz. realizado con seis coeficientes.

Filtro pasa-bajos	
Frecuencia de muestreo	50 Hz
Frecuencia de corte	1 Hz
Frecuencia de supresión	12 Hz
Rizado en banda transmitida	1 db
Atenuación	100 db
N	coeficiente
0	0.000216313943
1	0.001708293189
2	0.007096948054
3	0.020437565525
4	0.045217990231
5	0.080957510955
6	0.120756055191
7	0.152603096394
8	0.164834348339
9	0.152603096394
10	0.120756055191
11	0.080957510955
12	0.045217990231
13	0.020437565525
14	0.007096948054
15	0.001708293189
16	0.000216313943

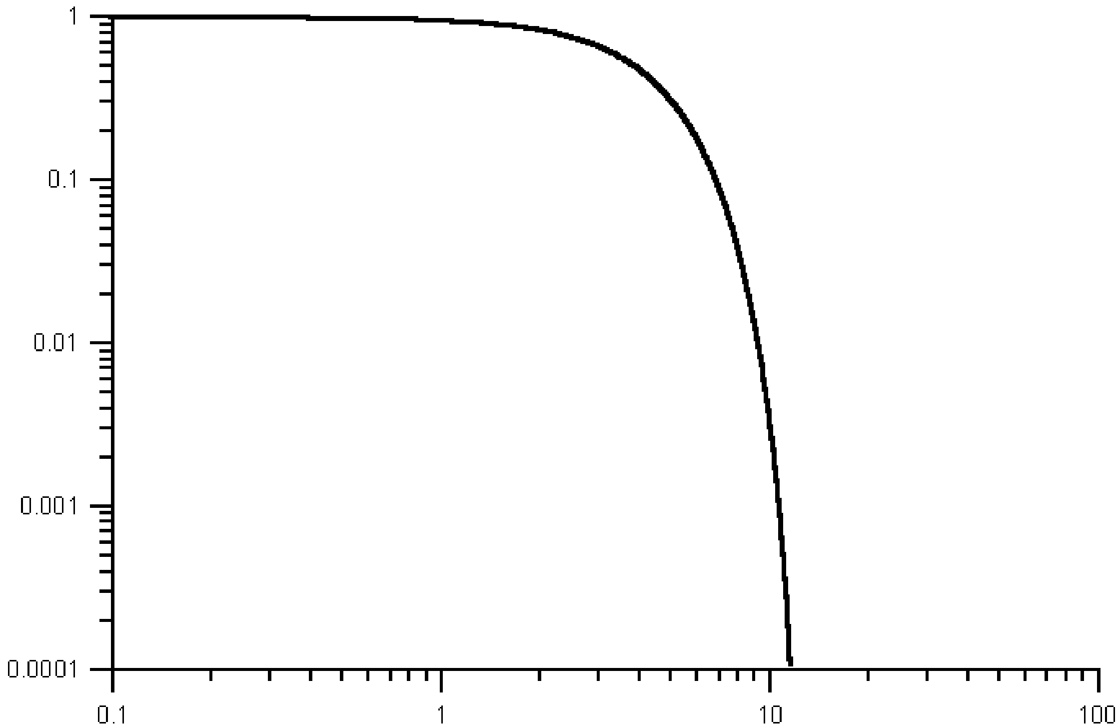


Figura 2-31. Filtro FIR pasa bajos de 1 Hz. realizado con diecisiete coeficientes.

No es fácil obtener la expresión analítica de la función de transferencia de este tipo de filtros y por consiguiente poder diseñar el filtro a partir de una serie de ecuaciones de diseño. Lo que lo más sencillo es implementar el filtro, obtener la respuesta a un impulso y calcular la FFT para obtener la respuesta en frecuencias, modificando los parámetros hasta obtener la respuesta deseada.

