# Digital Signal Processing

## Theory & Laboratory



# Chapter 1

# Introduction

# SCOPE OF DIGITAL SIGNAL PROCESSING

## BACKGROUND

### Applied area of DSP

engineering, science, medicine, economics, social science, ...

### Things that we can do with DSP?

- detect trends in a signal
- extract a wanted signal from noise-contaminated signal
- assess frequency components in a signal



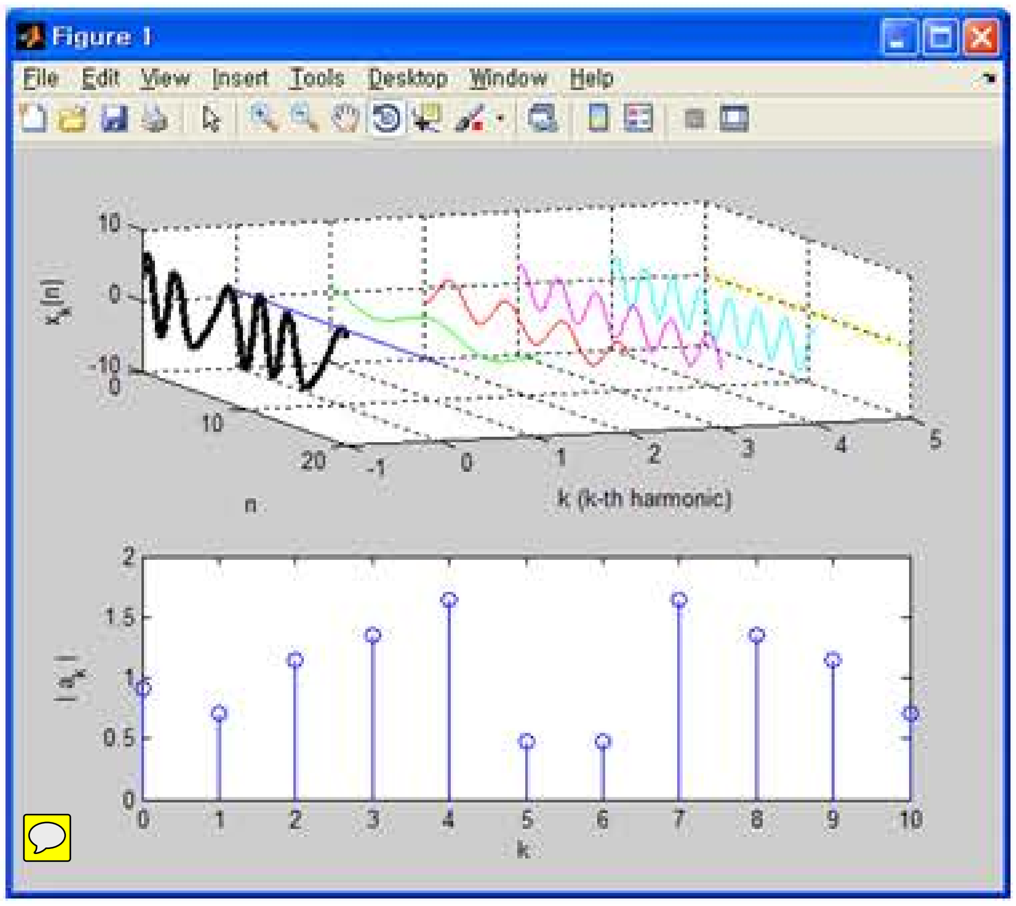
# SCOPE OF DIGITAL SIGNAL PROCESSING

### What is frequency component in a signal?

- discovery by Fourier
- every signal can be represented by combination of sinusoids

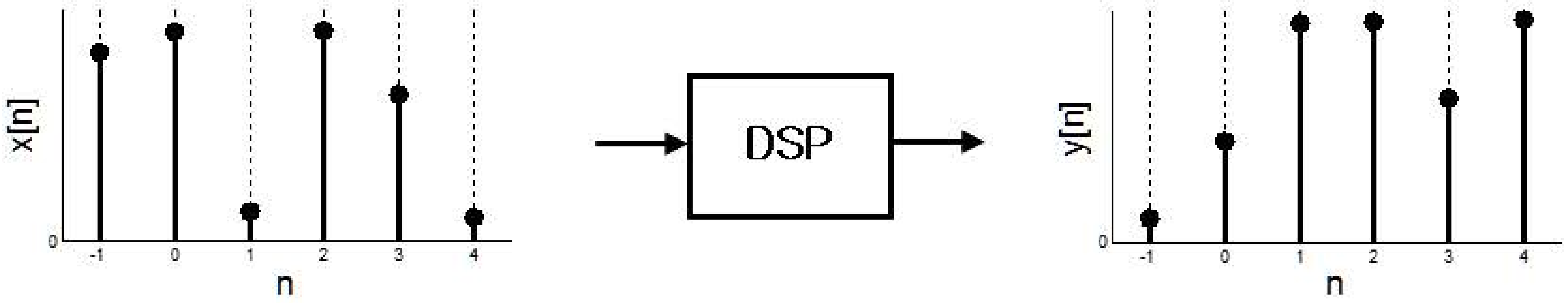
## Run m_ex_spectrum0.m

- shows relationship b/w time-domain and freq-domain representations of a signal
- k-th harmonic means freq component that completes k cycles in 1 period of origianl signal



# SCOPE OF DIGITAL SIGNAL PROCESSING

## Digital signal processors



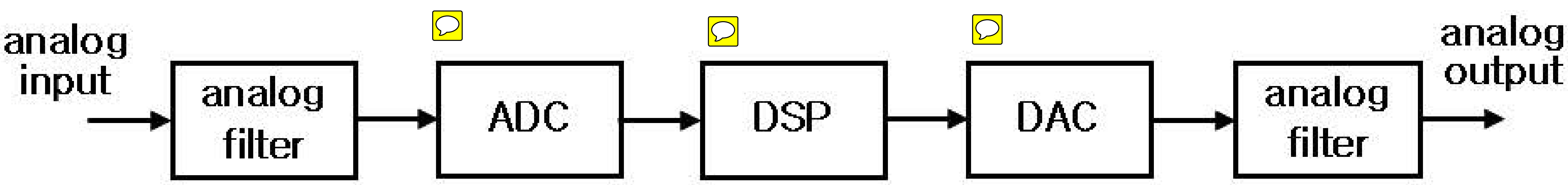

## What are they?

- difference equation



# SCOPE OF DIGITAL SIGNAL PROCESSING

## DSP scheme

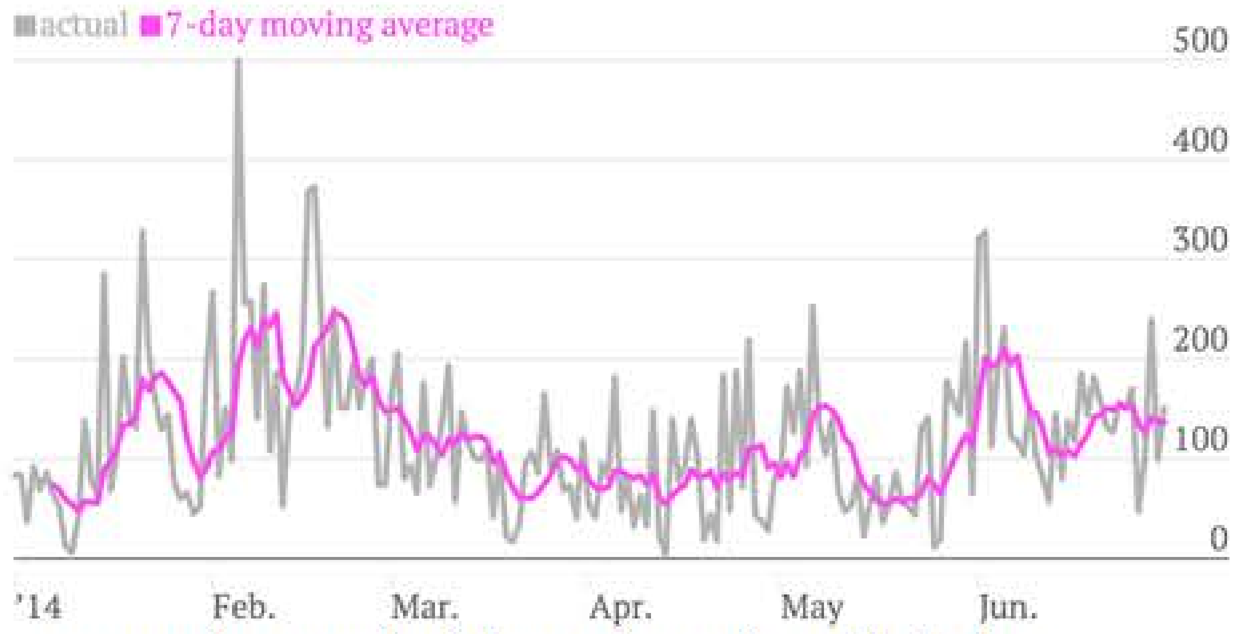



- ADC(analog-to-digital converter): sample and quantize
- DAC(digital-to-analog converter): zero-order hold
- Analog filter before ADC: limit freq. range of input signal before sampling
- Analog filter after DAC: remove sharp transitions from DAC output

# SCOPE OF DIGITAL SIGNAL PROCESSING

## SOME PRACTICAL APPLICATIONS

### Moving Average (MA)




- reveals underlying trends of data
- MA as DSP

x[n] → moving average → y[n]

- low pass filter



# SCOPE OF DIGITAL SIGNAL PROCESSING

### Equations for 200-Day MA

- nonrecursive form

$$y[n] = \frac{1}{200}\{x[n]+x[n-1]+x[n-2]+ \ldots x[n-199]\}$$
$$= 0.005\sum_{k=0}^{199} x[n-k]$$

