EDISP
(English) Digital Signal Processing

Jacek Misiurewicz
e-mail: jmisiure@elka.pw.edu.pl

Institute of Electronic Systemss
Warsaw University of Technology
Warsaw, Poland

General information

“Credits” 2h/week lecture + 2h/week $\approx 7 \cdot 4h$ **laboratory** exercises in groups ≤ 12 persons.

Next lecture Mon, 10:15-12 $\longrightarrow$ *the lab schedule will be announced then!!*

Textbook A. V. Oppenheim, R. W. Schafer, *Discrete-Time Signal Processing*, Prentice-Hall 1989 (or II ed, 1999; also acceptable previous editions entitled *Digital Signal Processing*).

Contact J. Misiurewicz, (jmisiure@elka.pw.edu.pl) room 453. A web page is under construction (http://staff.elka.pw.edu.pl/˜jmisiure/)

Homeworks Announced as a preparation for the labs.

Exams Two short tests within lecture hours (TBA with the lab schedule) and a final exam during the winter exam session (TBA).

Scoring:

2x10%	=	20%	tests
6x5%	=	30%	lab + homework (lab 0 – not scored)
		50%	final exam

Signal classification

Continuous or Discrete **amplitude** and **time**.

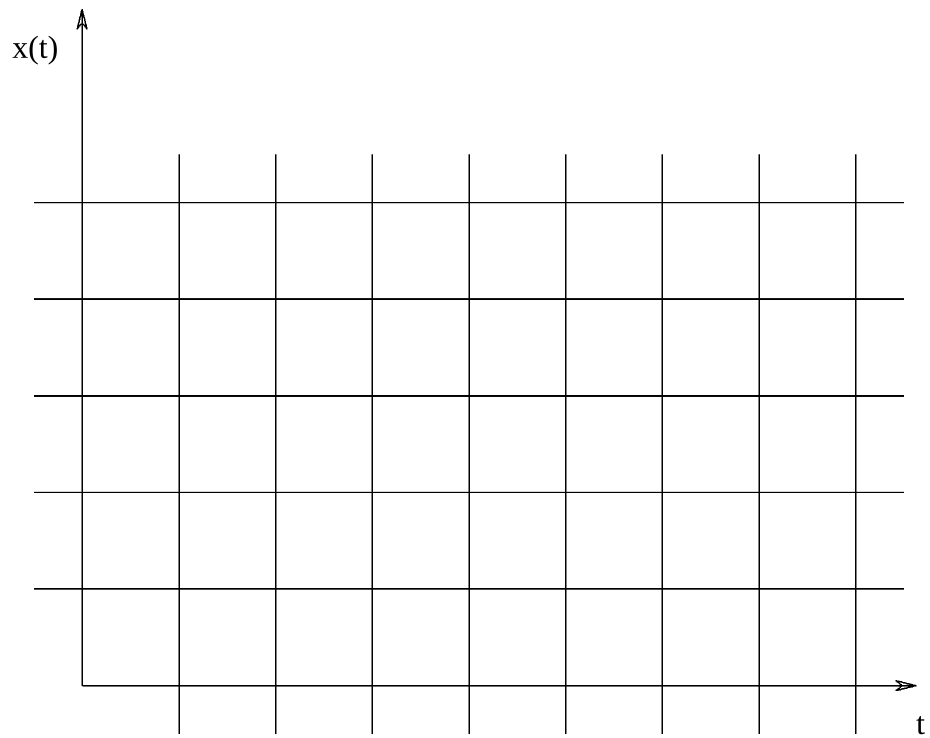
CA-CT ⟶ “analog” signals

DA-CT ⟶

CA-DT ⟶ CCD, SC, SAW devices

DA-DT ⟶ digital devices

We’ll speak mainly about DT properties; only in some subject DA will be of importance.