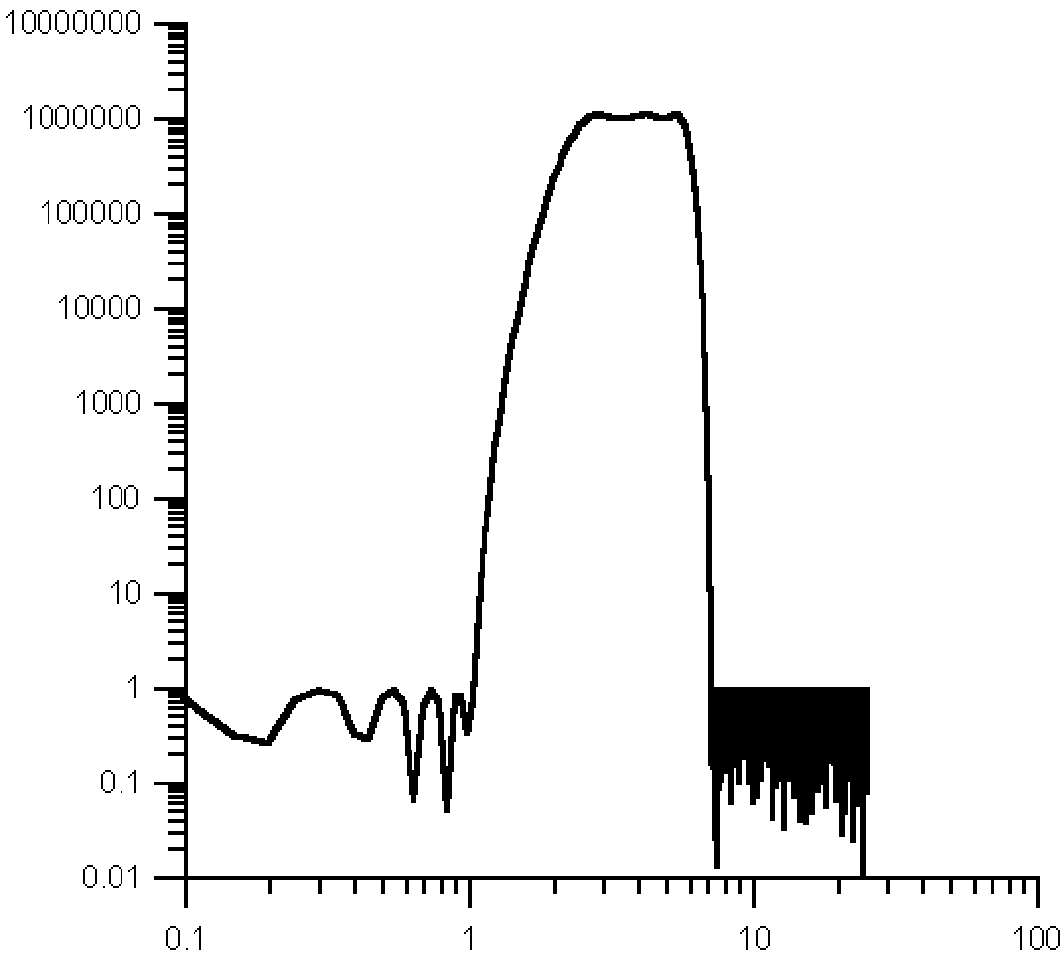
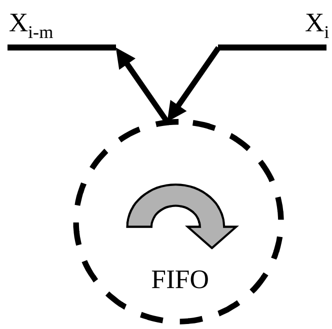


Figura 2-32. Filtro FIR pasa banda entre 2 Hz. y 6 Hz. realizado con ciento treinta y tres coeficientes. Obsérvese que fácilmente el número de coeficientes aumenta hasta hacerse inmanejable en aplicaciones en tiempo real que deben operar con pequeños sistemas de microproceso. Se ha incluido un número muy grande de décadas en el eje Y para poder mostrar la elevada atenuación que presenta el filtro fuera de la banda transmitida.

Figura 2-33. Implementación de un filtro digital de promedio utilizando una memoria FIFO (*First-In First -Out*) de longitud *m*. Primero se lee el contenido de la celda (X_{i-m}) y se resta del acumulador, después se escribe en dicha posición el dato (X_i) y se suma al acumulador S. El valor del acumulador es la salida del filtro. Este procedimiento es muy fácil de implementar, pero requiere memoria para almacenar m datos, lo cual puede ser problemático cuando se utilizan microcontroladores.



