BLM3620 Digital Signal Processing*

Erkan Uslu

euslu@yildiz.edu.tr

Yıldız Technical University – Computer Engineering

*Based on lecture notes from Ali Can Karaca & Ahmet Elbir

Lecture #1 – Introduction to DSP

- General Information about the Course
- Recommended Documents
- Introduction
- DSP Applications
- Basic Signal Operations

Course Materials

Important Materials:

- James H. McClellan, R. W. Schafer, M. A. Yoder, *DSP First Second Edition*, Pearson, 2015.

- Lizhe Tan, Jean Jiang, *Digital Signal Processing: Fundamentals and Applications*, Third Edition, Academic Press, 2019.

Auxilary Materials:

- Prof. Sarp Ertürk, *Sayısal İşaret İşleme*, Birsen Yayınevi.

- Prof. Nizamettin Aydin, DSP Lecture Notes.

- J. G. Proakis, D. K. Manolakis, *Digital Signal Processing Fourth Edition*, Peason, 2014.

- J. K. Perin, *Digital Signal Processing*, *Lecture Notes,* Standford University, 2018.

General Information

Ders Adı	Kodu	Yerel Kredi	AKTS	Ders (saat/hafta)	Uygulama (saat/hafta)	Laboratuar (saat/hafta)
Sayısal İşaret İşleme	BLM3620	3	5	3	0	0

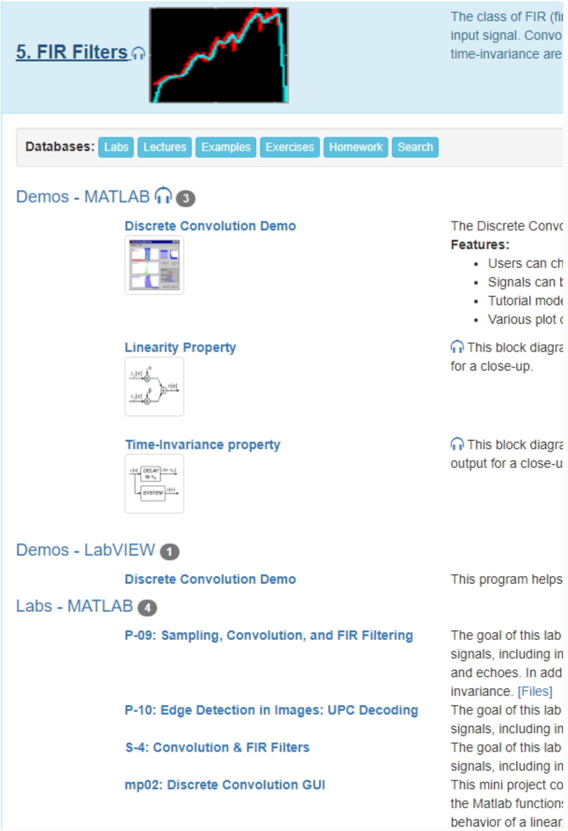
Değerlendirme Sistemi		
Etkinlikler	**Sayı**	**Katkı Payı**
Devam/Katılım		
Laboratuar		
Uygulama		
Arazi Çalışması		
Derse Özgü Staj		
Küçük Sınavlar/Stüdyo Kritiği		
Ödev	4	30
Sunum/Jüri		
Projeler		
Seminer/Workshop		
Ara Sınavlar	1	30
Final	1	40
Dönem İçi Çalışmaların Başarı Notuna Katkısı		60
Final Sınavının Başarı Notuna Katkısı		40

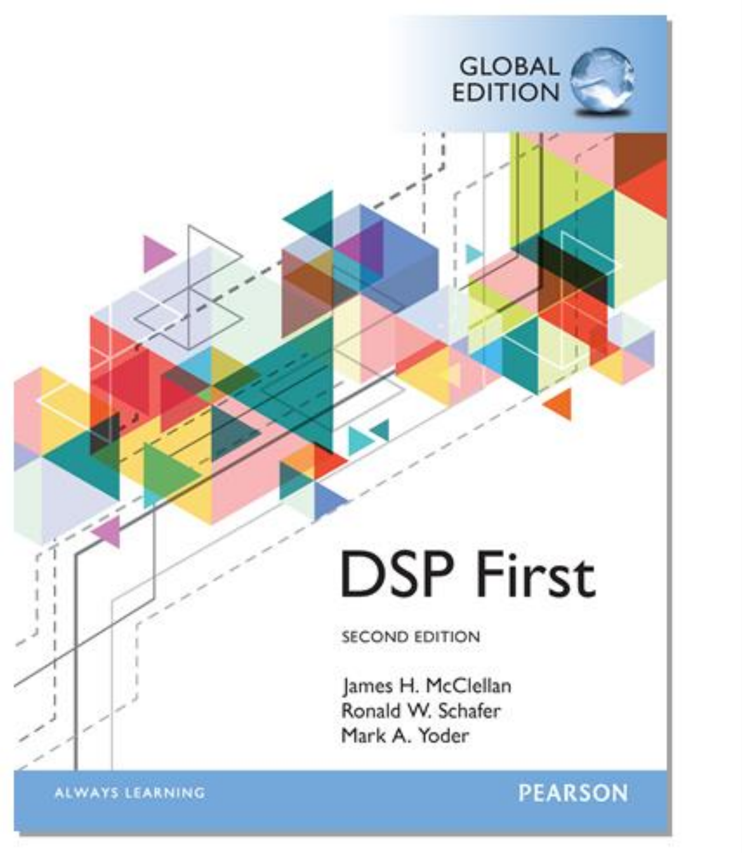
Course Materials

James H. McClellan, R. W. Schafer, M. A. Yoder, *DSP First Second Edition*, Pearson, 2015.

https://dspfirst.gatech.edu/

- Demos,
- Examples,
- Exercises,
- Lectures,
- Labs…





Syllabus

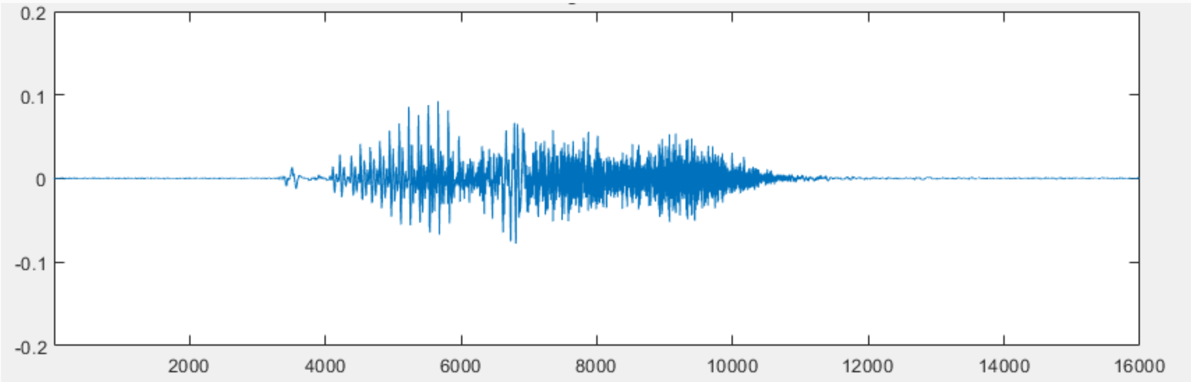
W.	Date	Topics
1	23. Feb. 2024	Introduction to DSP and MATLAB
2	1. Mar. 2024	Sinuzoids and Complex Exponentials
3	8. Mar. 2024	Spectrum Representation
4	15. Mar. 2024	Sampling and Aliasing
5	22. Mar. 2024	Discrete Time Signal Properties and Convolution
6	29. Mar. 2024	Convolution and FIR Filters
7	5. Apr. 2024	Frequency Response of FIR Filters
8	12. Apr. 2024	Ramadan Feast
9	19. Apr. 2024	Midterm
10	26. Apr. 2024	Discrete Time / Discrete Fourier Transform and Properties
11	3. May. 2024	Fast Fourier Transform and Windowing
12	10. May. 2024	z- Transforms
13	17. May. 2024	FIR Filter Design and Applications
14	24. May. 2024	IIR Filter Design and Applications

For more details -> Bologna page: http://www.bologna.yildiz.edu.tr/index.php?r=course/view&id=5730&aid=3

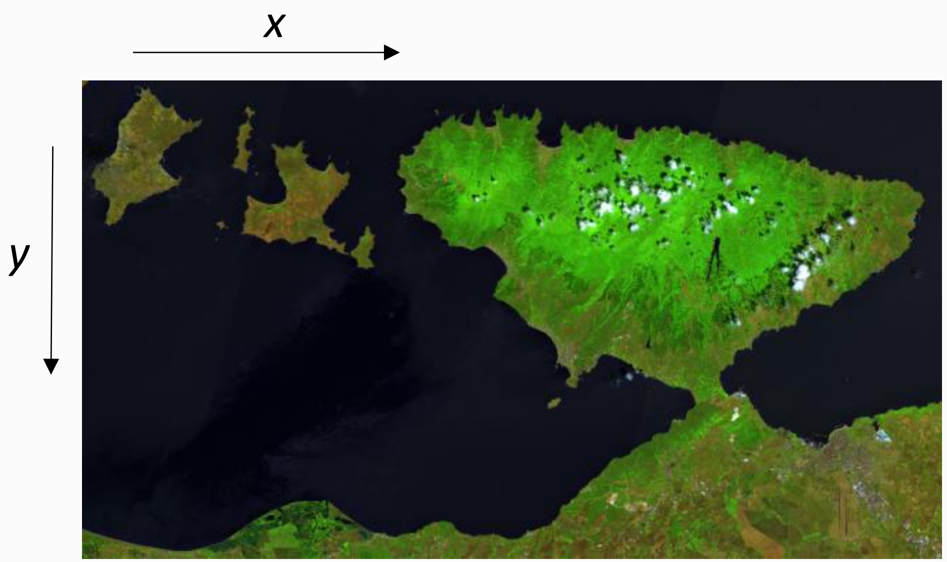
COURSE OBJECTIVE

- Students will be able to:
- Understand **mathematical** descriptions of signal processing **algorithms** and express those algorithms as computer **implementations** (MATLAB)
- What are your objectives?

«YES» Word Sound Signal



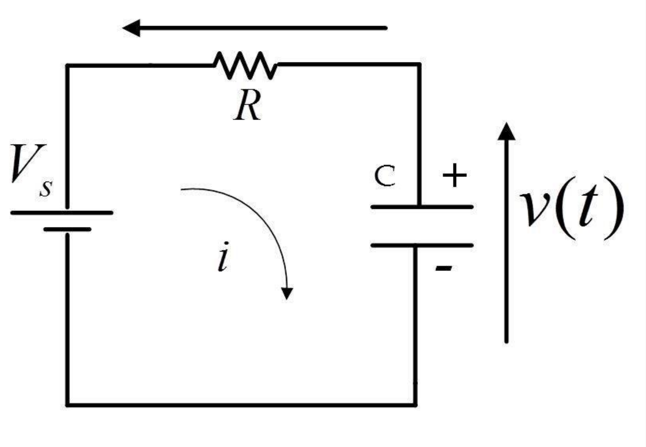
Time



Mathematical Function: I(x, y)

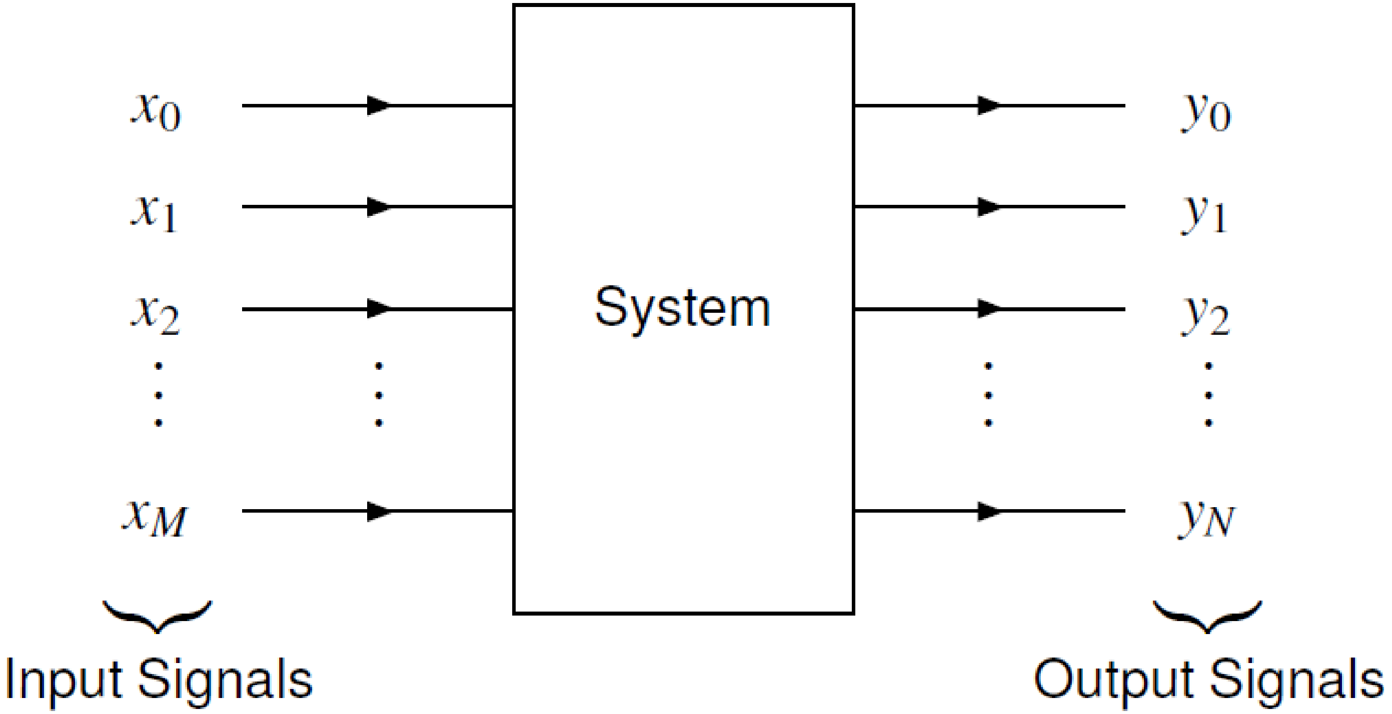
What are Signals and Systems?

