DISCRETE-TIME SYSTEMS

A system is a transformation of signals.

A system is an input-output relationship.

A SISO system

$$x[n] \longrightarrow \boxed{T\{\cdot\}} \longrightarrow y[n]$$

$$y[n] = T\{x\}$$

In general, the output sample $y[n]$ at discrete-time instant n is a function of a set of samples of the input signal $x[\cdot]$.

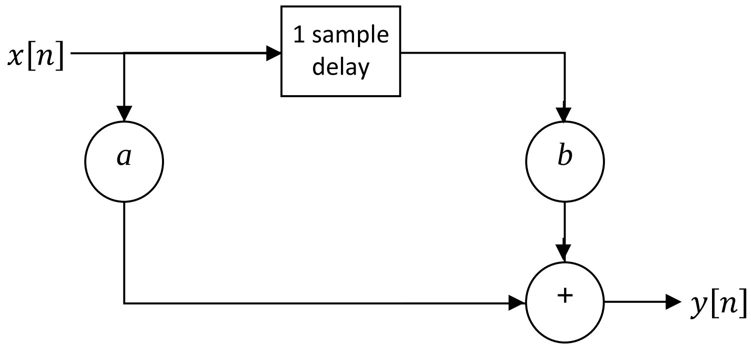
Ex: A delay system

$$y[n] = x[n - \Delta] ,$$

Δ is an integer.

Ex:

$$y[n] = ax[n] + bx[n-1]$$



This is an example of a linear, time-invariant, finite impulse response (FIR) system.

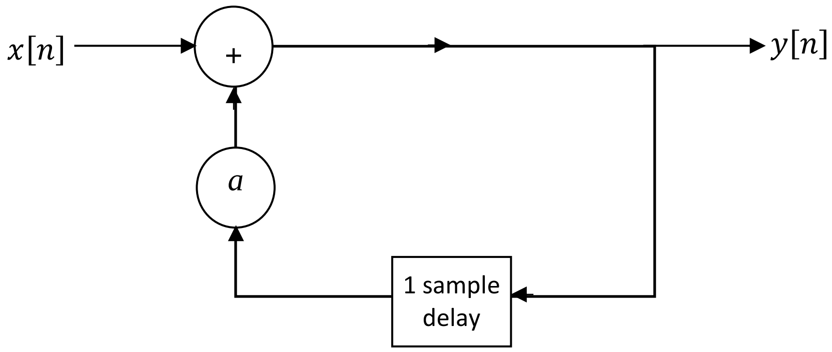
In general, an LTI, FIR system

$$y[n] = \sum_{N_1}^{N_2} a_k x[n-k]$$

Ex:

$$y[n] = ay[n-1] + x[n]$$

This is a recursive expression.



This is an example of a linear, time-invariant, infinite impulse response (IIR) system.

Equivalently its nonrecursive form is,

$$y[n] = \sum_{k=-\infty}^{n} x[k]$$

Note that not all recursive input-output relation expressions have their nonrecursive counterparts.

CLASSIFICATION OF SYSTEMS

- With memory - memoryless
- Linear - nonlinear
- Time-invariant – time-varying
- Causal-noncausal
- Stable-unstable

A Quotation from a Recent Research Paper:

“Null Space Component Analysis for Noisy Blind Source Separation”

“The solutions for the BSS problem were investigated under various source signal mixing models. Initially, linear instantaneous (**memoryless**) mixing models were used [3], followed by linear **convolution** mixing models [4]. More recently, nonlinear mixing models [5, 6, 7], bounded component analysis [8, 9], and the sparsity-based approach [10, 11] have been exploited.”

WITH MEMORY - MEMORYLESS

$$y[n] = x[n]$$

$$y[n] = 3x[n]$$

$$y[n] = 4^{x[n]}$$

are memoryless

whereas

$$y[n] = x[n-1]$$

$$y[n] = x[n+1]$$

$$y[n] = x[n] + x[n-1]$$

have memory

You have heard or you will hear about “dynamic systems”; they have memory.