$$S = S + X_i - X_{i-m}$$

$$Y_i = S$$

Filtros FIR de fácil implementación

La utilización práctica de un filtro FIR pasa por resolver dos problemas: el cálculo de los coeficientes y la implementación en el procesador. La solución práctica al primer punto pasa por la utilización de un herramienta específica, mientras que el segundo requiere utilizar un procesador con un mínimo de capacidad para la realización de cálculo aritméticos y si el número de coeficientes es muy elevado, fácilmente el sistema se saturará o habrá que acudir a frecuencias de reloj muy elevadas con el consiguiente aumento de consumo. Sin embargo es posible utilizar filtros FIR de fácil implementación donde el número de operaciones a realizar se ha simplificado al máximo. El filtro digital FIR más sencillo es aquel cuyos coeficientes tienen todos el valor 1. Este filtro equivale a promediar la señal a lo largo de una ventana móvil, es decir sustituye el valor de la señal por el valor medio de un determinado número de datos.

$$Y_i = \sum_{j=0}^{m} X_{i-j}$$

donde X_i es la serie de datos de entrada y Y_i la serie de datos filtrados. El índice m es el número de datos que se promedia. Para poder realizar fácilmente el valor medio se elige el número de datos igual a una potencia de 2.

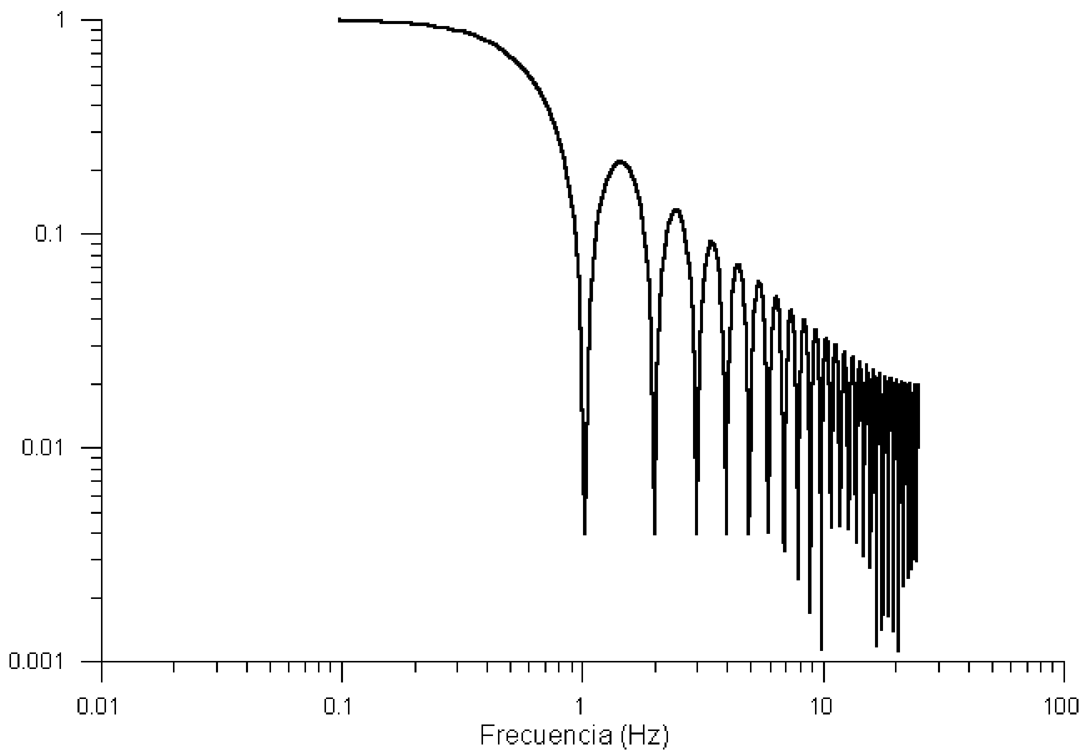


Figura 2-34. Respuesta de un filtro de promedio con una anchura de 50 muestras y frecuencia de muestreo de 50 Hz. Obsérvese la fuerte oscilación que se presenta fuera de la banda transmitida.

Una forma rápida de implementar este filtro consiste en utilizar una memoria circular de longitud m, que inicialmente tienen todos los datos a 0 y la salida (suma de todos los elementos) es obviamente 0. En cada paso el dato de entrada se suma a la salida y se resta el dato que entro hace m pasos. De esta forma sólo se requiere utilizar una operación de suma y una de resta. Es preciso destacar el comportamiento de este filtro en la banda suprimida, en la que aparecen lóbulos cuya amplitud v disminuyendo paulatinamente, respondiendo a una ley del tipo:

$$|H(s)| = \frac{|\sin(x)|^n}{x^n}$$

Este tipo de respuesta puede presentar problemas en algunas aplicaciones y en general no es tenido en cuenta. Por ejemplo muchos de sistemas de adquisición de datos por conteo de impulsos tienen este tipo de respuesta. La respuesta en frecuencia de muchos conversores de tecnología delta sigma es también de este tipo. En todos estos casos es conveniente limitar la respuesta en frecuencia, intercalando un filtro pasa bajos analógico entre el sensor y el conversor analógico digital.