- recursive form

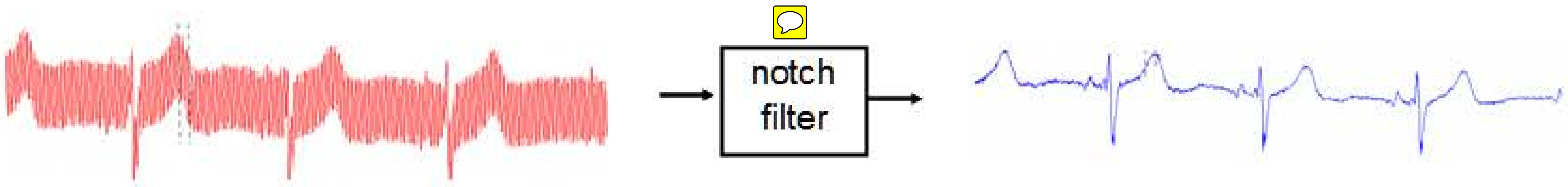
$$y[n+1] = 0.005\{x[n+1]+x[n]+x[n-1]+ \ldots x[n-198]\}$$
$$= y[n] + 0.005\{x[n+1]-x[n-199]\}$$

or

$$y[n] = y[n-1] + 0.005\{x[n]-x[n-200]\}$$

# SCOPE OF DIGITAL SIGNAL PROCESSING

## Medical application




heartbeat signal with 60Hz noise

filtered signal

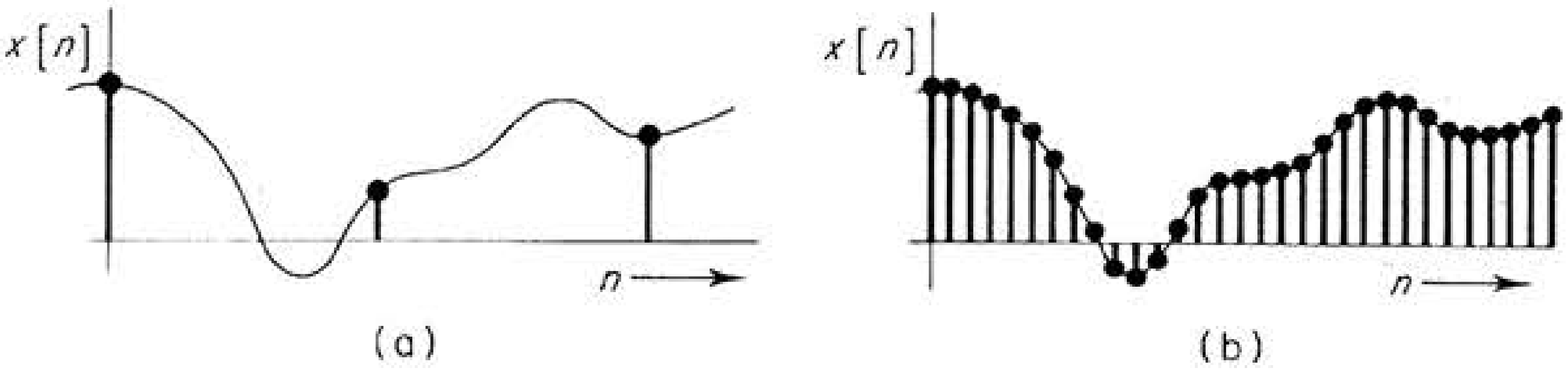
- filter equation (60Hz notch for 1.2KHz sampling)

$$y[n] = 1.8523y[n-1] - 0.94883y[n-2] + x[n] - 1.9021x[n-1] + x[n-2]$$



# SAMPLING & AD CONVERSION

## How often should we sample analog signal?




(a) too slow to pick out rapid fluctuations

(b) (maybe) too fast giving too many data

# SAMPLING & AD CONVERSION

## Sampling Theorem by Shannon

$f \geq 2f_{max}$ or $T \leq \dfrac{1}{2f_{max}}$

$f$ [Hz], $T$[s]: sampling frequency and interval
$f_{max}$[Hz]: maximum frequency component in analog signal

- Otherwise, aliasing occurs.

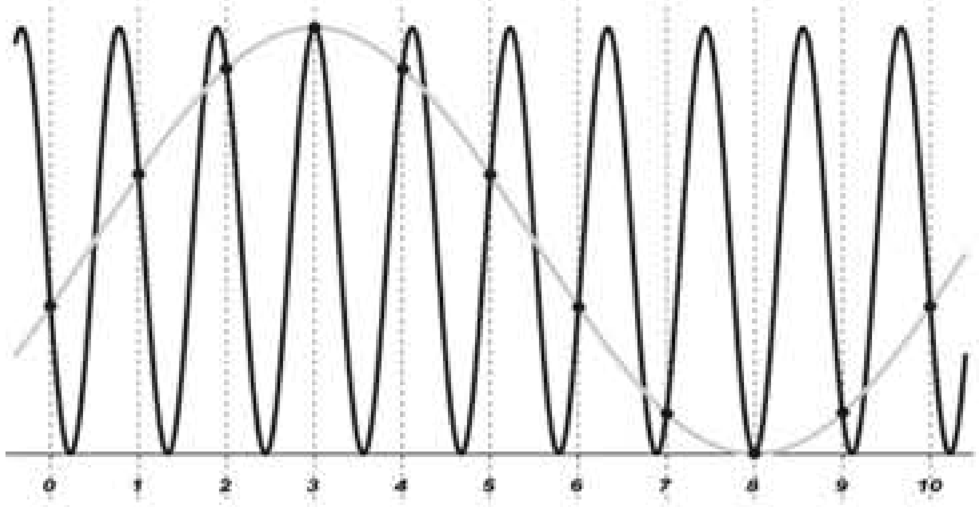
## Terminology

Nyquist frequency = half the sampling frequency $(=\dfrac{f}{2})$



# SAMPLING & AD CONVERSION

## Ambiguity in sampled sinusoid




$$\text{analog signal} \underset{\text{not unique}}{\overset{\text{unique (sampling)}}{\rightleftarrows}} \text{digital signal}$$

## Run m_sampledsinusoid.m

(1) f = 10Hz, f1 = 3Hz, f2 = 13Hz
(2) f = 10Hz, f1 = 3Hz, f2 = 7Hz

# SAMPLING & AD CONVERSION

## Aliasing

(Time Domain)

- With slow sampling, sampled signal (digital) looks slower than original signal (analog).

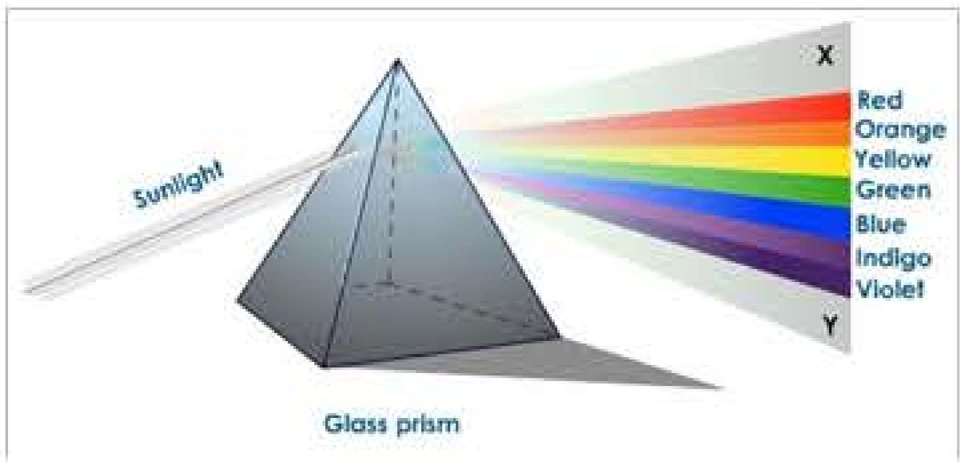
(Frequency Domain)

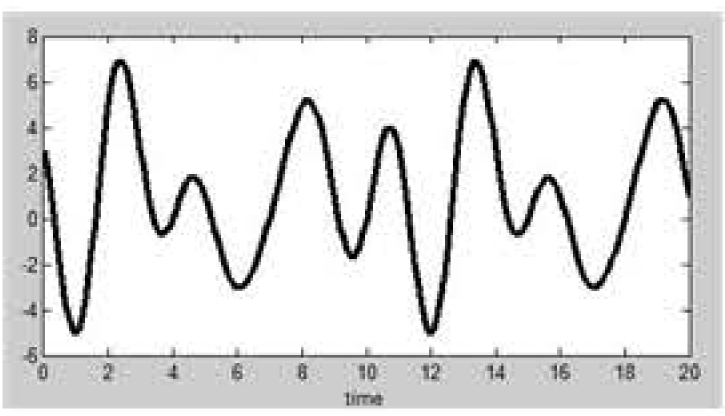
- Sampling causes original spectrum of analog signal to repeat around multiples of sampling frequency