- Signal:
  - any physical quantity that varies with time, space, or any other independent variable or variables
  - Examples: pressure as a function of altitude, sound as a function of time, color as a function of space, ...
  - $x(t) = \cos(2\pi t)$, $x(t) = 4\sqrt{t} + t^3$, $x(m,n) = (m+n)^2$
- System:
  - a physical device that performs an operation on a signal
  - Examples: analog amplifier, noise canceler, communication channel, transistor, ...
  - $y(t) = -4x(t)$, $\frac{dy(t)}{dt} + 3y(t) = -\frac{dx(t)}{dt} + 6x(t)$, $y(n) - \frac{1}{2}y(n-2) = 3x(n) + x(n-2)$



What is a system?

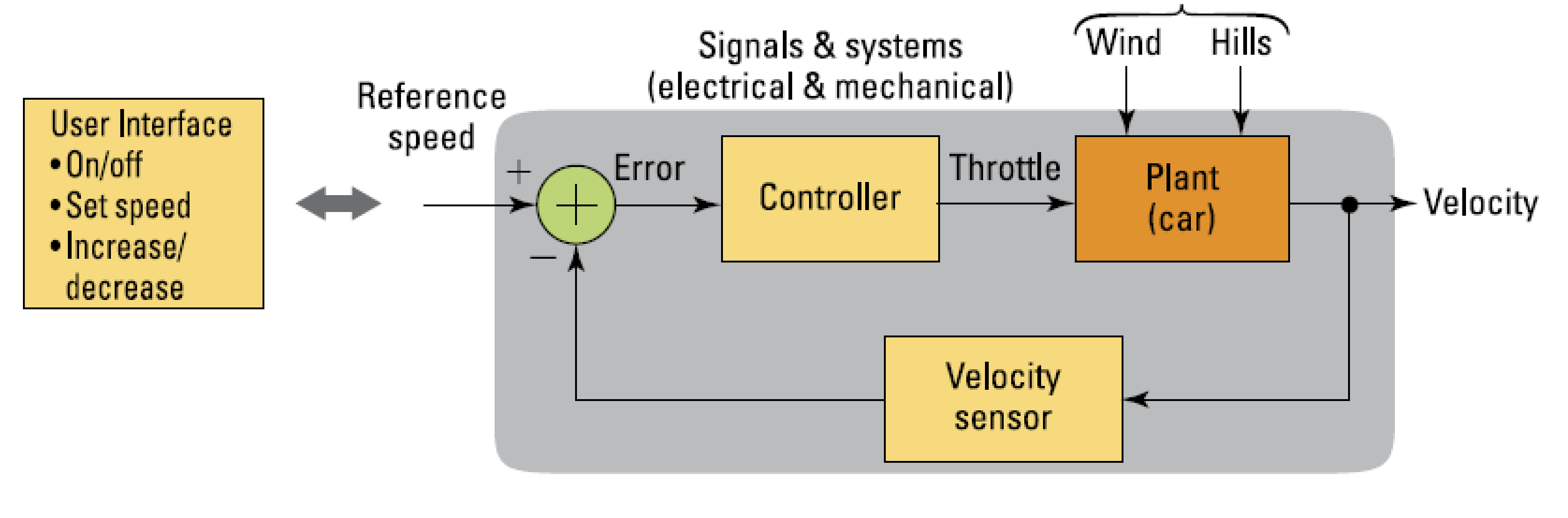
A **system** is an entity that processes one or more input signals in order to produce one or more output signals.



Example: Cruise Control System

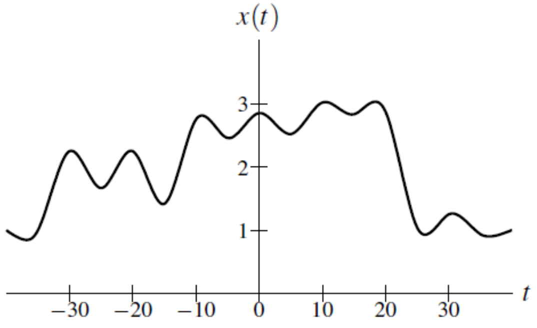
Input Signals: ?

Output Signals: Speed of car

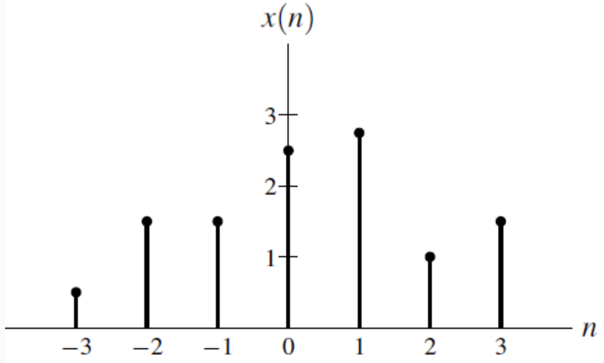


Classification of Signals

- Number of independent variables (i.e., dimensionality):
  - A signal with ***one*** independent variable is said to be **one dimensional** (e.g., audio).
  - A signal with ***more than one*** independent variable is said to be **multi-dimensional** (e.g., image).
- Continuous or discrete independent variables:
  - A signal with ***continuous*** independent variables is said to be **continuous time (CT)** (e.g., voltage waveform).
  - A signal with ***discrete*** independent variables is said to be **discrete time (DT)** (e.g., stock market index).



Continuous-Time (CT) Signal

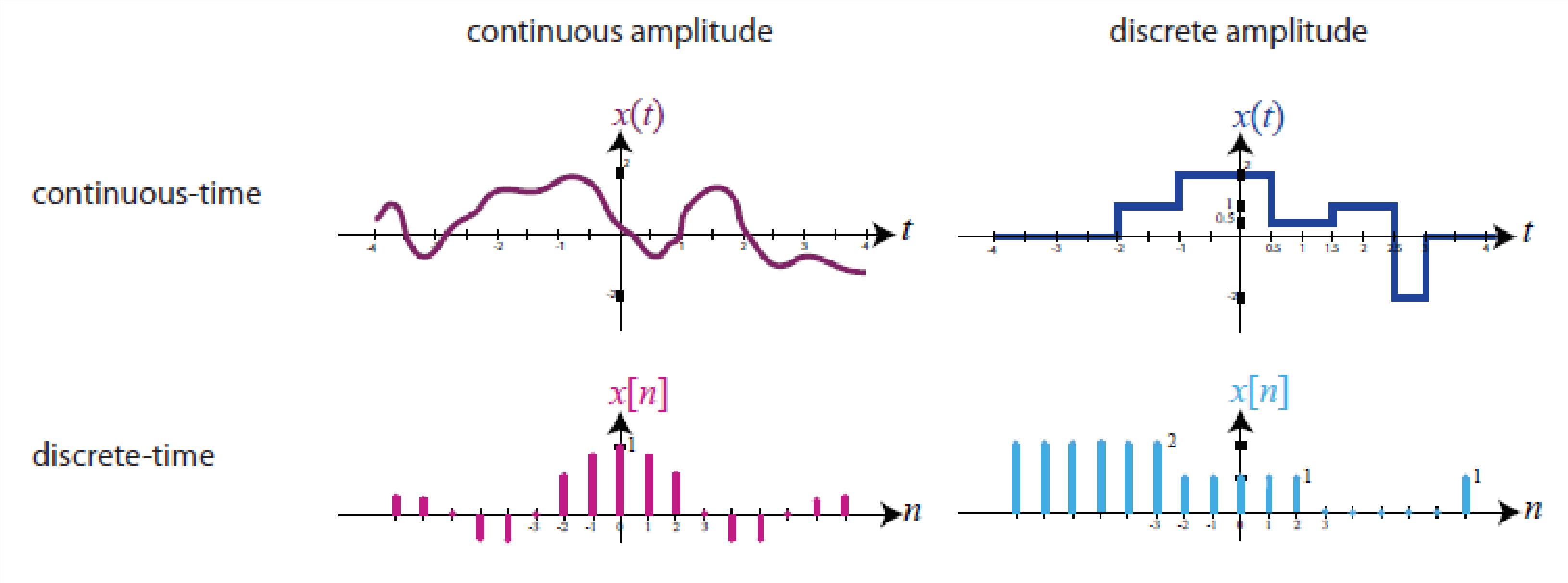


Discrete-Time (DT) Signal

Understanding Analog and Digital Signals

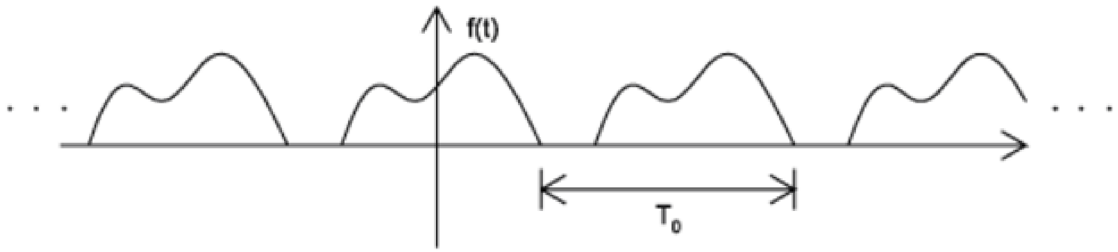
Analog signal-> continuous both in time and amplitude

Digital signal-> discrete both in time and amplitude

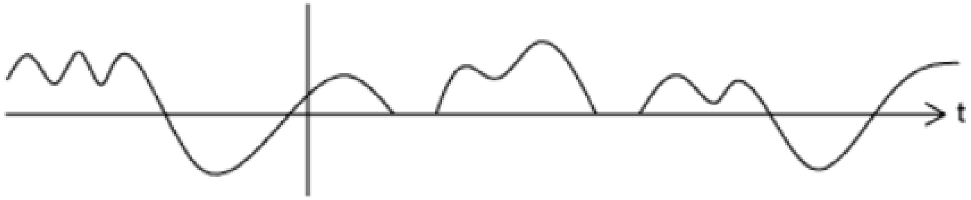


Classifications of Signals

Periodic vs. Aperiodic



3a: A periodic signal with period T_0



3b: An aperiodic signal

Periodic signals repeat with some **period** T, while aperiodic, or nonperiodic, signals do not. We can define a periodic function through the following mathematical expression, where t can be any number and T is a positive constant

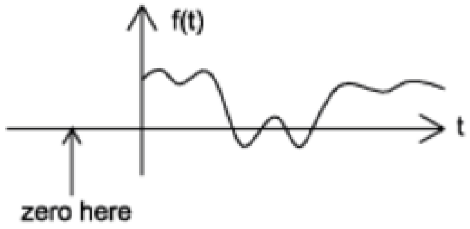
$$f(t) = f(t + T)$$

fundamental period of our function, f(t), is the smallest value of T that the still allows Equation to be true.

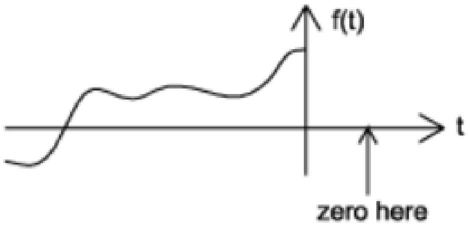
Classifications of Signals

Causal vs. Anticausal vs. Noncausal

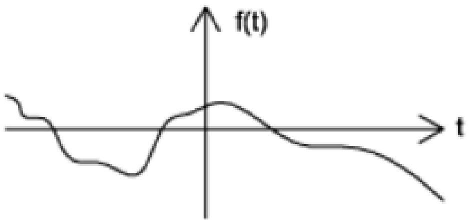
- **Causal** signals are signals that are zero for all negative time, while **anticausal** are signals that are zero for all positive time.
- **Noncausal** signals are signals that have nonzero values in both positive and negative time



4a: A causal signal



4b: An anticausal signal

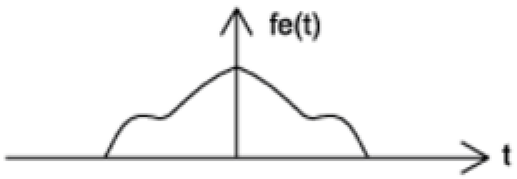


4c: A noncausal signal

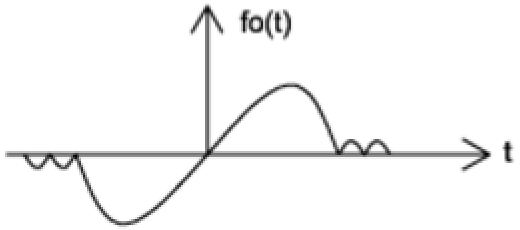
Classifications of Signals

- An **even signal** is any signal f such that f(t)=f(−t).
Even signals can be easily spotted as they are **symmetric** around the vertical axis.
- An **odd signal**, on the other hand, is a signal f such that f(t)=−f(−t)