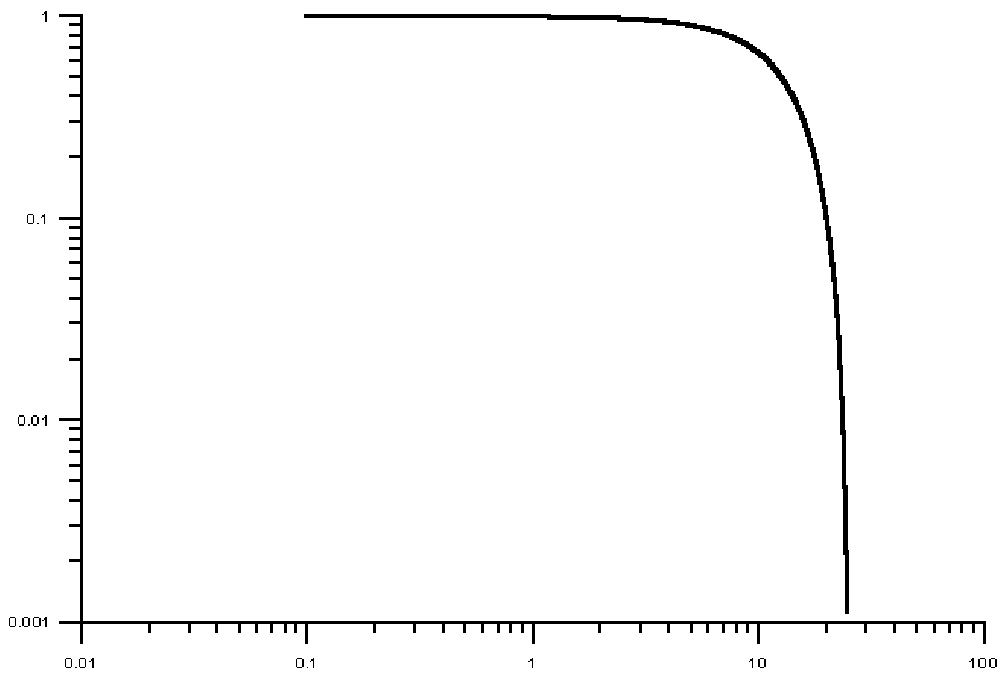


Figura 2-35. Respuesta de un filtro Daniel.

Otro filtro de FIR y también muy fácil de implementar es el filtro triangular o filtro Daniel. Este filtro tienen una longitud de 3 y los coeficientes son:

Filtro Daniel o triangular		
A0	A1	A2
0.25	0.5	0.25

Este filtro consiste en sumar un número impar de muestras divididas por una potencia de 2, por ejemplo:

$$Y_i = \frac{1}{4}(X_{i-1} + X_{i+1}) + \frac{1}{2}X_i$$

Como puede observarse en su realización práctica este filtro sólo requiere operaciones de suma y desplazamiento, por lo que es muy sencilla su codificación en el lenguaje ensamblador de los microcontroladores. Este filtro se utiliza frecuentemente para reducir el ruido de alta frecuencia en sistemas sencillos de registro sísmico.

