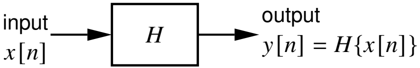
5. Discrete-time systems

- DT systems
- classifications of DT systems
- recursive solution of difference equations
- continuous to discrete signal processing

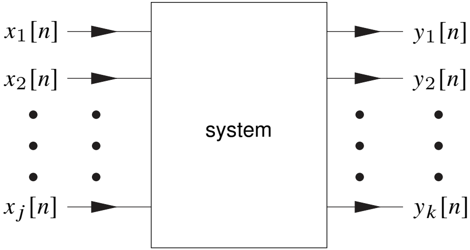
Discrete-time system

a *discrete-time system* is a system whose inputs and outputs are DT signals

Single-input single-output



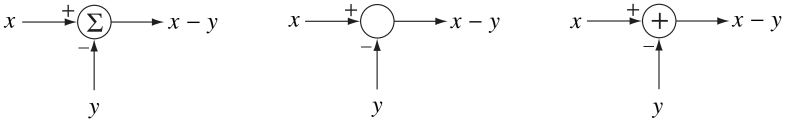
Multi-input multi-output



Block diagrams operations

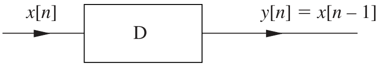
Amplifier (scalar multiplication)

Summation (addition)



(a) (b) (c)

Delay



Example 5.1 (saving accounts)

- a person makes a deposit in a bank regularly every $T = 1$ month
- the bank pays a interest rate r on the account balance during the period T

find the equation relating the output $y[n]$ (the balance) to the input $x[n]$ (the deposit)

Solution:

- the balance $y[n]$ is the sum of the previous balance $y[n-1]$, the interest on $y[n-1]$ during the period T, and the deposit $x[n]$:

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

or

$$y[n] - ay[n-1] = x[n], \quad a = 1 + r$$

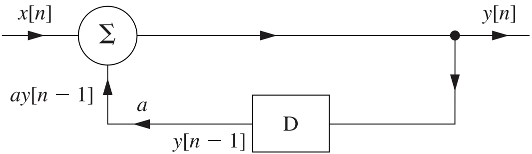
the equation

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

or

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

can be represented as:

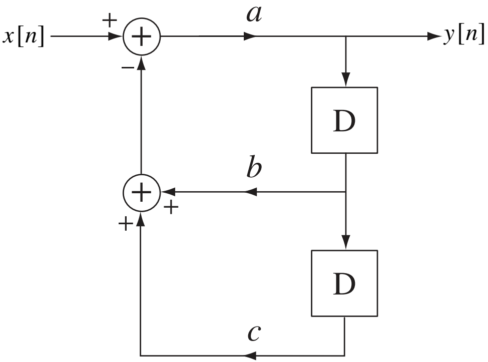


note that so we can replace n by $n+1$ to obtain the advance-form:

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

Example 5.2

find the input-output relation of the system described shown below



Solution:

$$y[n] = a(x[n] - by[n-1] - cy[n-2])$$

Example 5.3 (sales estimate)

- during semester n, $x[n]$ students enroll in a course requiring a textbook
- the publisher sells $y[n]$ new copies of the same book
- one-quarter of students with books in salable condition resell the texts at the end of the semester, and the book life is three semesters
- find the equation relating the new books sold , $y[n]$, to the number of students enrolled in the nth semester, $x[n]$, assuming that every student buys a book

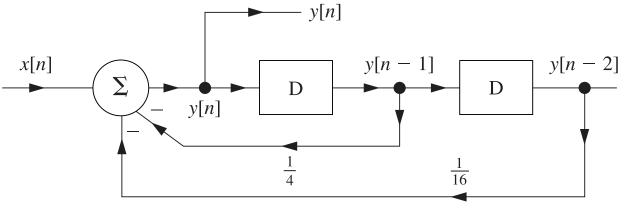
Solution: in the nth semester,

- the number of new sold books $y[n]$ and old resold books must be equal to number of students $x[n]$
- there are $y[n-1]$ new books sold in semester $(n-1)$, and $(1/4)y[n-1]$ books will be resold in the nth semester
- also, $y[n-2]$ new books are sold in semester $n-2$, and $(1/4)y[n-2]$ books will be resold in semester $(n-1)$; again, a quarter of these, that is, $(1/16)y[n-2]$, will be resold in the nth semester

hence, $x[n]$ is equal to:

$$y[n] + \frac{1}{4}y[n-1] + \frac{1}{16}y[n-2] = x[n]$$

the above equation is in delay-form; the block-diagram representation of the above equation is



in advance form, we can replace n by $n+2$ to obtain

$$y[n+2] + \frac{1}{4}y[n+1] + \frac{1}{16}y[n] = x[n+1]$$

Outline

- DT systems
- **classifications of DT systems**
- recursive solution of difference equations
- continuous to discrete signal processing

