PROCESAMIENTO DIGITAL DE SEÑALES

PROFESOR: JORGE ANTONIO POLANÍA PUENTES

CONTENIDO

1. BASE MATEMÁTICA

- **SEÑALES Y SISTEMAS DISCRETOS**
- **TRANSFORMADA Z**
- **TRANSFORMADA RÁPIDA DE FOURIER**

2. DISEÑO DE FILTROS

- **FILTROS DIGITALES IIR**
- **FILTROS DIGITALES FIR**
- **ESTRUCTURA DE FILTROS**

3. IMPLEMENTACIÓN Y APLICACIONES

- **MICROPROCESADORES DSP**
- **IMPLEMENTACIÓN DE FILTROS IIR**
- **IMPLEMENTACIÓN DE FILTRO FIR**

BIBLIOGRAFÍA

PROAKIS & MANOLAKIS. Tratamiento digital de señales. Printice Hall

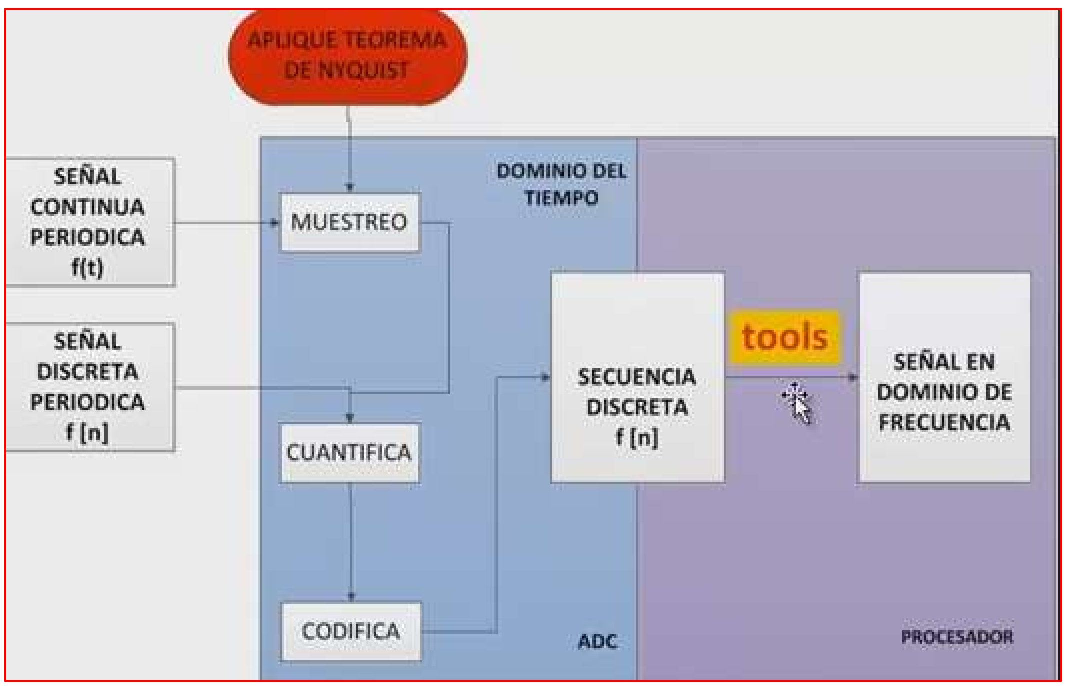
OPPENHEIM & SCHAFER. Digital signal processing. Prentice Hall.

BURRUS & OPPENHEIM. Ejercicios de tratamiento de la señal utilizando Matlab. Prentice Hall.

LUDEMAN LONNIE. Fundamentos de procesamiento digital de señales. Harper &Roe Publishers.

PARKS & BURRUS. Digital filter design. John Wiley & sons.

INTRODUCCIÓN



El procesamiento digital de señales es la aplicación de una serie de operaciones lógicas y matemáticas a un conjunto de datos provenientes de una señal que se implementa en un microprocesador de señales.

El objetivo de este procesamiento pueden ser, por ejemplo, añadir una información a una señal portadora para que sea transmitida y se pueda recuperar más tarde en otro lugar (transmisión de radio o una conversación telefónica) o procesar una señal de voz (reconocimiento de palabras), proteger información sensible (encriptación) o detectar piezas defectuosas, mediante imágenes.

En el mundo tecnológico en el que vivimos estamos rodeados de sistemas de procesamiento de datos, como identificadores de productos del supermercado con un código de barras que es leído por una luz láser, modernas ecografías, en la industria del automóvil gracias a precisos autómatas que procesan y ejecutan complicadas tareas.

Pero el procesamiento de señal, por sí solo, no puede hacer muchas cosas. De entrada, hemos dicho que se trata de manipular señales. Por lo tanto hay que obtener primero estas señales, por lo que necesitaremos sensores que nos permitan leerlas. Otro paso será traducir estas señales que nos dan los sensores en señales eléctricas. Esto es lo que realizaremos mediante un transductor. Además, si la señal es analógica, es decir, continua en el tiempo y en amplitud, necesitaremos digitalizar la señal para acabar teniendo un conjunto de valores numéricos cada determinado intervalo de tiempo, que se podrán registrar en un microprocesador. Esta es la tarea de los dispositivos convertidores analógico-digitales.