LINEARITY

A system, $T\{\cdot\}$, is said to be linear if it satisfies

a) additivity: $T\{x_1[n] + x_2[n]\} = T\{x_1[n]\} + T\{x_2[n]\}$

b) homogeneity: $T\{ax[n]\} = aT\{x[n]\}$

Ex:

$$y[n] = \sum_{k=-\infty}^{n} x[k] \qquad \text{is linear.}$$

$$y[n] = \log(|x[n]|) \qquad \text{is nonlinear.}$$

$$y[n] = x[n] + 3 \qquad \text{is nonlinear.}$$

Proof: Exercise

TIME-INVARIANCE

Let

$$y_1[n] = T\{x[n]\}$$

and

$$y_2[n] = T\{x[n - N_0]\}$$

be the outputs of the system to $x[n]$ and $x[n - N_0]$, respectively.

Then, if

$$y_2[n] = y_1[n - N_0]$$

the system is said to be time-invariant.

Ex: Compressor (downsampler!)

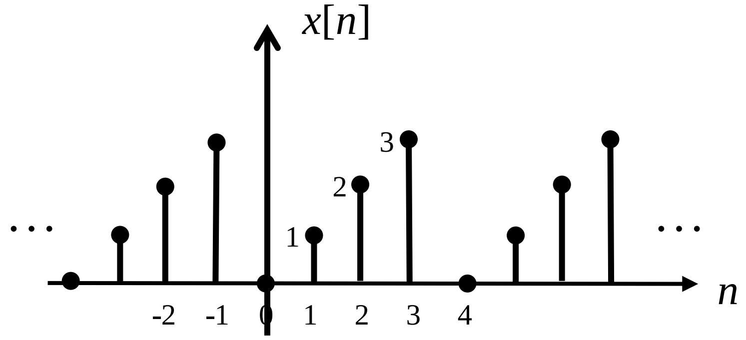
$$y[n] = x[Mn] \qquad M \in Z,\ M > 1$$

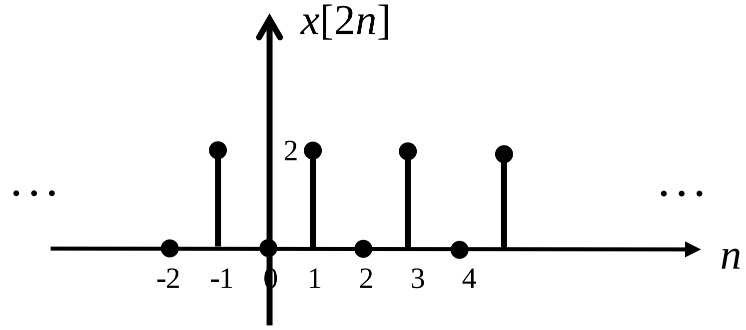
$$x[n] \longrightarrow \boxed{\downarrow M} \longrightarrow y[n]$$

Is it time-invariant?

Before answering the question, let's see what a compressor is.

Compressor:





$M = 2$

$$
\begin{gathered}
\vdots \\
y[-3] = x[-6] \\
y[-2] = x[-4] \\
y[-1] = x[-2] \\
y[0] = x[0] \\
y[1] = x[2] \\
y[2] = x[4] \\
y[3] = x[6] \\
\vdots
\end{gathered}
$$

Is it time-invariant?
Answer:

Following the above definition, let

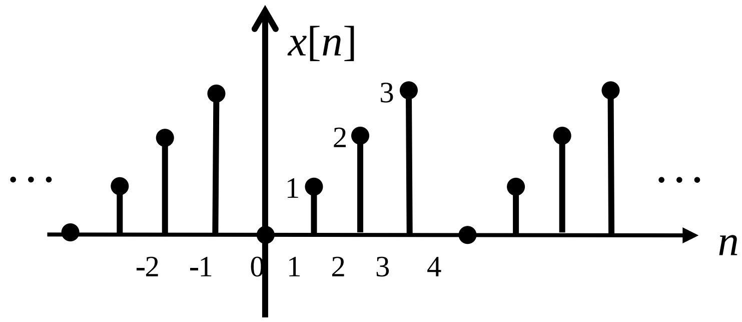
$$y_1[n] = x[Mn]$$
$$y_2[n] = x[Mn - N_0]$$