Linear systems

a system is

- *homogeneous* if $x \to y$, then $ax \to ay$ for any number a
- *additive* if for $x_1 \to y_1$ and $x_2 \to y_2$, we have $x_1 + x_2 \to y_1 + y_2$

a system is **linear** if it satisfies the *superposition* property:

$$x_1 \longrightarrow y_1, \quad x_2 \longrightarrow y_2$$

then for any numbers a_1, a_2:

$$a_1 x_1 + a_2 x_2 \longrightarrow a_1 y_1 + a_2 y_2$$

in other words, the system is both homogeneous and additive

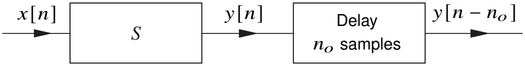
(a system that does not satisfy either the homogeneity or additivity property is *nonlinear*)

Time-invariant systems

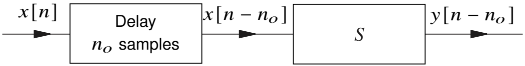
a system S is *time-invariant* (*shift invariance*) if

$$x[n] \rightarrow y[n] \quad \text{implies that} \quad x[n-n_o] \rightarrow y[n-n_o]$$

for any integer n_o (assuming initial conditions are also delayed by n_o)



(a)



(b)

- a system is *time-varying* if the the above does not hold
- *linear time-invariant discrete system* (LTID) is a DT system that is both linear and time-invariant

Example 5.4

determine whether the system described by $y[n] = e^{-n}x[n]$ is

(a) linear or non-linear

(b) time-invariant or time-varying

Solution:

(a) for inputs $x_1[n]$ and $x_2[n]$, the outputs are $y_1[n] = e^{-n}x_1[n]$ and $y_2[n] = e^{-n}x_2[n]$; for input $x[n] = a_1x_1[n] + a_2x_2[n]$, the output is

$$e^{-n}[a_1x_1[n] + a_2x_2[n]] = a_1e^{-n}x_1[n] + a_2e^{-n}x_2[n] = a_1y_1[n] + a_2y_2[n]$$

hence the system satisfies the superposition property, hence linear

(b) the input $x_1[n]$ gives output $y_1[n] = e^{-n}x[n]$ and the input $x_2[n] = x_1[n-n_0]$ gives output

$$y_2[n] = e^{-n}x_2[n] = e^{-n}x_1[n-n_0] \neq y_1[n-n_0]$$

hence, the system is time-varying

Causal and static systems

Causal systems: a *causal* (or *physical* or *nonanticipative*) system is one where the output at $n = k$ depends only on the input $x[n]$ for $n \leq k$

- the output at the present instant depends only on the past and present values of the input
- output does not depends on future inputs
- a system that violates the condition of causality is called a *noncausal* (or *anticipative*) system

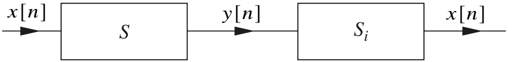
Static (memoryless) systems: a system is *static* (*instantaneous*) or *memoryless* if the output at any instant n depends input at the same time n only; otherwise, the system is *dynamic* or *with memory*

Examples

- $y[n] = \sin(x[n])$ is a causal and static (memoryless) system
- the system $y[n] = x[n] + y[n-1]$ is a causal and dynamic system (with memory)

Invertible systems

a discrete-time system S is *invertible* if we can recover the input $x[n]$ from the output $y[n]$ by another system (called the *inverse system*)



- for an invertible system, every input have a unique output
- example: $y[n] = x[n+k]$ is invertible since $x[n] = y[n-k]$

Noninvertible system: a system is *noninvertible* when it is impossible to obtain the input from the output