Algunas de las principales aplicaciones del procesamiento de señal son, por ejemplo, el procesamiento de señales de audio, el procesamiento de imágenes digitales, la compresión de vídeo, el procesamiento del habla, el reconocimiento de voz, las comunicaciones digitales, el radar, la sismología y la biomedicina. Y como ejemplos concretos se puede mencionar la compresión y transmisión de la voz en teléfonos móviles digitales, la ecualización del sonido en equipos de alta fidelidad, el procesamiento de datos sísmicos, el control de procesos industriales, la minería de datos para la detección de fraudes en las tarjetas de crédito, las animaciones generadas por computador en las películas, la compresión MP3 o las imágenes médicas como las de resonancia magnética funcional.

A menudo es deseable que estos sistemas funcionen en tiempo real, lo que significa que el sistema en tiempo discreto se implementa de forma que las muestras de salida se calculan a la misma velocidad a la que se muestrea la señal en tiempo continuo. El tratamiento en tiempo discreto y en tiempo real de señales en tiempo continuo es práctica común en sistema de control, comunicaciones, radar, sonar, codificación y realce de voz y vídeo, ingeniería biomédica y un largo etcétera.

Otro tipo de problemas del tratamiento de señales al que se enfrenta es la interpretación de señales. Por ejemplo, en un sistema de reconocimiento de voz el objetivo es comprender la señal de entrada. Típicamente, un sistema como éste aplicará un procesado digital previo (filtrado, estimación de parámetros, etc.) seguido por un sistema de reconocimiento de patrones que produzca una representación simbólica.

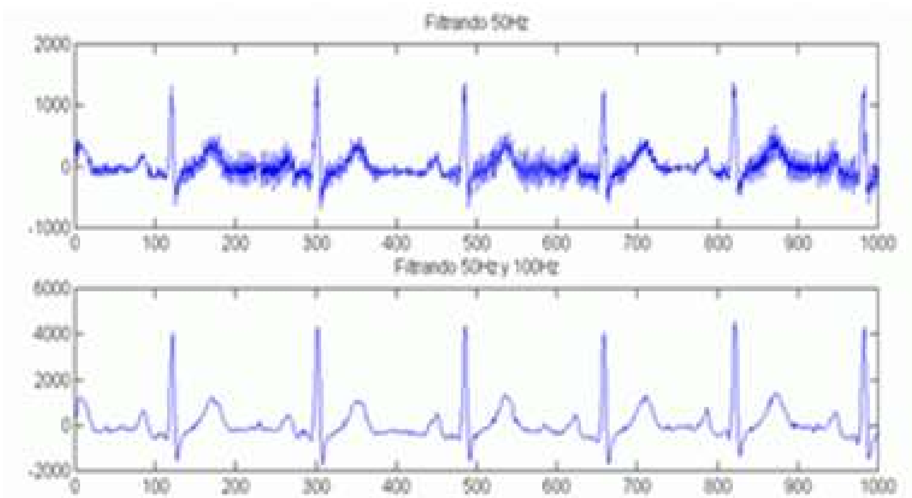
Los problemas de tratamiento de señales no están confinados, por supuesto, a señales unidimensionales. Aunque hay algunas diferencias fundamentales entre las teorías del tratamiento de señales unidimensionales y multidimensionales, una buena parte del material que se presenta aquí tiene su contrapartida en sistemas multidimensionales. Entre ellas destaca las aplicadas al procesamiento de imágenes digitales.

Aunque el procesamiento de señales mediante computadores digitales ofrecía tremendas ventajas de flexibilidad, sin embargo, el procesado no se podía realizar en tiempo real. Las aportaciones de Cooley y Tukey (1965) de un algoritmo eficiente para el cálculo de las transformadas de Fourier aceleró el uso del computador digital. Muchas aplicaciones desarrolladas requerían del análisis espectral de la señal y con las nuevas transformadas rápidas se redujo en varios órdenes de magnitud el tiempo de cómputo. Además, se dieron cuenta de que el nuevo algoritmo se podría implementar en hardware digital específico, por lo que muchos algoritmos de tratamiento digital de señales que previamente eran impracticables comenzaron a verse como posibles.

Otro desarrollo importante en la historia del Procesamiento de Señales ocurrió en el terreno de la Microelectrónica. Aunque los primeros microprocesadores eran demasiado lentos para implementar en tiempo real la mayoría de los sistemas en tiempo discreto, a mediados de los ochenta la tecnología de los circuitos integrados había avanzado hasta el nivel de permitir la realización de microcomputadores en punto fijo y punto flotante con arquitecturas especialmente diseñadas para realizar algoritmos de procesamiento de señales en tiempo discreto. A estos procesadores se les conocen como DSP (*Digital Signal Processor*). Con esta tecnología llegó, por primera vez, la posibilidad de una amplia aplicación de las técnicas de tratamiento de señales en tiempo discreto.