# SAMPLING & AD CONVERSION

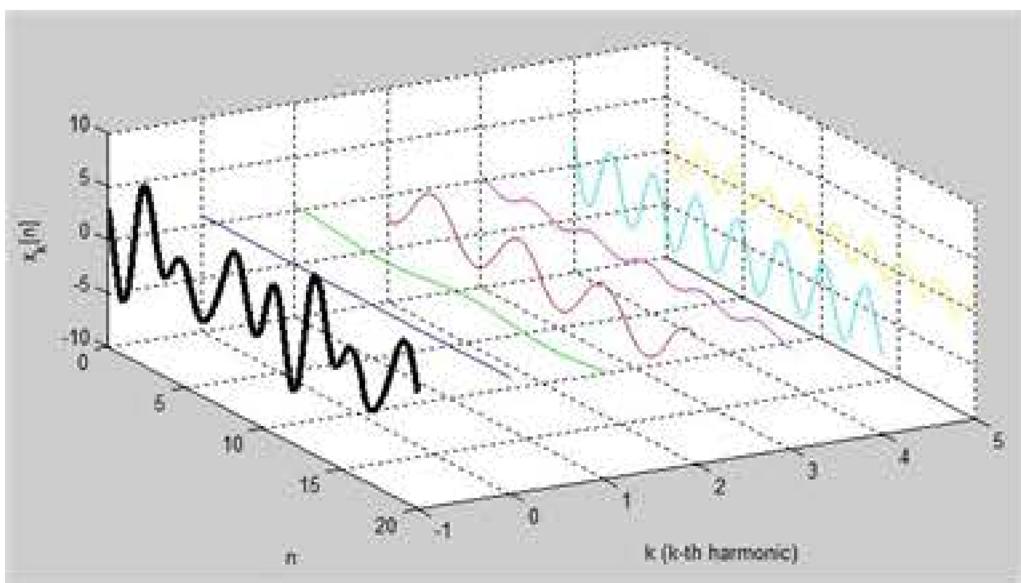
## Spectrum




time history

Fourier transform

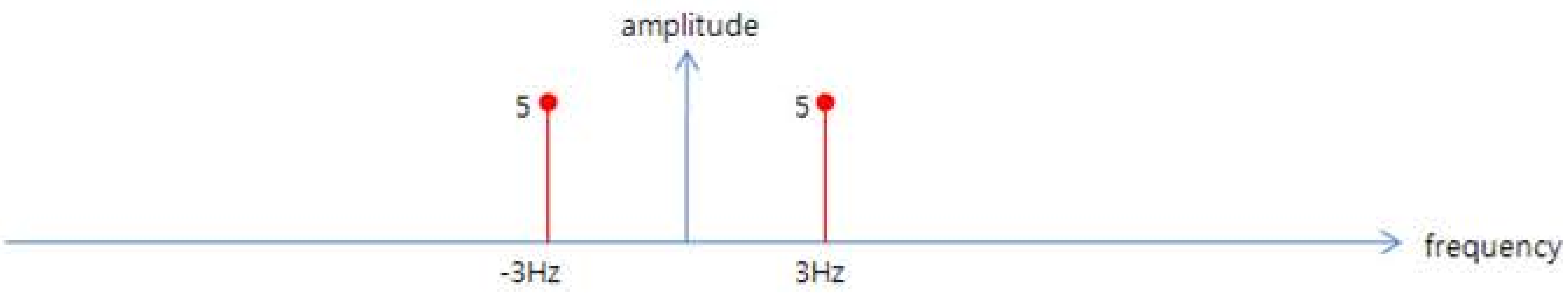



frequency spectrum

# SAMPLING & AD CONVERSION

## Effect of sampling in frequency domain

- consider pure sinusoid: $10\cos(2\pi f_a t)$ where $f_a = 3Hz$
- frequency spectrum




Q: Negative frequencies? Half amplitude?
A: Come from expressing sinusoid as sum of two exponentials

$$A\cos(\omega t) = \frac{A}{2}exp(j\omega t) + \frac{A}{2}exp(j(-\omega)t)$$

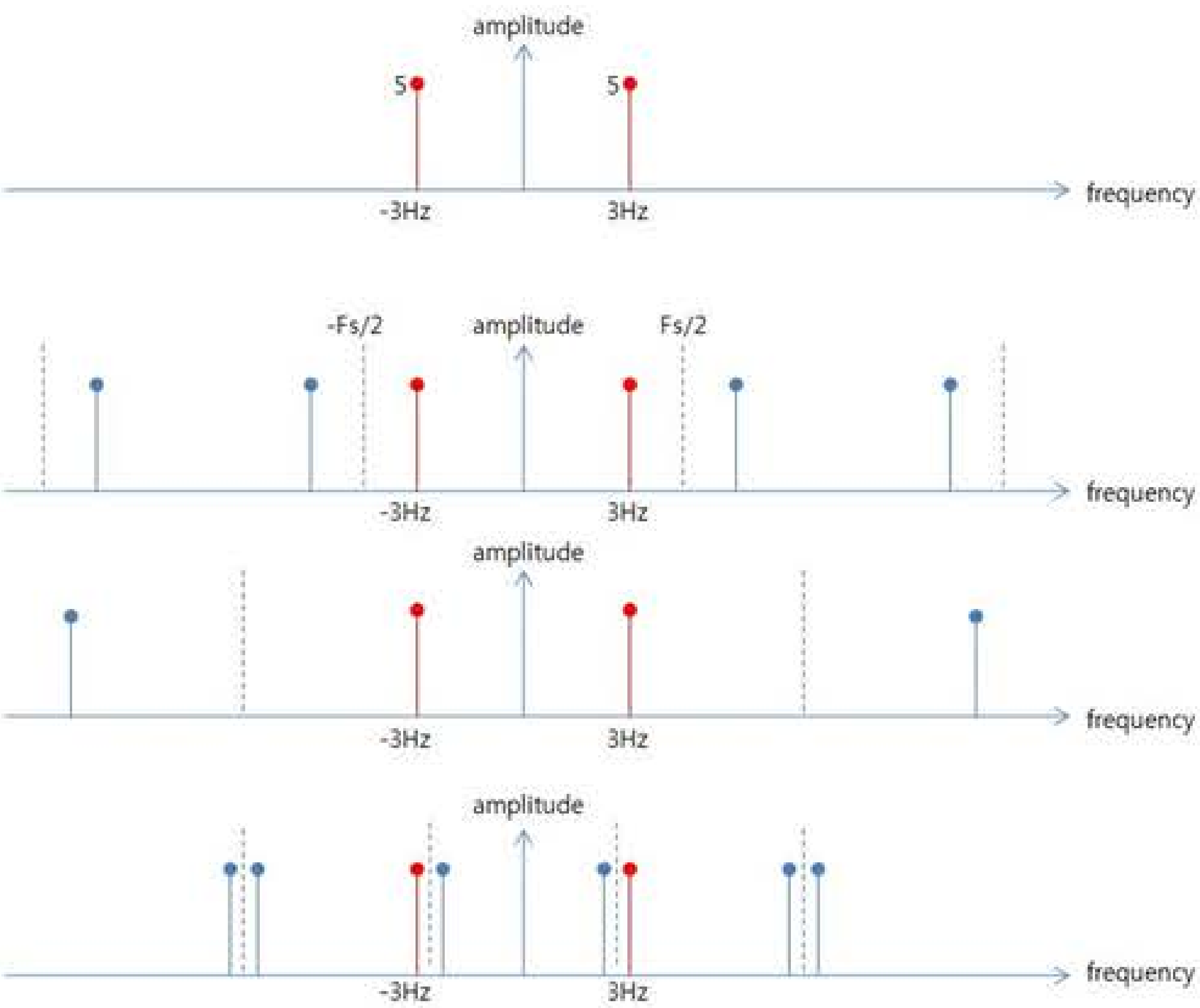
⇨ $\{A, \omega\}$ can be represented by $\left\{\frac{A}{2}, -\omega\right\}, \left\{\frac{A}{2}, \omega\right\}$

⇨ amplitude spectrum is always even



# SAMPLING & AD CONVERSION

- frequency spectrum after sampling (Fs: sampling freq)




→ amplitudes repeats by sampling freq!

# SAMPLING & AD CONVERSION

## Effect of Sampling in time domain

Q: We saw that sampled sinusoid contains so many freq components.
Which one among them does sampled sinusoid look like in time domain?

A: The lowest frequency.



# SAMPLING & AD CONVERSION

## Ex. Aliasing Examples

1) sinusoid freq = 120Hz, sampling freq = 100Hz
amplitudes repeats by sampling freq: ..., -180, -80, 20, 120, ...
amplitude spectrum is even: ..., -180, -120, -80, -20, 20, 120, 180, ...
lowest freq = 20Hz

2) sinusoid freq = 120Hz, sampling freq = 150Hz
120 - 150 = -30, 30Hz with opposite phase

3) sinusoid freq = 430Hz, sampling freq = 150Hz
430 - 150 - 150 - 150 = -20, 20Hz with opposite phase

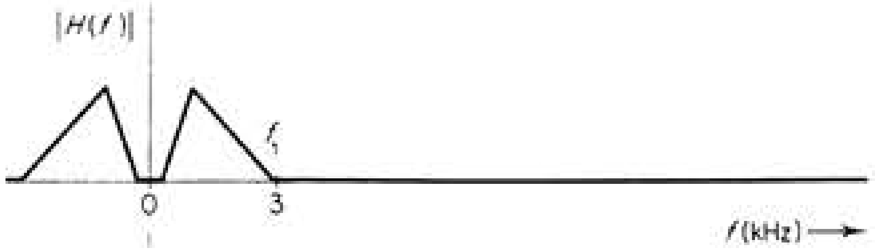
**Run again m_sampledsinusoid.m for above cases.**