- example: $y[n] = x[Mn]$ loses all but every Mth sample of the input, hence, noninvertible
- systems where two different inputs give same output are noninverible
  - $y[n] = \cos(x[n])$
  - $y[n] = |x[n]|$

BIBO stable

a system is *bounded-input-bounded-output (BIBO) stable* (externally stable) if every bounded input applied at the input terminal results in a bounded output

Examples:

- $y[n] = x[-n]$ is BIBO stable
- $y[n] = e^{-n}x[n]u[n]$ is BIBO stable
- $y[n] = nx[n]$ is BIBO unstable
- $y[n] = e^{-n}x[n]$ is BIBO unstable

Example 5.5

consider a DT system described as $y[n] = x[n]x[n-1]$; determine whether the system is

(a) linear

(b) time-invariant

(c) causal

(d) memoryless (static)

(e) invertible

(f) BIBO-stable

Solution: suppose $x[n] \Rightarrow y[n]$

(a) we have

$$ax[n] \Rightarrow a^2x[n]x[n-1] = a^2y[n] \neq ay[n],$$

hence, system is nonlinear since it does not satisfy the homogeneity property

(b) we have

$$x[n-n_0] \Rightarrow x[n-n_0]x[n-1-n_0] = x[n-n_0]x[(n-n_0)-1] = y[n-n_0]$$

hence, the system is time-invariant

(c) causal because the current output does not depend on future input values

(d) the output y at time n depends on the past values of input x (at $n-1$); hence not memoryless (has memory, dynamic)

(e) since two different inputs $x_1[n] = 1$ and $x_2[n] = -1$ give same output $y_1[n] = y_2[n] = 1$, the system is noninvertible

(f) for $|x[n]| \leq M_x < \infty$, we have $|y[n]| \leq M_x^2 < \infty$; hence, system is BIBO-stable

Exercises

- show that system described by the first-order difference equation $y[n+1] + 2y[n] = x[n+1] - x[n]$ is linear and time-invariant
- determine whether the following systems are static or dynamic; which of the systems are causal?

  (a) $y[n+1] = x[n]$

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

  (c) $y[n+1] = x[n+1]$

  (d) $y[n-1] = x[n]$

- show that a system specified by equation $y[n] = ax[n] + b$ is invertible
- show that the system $y[n] = nx[n]$ is linear, time varying, and non-invertible
- show that the system $y[n] = x[-n]$ is linear, time variant, dynamic, and noncausal
- show that the system $y[n] = |x[n]|^2$ is not linear, time invariant, and noninvertible
- explain why the continuous-time system $y(t) = x(2t)$ is always invertible and yet the corresponding discrete-time system $y[n] = x[2n]$ is not invertible

Outline

- DT systems
- classifications of DT systems
- **recursive solution of difference equations**
- continuous to discrete signal processing

Linear difference equation

Advance-form

$$y[n+N] + a_1 y[n+N-1] + \cdots + a_{N-1} y[n+1] + a_N y[n] =$$
$$b_0 x[n+M] + b_1 x[n+M-1] + \cdots + b_{M-1} x[n+1] + b_M x[n]$$

- order is N
- system is linear
- system is time-invariant if coefficients a_i, b_i are constants (independent of n)
- the system is causal if $M \leq N$
- many systems can be modeled as linear difference systems

Causal advance-form: for $M = N$, we get the Nth order causal advance-form

$$y[n+N] + a_1 y[n+N-1] + \cdots + a_{N-1} y[n+1] + a_N y[n] =$$
$$b_0 x[n+N] + b_1 x[n+N-1] + \cdots + b_{N-1} x[n+1] + b_N x[n]$$

Causal delay-form: if we replace n by $n - N$ throughout the equation, we get the delay-form:

$$y[n] + a_1 y[n-1] + \cdots + a_{N-1} y[n-N+1] + a_N y[n-N] =$$
$$b_0 x[n] + b_1 x[n-1] + \cdots + b_{N-1} x[n-N+1] + b_N x[n-N]$$

- delay form is more natural because delay operation is causal, hence realizable
- advance form is more mathematically convenience compared to delay form