De los diferentes algoritmos que involucra el procesamiento digital de señales, en este curso sólo se va a tratar los filtros digitales lineales como sistemas invariantes en el tiempo, abreviados por la sigla LTI (*Linear Time Invariant*).

En la figura se observa un electrocardiograma al que se le ha aplicado un filtrado para quitar el ruido de la señal.



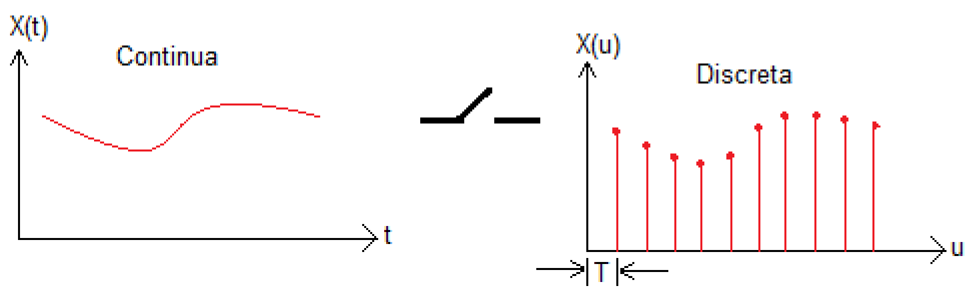
UNIDAD 1. BASE MATEMÁTICA

- **SEÑALES Y SISTEMAS DISCRETOS**
- **TRANSFORMADA Z**
- **TRANSFORMADA RÁPIDA DE FOURIER**

1. SEÑALES Y SISTEMAS DISCRETOS

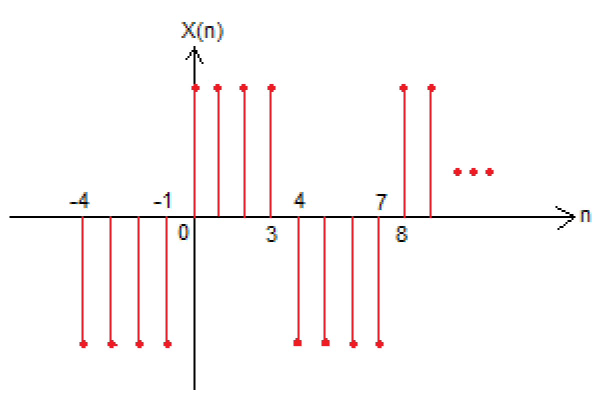
1.1 SEÑALES DISCRETAS

Una señal discreta es una señal continua muestreada a una tasa uniforme. Es una sucesión que es una función definida para números enteros positivos y negativos de tiempo.



T: Periodo de muestreo

SEÑAL PERIÓDICA



N=8

$$x(n) = x(n + N)$$

$N: Periodo$

$\Omega: Frecuencia\ fundamental$

$$\Omega = \frac{2\pi}{N} \rightarrow \Omega = \frac{2\pi p}{q}; \quad p\ y\ q = Z\ (Enteros)$$

Energía de una secuencia

$$E = \sum_{n=-\infty}^{\infty} x^2(n)$$

Potencia promedio

$$P = \lim_{N \to \infty} \frac{1}{2N} \sum_{n=-N}^{N} x^2(n)$$

Potencia de una señal periódica

$$P = \frac{1}{N} \sum_{n=0}^{N-1} x^2(n)$$

OPERACIONES DE LAS SEÑALES

$y(n) = cx(n), \qquad\qquad\qquad Escalonamiento\ en\ magnitud$

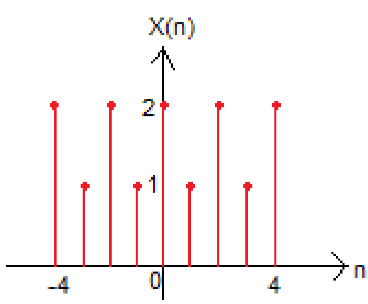
$y(n) = x_1(n) + x_2(n), \qquad Suma$

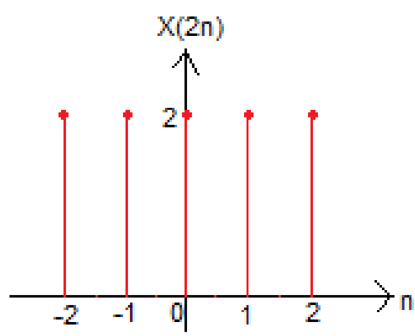
$y(n) = x_1(n) * x_2(n), \qquad Convolución$

$y(n) = x(kn), \qquad\qquad\qquad Escalamiento\ en\ tiempo$

EJEMPLO DE ESCALAMIENTO

Dada una secuencia x(n) hallar x(2n)