- signal freq can be negative

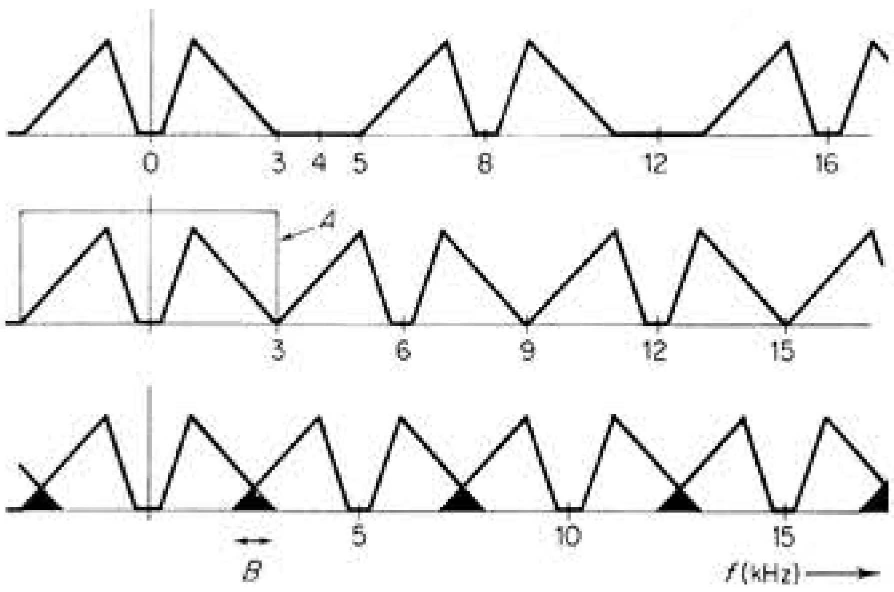
# SAMPLING & AD CONVERSION

## Aliasing for general signal

- consider analog signal with $f_{\max}$= 3kHz




- after sampling




(a) sampled signal ($f$ = 8kHz)

(b) sampled signal ($f$ = 6kHz)

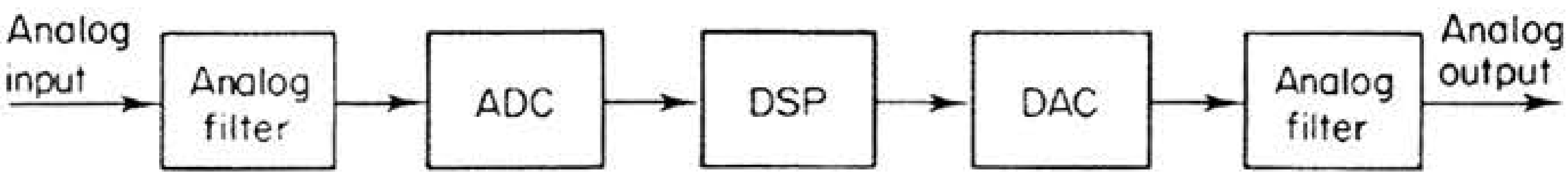
(c) sampled signal ($f$ = 5kHz) ⇨ aliasing



# SAMPLING & AD CONVERSION

## Anti-aliasing filter

- analog filter before ADC




- ensures $f_{\max} \le \frac{f}{2}$

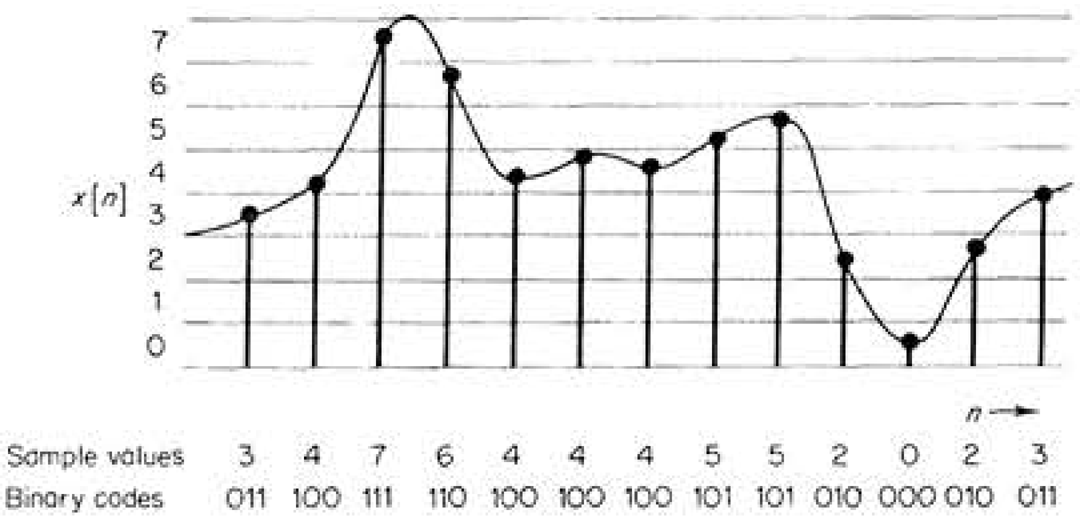
Q: Why must anti-aliasing filter be an analog filter?

A: If it is a digital filter, sampling must come before this filter and sampling causes aliasing.

# SAMPLING & AD CONVERSION

## QUANTIZATION

continuous amplitude --------quantization--------> finitely many amplitudes (binary code)



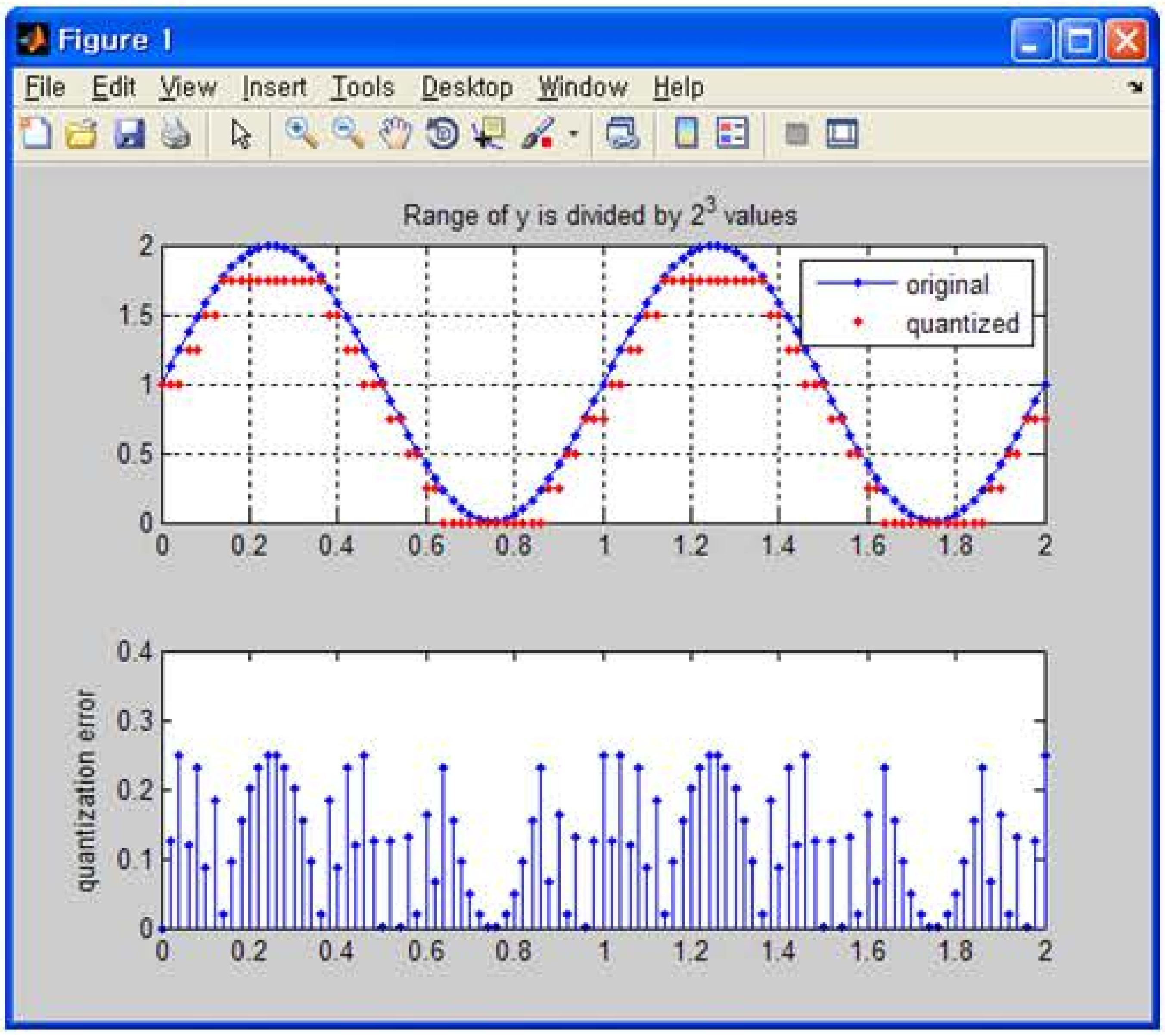

### quantization error

- difference b/w actual data and quantized data
- more bits (= better resolution) gives smaller error



# SAMPLING & AD CONVERSION

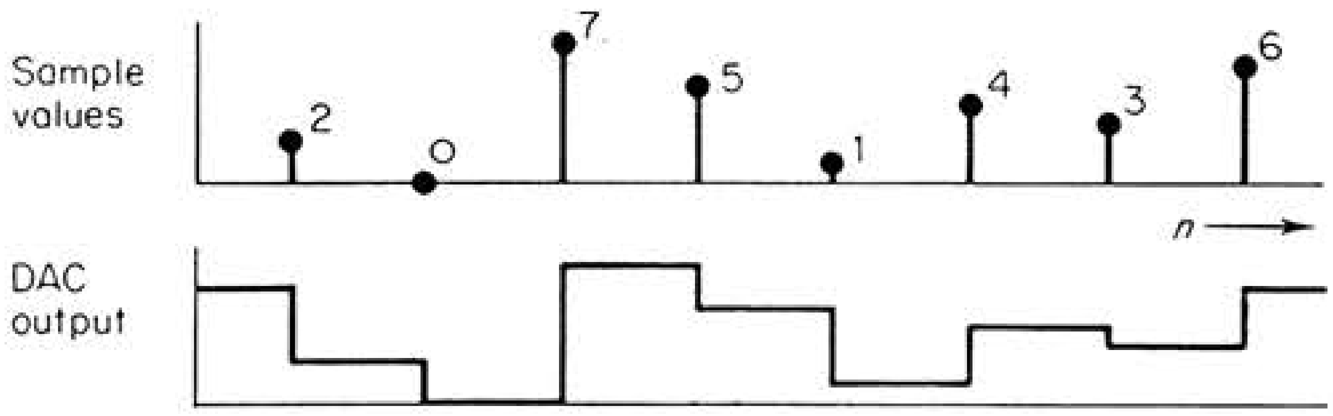
## Run m_quantize.m to understand quantization.