Even vs. Odd

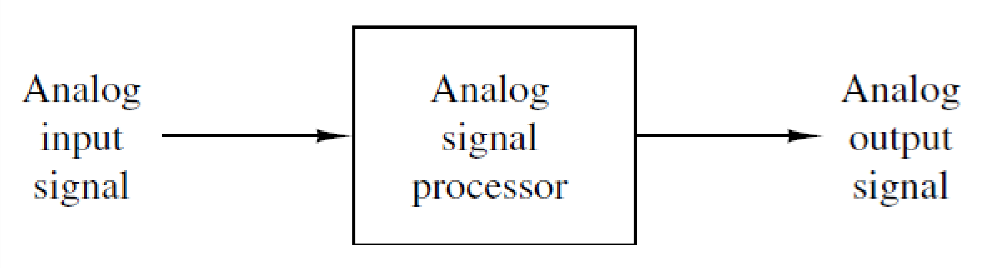


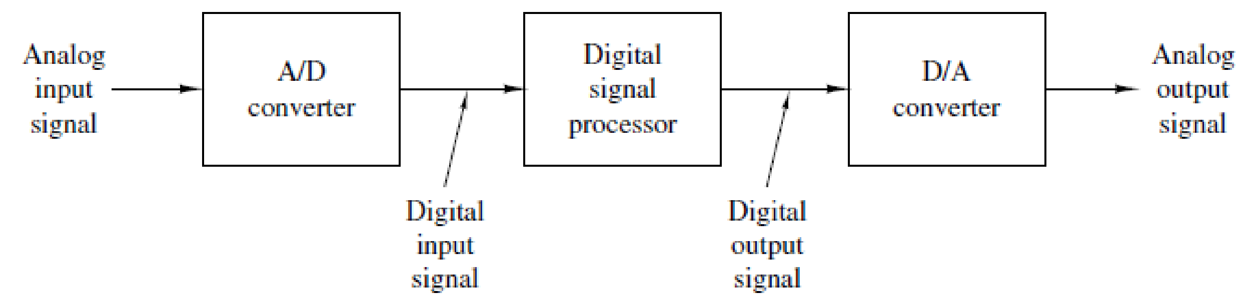
5a: An even signal



5b: An odd signal

Analog Systems vs. Digital Systems





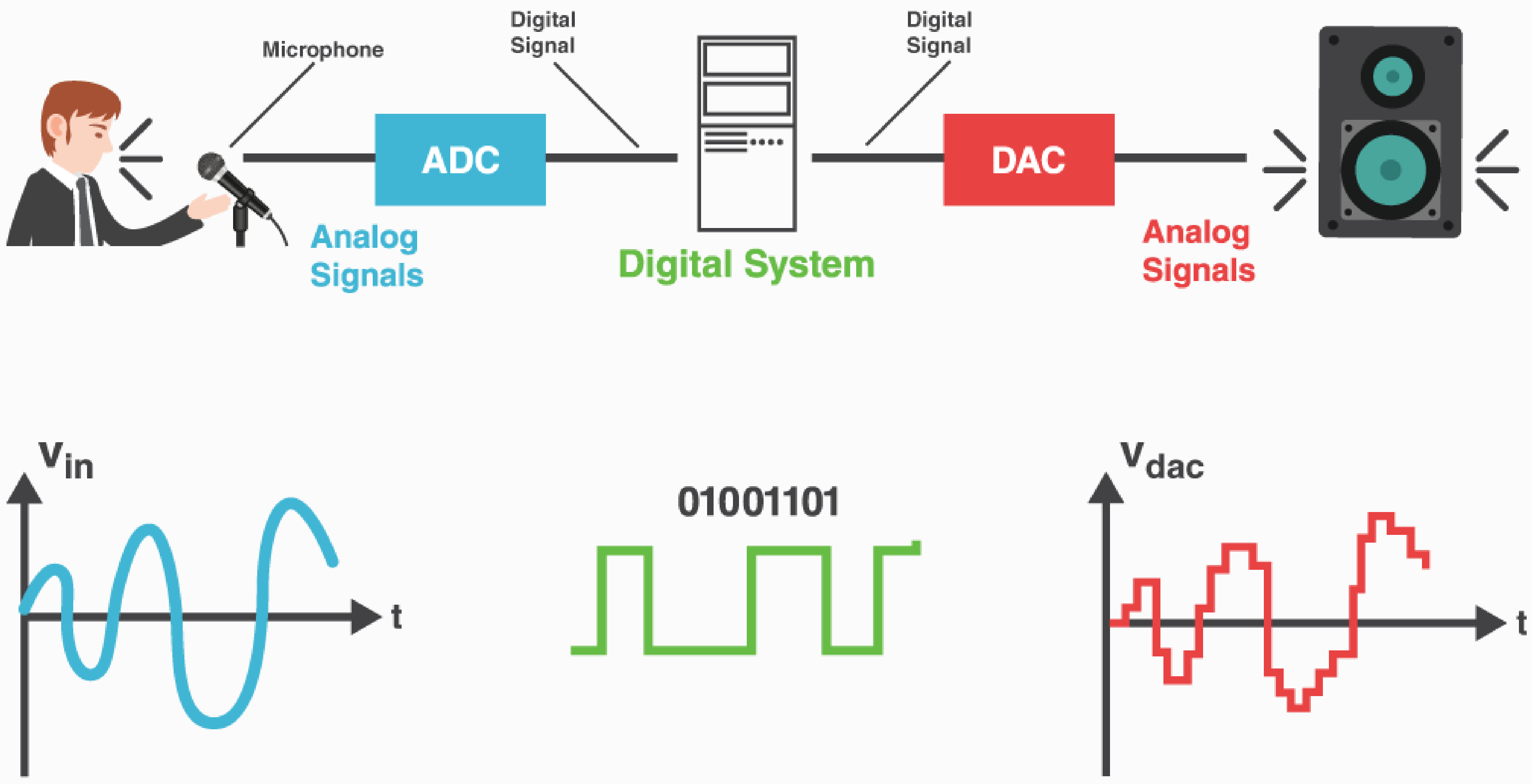
Analog Systems:

- *Directly use real-world signals.*
- *Do not need to an ADC or a DAC.*
- *Give the fastest application results.*

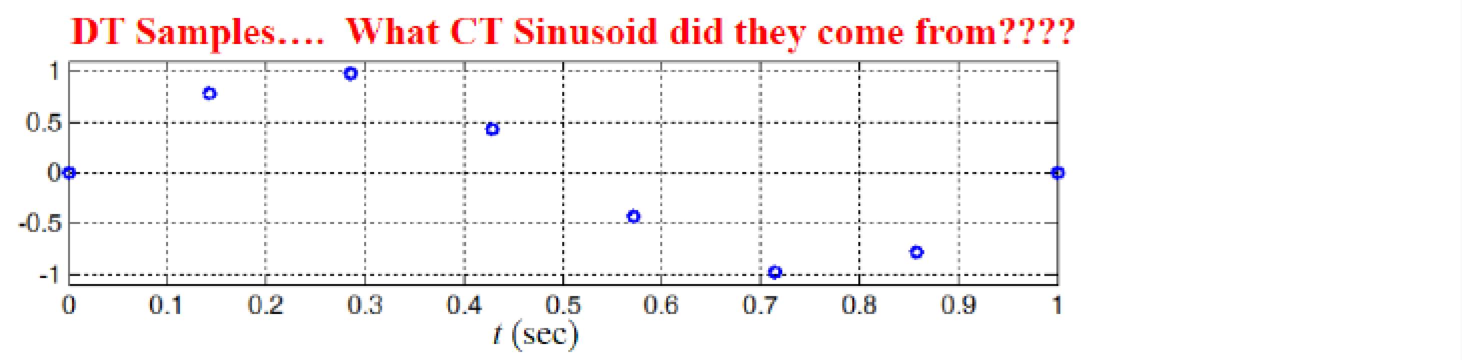
Digital Systems:

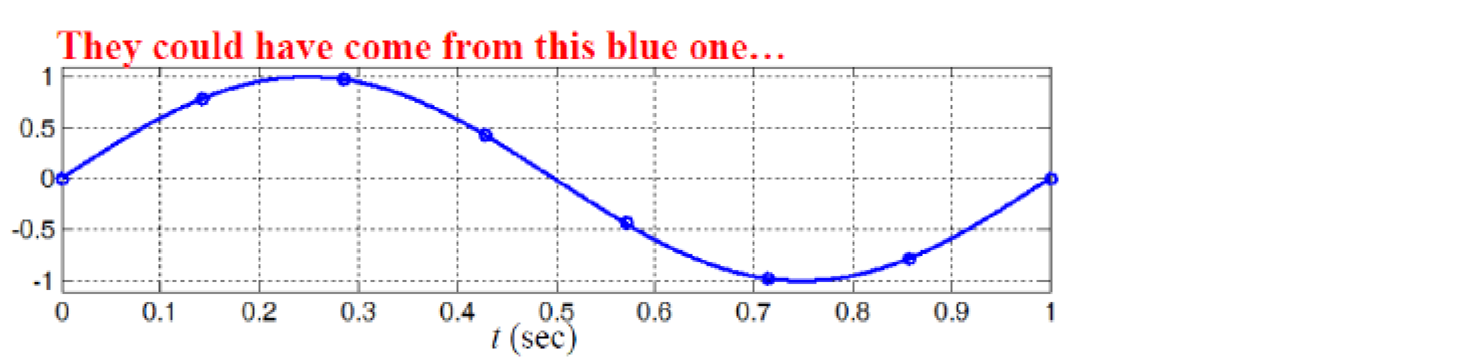
- *have lower distortions thanks to error correction.*
- *Digital signals can easily be compressed and saved.*
- *Do not include any R, L, C elements. (programmed on software)*
- *Are more stable and robust to the environmental conditions.*
- *Can be ported different hardware*

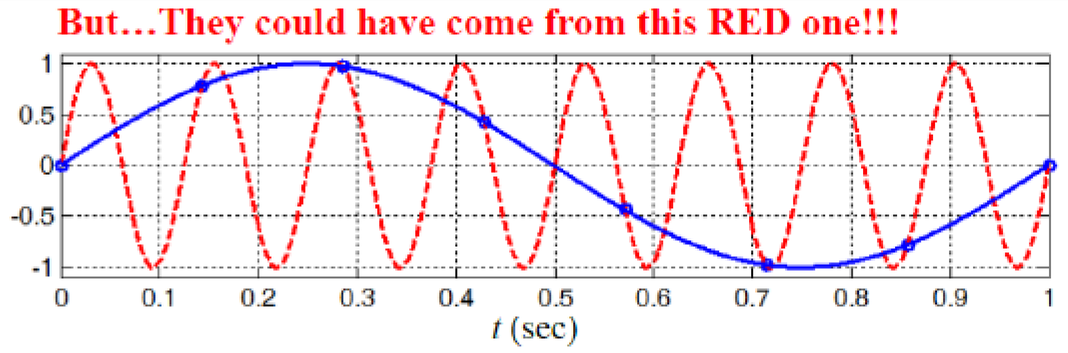
General Concept in DSP Applications



Sampling is very important !



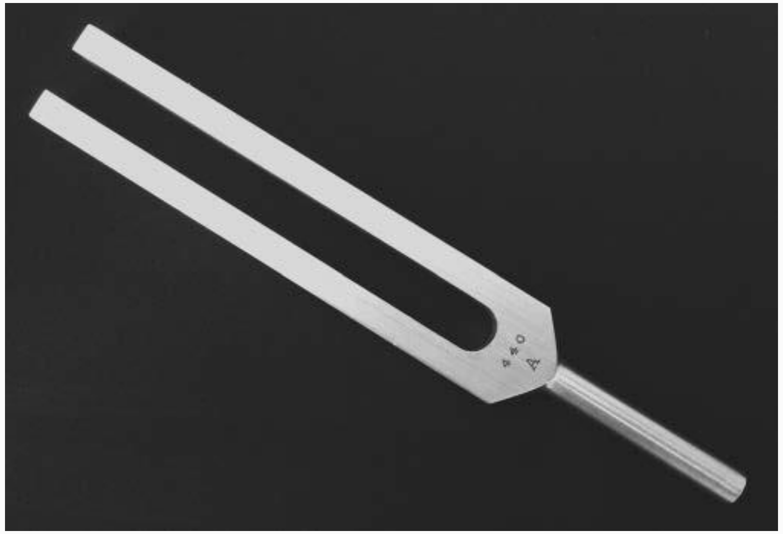




Thus… if we want to be able to tell these two apart we need to sample faster!!