Recursive (iterative) solution

the equation

$$\begin{aligned} & y[n] + a_1 y[n-1] + \cdots + a_{N-1} y[n-N+1] + a_N y[n-N] \\ & = b_0 x[n] + b_1 x[n-1] + \cdots + b_{N-1} x[n-N+1] + b_N x[n-N] \end{aligned}$$

can be expressed in *recursive form*:

$$\begin{aligned} y[n] = & -a_1 y[n-1] - a_2 y[n-2] - \cdots - a_N y[n-N] \\ & + b_0 x[n] + b_1 x[n-1] + \cdots + b_N x[n-N] \end{aligned}$$

- to find $y[0]$, we need to know the N initial conditions $y[-1], y[-2], \ldots, y[-N]$
- to find $y[1]$, we need to know the N initial conditions $y[0], y[-1], \ldots, y[-N+1]$ ...etc
- knowing the N initial conditions and the input, we can determine recursively the entire output $y[0], y[1], y[2], y[3], \ldots$, one value at a time

Example 5.6

solve iteratively (recursively):

$$y[n] - 0.5y[n-1] = x[n]$$

given $y[-1] = 16$ and causal input $x[n] = n^2 u[n]$

Solution: the equation can be expressed as

$$y[n] = 0.5y[n-1] + x[n]$$

if we set $n = 0$, we obtain

$$\begin{aligned}
y[0] &= 0.5y[-1] + x[0] = 0.5(16) + 0 = 8 \\
y[1] &= 0.5(8) + (1)^2 = 5 \\
y[2] &= 0.5(5) + (2)^2 = 6.5 \\
y[3] &= 0.5(6.5) + (3)^2 = 12.25 \\
y[4] &= 0.5(12.25) + (4)^2 = 22.125 \\
&\vdots
\end{aligned}$$

Example 5.7

solve iteratively

$$y[n+2] - y[n+1] + 0.24y[n] = x[n+2] - 2x[n+1]$$

with initial conditions $y[-1] = 2, y[-2] = 1$ and a causal input $x[n] = nu[n]$

Solution: the system equation can be expressed as

$$y[n+2] = y[n+1] - 0.24y[n] + x[n+2] - 2x[n+1]$$

hence

$$y[0] = y[-1] - 0.24y[-2] + x[0] - 2x[-1] = 2 - 0.24(1) + 0 - 0 = 1.76$$
$$y[1] = y[0] - 0.24y[-1] + x[1] - 2x[0] = 1.76 - 0.24(2) + 1 - 0 = 2.28$$
$$y[2] = y[1] - 0.24y[0] + x[2] - 2x[1] = 2.28 - 0.24(1.76) + 2 - 2(1) = 1.8576$$
$$\vdots$$

Matlab example

we can use Matlab to recursively solve difference equations

- the previews example, can be solved via Matlab code:

```
n = (-2:4); x = n.*(n>=0);
y = zeros(size(n)); y(n<0) = [1,2];
for ind = find(n>=0),
y(ind) = y(ind-1)-0.24*y(ind-2)+x(ind)-2*x(ind-1);
end
y(n>=0)
```

  [output: `y = 1.76 2.28 1.86 0.31 -2.14`]

- in the example on page 5.4, we can determine the money earned by investing $\$100$ monthly at 0.5% interest per month for 100 months:

```
r = 0.005; a1 = -(1+r); y(1) = 100;
for n = 2:100, y(n) = -a1*y(n-1)+100; end
y(100)-100*100
```

  [output: `ans = 2933.37`]

Exercises

- using the iterative method, find the first three terms of $y[n]$ for

$$y[n+1] - 2y[n] = x[n]$$

  the initial condition is $y[-1] = 10$ and the input $x[n] = 2$ starting at $n = 0$
  [**Answer:** $y[0] = 20$, $y[1] = 42$, and $y[2] = 86$]