# SAMPLING & AD CONVERSION

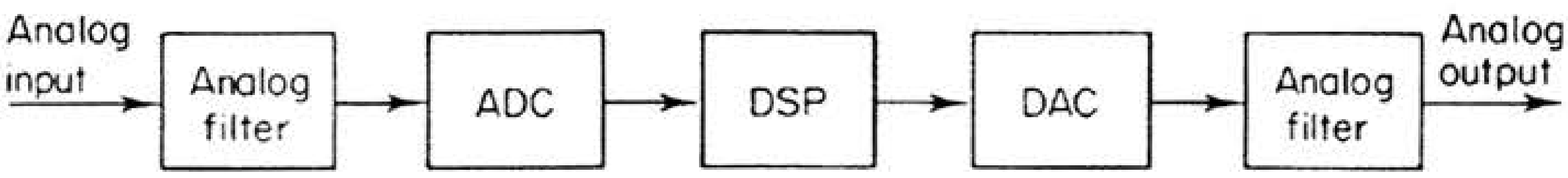
## DAC

- input: digital data
- output: zero-order-hold of input




### Smoothing Filter

- analog filter after DAC
- used when staircase waveform is not acceptable






# DIGITAL SIGNAL

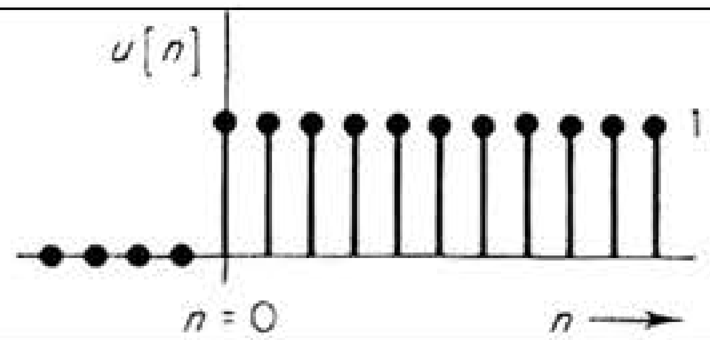
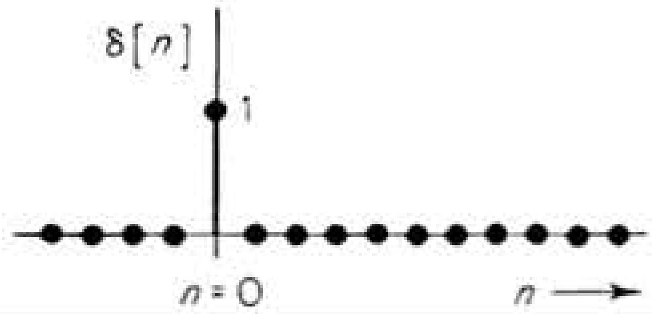
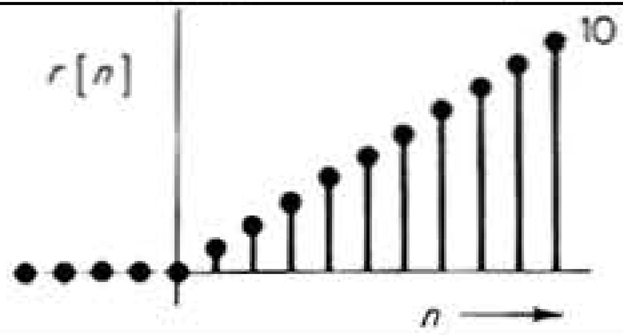
## DIGITAL SIGNALS

### Most important signals

- impulse
- step
- ramp
- exponential
- sinusoid

# DIGITAL SIGNAL

## STEP, IMPULSE, AND RAMP

| | | |
|---|---|---|
| step, $u[n]$ | $u[n]=0, n<0$<br>$u[n]=1, n \geq 0$ |  |
| impulse, $\delta[n]$ | $\delta[n]=0,\ n \neq 0$<br>$\delta[n]=1,\ n=0$ |  |
| ramp, $r[n]$ | $r[n]=n\,u[n]$ |  |

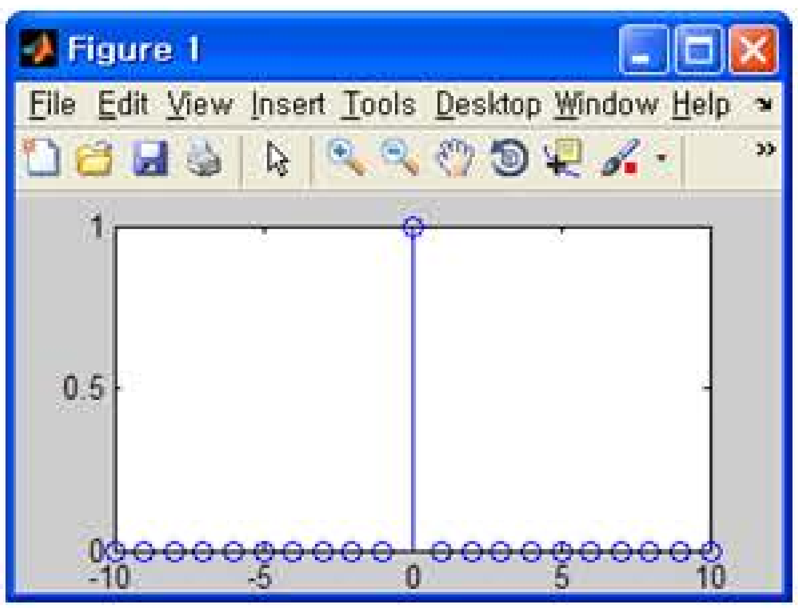
Note: $n$ is an integer.

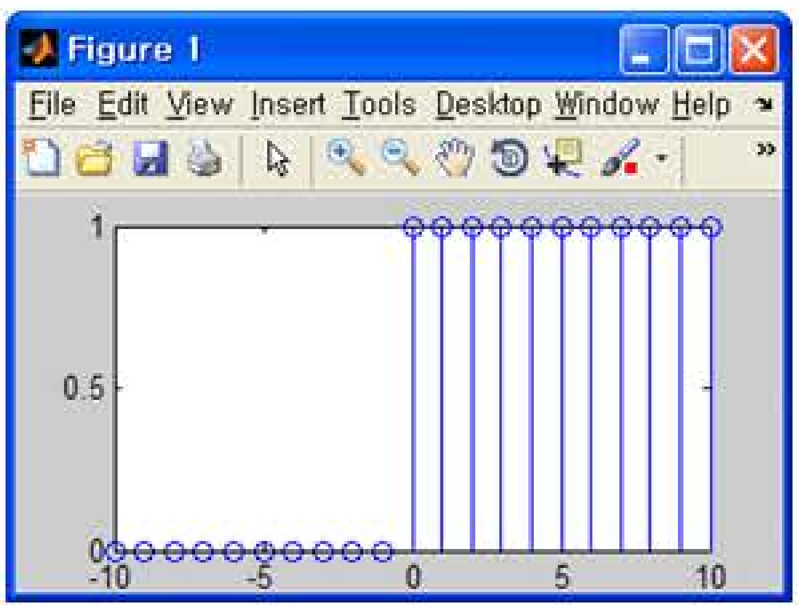


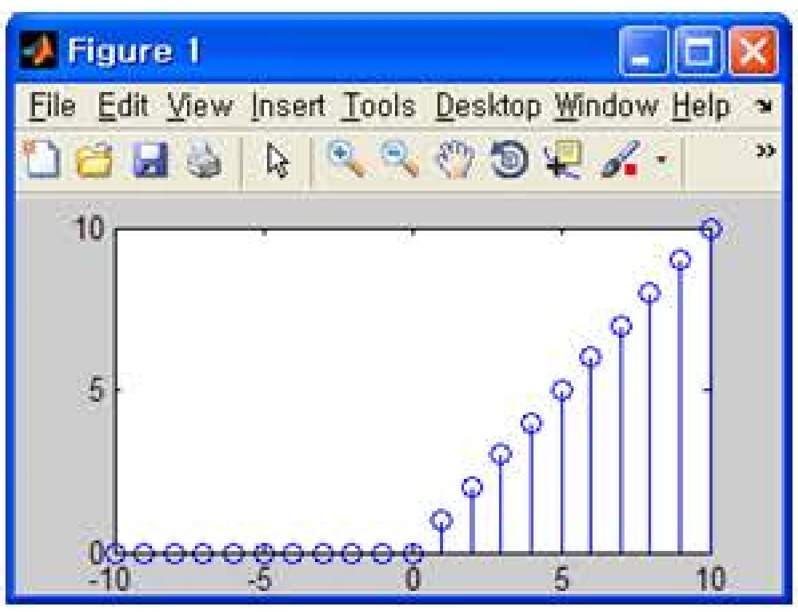
# DIGITAL SIGNAL

## Plot basic digital signals.

```
n = -10 : 10
n = n'
y1 = f_impulse(n)
y2 = f_step(n)
y3 = f_ramp(n)

stem(n,y1)
stem(n,y2)
stem(n,y3)
```