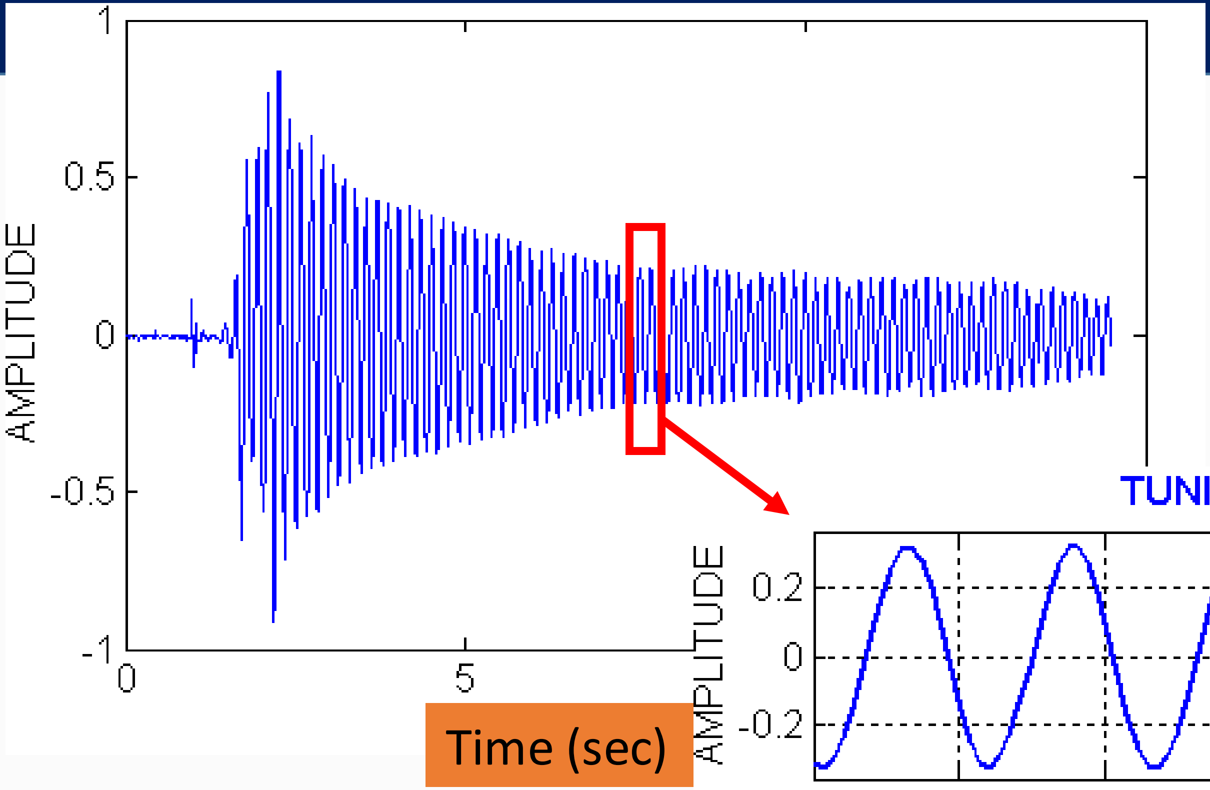
TUNING FORK EXAMPLE

- CD-ROM demo
- "A" is at 440 Hertz (Hz)
- Waveform is a SINUSOIDAL SIGNAL
- Computer plot looks like a sine wave
- This should be the mathematical formula:

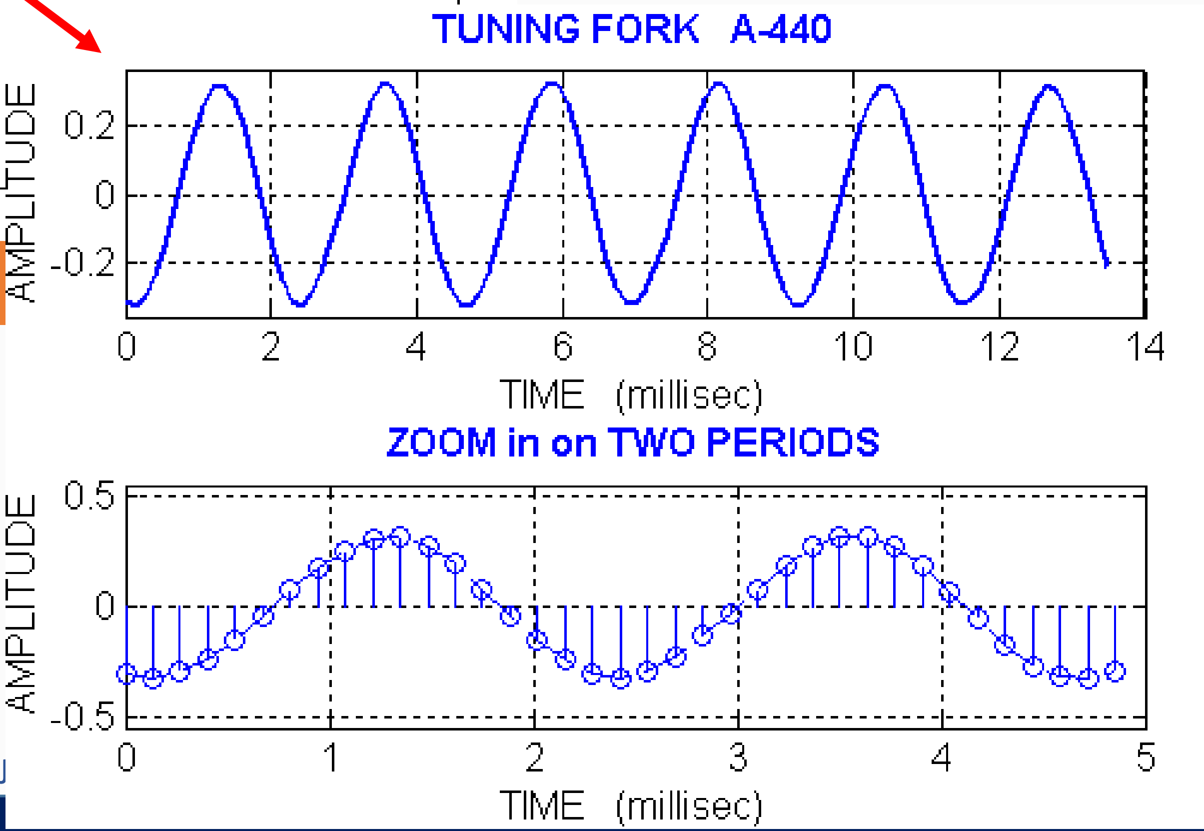


$$A\cos(2\pi(440)t + \varphi)$$

Tunning Fork: A-440 Waveform



$$T \approx 8.15 - 5.85$$
$$= 2.3\,\mathrm{ms}$$



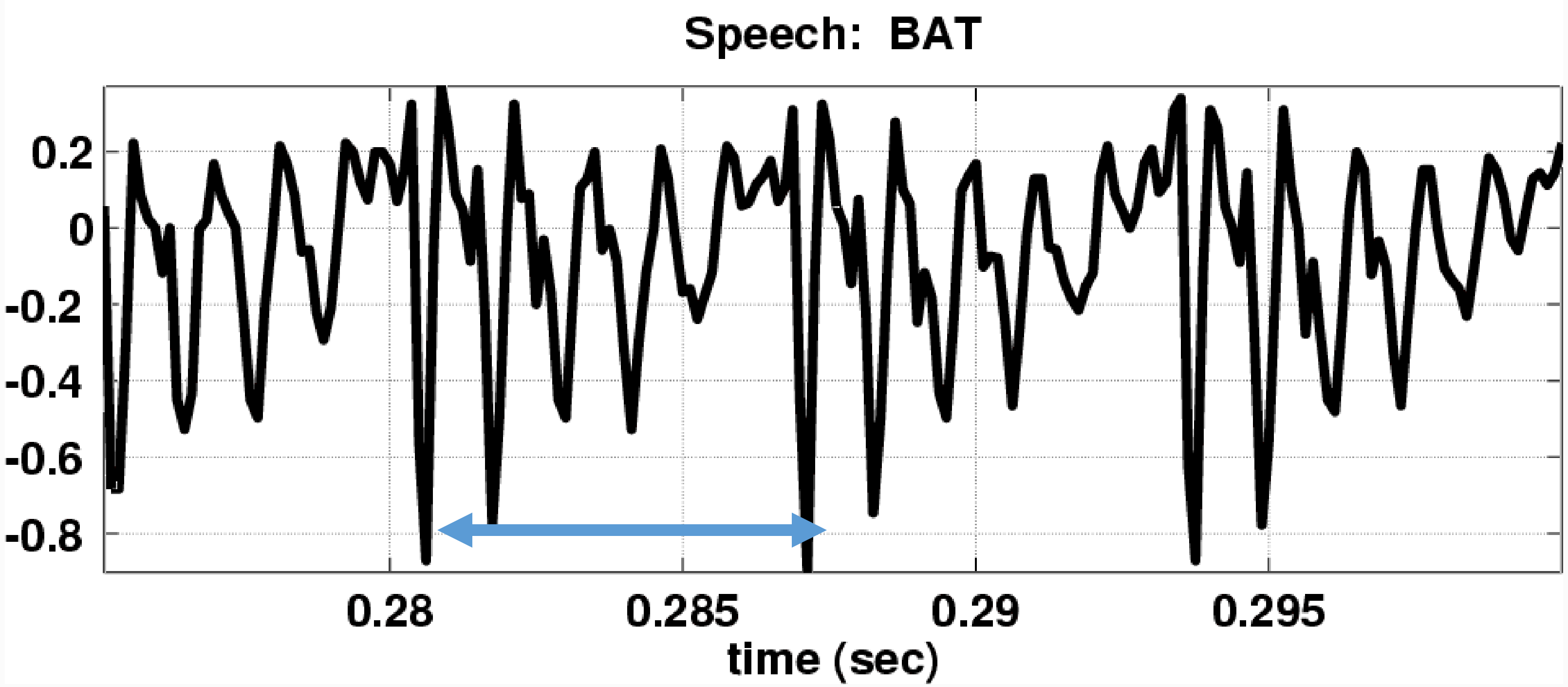
$$f = 1/T$$
$$= 1000/2.3$$
$$\approx 435\,\mathrm{Hz}$$

What about speech?

- More complicated signal (BAT.WAV)
- Waveform ***x*(*t*)** is NOT a Sinusoid
- Theory will tell us
  - ***x*(*t*)** is approximately a sum of sinusoids
  - FOURIER ANALYSIS
    - Break ***x*(*t*)** into its sinusoidal components
  - Called the FREQUENCY SPECTRUM

What about speech?

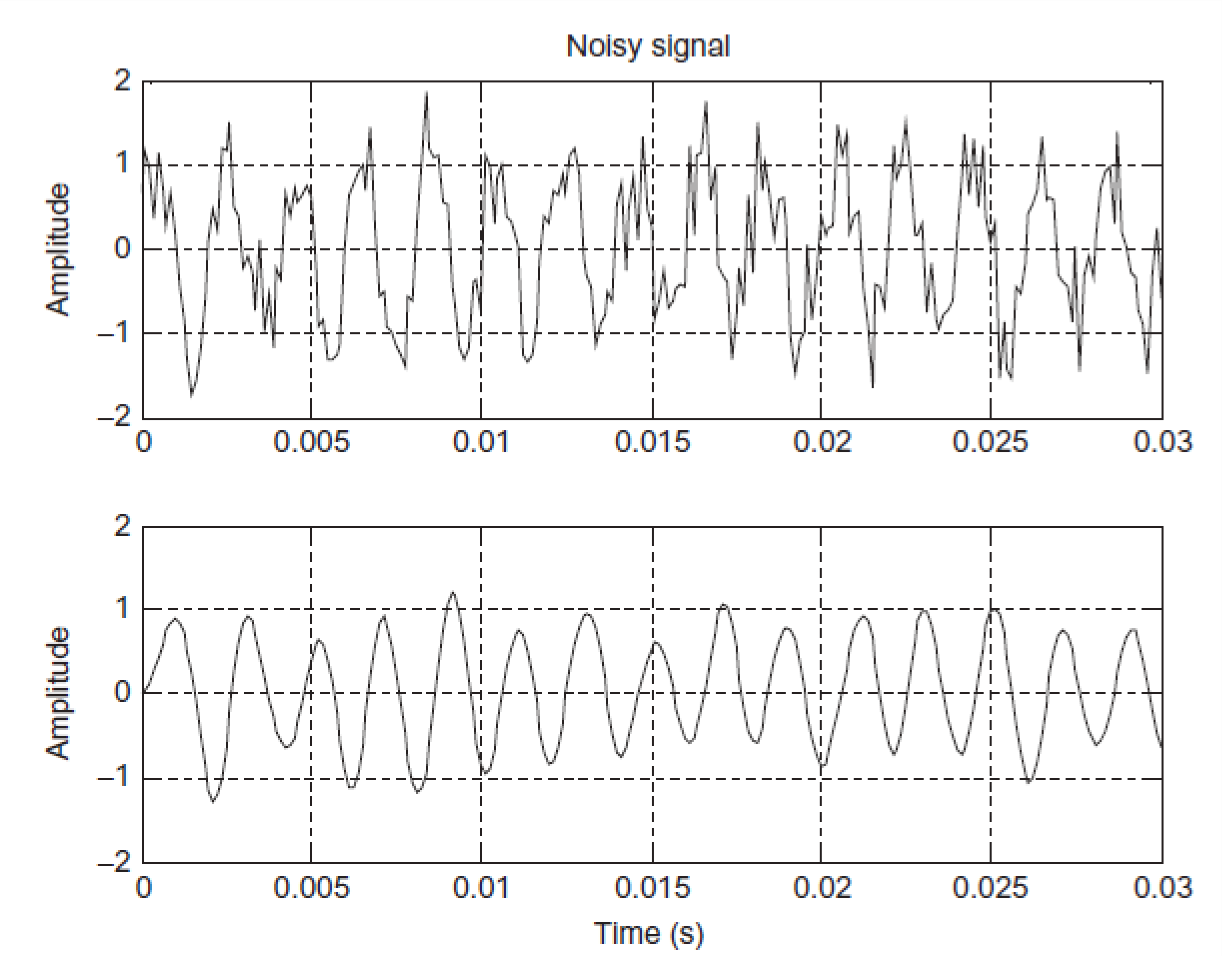
- Nearly **Periodic** in Vowel Region
  - Period is (Approximately) T = 0.0065 sec