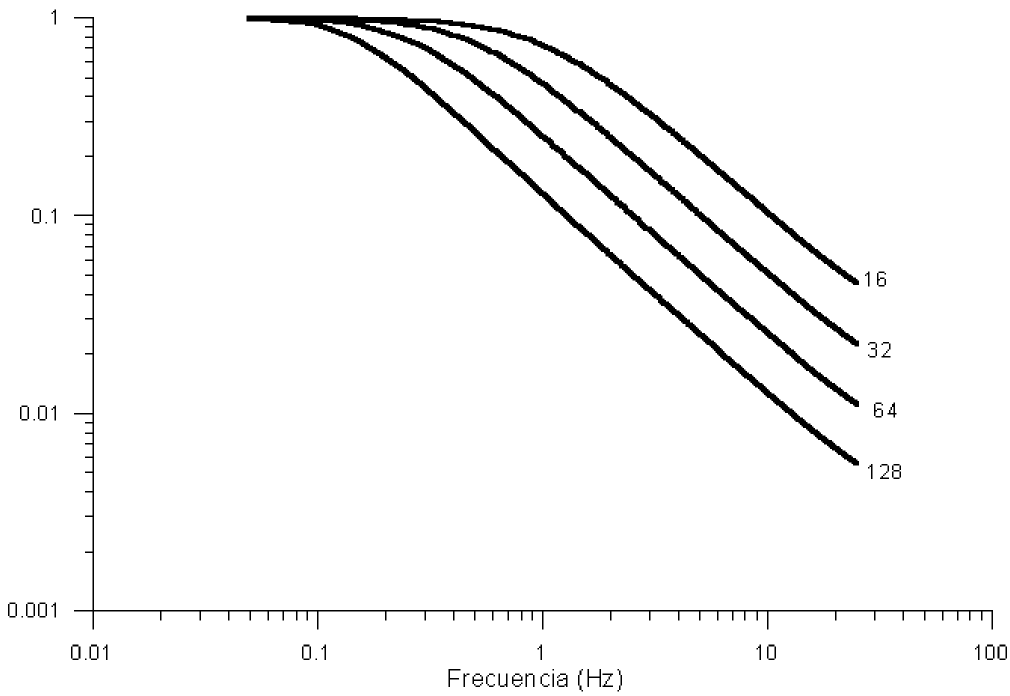


Figura 2-36. Respuesta de cuatro filtros de recurrencia Y = Y – (X – Y)/K con constantes de 16, 32, 64 y una frecuencia de muestreo de 50 Hz. Obsérvese la regularidad que presenta fuera de la banda transmitida.

Filtros IIR

Esta familia de filtros (IIR *infinite-duration impulse response*) presenta respuesta impulsional infinita. En su forma más general un filtro IIR está definido como:

$$Y_i = \sum_k a_k \; X_{i-k} \quad + \quad \sum_k b_k \; Y_{i-k}$$

Es decir la señal filtrada se obtiene a partir de la serie de datos de la señal de entrada y de la serie filtrada. Mientras que los filtros FIR eran intrínsecamente estables, los filtros IIR pueden presentar oscilaciones o bien la salida tender a infinito (saturaciones). El filtro IIR más sencillo está definido mediante una relación de recurrencia de la forma:

$$Y_i = Y_{i-1} + \frac{1}{K}(X_i - Y_{i-1})$$

donde *K* es la constante del filtro. En general se hace K igual a una potencia de dos para poder realizar la división por simple desplazamiento de bits. De acuerdo con la definición de los coeficientes del filtro IIR sería:

Filtro de desplazamiento	
A0	B0
1/K	1-1/K

Es muy fácil realizar un filtro pasa altos a partir de un filtro pasa bajos (siempre que la amplificación en la banda transmitida sea 1), pues basta con restar de la señal original la señal filtrada:

$$Z_i = X_i - Y_{i-1}$$

Este tipo de filtros de muy sencilla implementación se utiliza frecuentemente para cancelar el offset en sistemas de registro sísmico con valores de 1/K superiores a varios minutos. También se utiliza para calcular rápidamente los valores medios de la señal en algoritmos de disparo del tipo LTA/STA.

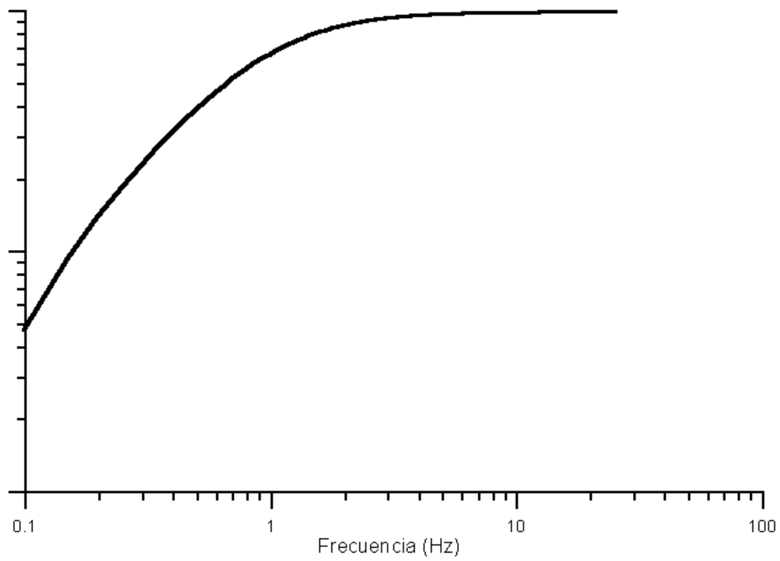


Figura 2-37. Respuesta de un filtro pasa altos construido a partir de un filtro de recurrencia Y = Y – (X – Y)/K con K = 16.