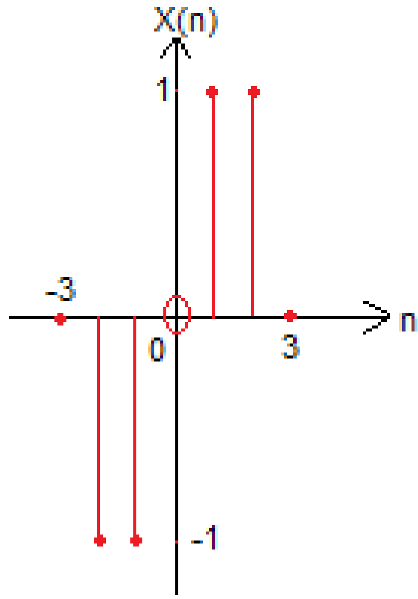
CORRIMIENTO

$y(n) = x(n + a) \rightarrow Izquierda$

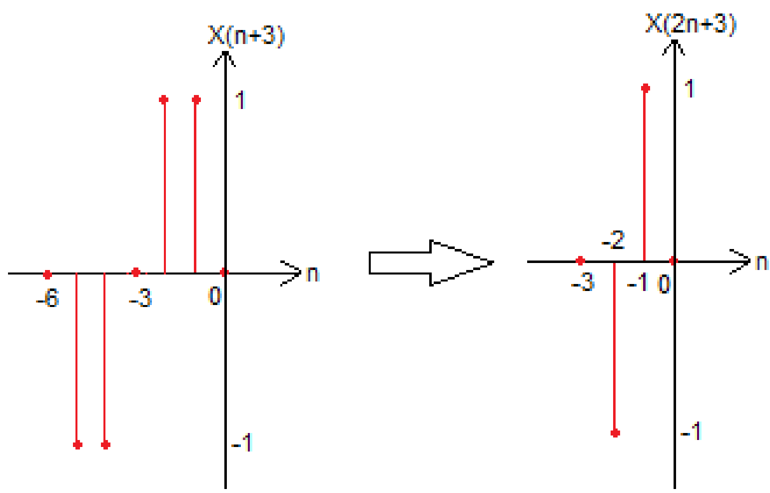
$y(n) = x(n - a) \rightarrow Derecha$

EJEMPLO DE CORRIMIENTO

Dado x(n), hallar $y(n) = x(2n + 3)$

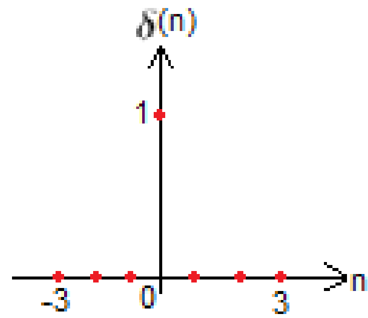


Nota: Primero se corre y luego se escala



SEÑALES FUNDAMENTALES

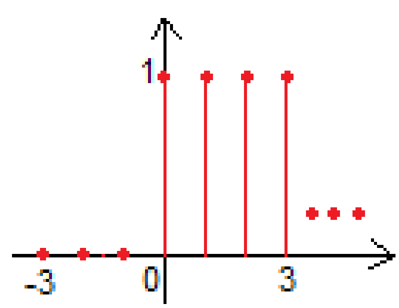
(a) Impulso unitario: $\delta(n)$



$\delta(0) = 1$

$\delta(n) = 0, si \neq 0;\quad \delta: Delta\ Kronecter$

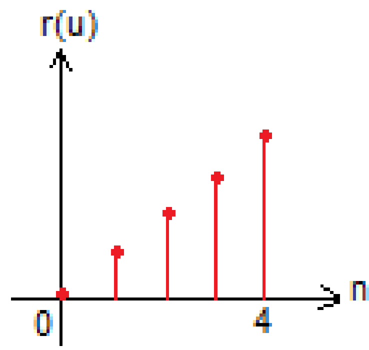
(b) Escalón unitario: $u(n)$



$u(n) = 1\ , n \geq 0$

$u(n) = 0\ , n < 0$

(c) Rampa unitaria: r(n)



$r(n) = n\ , n \geq 0$

$r(n) = 0\ , n < 0$

(d) Potencial: $x(n) = a^n$
(e) Exponencial: $x(n) = e^{-an}$
(f) Senoidal: $x(n) = Asen(\Omega n + \emptyset)$

EJEMPLO DE SUMA DE SECUENCIAS

Sean las señales:

$x_1(n) = sen(5\pi n), \quad x_2(n) = \sqrt{3}\,cos(5\pi n),$

Hallar $y(n) = x_1(n) + x_2(n)$

Solución

$y(n) = sen(5\pi n) + \sqrt{3}\,cos(5\pi n)\ \ (1)$

Si $y(n) = Asen(\Omega n + \emptyset).\ entonces,$

$y(n) = A[sen(\Omega n)cos\emptyset + sen\emptyset\,cos(5\pi n)]$

$y(n) = Acos\emptyset sen(\Omega n) + Asen\emptyset cos(\Omega n),$

$por\ comparación\ de\ (1)$

$\Omega = 5\pi$

$$y(n) = Acos\emptyset sen(5\pi n) + Asen\emptyset cos(5\pi n)$$

Nuevamente por comparación,

$$Acos\emptyset = 1\,; Asen\emptyset = \sqrt{3}\,; Dividiendo\ tan\emptyset = \sqrt{3} \rightarrow\ \emptyset = \pi/3$$