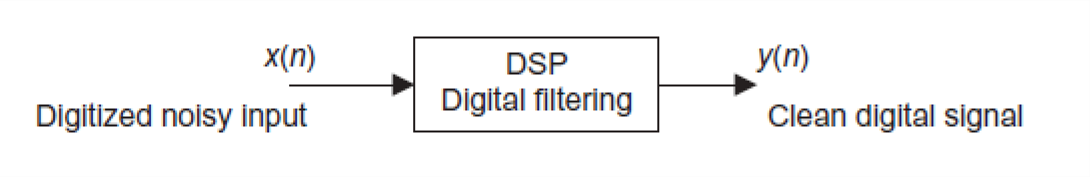


STORING DIGITAL SOUND

- $x[n]$ is a SAMPLED SINUSOID
  - A list of numbers stored in memory
- CD rate is 44,100 samples per second
- 16-bit samples
- Stereo uses 2 channels
- Number of bytes for 1 minute is
  - 2 X (16/8) X 60 X 44100 = 10.584 Mbytes

DSP Applications: Digital Filtering

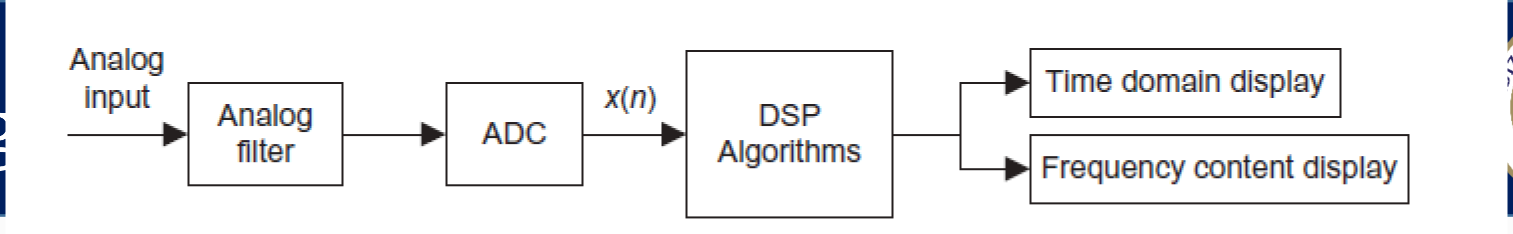




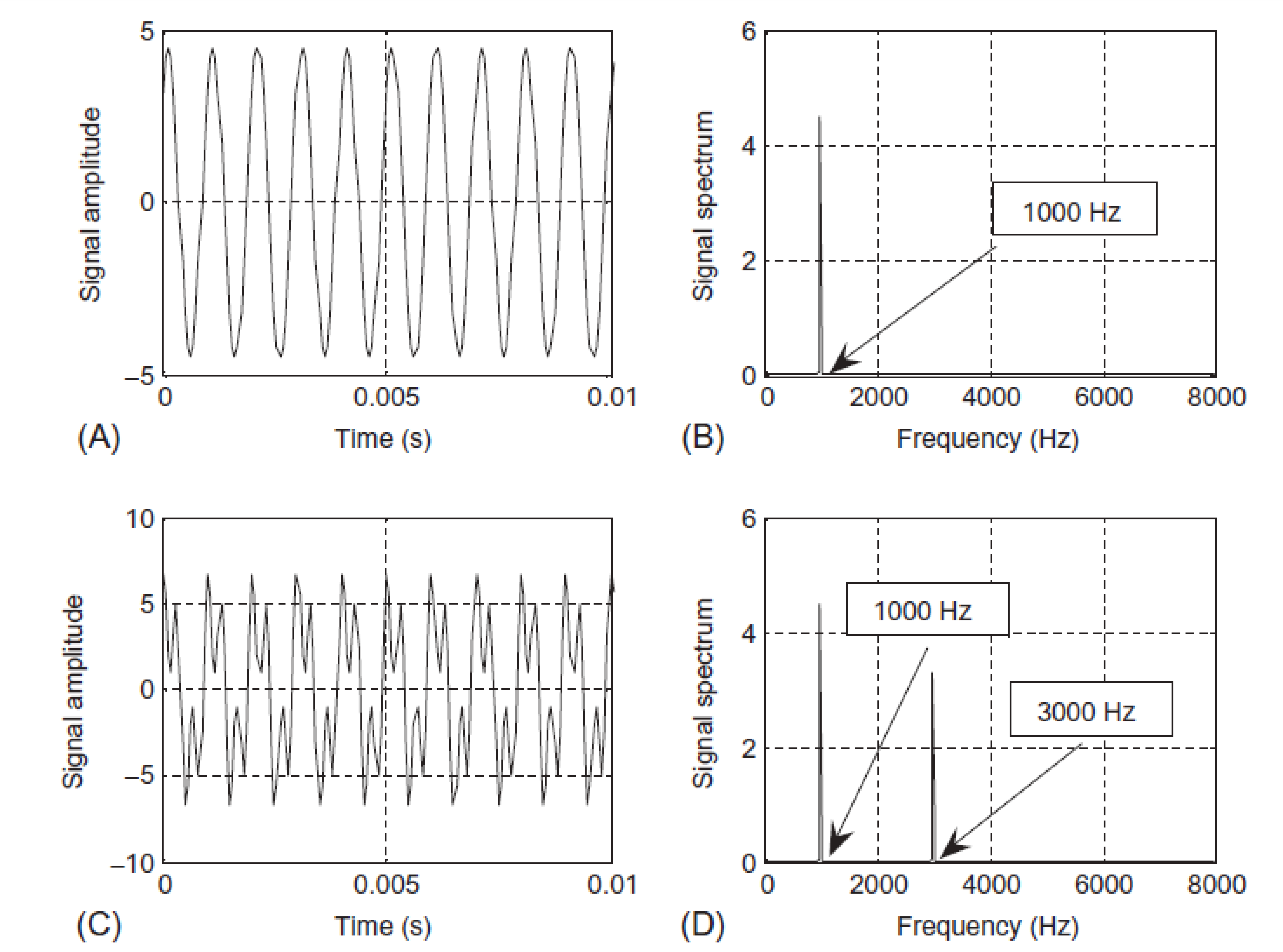
Which filter type should we use?

Ref. J. Jiang and L. Tan

DSP Applications: Dig



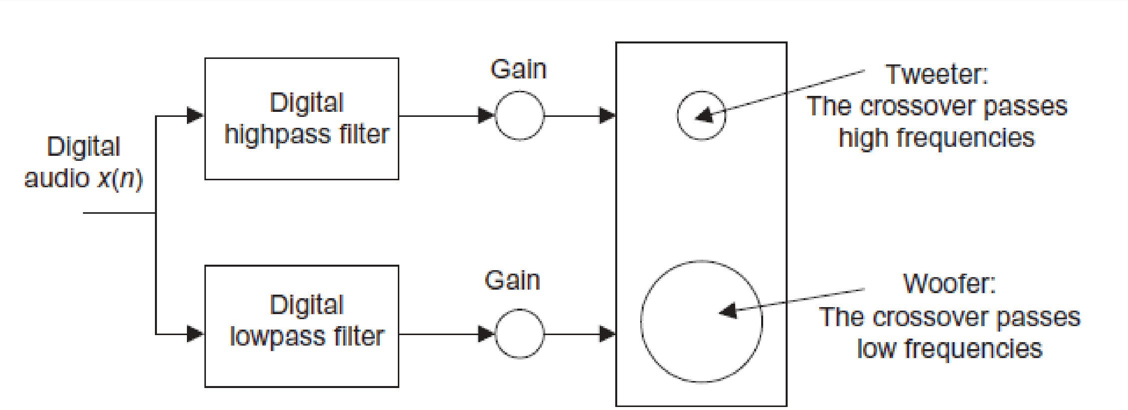
We should go to frequency domain…



Ref. J. Jiang and L. Tan

Some Real-World Apps: Digital Crossover Audio System

The incoming digital audio signal is split into two bands using a digital lowpass filter and a digital highpass filter in parallel. Then the separated audio signals are amplified. Finally, they are sent to their corresponding speaker drivers.

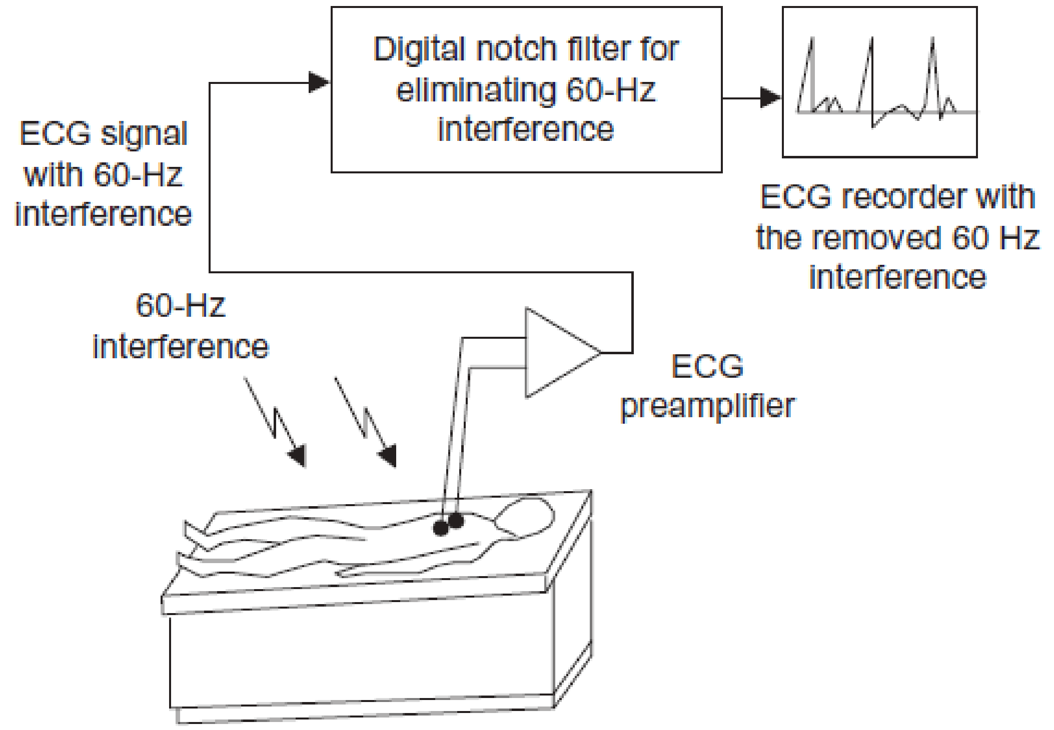


Although the traditional crossover systems are designed using the analog circuits, the digital crossover system offers a cost-effective solution with programmable
ability, flexibility, and high quality.

Ref. J. Jiang and L. Tan, DSP: Fundamentals and Apps

Some Real-World Apps: Filtering ECG Signal

Design a notch filter and apply it to digital signal



Ref. J. Jiang and L. Tan, DSP: Fundamentals and Apps

Some Real-World Apps: Vibration Signature for Defect Det.

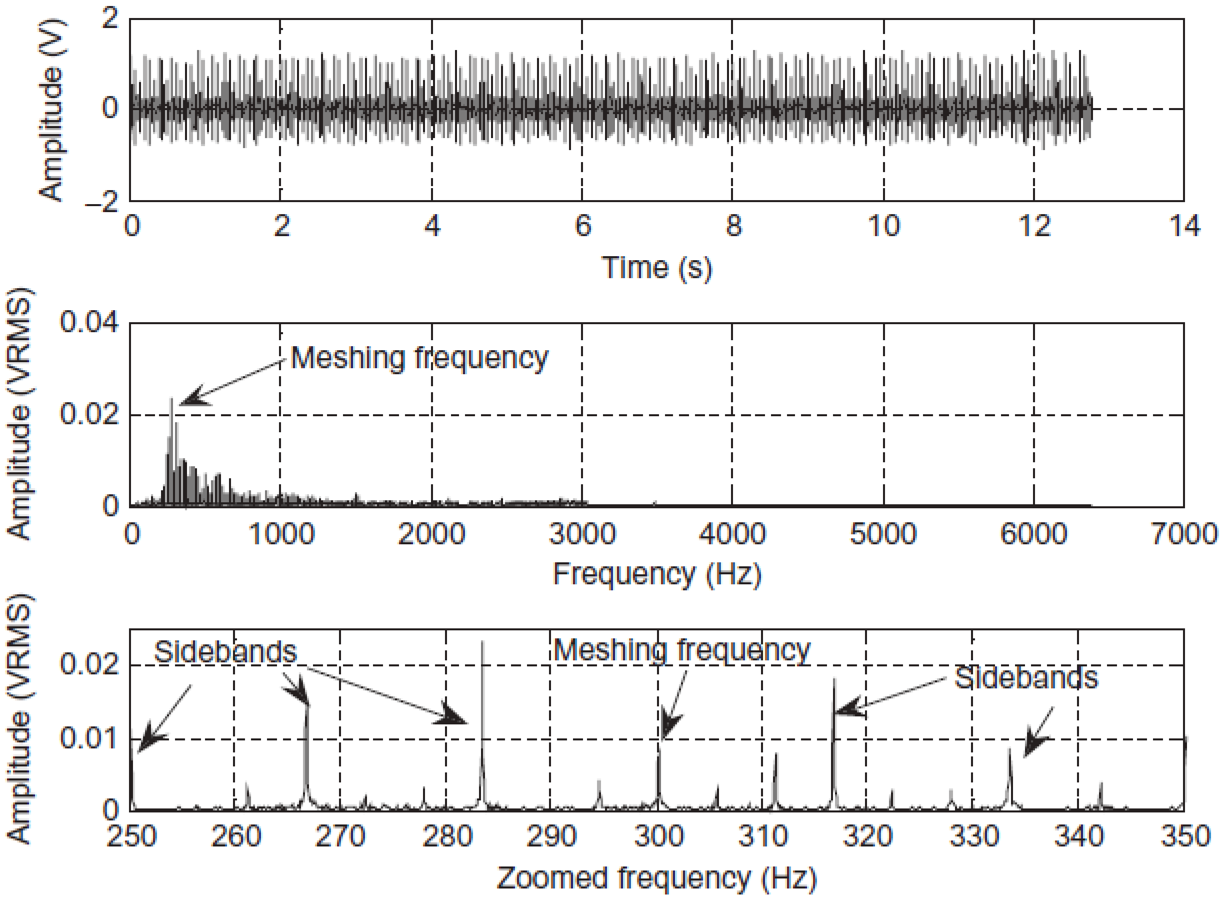


FIG. 1.13

Vibration signal and spectrum from the damaged gearbox.