DT signal representations

DT signal $\longleftarrow\longrightarrow$ a number sequence

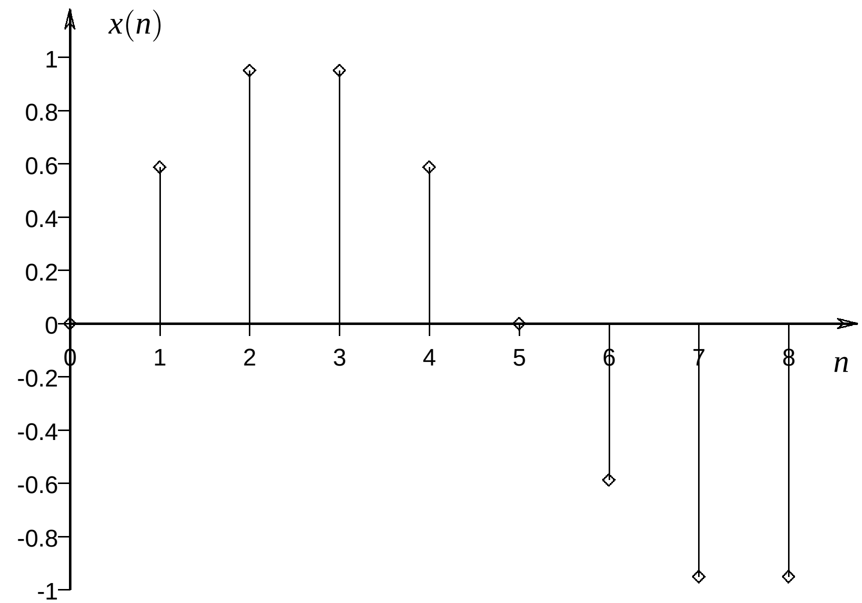
$$x[n] = \{x(n)\}$$

$x[n]$ is a number sequence (or ...)

$x(n)$ is a $n-$th sample

$\longrightarrow$ $x(n)$ is *undefined* for $n \notin \mathrm{I}$

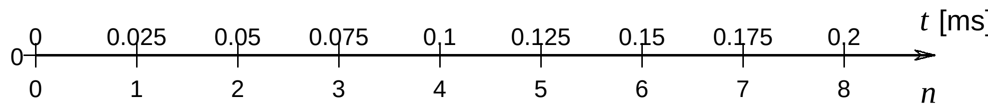
- it *may* come from sampling of analog signal
- but it may also be inherently discrete
- n may correspond to: time, space, ...



However, the most popular interpretation is: periodic sampling in time.

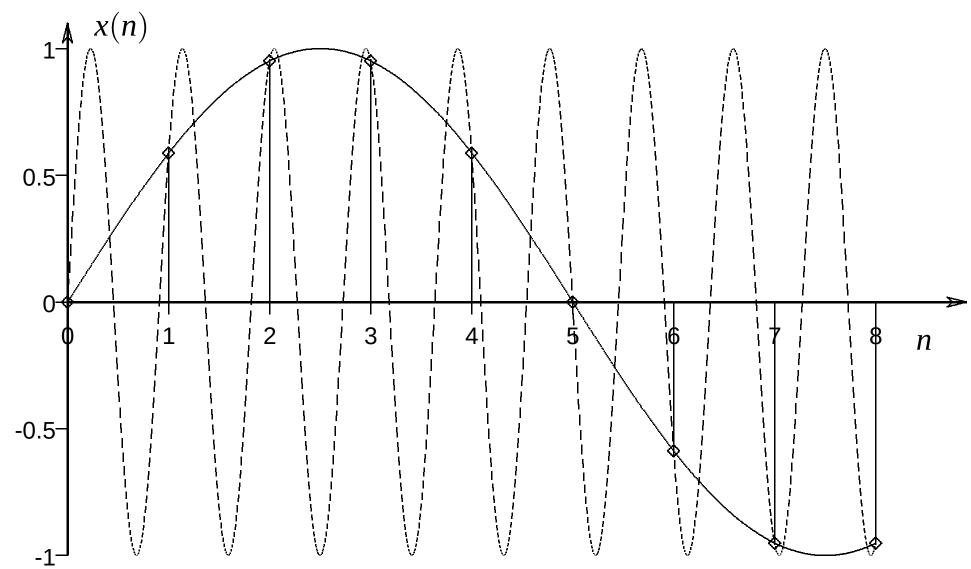
Periodic sampling

$$n \longleftarrow\longrightarrow n \cdot T_s$$



$$x(n) = x_a(nT_s)$$

$$n = t/T_s, \;\; T_s = 0.025 \; [\mathrm{ms}]$$



Misinterpretations

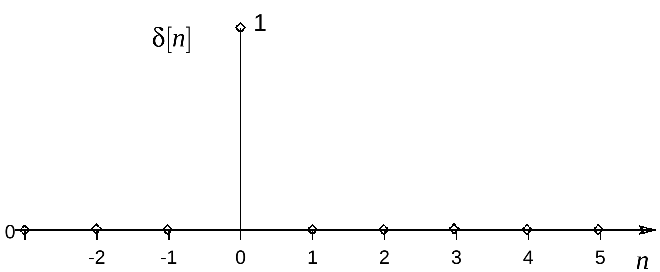
$\longrightarrow$ we do not know what is between points