Elevando al cuadrado

$$A^2cos^2\emptyset = 1$$

$$A^2sen^2\emptyset = 3$$

Entonces:

$$A^2 = 4 \rightarrow A = 2$$

$$y(n) = 2sen\left(5\pi n + \frac{3}{\pi}\right)$$

REPRESENTACIÓN DE UNA SEÑAL DISCRETA

Una señal discreta se puede representar mediante impulsos unitarios de la forma,

$$x(n) = \sum_{k=-\infty}^{\infty} x(k)\ \delta(n-k),$$

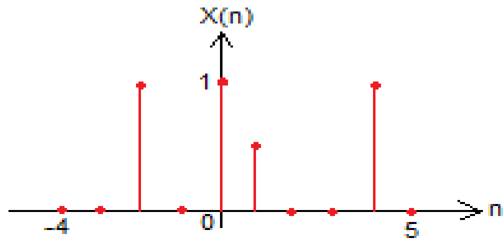
$$x(k)\ es\ la\ magnitud\ y\ \delta(n-k) es\ el\ corrimiento$$

Para el caso de un escalón unitario:

$$x(n) = \sum_{k=-\infty}^{\infty} \delta(n-k)$$

EJEMPLO

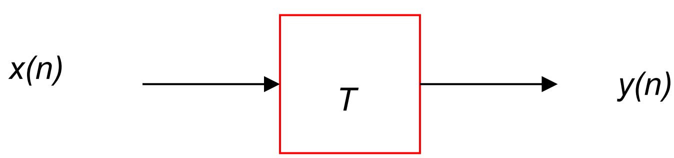
Representar la señal x(n) dada en forma de serie de impulsos



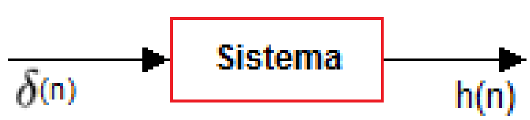
$x(n) = x(-2)\,\delta(n+2) + x(0)\,\delta(n) + x(1)\,\delta(n-1) + x(4)\,\delta(n-4)$

1.2 SISTEMAS DISCRETOS

Un sistema discreto es aquel que establece una correspondencia o transformación (T) entre una secuencia de entrada x(n) y una secuencia de salida y(n).

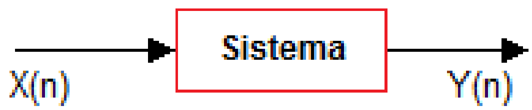


Respuesta del sistema al impulso unitario δ(n): h(n)



CONVOLUCIÓN

La salida de un sistema discreto y(n), con entrada x(n), se obtiene fácilmente a partir de la convolución de la entrada con su respuesta al impulso unitario h(n).



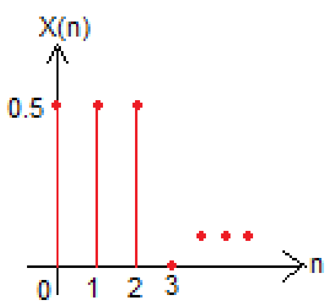
$$y(n) = x(n) * h(n) \rightarrow Convolución$$

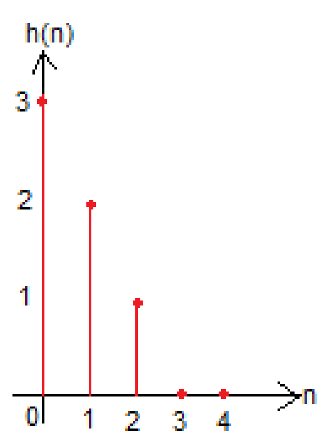
$$y(n) = \sum_{k=-\infty}^{\infty} x(k)h(n-k)$$

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

EJEMPLO DE APLICACIÓN DE LA CONVOLUCIÓN

Conocida la respuesta al impulso unitario h(n), hallar la respuesta del sistema $y(n)$ para una entrada x(n).

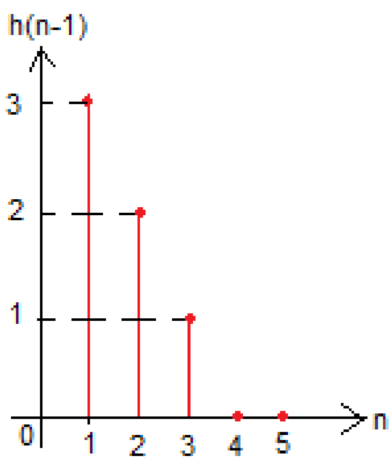


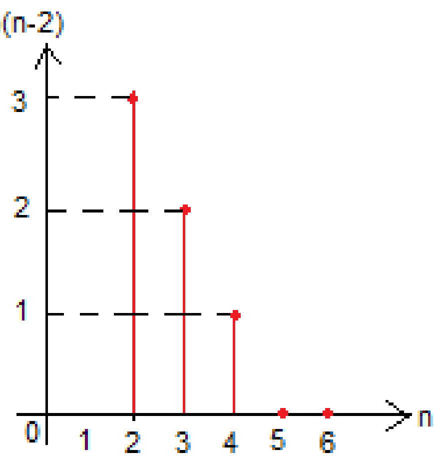


Solución

$$y(n) = x(n) * h(n) = \sum_{k=-\infty}^{\infty} x(k)h(n-k) = \sum_{k=0}^{2} x(k)h(n-k)$$