(D) Damaged pinion

Ref. J. Jiang and L. Tan, DSP: Fundamentals and Apps

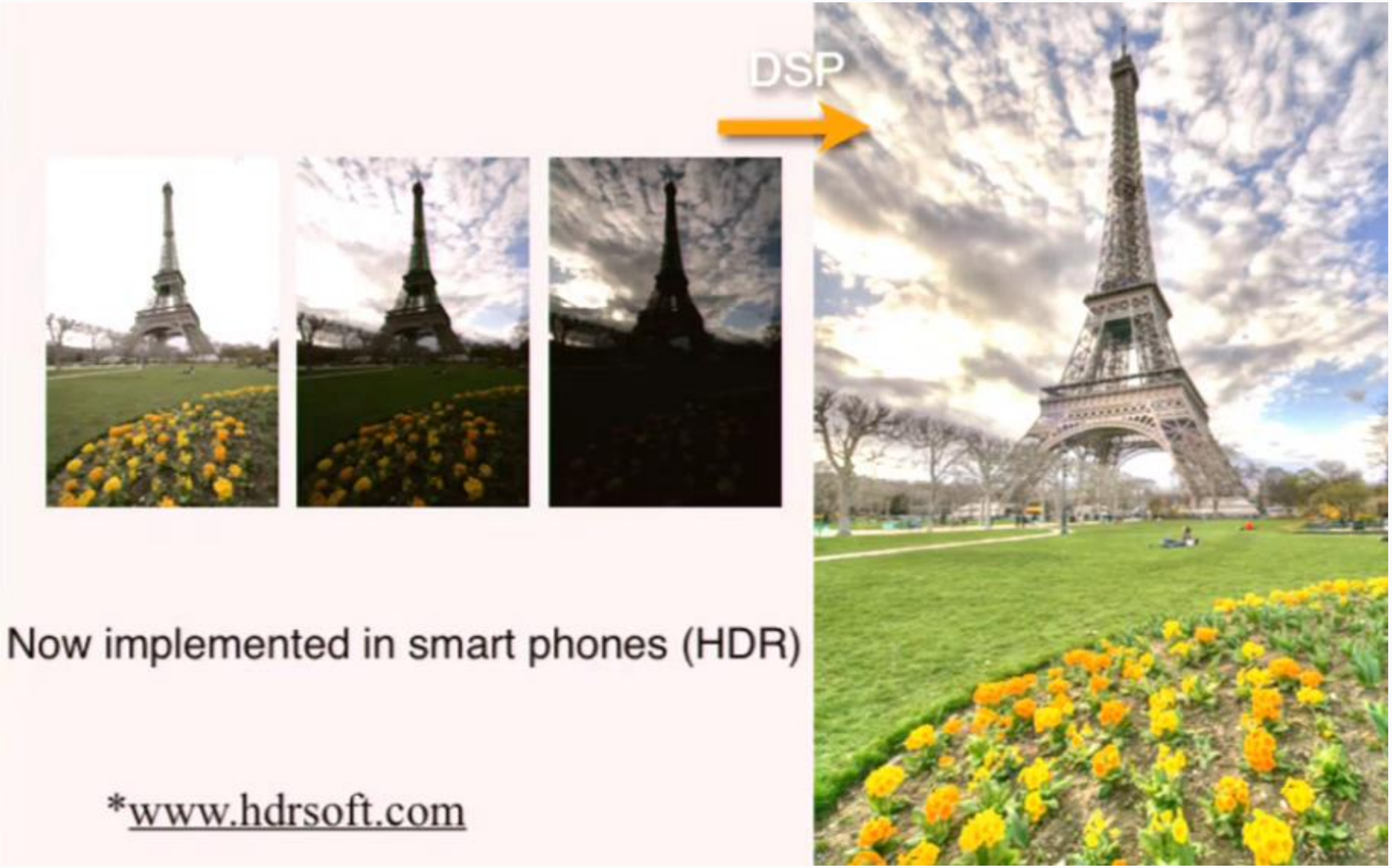
Other Applications

DSP = Swiss-Army-Knife of modern CEng

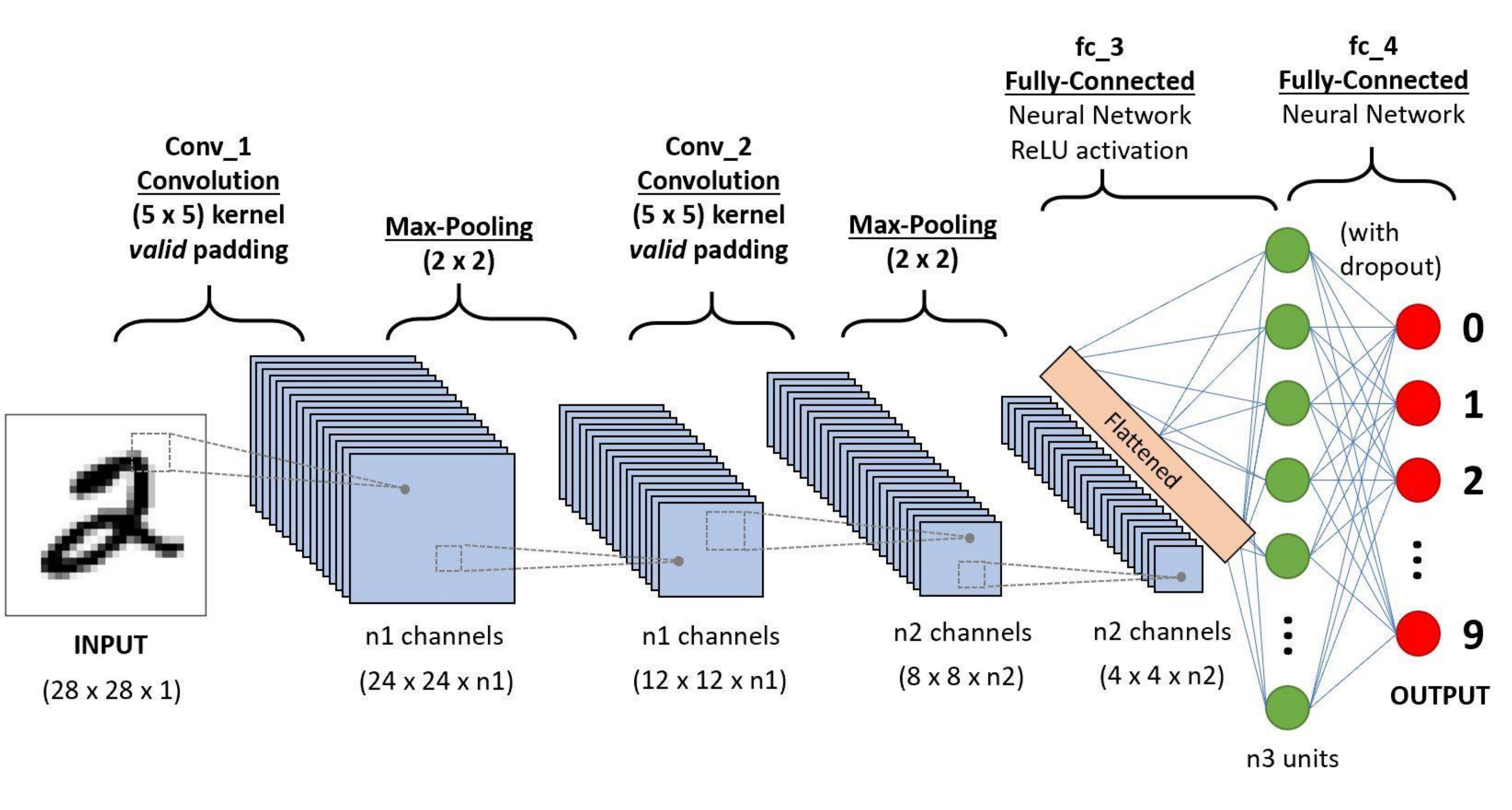
- Communications (wireless, internet, GPS),
- Control and monitoring (cars, machines…),
- Multimedia (videos, cameras, …),
- Healthcare (medical devices),
- More…

Satellite Image Compression

HDR (High Dynamic Range)



Some Real-World Apps: Convolutional NNs



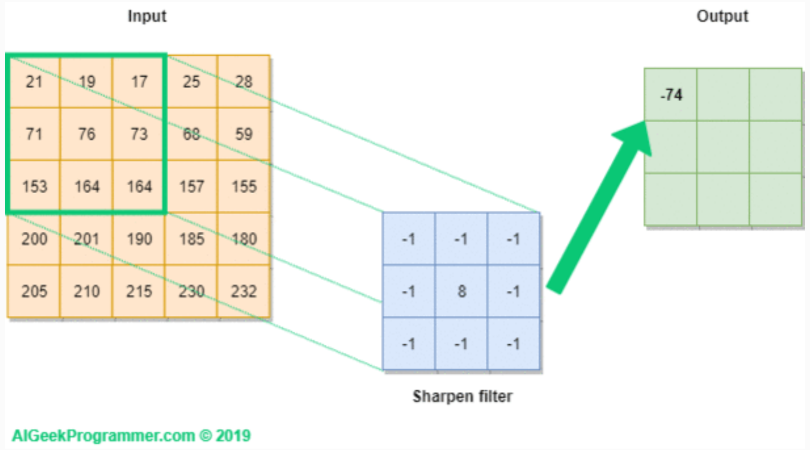
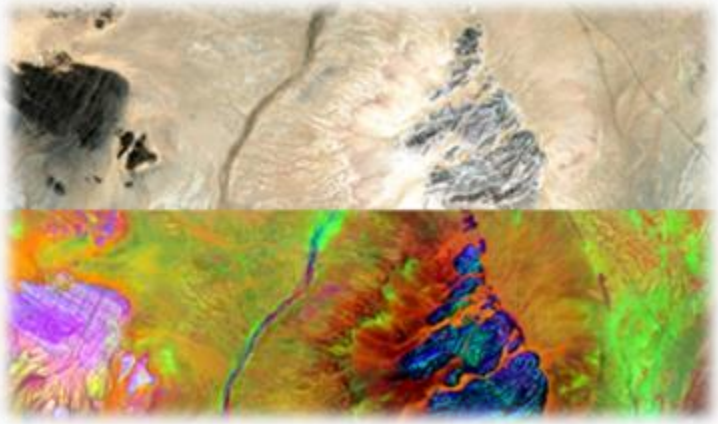