# DIGITAL SIGNAL

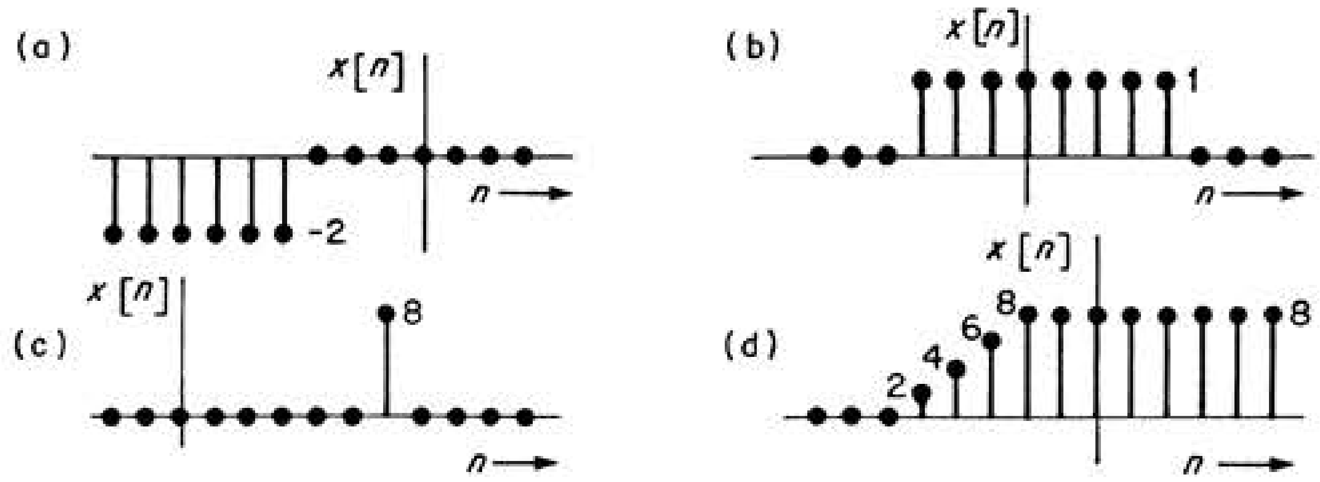
## Ex. Find expression.

(a) $x[n] = -2u[-n-4]$

(b) $x[n] = u[n+3] - u[n-5]$

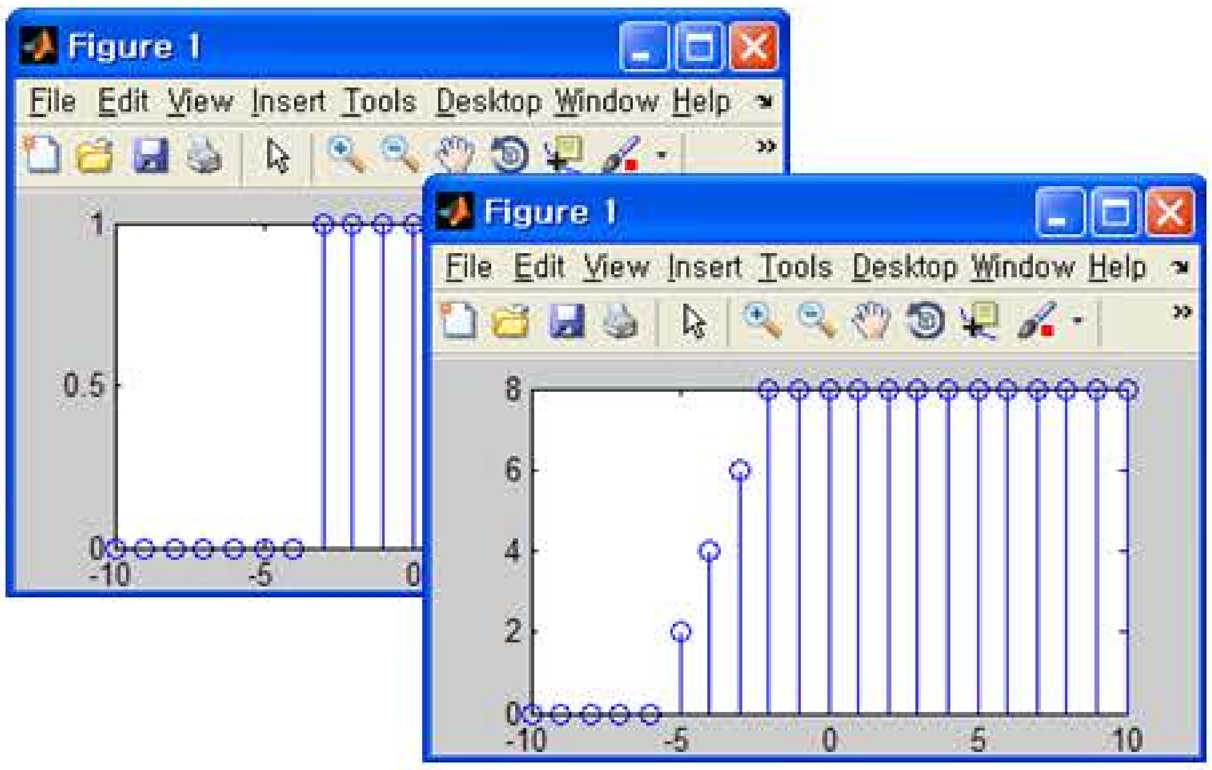
(c) $x[n] = \delta[n-8]$

(d) $x[n] = 2r[n+6] - 2r[n+2]$




## Plot (b) and (d).

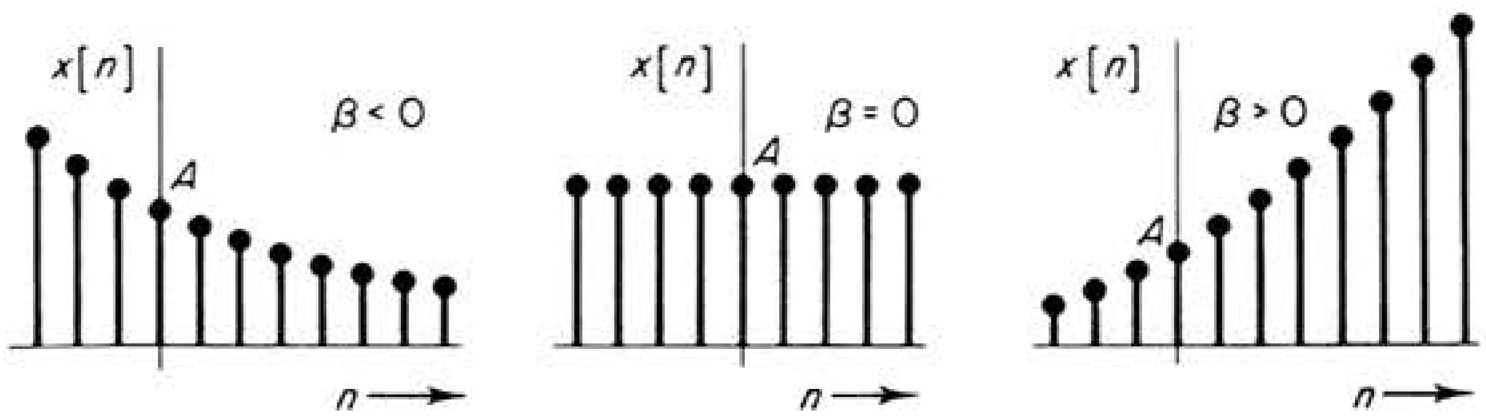
```
y = f_step(n+3) - f_step(n-5);
stem(n,y)

y = 2*f_ramp(n+6) - 2*f_ramp(n+2);
stem(n,y)
```






# DIGITAL SIGNAL

## EXPONENTIAL




### Exponential signal

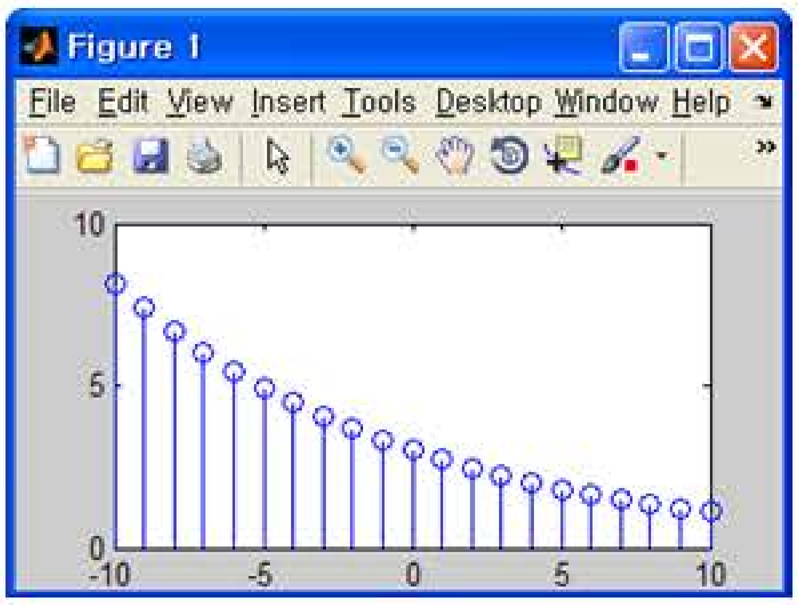
$x[n] = A\exp(\beta n)$

### Practical exponential signal

$$x[n] = A\exp(\beta n), \; n \geq 0 \qquad \text{or} \qquad x[n] = A\exp(\beta)u[n]$$
$$= 0, \qquad n < 0$$