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





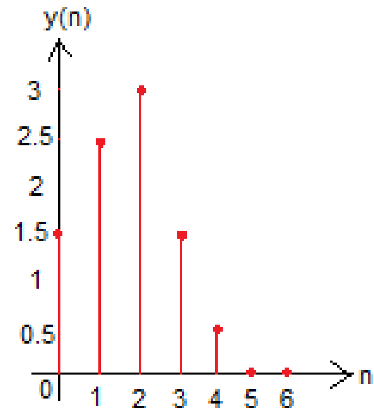
$$y(n) = 0.5[h(n) + h(n-1) + h(n-2)$$

$$y(n) = 1.5\,\delta(n) + 2.5\,\delta(n-1) + 3\,\delta(n-2) + 1.5\,\delta(n-3)$$

$$+0.5\,\delta(n-4)$$

Por ejemplo, para y(1),

y1)=0.5[h(1)+h(0)+h(-1)] = 0.5 [2 + 3 +0]= 2.5



PROPIEDADES DE LOS SITEMAS DISCRETOS

1. Una sucesión x(n) es <u>limitada</u> si existe un número M finito tal que,

$$|x(n)| \leq M$$

2. Un sistema discreto en el tiempo es <u>estable</u> si a una sucesión limitada x(n) se produce una sucesión de salida limitada y(n). La respuesta al impulso debe ser acotada.

$$\sum_{k=-\infty}^{\infty} |h(n)| < \infty$$

3. Un sistema DT es <u>causal</u> si la salida depende totalmente de las entradas presentes y pasadas, no de las entradas y salidas futuras.

$$h(n) = 0 \qquad para\; n < 0$$

4. Un sistema DT es <u>lineal</u> si,

$$T[a_1 x_1(n) + a_2 x_2(n)] = a_1 T[x_1(n)] + a_2 T[x_2(n)]$$

5. Un sistema es invariante en el tiempo, si

$$T[x(n] = y(n) \quad \rightarrow \quad T[x(n-n_0] = y(n-n_0)$$

6. Un sistema es sin memoria si salida depende únicamente de la entrada presente. Con memoria, la salida depende de los valores de las entradas pasadas.

EJEMPLOS

1. Determinar si la señal es periódica

a) $x(n) = 4cos(\pi n)$

$x(n+N) = 4cos[\pi(n+N)] = 4\cos(\pi n + \pi N)$

Si la señal es periódica, se debe cumplir que,

$4\cos(\pi n + \pi N) = 4\cos(\pi n + 2\pi k)$

Por comparación,

$\pi N = 2\pi k, \;\; entonces, \qquad N = 2k, \quad k = 1\ (entero\ mínimo)$

Por tanto el periodo N = 2, es periódica.

b) $x(n) = 4cos(3n)$

$$4\,cos\big(3(n+N)\big) = 4\cos(3n+3N) = 4\cos(3n+2\pi k)$$

No hay un valor entero de k para que 3N=2πk, No es periódica

2. Determinar las propiedades de los sistemas.

a) $y(n) + 2y(n-1) = x(n+1)$

Ecuación de coeficientes constantes: es lineal, invariante en el tiempo y no causal.

b) $y(n+1) + 4y(n) = 3x(n+1) - x(n-1)$

Es un LTI causal.