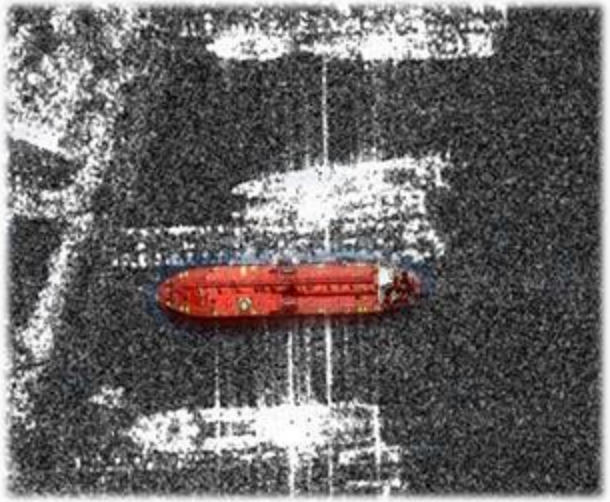
Image Processing – Remote Sensing

3D mapping

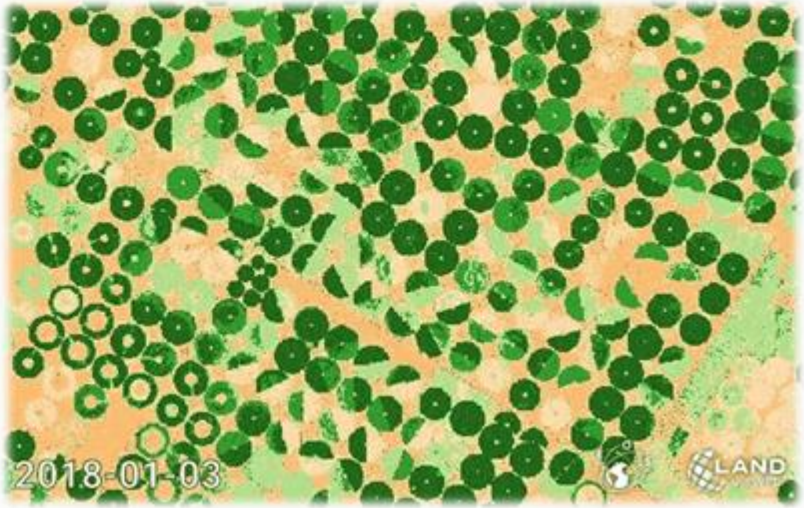


Mineral mapping

Change detection

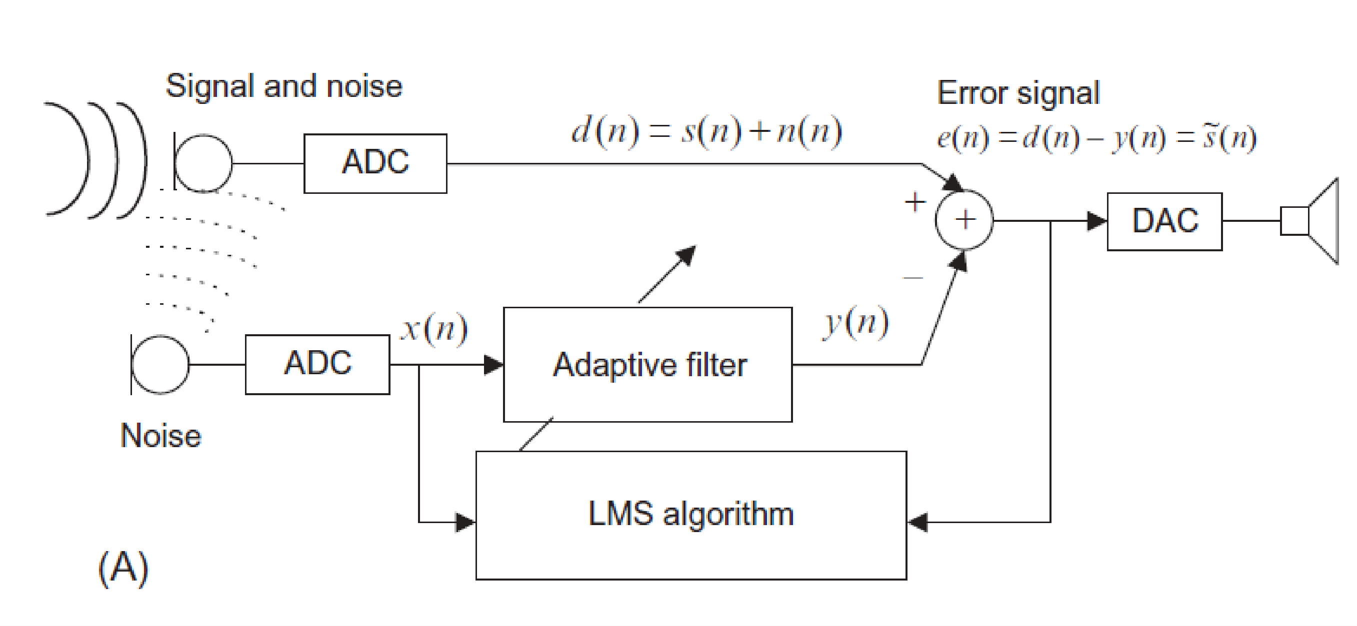


Target detection



agriculture

Some Real-World Apps: Noise Cancellation



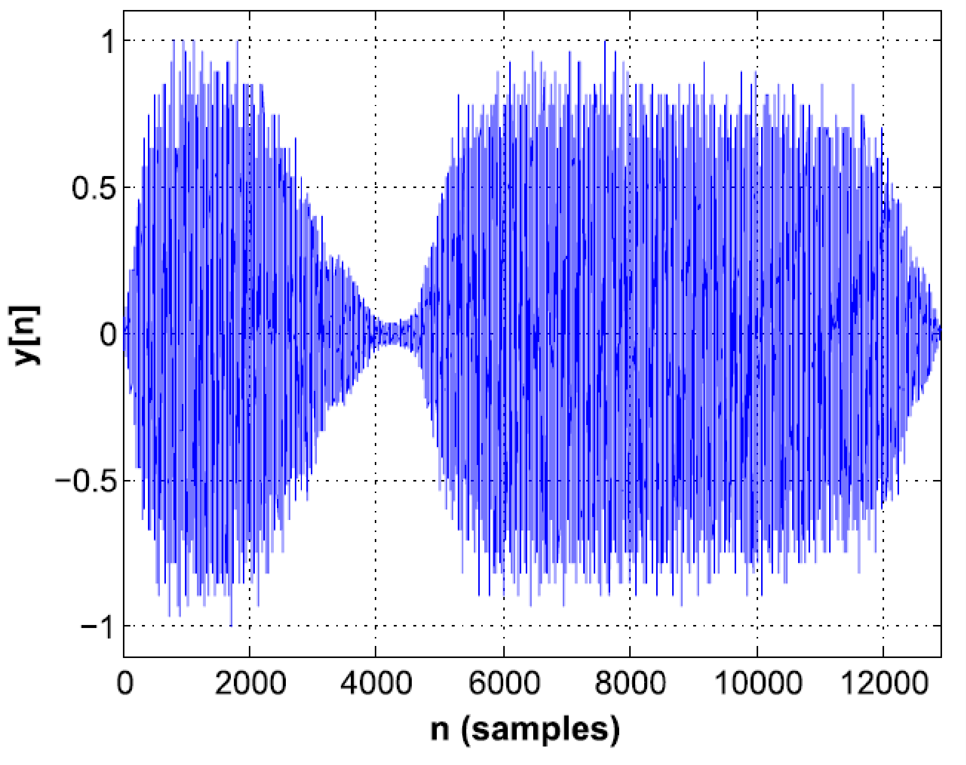
Ref. J. Jiang and L. Tan, DSP: Fundamentals and Apps

Let's take a look at train signal...

```
clear all;
load train;

%% Example---Listening to/plotting train signal
sound(y,Fs)
t=0:1/Fs:(length(y)-1)/Fs;
figure(2); plot(t,y'); grid
ylabel('y[n]'); xlabel('n')

%% Example---Using stem to plot 200 samples of train
figure(3)
n=100:299;
stem(n,y(100:299)); ylabel('y[n]'); xlabel('n')
title('Segment of train signal')
axis([100 299 -0.5 0.5])
```



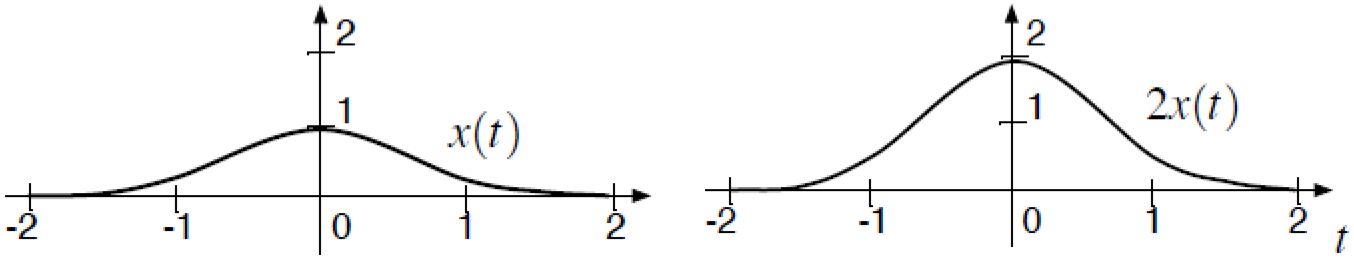
How can we decrease the amplitude of the sound??

What about noisy signals??

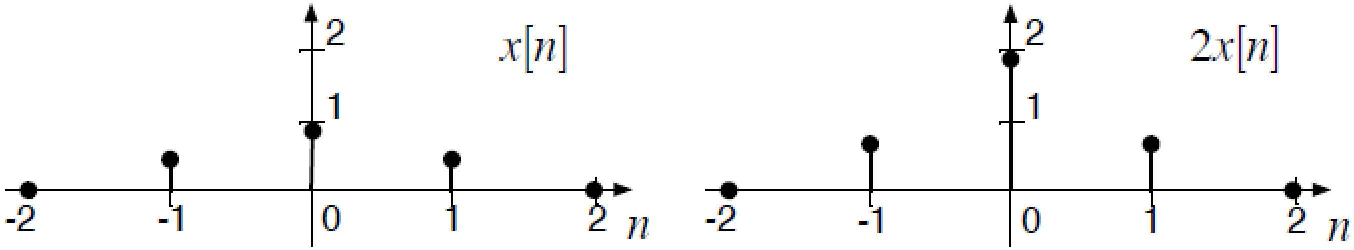
Operations on Signals

Amplitude Scaling

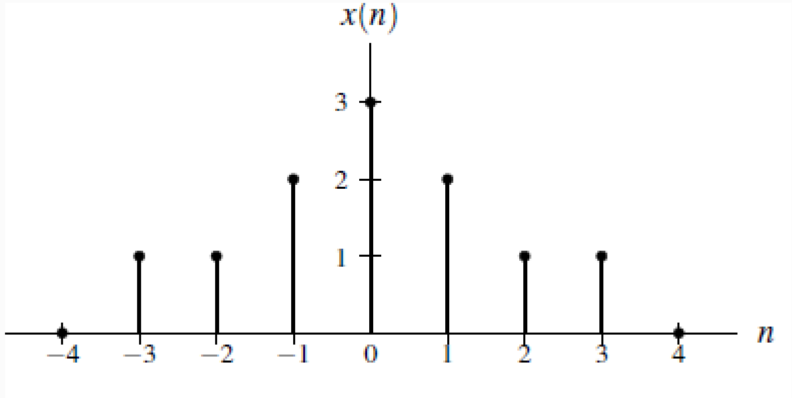
- The scaled signal $ax(t)$ is $x(t)$ multiplied by the constant a



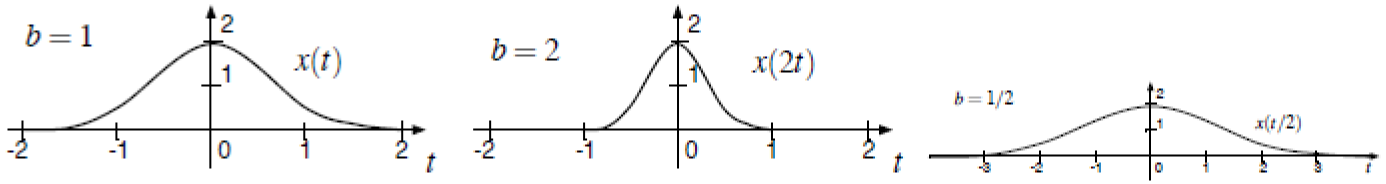
- The scaled signal $ax[n]$ is $x[n]$ multiplied by the constant a

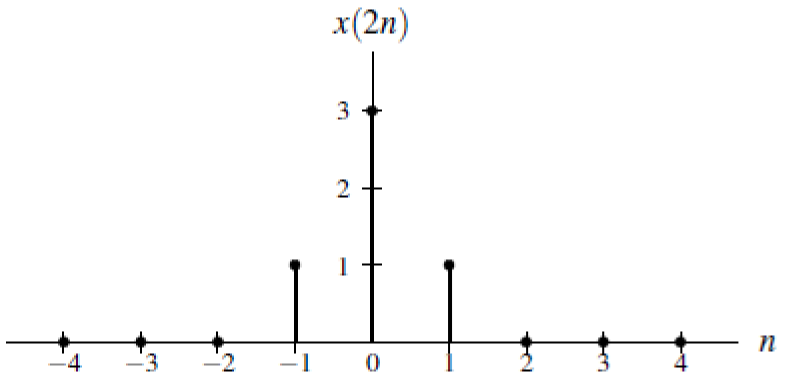


Time Scaling, Continuous Time



A signal $x(t)$ is scaled in time by multiplying the time variable by a positive constant b, to produce $x(bt)$. A positive factor of b either expands $(0 < b < 1)$ or compresses $(b > 1)$ the signal in time.



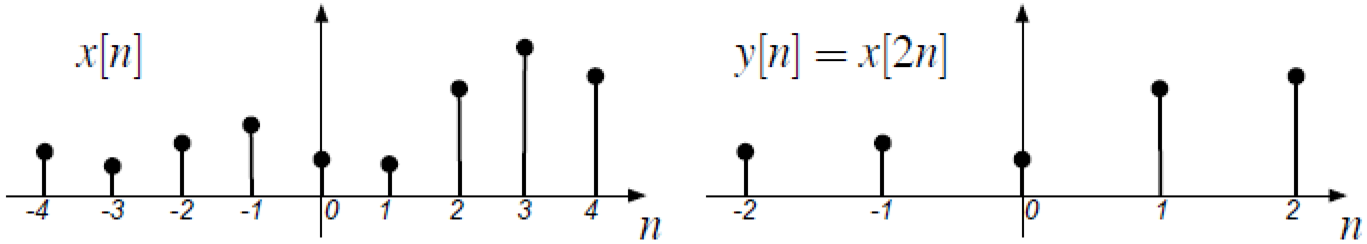


Operations on Signals

Time Scaling, Discrete Time

The discrete-time sequence $x[n]$ is *compressed* in time by multiplying the index n by an integer k, to produce the time-scaled sequence $x[nk]$.

- This extracts every k^{th} sample of $x[n]$.
- Intermediate samples are lost.
- The sequence is shorter.

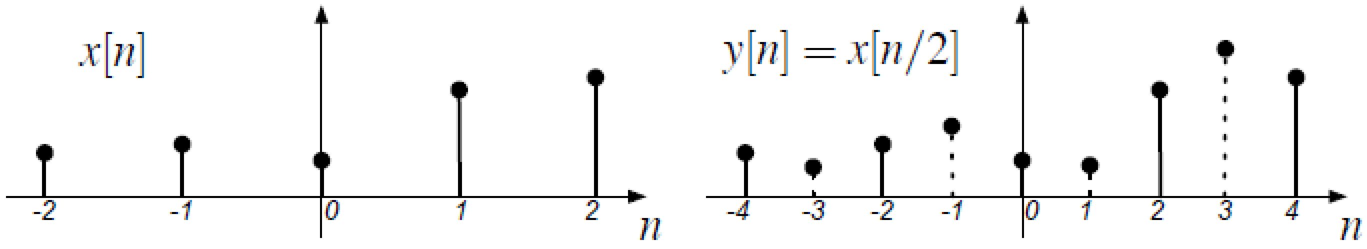


Called *downsampling*, or *decimation*.