a) $sin(n \cdot (1/5) \cdot \pi)$ or

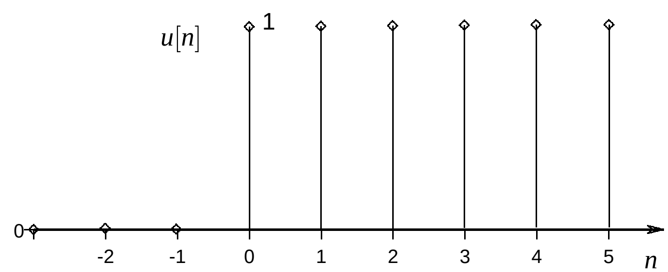
b) $sin(n \cdot (2 + 1/5) \cdot \pi)$?

Number sequence (or DT signal) operations; basic sequences

operation	notation	definition
sum	$z[n] = x[n] + y[n]$	$\forall n \;\; z(n) = x(n) + y(n)$
scale	$z[n] = \alpha \cdot y[n]$	$\forall n \;\; z(n) = \alpha \cdot y(n)$
shift	$z[n] = x[n - n_0]$	$\forall n \;\; z(n) = x(n - n_0)$
difference	$z[n] = x[n] - y[n]$	$\forall n \;\; z(n) = x(n) - y(n)$
product	$z[n] = x[n] \cdot y[n]$	$\forall n \;\; z(n) = x(n) \cdot y(n)$



Unit sample sequence (DT impulse)



Unit step sequence

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

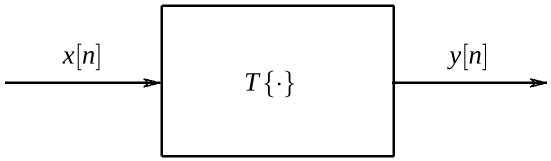
$$u[n] = \sum_{k=0}^{\infty} \delta[n-k]$$

DT systems

A DT system: an operator mapping an input sequence $x[n]$ into an output sequence $y[n]$.

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

$\longrightarrow$ A rule (formula) for computing output sequence values $y(n)$ from the input sequence values $x(n)$.



Implementations:

- PC program
- matlab m-file
- custom VLSI or FPGA
- programmable digital signal processor

Examples:

$$y(n) = 3 \cdot x(n)$$

$$y(n) = \frac{x(n) + x(n-1)}{2}$$

$$y(n) = \frac{1}{M} \sum_{k=0}^{M-1} x(n-k)$$

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

Linear & time-invariant DT systems

Linearity property

$$T\{\alpha_1 x_1[n] + \alpha_2 x_2[n]\} = \alpha_1 T\{x_1[n]\} + \alpha_2 T\{x_2[n]\}$$

in other words:

if

$$x_1[n] \longrightarrow y_1[n]$$

$$x_2[n] \longrightarrow y_2[n]$$

then

$$\alpha x_1[n] \longrightarrow \alpha y_1[n]$$ (scaling, homogeneity)

$$x_1[n] + x_2[n] \longrightarrow y_1[n] + y_2[n]$$ (additivity)

Time invariance (shift invariance)

If

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

then

$$\forall n_0, \quad T\{x[n-n_0]\} = y[n-n_0]$$

Linear systems - examples

- $y(n) = 3 \cdot x(n)$ – is linear; it is also *memoryless*
- $y(n) = \frac{x(n)+x(n-1)}{2}$ (not memoryless):

$$T\{\alpha_1 x_1(n) + \alpha_2 x_2(n)\} = \frac{[\alpha_1 x_1(n) + \alpha_2 x_2(n)] + [\alpha_1 x_1(n-1) + \alpha_2 x_2(n-1)]}{2} =$$

$$= \frac{\alpha_1 x_1(n) + \alpha_1 x_1(n-1)}{2} + \frac{\alpha_2 x_2(n) + \alpha_2 x_2(n-1)}{2} =$$

$$= \alpha_1 \frac{x_1(n) + x_1(n-1)}{2} + \alpha_2 \frac{x_2(n) + x_2(n-1)}{2} = \alpha_1 y_1(n) + \alpha_2 y_2(n) \text{ cnd}$$