3. Encontrar el valor $y(n)$ de:

a) $y(n) + 0.5y(n-1) = 2x(n-1);\ \ x(n) = \delta(n);\ y(-1) = 0$

$y(n) = -0.5y(n-1) + 2x(n-1)$

$y(0) = -0.5y(-1) + 2\delta(-1) = 0$

$y(1) = -0.5y(0) + 2\delta(0) = 2$

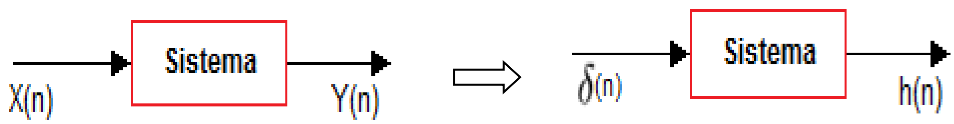
$y(2) = -0.5y(1) + 2\delta(1) = -1$

$\vdots$

4. Encontrar la respuesta al impulso: $h(n)$

a) $y(n) + 0.2y(n-1) = x(n) - x(n-1)$

$y(n) = -0.2y(n-1) + x(n) - x(n-1)$



$$h(n) = -0.2h(n-1) + \delta(n) - \delta(n-1)$$

$$h(0) = -0.2h(-1) + \delta(0) - \delta(-1) = 0 + 1 = 1;$$

$$el\ h(n) \rightarrow Siempre\ se\ considera\ despues\ de\ 0$$

$$h(1) = -0.2h(0) + \delta(1) - \delta(0) = -0.2(1) - 1 = -1.2$$

$$h(2) = -0.2h(1) + \delta(2) - \delta(1) = -0.2(-1.2) = 0.24$$

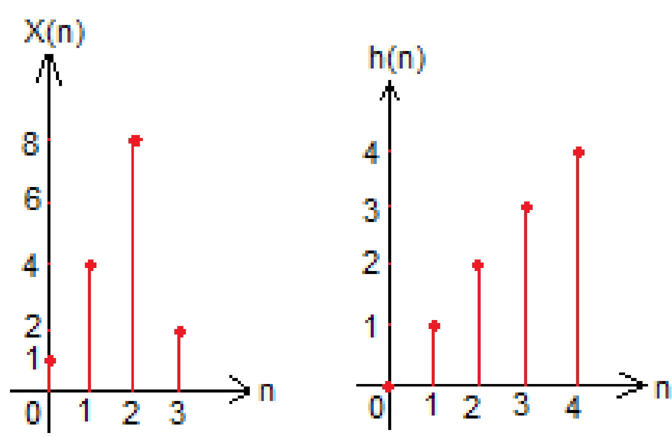
$$h(3) = -0.2h(2) = -0.2(0.24) = -0.048$$

$$\vdots$$

$$h(n) = (-0.2)^{n-1}(-1.2),\ \ para\ n \geq 1, \qquad h(n) = 1, n = 0$$

5. Realizar las convoluciones

a)

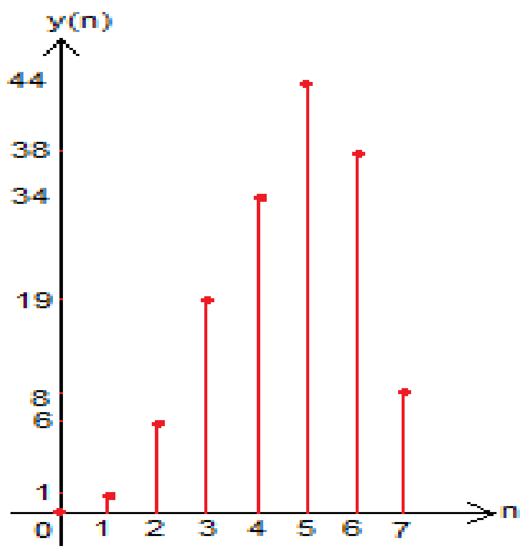


$$y(n) = x(n) * h(n) = \sum_{k=0}^{3} x(k)h(n-k)$$

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

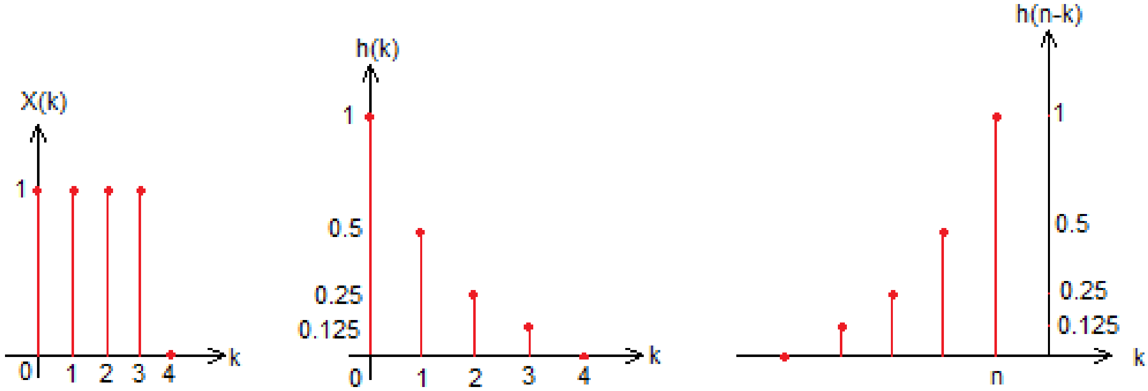
$$+x(3)h(n-3)$$

n	*0*	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*X(n)*
h(n)	*0*	*1*	*2*	*3*	*4*	-	-	-	*1*
h(n-1)	*0*	*0*	*1*	*2*	*3*	*4*	-	-	*4*
h(n-2)	*0*	*0*	*0*	*1*	*2*	*3*	*4*	-	*8*
h(n-3)	*0*	*0*	*0*	*0*	*1*	*2*	*3*	*4*	*2*
y(n)	*0*	*1*	*6*	*19*	*34*	*44*	*38*	*8*	-



b) $x(n) = u(n) - u(n-4)$

$h(n) = 0.5^n u(n)$



Para $0 \leq n < 4$

$$y(n) = \sum_{k=0}^{n} 0.5^{n-k} = \sum_{k=0}^{n} 0.5^{n} * 2^{k} = 0.5^{n} \sum_{k=0}^{n} 2^{k}$$

$$\Sigma a^{k} = a^{0} + a^{1} + a^{2} + \cdots + a^{n} = s$$

$$\underline{a^{1} + a^{2} + a^{3} + \cdots + a^{n+1} = as}$$, restando

$$1 - a^{n+1} = s - as$$

$$s(1-a) = 1 - a^{n+1}, \qquad s = \sum_{k=0}^{n} a^{k} = \frac{-1 + a^{n+1}}{a-1}$$