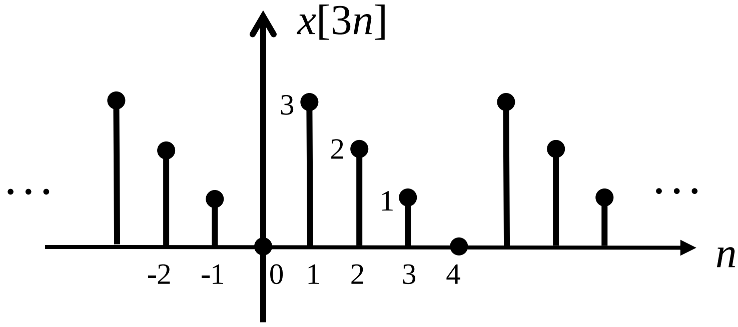
Since

$$y_2[n] \neq y_1[n - N_0] = x[Mn - MN_0]$$

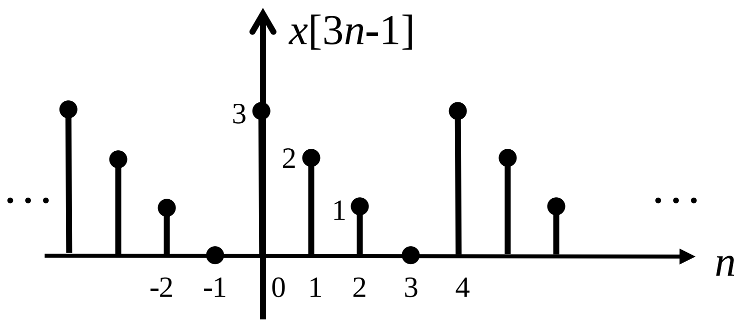
compressor is time-varying.

Compressor is time-varying:

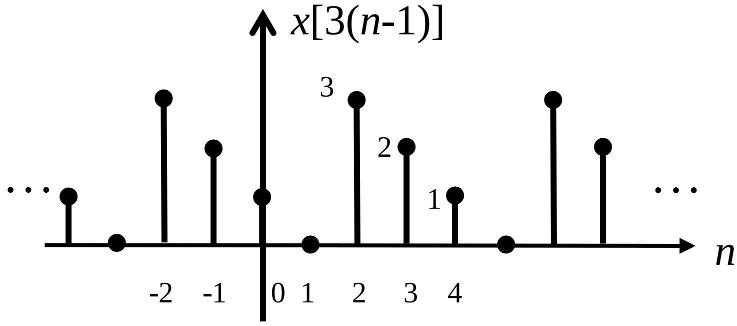




$M = 3$



The response to $x[n-1]$ when $M = 3$



$x[3n]$ delayed by 1 sample

```
clear all
close all

k = 0:10;
a = (-1).^k
aa = upsample(a,2)
x = [0 aa 0] %input
n = 0:length(x)-1;
stem(n,x)
xlabel('n, sample index')
ylabel('x[n]')
title('input')
x_1 = circshift(x,[0,1])  % input delayed by one sample
figure
n = 0:length(x_1)-1;
stem(n,x_1)
xlabel('n, sample index')
ylabel('x[n-1]')
title('input delayed by one sample')
y = downsample(x,2)
figure
n = 0:length(y)-1;
stem(n,y)
xlabel('n, sample index')
ylabel('y[n]')
title('response to x[n]')
yy = downsample(x_1,2)
figure
n = 0:length(yy)-1;
stem(n,yy)
xlabel('n, sample index')
ylabel('z[n]\neq y[n-1]')
title('response to x[n-1]')
```

Show that it is linear! (exercise)

Ex: Expander (upsampler!)

$$y[n] = \begin{cases} x\left[\frac{n}{L}\right] & n = kL \\ 0 & n \neq kL \end{cases} \qquad L \in Z, \qquad L > 1$$

$$x[n] \longrightarrow \boxed{\uparrow L} \longrightarrow y[n]$$

How does it work?