- iteratively solve

$$y[n] - 0.5y[n-1] - x[n] - 0.5x[n-1]$$

  with $y[-1] = 6$ and $x[n] = \cos(n\pi/2)u[n]$; compute the first six values

- *Nonrecursive difference equation:* consider an input $x[n] = \cos(\pi n/2)u[n]$ and a system represented as

$$y[n] = (1/5)(x[n+2] + x[n+1] + x[n] + x[n] + x[n-1] + x[n-2])$$

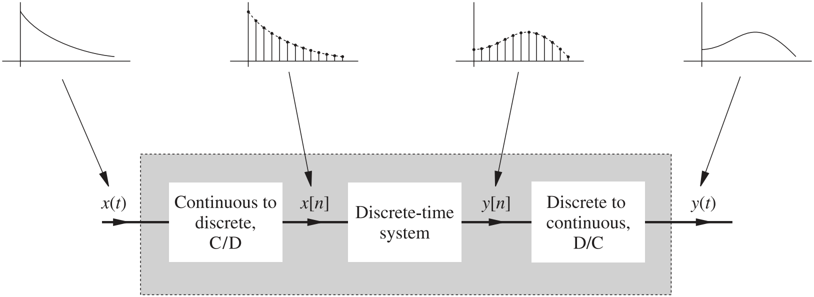
  determine the output $y[n]$ at times $n = 0$ and $n = 1234$

Outline

- DT systems
- classifications of DT systems
- recursive solution of difference equations
- **continuous to discrete signal processing**

Continuous to discrete C/D and discrete to continuous D/C

- DT system can be used to process a continuous-time signal
- DT signals are more efficient to send, receive, and store information
- DT systems are easier to handle and manipulate compared to CT systems

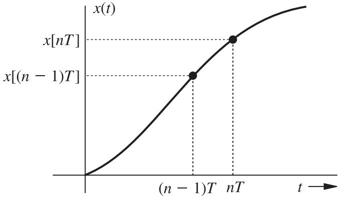


Example: digital differentiator

a *digital differentiator* is a system that differentiates a continuous-time signals by discrete-time processing

- output $y(t)$ is derivative of input $x(t)$: $y(t) = \frac{dx(t)}{dt}$
- let $x[n] = x(nT)$ and $y[n] = y(nT)$

$$y(nT) = \left.\frac{dx(t)}{dt}\right|_{t=nT}$$
$$= \lim_{T\to 0} \frac{1}{T}[x(nT) - x[(n-1)T]]$$

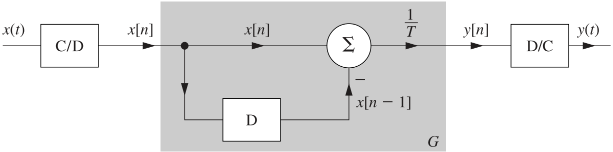


hence

$$y[n] = \lim_{T\to 0} \frac{1}{T}(x[n] - x[n-1])$$

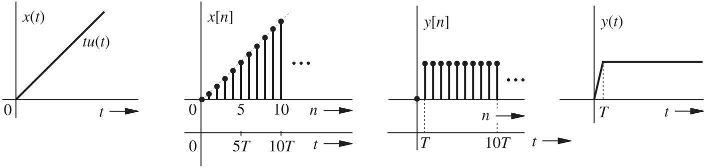
assuming $T > 0$ to be sufficiently small, we can approximate the differentiator by the *backward difference* system:

$$y[n] = \frac{1}{T}(x[n] - x[n-1])$$



for example, on ramp function $x[n] = nTu[n]$, we get

$$y[n] = \frac{1}{T}(x[n] - x[n-1]) = u[n-1]$$



as T approaches zero, $y(t)$ approaches the desired output $u(t)$

Example: digital integrator

suppose we want to compute $y(t) = \int_{-\infty}^{t} x(\tau)d\tau$ using discrete-time systems; at $t = nT$,

$$y(nT) = \lim_{T\to 0} \sum_{k=-\infty}^{n} x(kT)T \quad \text{or} \quad y[n] = \lim_{T\to 0} T \sum_{k=-\infty}^{n} x[k]$$

assuming T is small enough, we get the approximation

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

- the above is an example of *accumulator system*
- digital integrator equation can be expressed in the *recursive form*:

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

Differentiation to difference equation

for small enough T, we can approximate

$$\left.\frac{dy(t)}{dt}\right|_{t=nT} \approx \frac{y[n]-y[n-1]}{T}$$

$$\begin{aligned}\left.\frac{d^2y(t)}{dt^2}\right|_{t=nT} &= \lim_{T\to 0}\frac{1}{T}\left(\left.\frac{d}{dt}y(t)\right|_{t=nT} - \left.\frac{d}{dt}y(t)\right|_{t=(n-1)T}\right)\\ &\approx \frac{1}{T}\left(\frac{y[n]-y[n-1]}{T} - \frac{y[n-1]-y[n-2]}{T}\right)\\ &= \frac{1}{T^2}(y[n]-2y[n-1]+y[n-2])\end{aligned}$$

similarly, we can show that a Kth-order derivative can be approximated by

$$\left.\frac{d^K}{dt^K}y(t)\right|_{t=nT} \approx \frac{1}{T^K}\sum_{k=0}^{K}(-1)^k \binom{K}{k} y[n-k]$$