$$\sum_{k=0}^{n} 2^{k} = \frac{-1 + 2^{n+1}}{2-1} = -1 + 2^{n+1}$$

$$y(n) = 0.5^{n}[-1 + 2^{n+1}] = -0.5^{n} + 0.5^{n} * 2^{n+1}$$

$$y(n) = -0.5^{n} + 2$$

Para $n \geq 4$

$$y(n) = \sum_{k=0}^{3} 0.5^{n-k} = \sum_{k=0}^{3} 0.5^{n} * 2^{k} = 0.5^{n} \sum_{k=0}^{3} 2^{k}$$

$$y(n) = \frac{-1 + 2^{4}}{2 - 1} = 15(0.5)^{n}$$

LABORATORIO 1: SEÑALES Y SISTEMAS DISCRETAS

1. GENERACIÓN DE SEÑALES

a) Pulso unitario:

```
n = [0:10]';                   % vector tiempo
imp = [1; zeros(10,1)];        %vector impulso
stem(n, imp)                   %gráfica señal discreta
axis([0 10 0 2])
```

b) Paso unitario:

```
n = [0:10]';                    % vector tiempo
paso= ones(11,1);               %vector paso unitario
stem(n, paso)                   %gráfica señal
discreta
axis([0 10 0 2])
```

c) Rampa unitaria:

```
n = [0:0.1:1]';                % vector tiempo
ramp = n;                      %vector rampa unitaria
stem(n, ramp)                  %gráfica señal discreta
axis([0 1 0 2])
xlabel(' Tiempo (n)')
ylabel('r(n)')
title('Rampa')
```

d)Señal cuadrática:

```
n = [0:0.1:0.9]';             % vector tiempo
cuadr = n.^2;                 %cuadrática
stem(n, cuadr)                %gráfica señal discreta
axis([0 1 0 2])
xlabel(' Tiempo (n)')
ylabel('x(n)')
title('cuadratica')
```

e)Señal exponencial:

```
n = [0:0.1:9.9]';
xn= (0.5).^n;
stem(n, xn)
axis([0 10 0 2])
xlabel(' Tiempo (n)')
ylabel('x(n)')
title('x(n)=a^n')
```

2. GENERACIÓN DE SEÑALES PERIÓDICAS

a)Generar un diente de sierra de 1.5 seg de 50 Hz con tasa de muestreo de 1KHz y graficarla 0.2 seg.

```
fs = 1000;                   %frecuencia de muestreo
f = 50;                      %frecuencia de la señal
n = [0:1/fs:1.5]';           %vector tiempo
x = sawtooth(2*pi*f*n);
stem(n, x,'g')
axis([0  0.2  0  1]);
xlabel(' Tiempo (n)')
ylabel('x(n)')
title('Diente de sierra')
```

b)Generar una señal senoidal de 30 Hz de frecuencia y amplitud de 2, frecuencia de muestreo de 1000 Hz. Grafique los primeros 50 valores.

```
fs = 1000;
f = 30;
t = [0:1/fs:1.0]';
y = 2*sin(2*pi*f*t);
```

```
stem(t(1:50), y(1:50))
```

c)Adicionar a la señal un ruido senoidal de f = 400 Hz de amplitud de 0.5

```
fr = 400;
ruido = 0.5*sin(2*pi*fr*t)
yr = y + ruido
plot(t(1:50), yr(1:50))
```

d)Graficar la señal x(n)=0, para n<2; x(n)=2n-4, para 2<=n<4; x(n)=4-n, para n>=4

```
n1=-6:1;     %Vector primera condición
x1=zeros(1,length(n1)); %Coloca ceros desde -6 a 1
n2=2:3;      %Vector segunda condición
x2=2*n2-4;
n3=4:8;      %Vector tercera condición
x3=4-n3;
n=[n1 n2 n3]; %Vector de valores que puede tomar n
x=[x1 x2 x3]; %vector de la evaluación de la función
stem(n,x)
axis([-2 10 -5 3])  %límites de los ejes
grid                %rejilla de la grafica
```

3. CONVOLUCIÓN

a)Realizar la convolución de: a = [0.5 0.5 0.5] y b = [3.0 2.0 1.0]

```
n = 0:10;
a = [0.5  0.5  0.5];
b = [3.0  2.0  1.0];
c = conv(a,b);
stem(c)
```

b) Respuesta de un sistema conocido h(n)

```
xn=[1 4 8 2]; %Vector de la función Xn
hn=[0 1 2 3 4]; %Vector de la función hn
yn=conv(xn,hn)
```

```
c)Hallar la respuesta en frecuencia y(n), si
x(n)=u(n)-u(n-4), h(n)=0.5^n.u(n) para 10 muestras
de xn

nx=0:9;
L=length(nx);
u1=[ones(L,1)]';
u2=[zeros(4,1); ones(L-4,1)]';
xn=u1-u2;
for k=1:10
    h(k)=(0.5).^(k-1);
end
yn=conv(xn,h);
m=length(yn);
ny=0:(m-1);
stem(ny,yn)
```

NOTA: Informe de los resultados al correo: joanpola@gmail.com