Let $L = 2$

$$\begin{aligned}
&\vdots \\
y[-3] &= 0 \\
y[-2] &= x[-1] \\
y[-1] &= 0 \\
y[0] &= x[0] \\
y[1] &= 0 \\
y[2] &= x[1] \\
y[3] &= 0 \\
&\vdots
\end{aligned}$$

Let $L = 3$

$$\begin{aligned}
&\vdots \\
y[-6] &= x[-2] \\
y[-5] &= 0 \\
y[-4] &= 0 \\
y[-3] &= x[-1] \\
y[-2] &= 0 \\
y[-1] &= 0 \\
y[0] &= x[0] \\
y[1] &= 0 \\
y[2] &= 0 \\
y[3] &= x[1] \\
y[4] &= 0 \\
y[5] &= 0 \\
y[6] &= x[2] \\
&\vdots
\end{aligned}$$

Is it time-invariant?

$$y_1[n] = \begin{cases} x\left[\frac{n}{L}\right] & n = kL \\ 0 & n \neq kL \end{cases}$$

$$y_2[n] = \begin{cases} x\left[\frac{n}{L} - N_0\right] & n = kL \\ 0 & n \neq kL \end{cases}$$

Since

$$y_2[n] \neq y_1[n - N_0] = \begin{cases} x\left[\frac{n - N_0}{L}\right] & n - N_0 = kL \\ 0 & n - N_0 \neq kL \end{cases}$$

expander is time-varying.

Show that it is linear! (exercise)

CAUSALITY

A system is said to be causal if the two output signals $y_1[n]$ and $y_2[n]$ (due to two input signals $x_1[n]$ and $x_2[n]$) satisfy

$$y_1[n] = y_2[n] \qquad\qquad n \leq n_0$$

whenever

$$x_1[n] = x_2[n] \qquad\qquad n \leq n_0$$

Ex:

$y[n] = x[n+1] - x[n]$ is noncausal

$y[n] = x[n] + x[n-1]$ is causal

$y[n] = x[n] + 5$ is causal

Proof: Exercise

STABILITY (BIBO)

A system is said to be BIBO stable if “bounded inputs yield bounded outputs.”, i.e.,

$$|x[n]| \leq b_x < \infty \qquad \Rightarrow \qquad |y[n]| \leq b_y < \infty$$

for arbitrary finite b_x and b_y .

Ex:

$$y[n] = \sum_{k=-\infty}^{n} x[k]$$

$$= y[n-1] + x[n]$$

UNSTABLE

For example, for $x[n] = u[n]$ the output is $y[n] = \begin{cases} n+1 & n \geq 0 \\ 0 & n < 0 \end{cases}$

Therefore bounded input does not yield bounded output.

Computation of

$$y[n] = ay[n-1] + x[n]$$

in MATLAB.

(There are other possibilities!)

```
clear all
close all

N = 1000;   % signal length
a = 0.97;     % system parameter
b = 1;      % system parameter
k = 1:N;  % discrete-time index vector
w = pi/2;   % frequency of input sinuoid

% x = 2*(rand(1,N)-0.5); % random input

x = sin(w*k);            % sinusoidal input

x = [x zeros(1,N)];      % zero padding (why?)

y = zeros(1,2*N);        % output signal

for n = 2:2*N
    y(n) = a*y(n-1)+b*x(n);
end

plot(x)
hold on
plot(y,'r')
```