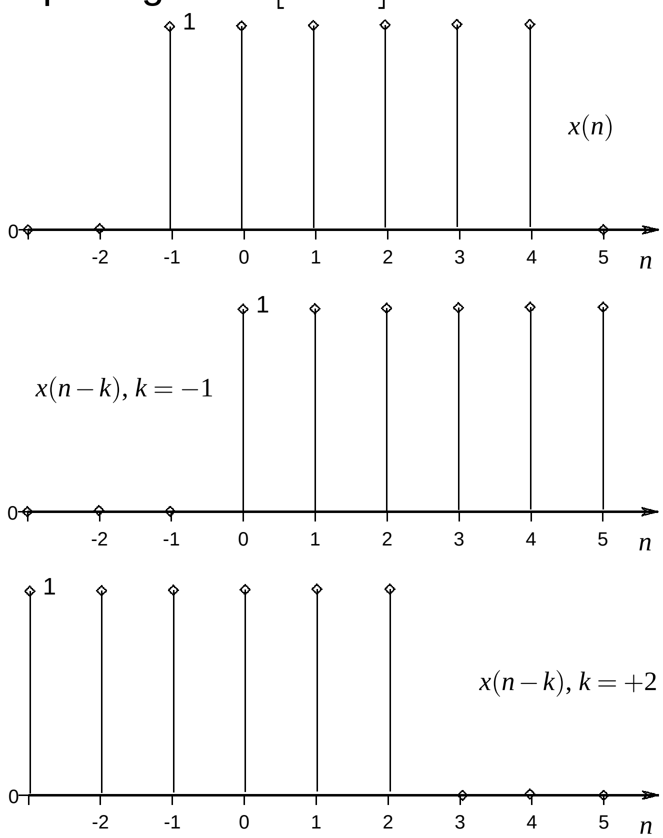
(not L) $y(n) = (x(n))^2$ because

$$T\{x_1(n) + x_2(n)\} = (x_1(n) + x_2(n))^2 = (x_1(n))^2 + (x_2(n))^2 + [2 \cdot x_1(n) x_2(n)]$$

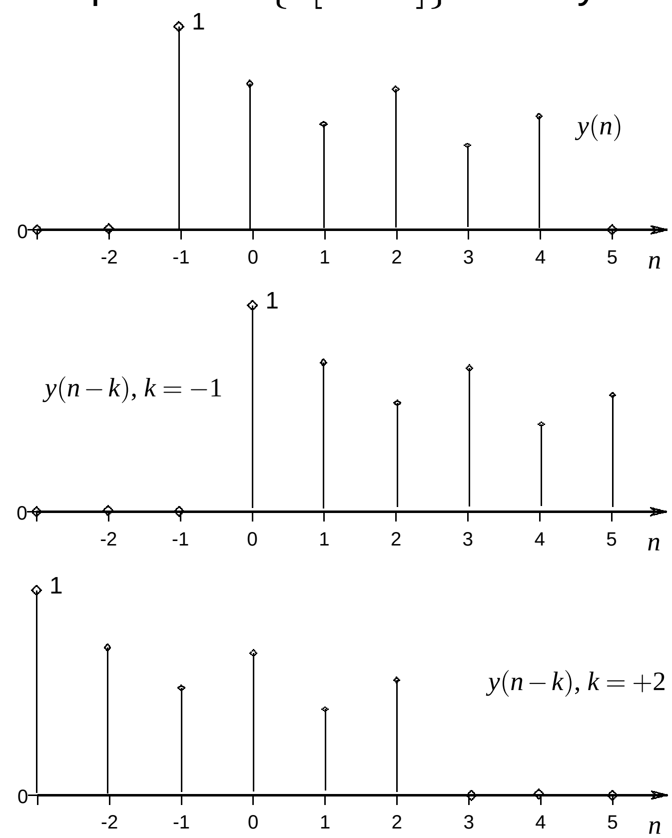
---

shift example

Input signals $x[n-k]$.