## Plot exponential signal.

```
y = 3*exp(-0.1*n);
stem(n,y)
```



# DIGITAL SIGNAL

## SINUSOID

### digital sinusoid

cosine: $A\cos(\Omega n)$ , sine: $A\sin(\Omega n)$

### can be expressed as an exponential with imaginary power

$$x[n] = A\exp(j\Omega n) = A\cos(\Omega n) + jA\sin(\Omega n)$$

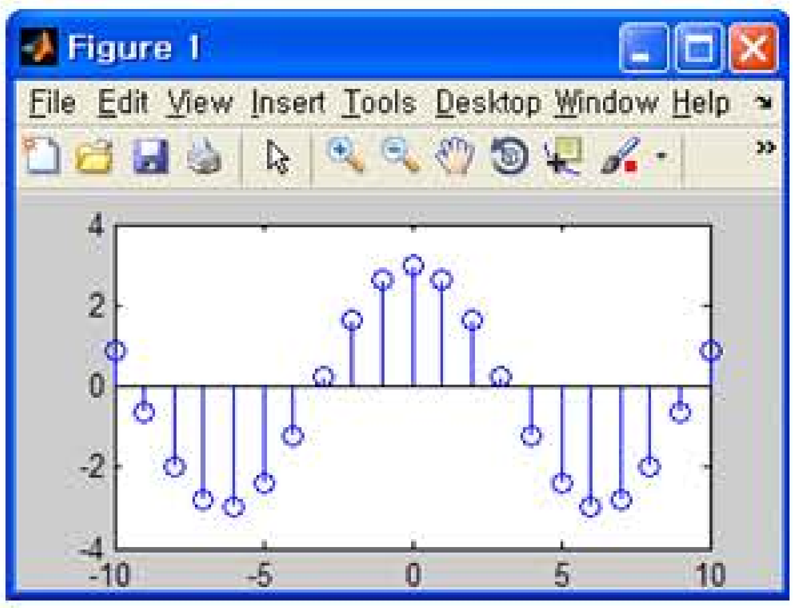
$$\Rightarrow \left\{ \begin{aligned} A\cos(\Omega n) &= Re(A\exp(j\Omega n)) \\ &= \frac{A}{2}exp(j\Omega n) + \frac{A}{2}exp(-j\Omega n) \\ A\sin(\Omega n) &= Im(A\exp(j\Omega n)) \\ &= \frac{A}{2j}exp(j\Omega n) - \frac{A}{2j}exp(-j\Omega n) \end{aligned} \right.$$

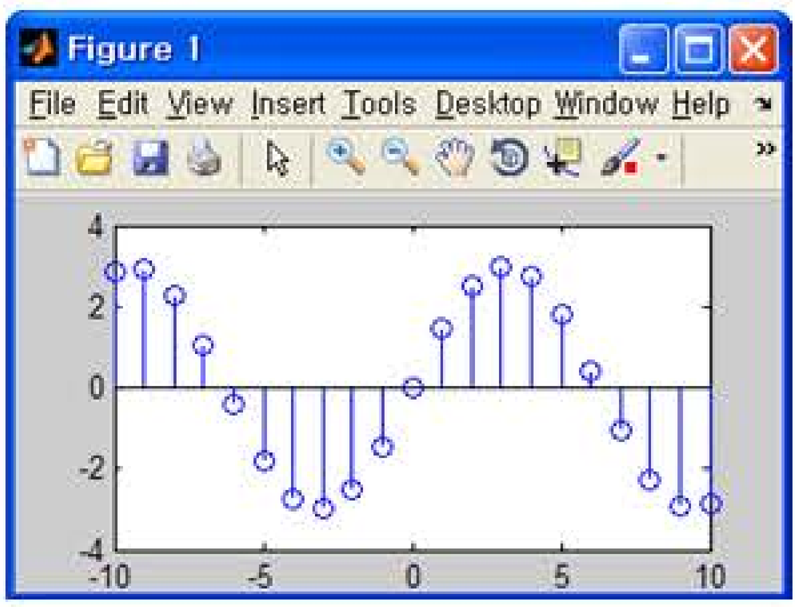


# DIGITAL SIGNAL

## Plot 3cos(0.5n) and 3sin(0.5n) using exp.

```
y = 3*exp(0.5i*n)
stem(n,real(y)); stem(n,imag(y))
```



# DIGITAL SIGNAL

## Time concept in sampled sinusoid

$$x(t) = \sin(\omega_a t)$$

$$\downarrow \quad t = nT,\ n = \ldots, 0, 1, 2, \ldots$$

$$x[n] = \sin(n\omega_a T) \quad \text{or} \quad x[n] = \sin\left(\frac{n2\pi f_a}{f}\right)$$

$$\downarrow \quad \text{let } \Omega = \omega_a T$$

$$x[n] = \sin(\Omega n)$$

where

$\omega_a$ [rad/s], $f_a$ [Hz]: analog sinusoid

$T$[s], $f$ [Hz]: sampling

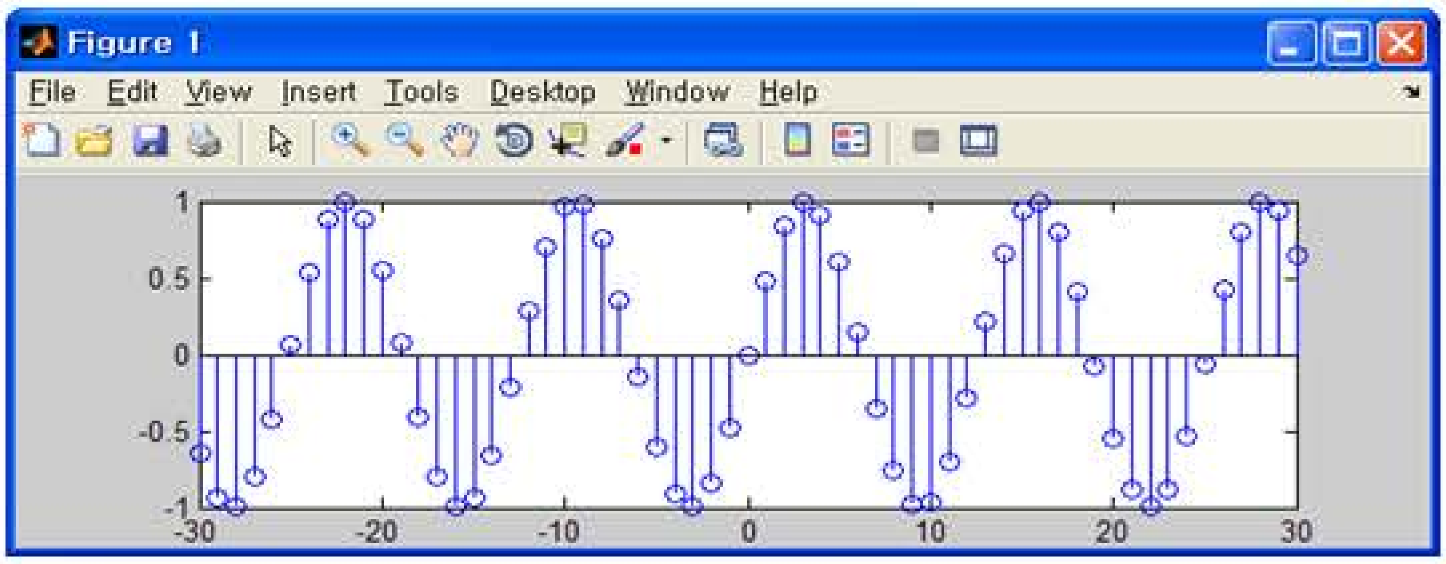


# DIGITAL SIGNAL

## Sampled sinusoid

- periodic only when $\Omega/2\pi$ is a rational number (ratio of two integers)

```
n = -30 : 30
stem(n,sin(0.5*n)) % sin(0.5n) is not periodic
```




## Period

- period is represented by N ( = # of samples/cycle)
- $\Omega$ (frequency) has unit [rad].

# DIGITAL SIGNAL

## Ex. Find period

1) $x[n] = \cos(0.2\pi n)$

2) $x[n] = \sin(\frac{7\pi}{6} n)$

3) $x[n] = \sin(0.5n)$

4) $x[n] = \cos(\frac{2\pi}{3} n) + 0.7\sin(\frac{\pi}{2} n)$

## Sol.

1) N = 10

2) N = 12

3) aperiodic ($0.5n$ ($n$: integer) never becomes a multiple of $2\pi$)

4) first term: N = 3, second term: N = 4 ⇨ N = 12



# DIGITAL SIGNAL

## Sinusoid with exponential amplitude

⇦ when $\beta$ is complex (= $\beta_0 + j\Omega$) in $A\exp(\beta n)$,

$$x[n] = A\exp(\{\beta_0 + j\Omega\}n) = A\exp(\beta_0 n)\exp(j\Omega n)$$

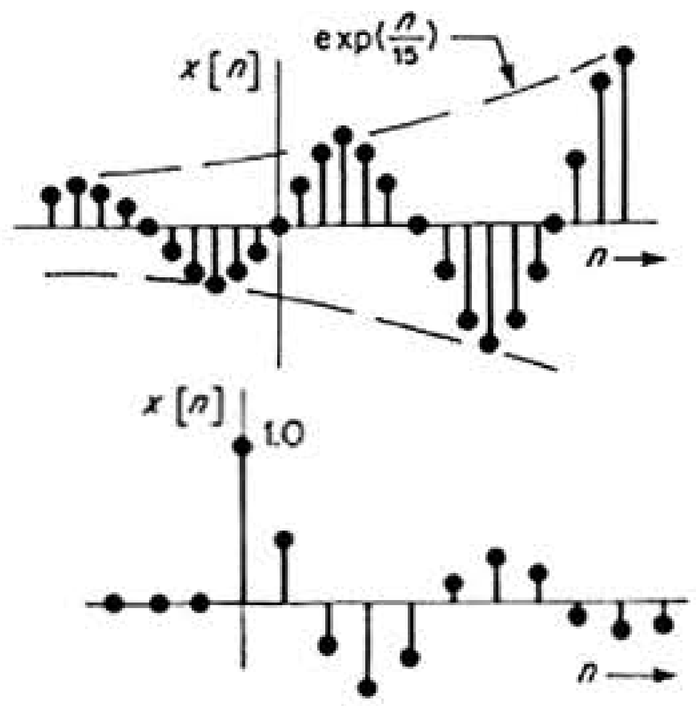
$$\Rightarrow \begin{cases} x[n] = A\exp(\beta_0 n)\sin(\Omega n) \\ x[n] = A\exp(\beta_0 n)\cos(\Omega n) \end{cases}$$

## Ex.

$$x[n] = \exp\left(\frac{n}{15}\right)\sin\left(\frac{\pi n}{6}\right)$$

```
y = exp((1/15+1/6*pi*1i)*n);
stem(n,imag(y))
```

$$x[n] = \exp\left(\frac{-n}{5}\right)\cos(n)u[n]$$



# DIGITAL SIGNAL

## Digital Sinusoid

- A digital sinusoid with frequency $\Omega$ is identical to those with $\Omega \pm 2\pi$, $\Omega \pm 4\pi$, ...