Consideraciones sobre filtrado

Un filtro modifica el contenido espectral de una señal, pero no hay que olvidar que ello también supone importantes cambios en la representación temporal de la señal y esto no siempre se tiene en cuenta. Por ejemplo una señal muy impulsiva aparece como emergente después de ser filtrada o se produce un desplazamiento temporal del máximo de la señal en función de la longitud del filtro y de su implementación. Estos efectos son especialmente dramáticos cuando se utilizan filtros con pendientes de corte elevadas y pueden conducir a importantes errores en la estimación de los tiempos de llegada o de las polaridades en el análisis de las señales sísmicas.

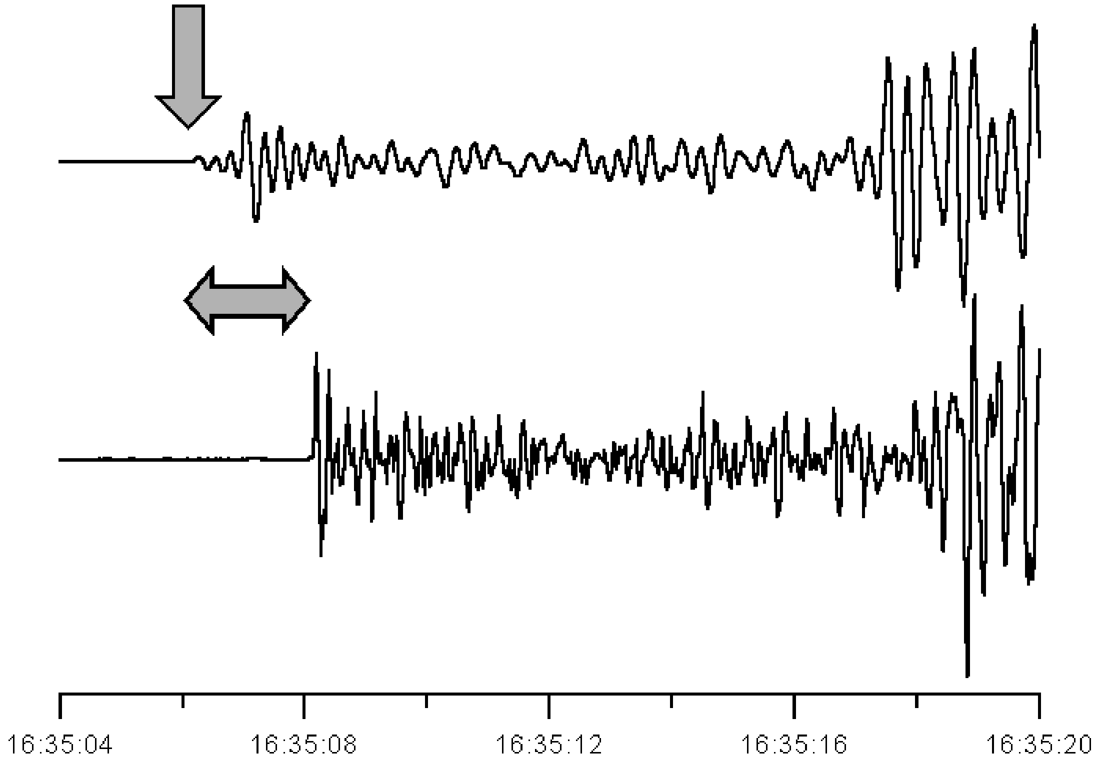


Figura 2-38. Efectos del filtrado mediante un filtro FIR pasa bajos de 111 coeficientes del inicio de un sismo tectónico ocurrido en Lanzarote el 22 de enero de 2001 a las 16:35. Las flechas muestran la transformación del inicio de la señal de impulsiva a emergente y el desplazamiento temporal debido a una mala implementación del filtro.

Finalmente, recordar que estos efectos también los producen los filtros analógicos, pues son inherentes al propio concepto de filtrado como modificación del contenido espectral de la señal.