Responses $T\{x[n-k]\}$ of TI system $T\{.\}$



Other properties: causality, stability

causality

$\longrightarrow y(n_0)$ depends only on $x(n),\ n \leq n_0$ (*usually less important in DT implementations)*

stability

$\longrightarrow$ bounded input causes bounded output [BIBO]

bounded $\longrightarrow \exists B_x : \ \ \forall n \ \ |x(n)| \leq B_x < \infty$

Examples

Decimator (compressor)

$$y(n) = x(Mn)$$

$\longrightarrow$ L, but not TI (*prove it!*)

1-st order difference

forward: $y(n) = x(n+1) - x(n) \longrightarrow$ noncausal

backward: $y(n) = x(n) - x(n-1) \longrightarrow$ causal

Accumulator

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

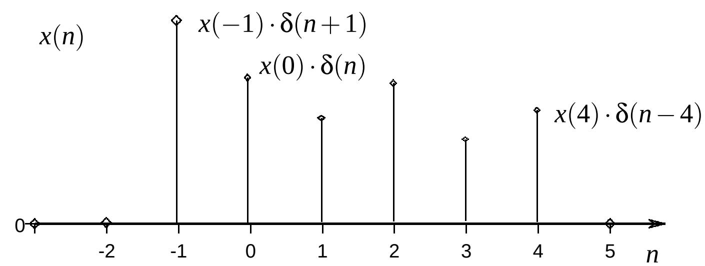
$\longrightarrow$ unstable; (*hint: feed it with* $u[n]$)

LTI systems: impulse response

$h[n] = T\{\delta[n]\} \longrightarrow$ impulse response of $T\{.\}$
$h[n]$ characterizes completely system $T\{.\}$ – we may compute its response for any input $x[n]$.

- Decompose $x[n]$ into weighted sum of impulses $\delta[n-k]$

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



- Superpose responses (use LTI properties)

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

$\longrightarrow$ this is a **convolution sum**

Convolution example

Convolution properties

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

we denote as

$$y[n] = x[n] * h[n]$$

Properties of "$*$"

"$*$" is commutative: $x[n] * h[n] = h[n] * x[n]$

"$*$" distributes over addition $x[n] * (h_1[n] + h_2[n]) = x[n] * h_1[n] + x[n] * (h_2[n])$

System and $h[n]$

- causality $\Leftrightarrow h[n] = 0, \ \ n < 0$. A hint: $y(n) = \sum_{k=-\infty}^{\infty} h(k)x(n-k)$
- stability $\Leftrightarrow S = \sum_{k=-\infty}^{\infty} |h(k)| < \infty$

Linear difference equations

. . . describe an important class of LTI systems.

$$\sum_{k=0}^{N} a_k y(n-k) = \sum_{k=0}^{M} b_k x(n-k), \quad a_0 = 1 \text{ (traditionally)}$$

or

$$\begin{aligned} y(n) &= a_1 \cdot y(n-1) + a_2 \cdot y(n-2) + \ldots + a_n \cdot y(n-N) + \\ &+b_0 \cdot x(n) \quad +b_1 \cdot x(n-1) + b_2 \cdot x(n-2) + \ldots + b_n \cdot x(n-M) \end{aligned}$$

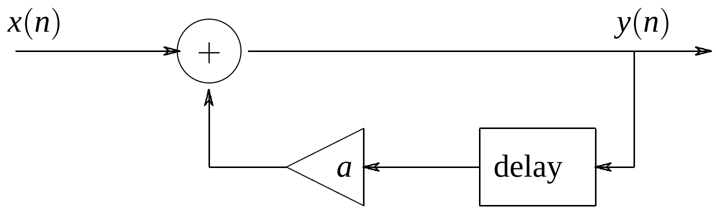
Note: if, for a given input $x_p[n]$, an output sequence $y_p[n]$ satisfies given differential equation,

$$y[n] = y_p[n] + y_h[n]$$

will also satisfy the equation, if $y_h[n]$ is a solution to $\sum_{k=0}^{N} a_k y(n-k) = 0$ (homogenous equation).

Differential equation – example

An equation: $y(n) = a \cdot y(n-1) + x(n)$
with input
$x(n) = 0,\ n < 0$
$x(n) \neq 0,\ n > 0$.



$$
\begin{aligned}
y(0) &= a \cdot \mathbf{y}(-\mathbf{1}) + x(0) \\
y(1) &= a \cdot y(0) + x(1) \\
y(2) &= a \cdot y(1) + x(2) \\
&\dots
\end{aligned}
$$

Initial condition: $y(-1) = \alpha$
Let $x[n] = \delta[n]$

$$
\begin{aligned}
y(0) &= a \cdot \alpha + 1 \\
y(1) &= a(a \cdot \alpha + 1) = a^2\alpha + a \\
y(2) &= a^3\alpha + a^2 \\
&\dots \\
y(n) &= a^{n+1}\alpha + a^n
\end{aligned}
$$

example – continued