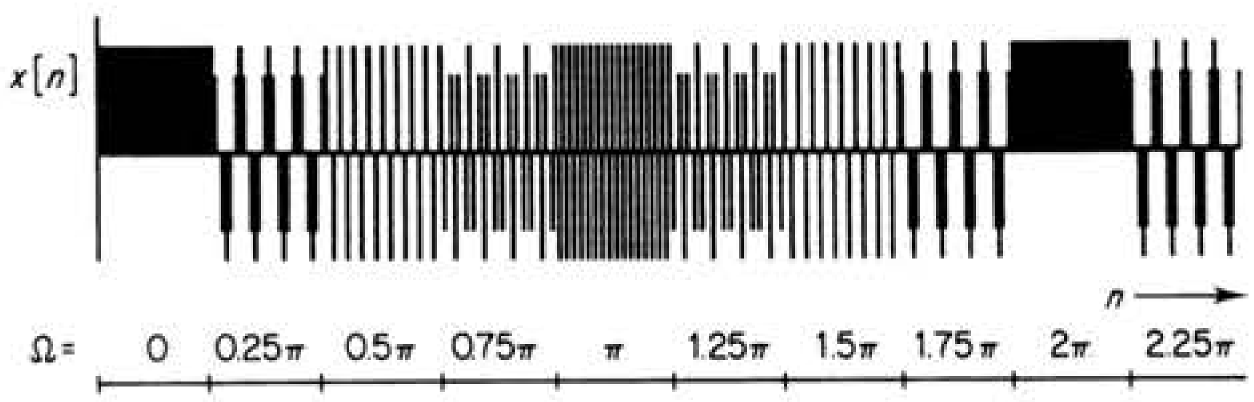
$$x[n] = \exp(j\Omega n) = \exp(j\{\Omega + 2\pi\}n)$$



# DIGITAL SIGNAL

## Effect of frequency increase in digital sinusoid

$$x[n] = \cos(\Omega n) = \cos\left(\frac{2\pi m}{8} n\right), \quad (m = 0,1,2, ...)$$




$\Omega = 0 \sim \pi \ (m = 0 \sim 4)$ ⇨ rate of oscillation increases

$\Omega = \pi \sim 2\pi \ (m = 4 \sim 8)$ ⇨ rate of oscillation decreases

$\Omega > 2\pi$ ⇨ pattern for $\Omega \in [0, 2\pi]$ repeats

**Run m_freqincrease.m**

- shows effect when increasing $m$ in digital sinusoid $x[n] = \cos\left(\frac{2\pi m}{8} n\right)$

# DIGITAL SIGNAL

### Summary of digital sinusoid

- In $x[n]=\cos(\Omega n)$, no ambiguity when $\Omega \leq \pi$, but aliasing when $\Omega > \pi$ (such effect does not happen in analog sinusoid, $x(t)=\cos(\omega t)$)
- $\Omega = \pi$ corresponds to $f_a = \frac{f}{2}$ (Nyquist freq) i.e., fastest freq for a given sampling.

### Ex.

$x[n]=\cos(5\pi n)$ was obtained by sampling analog sinusoid with 10Hz freq. Find freq of original analog signal.

Sol) $\Omega = \omega_a T$ ⇨ $5\pi = \omega_a \times 0.1 = 2\pi f_a \times 0.1$ ⇨ $f_a = 25Hz$

We sampled 25Hz sinusoid by 10Hz sampling! Aliasing!
It will look like 5Hz sinusoid after sampling. (5Hz = 25 - 10 - 10)



# DIGITAL SYSTEM

## LINEAR TIME-INVARIANT(LTI) SYSTEMS

### Form of digital processor (= system)

- CPU that is programmed for DSP task
- dedicated digital H/W

### linearity

linear system = system that obeys Principle of Superposition

known: $x_1[n] \rightarrow$ linear system $\rightarrow y_1[n]$, $x_2[n] \rightarrow$ linear system $\rightarrow y_2[n]$

unknown: 1) $x_1[n]+x_2[n] \rightarrow$ linear system $\rightarrow$ ?

2) $ax_1[n] \rightarrow$ linear system $\rightarrow$ ?

ans: 1) $y_1[n]+y_2[n]$ 2) $ay_1[n]$

# DIGITAL SYSTEM

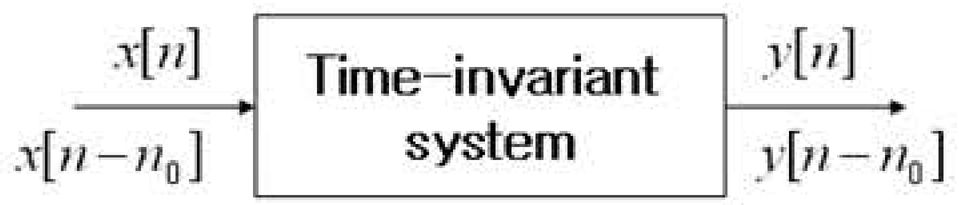
## frequency preservation

$$A\cos(\Omega n) \rightarrow \boxed{\text{linear system}} \rightarrow B\cos(\Omega(n+M))$$

output contains only those frequencies present in the input

## time invariance

⇨ system properties don't change with time

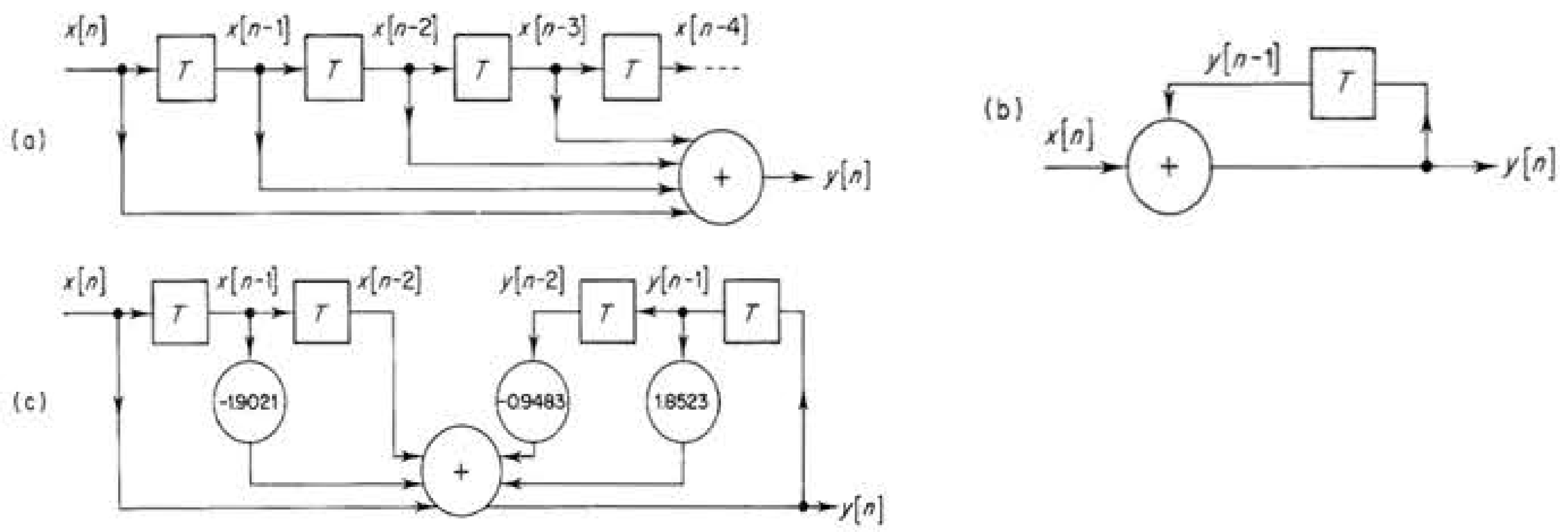





# DIGITAL SYSTEM

## Block diagram (used for dedicated HW implementation)

- elements: summation, gain, delay $\boxed{T}$




(a) $y[n] = x[n] + x[n-1] + x[n-2] + \cdots$

(b) $y[n] = y[n-1] + x[n]$

(c) $y[n] = 1.8523y[n-1] - 0.94833y[n-2] + x[n] - 1.9021x[n-1] + x[n-2]$

# DIGITAL SYSTEM

## MORE SYSTEM PROPERTIES

### causality

Output depends only on present and previous values of input.

### stability

System gives bounded output in response to bounded input.

### invertibility

If system gives y[n] in response to x[n], its inverse gives x[n] in response to y[n].

### memory

Output depends on previous input values.



# DIGITAL SYSTEM

### Ex. Determine properties of following systems

(linearity, time invariance, causality, stability, invertibility, memory)

(a) $y[n] = 3x[n] - 4x[n-1]$

⇨ all six properties hold

(b) $y[n] = 2y[n-1] + x[n+2]$

⇨ noncausal, unstable

(c) $y[n] = n\,x[n]$

⇨ time-variant, no memory

(d) $y[n] = \cos(x[n])$

⇨ nonlinear, time-invariant, noninvertible, no memory