Operations on Signals

The discrete-time sequence $x[n]$ is *expanded* in time by dividing the index n by an integer m, to produce the time-scaled sequence $x[n/m]$.

- This specifies every m^{th} sample.
- The intermediate samples must be synthesized (set to zero, or interpolated).
- The sequence is longer.

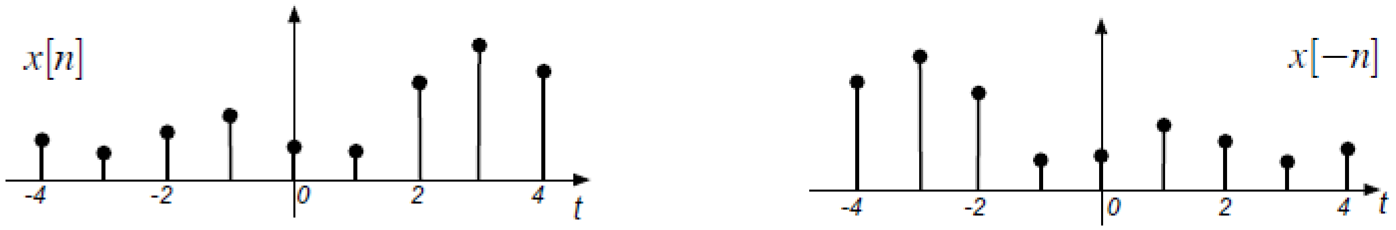


Called *upsampling*, or *interpolation*.

Time Reversal

- Continuous time: replace t with $-t$, time reversed signal is $x(-t)$

- Discrete time: replace n with $-n$, time reversed signal is $x[-n]$.

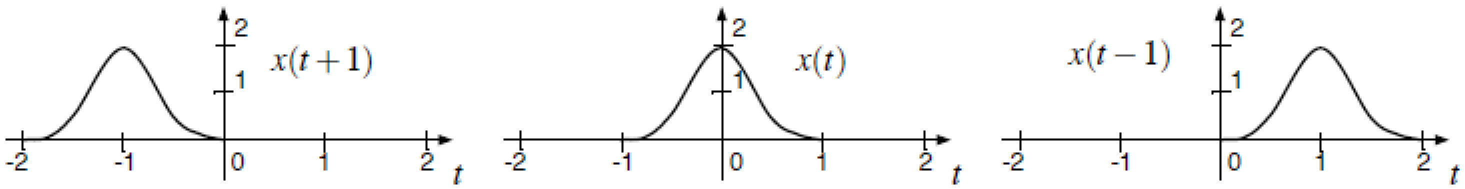


- Same as time scaling, but with $b = -1$.

Time Shift

For a continuous-time signal $x(t)$, and a time $t_1 > 0$,

- Replacing t with $t - t_1$ gives a *delayed* signal $x(t - t_1)$
- Replacing t with $t + t_1$ gives an *advanced* signal $x(t + t_1)$

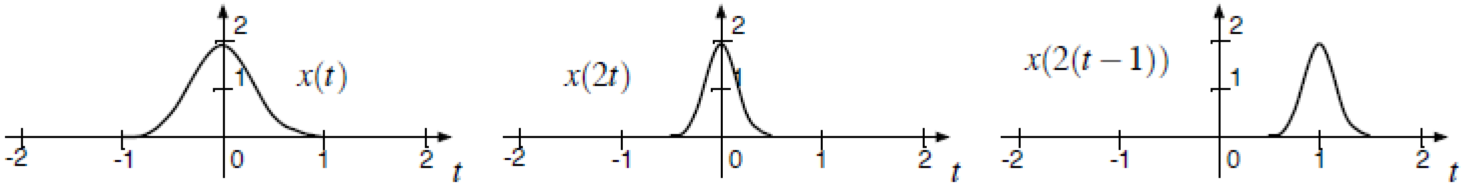


- May seem counterintuitive. Think about where $t - t_1$ is zero.

Combinations of Operations

- Time scaling, shifting, and reversal can all be combined.
- Operation can be performed in any order, but care is required.
- This *will* cause confusion.
- Example: $x(2(t-1))$

Scale first, then shift
Compress by 2, shift by 1

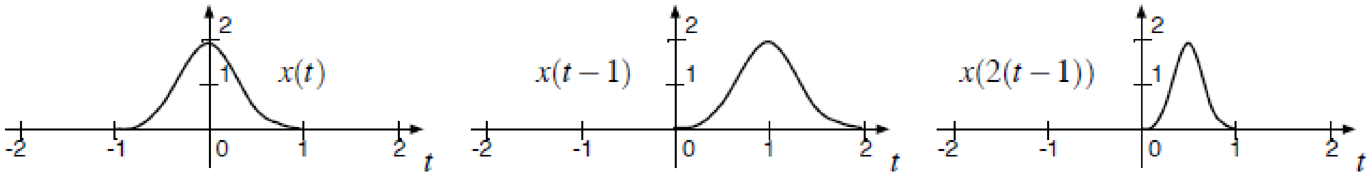


Example-1

Example $x(2(t-1))$, continued

Shift first, then scale

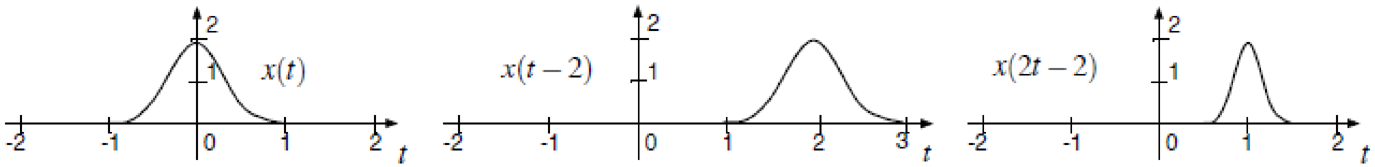
Shift by 1, compress by 2 *Incorrect*



Shift first, then scale

Rewrite $x(2(t-1)) = x(2t-2)$

Shift by 2, scale by 2 *Correct*

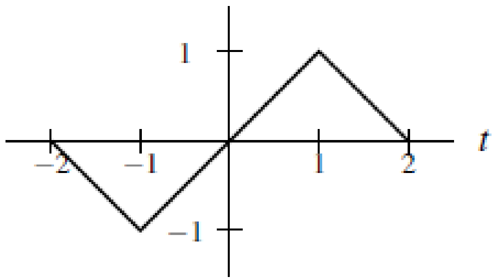


Where is $2(t-1)$ equal to zero?

Example-2

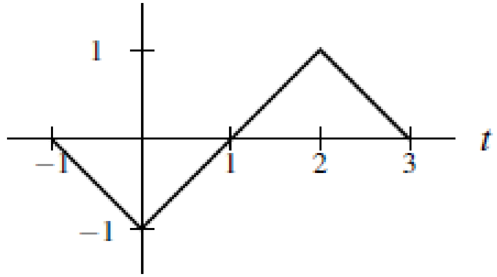
Given $x(t)$ as shown below, find $x(2t-1)$.

$x(t)$

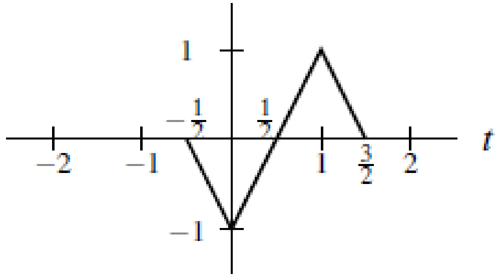


time shift by 1 and then time scale by 2

$p(t) = x(t-1)$

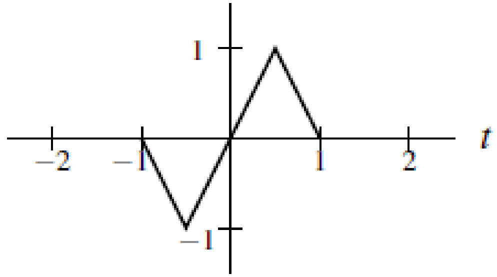


$p(2t) = x(2t-1)$

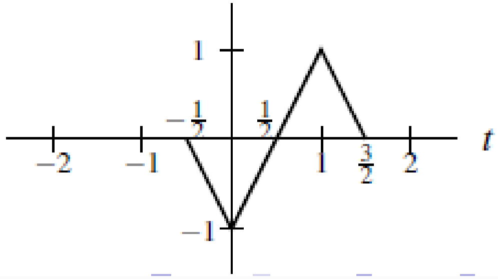


time scale by 2 and then time shift by $\frac{1}{2}$

$q(t) = x(2t)$



$q(t-1/2) = x(2(t-1/2)) = x(2t-1)$



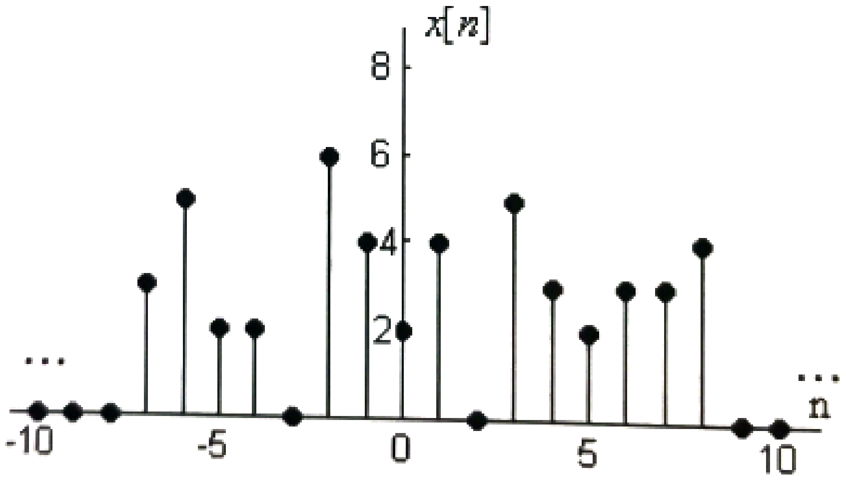
Homework (Just for you, do not e-mail)

Problemler:

2.1. x[n] dizisi Şekil 2.23'de gösterilmektedir. Aşağıdaki dizileri çiziniz.

(a) y[n]=x[n-3] (b) y[n] = x[-n]-x[n+2] (c) y[n]=x[-n+3] (d) y[n]=x[6-3n]

(e) y[n]=x[n/2-4]



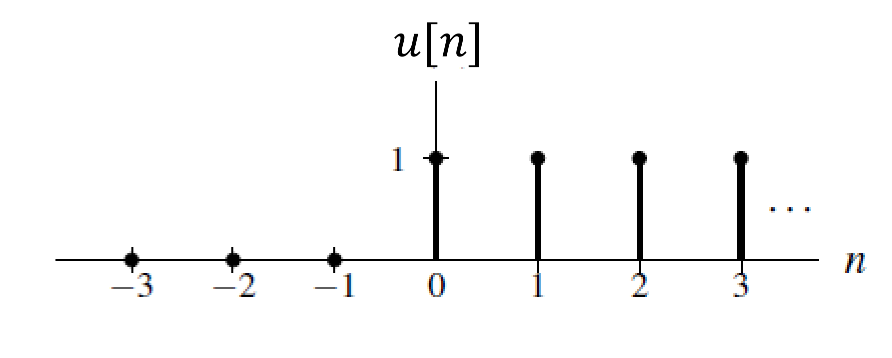
Şekil 2.23. Problem 2.1. için x[n] işareti

@Sayısal İşaret İşleme, Sarp ERTÜRK

Delta Function and Unit Step Function (Please Do Not Forget it!)

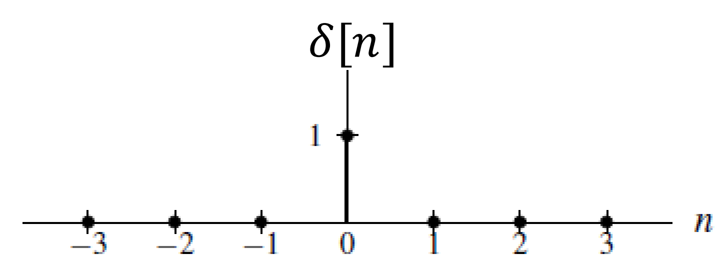
Unit Step Function

$$u(n) = \begin{cases} 1 & \text{if } n \geq 0 \\ 0 & \text{otherwise} \end{cases}$$



Unit Impulse Function

$$\delta(n) = \begin{cases} 1 & \text{if } n = 0 \\ 0 & \text{otherwise} \end{cases}$$



Relations

$$\delta[n] = u[n] - u[n-1]$$

Plot the signals given below:

1) $x[n] = 2\delta[n-1] + 3\delta[\mathrm{n}-2] + 4\delta[n-3]$ işaretini çiziniz.

2) $y[n] = 3\delta[n-2] + u[n-5] - u[n-7]$ işaretini çiziniz.

3) $g[n] = y[n]x[n]$ ifadesini bulunuz.

4) $h[n] = y[n] + x[n]$ ifadesini bulunuz.

MATLAB kodu:

```
clc; clear all;
%%
n = [0 1 2 3 4 5 6 7];
x = [0 2 3 4 0 0 0 0];
figure(1), stem(n,x,'filled');

%%
n = [0 1 2 3 4 5 6 7];
y = [0 0 3 0 0 1 1 0];
figure(2), stem(n,y,'filled');

%%
g=y.*x;
figure(3), stem(n,g,'filled');

%%
h=y+x;
figure(4), stem(n,h,'filled');
```