Example: differential equation to difference equation

$$\frac{dy(t)}{dt} + cy(t) = x(t)$$

from the definition of a derivative, we can express the above at $t = nT$ as

$$\lim_{T \to 0} \frac{y[n] - y[n-1]}{T} + cy[n] = x[n]$$

assuming very small $T \neq 0$, we can approximate the above as:

$$y[n] + \alpha y[n-1] = \beta x[n]$$

where

$$\alpha = \frac{-1}{1 + cT}, \qquad \beta = \frac{T}{1 + cT}$$

(a computer solves differential equations by solving an equivalent difference equation; hence, it is important to know how to solve difference systems)

Example 5.8

assuming a sampling interval $T = 0.1$, determine a difference equation model for the differential equation

$$\ddot{y}(t) + 4\dot{y}(t) + 3y(t) = 100x(t)$$

with initial conditions $y(0) = 0$ and $\dot{y}(0) = 10$

Solution: the differential equation is approximated as

$$\frac{1}{T^2}(y[n] - 2y[n-1] + y[n-2]) + \frac{4}{T}(y[n] - y[n-1]) + 3y[n] = 100x[n]$$

combining terms and substituting $T = 0.1$ yield

$$143y[n] - 240y[n-1] + 100y[n-2] = 100x[n]$$

to compute the equivalent initial conditions, we note that $y[0] = y(0) = 0$; further,

$$10 = \dot{y}(0) = \frac{d}{dt}y(t)\bigg|_{t=0} \approx \frac{y(0) - y(0-T)}{T} = \frac{y[0] - y[-1]}{0.1} = -10y[-1]$$

assuming that T is sufficiently small, this leads to $y[-1] = -1$; following normalization of the coefficient for $y[n]$, the differential equation is therefore modeled as

$$y[n] - \frac{240}{143}y[n-1] + \frac{100}{143}y[n-2] = \frac{100}{143}x[n]$$

with initial conditions $y[0] = 0$ and $y[-1] = -1$

Example: estimation position from video

- a car is filmed using a camera operating at 60 frames per second
- let n designate the film frame, where $n = 0$ corresponds to engine ignition
- by analyzing each frame of the film, we can determine the car position $x[n]$, measured in meters, from the original starting position $x[0] = 0$
- from physics, we know that velocity is $v(t) = \frac{d}{dt}x(t)$
- furthermore, we know that acceleration is $a(t) = \frac{d}{dt}v(t)$
- we can estimate the car velocity from the film data by using a simple difference equation $v[n] = 60(x[n] - x[n-1])$
- we can now estimate the car acceleration from the film data by using $a[n] = 60(v[n] - x[v-1])$

- combining last two equations, we get

$$a[n] = 60(60(x[n] - x[n-1]) - 60(x[n-1] - x(n-2]))$$

or

$$a[n] = 3600(x[n] - 2x[n-1] + x[n-2])$$

- this estimate of acceleration has two primary advantages;
  - it is simple to calculate
  - it is a causal, stable, LTI system (easy to analyze)

Digital signal processing (DSP)

advantages of DSP

- digital systems are less sensitive to changes in signal values, thus less sensitive changes in the component parameter values due to temperature variation, aging, and other factors
- digital systems are extremely flexible and easy to implement; digital filter function is easily altered by simply changing the program
- even in the presence of noise, reproduction with digital messages is extremely reliable, often without any deterioration; further, digital signals can be coded to yield extremely low error rates, high fidelity, error correction capabilities, and privacy
- digital signals can be coded to yield extremely low error rates and high fidelity, as well as privacy; also, more sophisticated signal-processing algorithms can be used to process digital signals

- digital filters can be easily time-shared and therefore can serve a number of inputs simultaneously; moreover, it is easier and more efficient to multiplex several digital signals on the same channel
- reproduction with digital messages is extremely reliable without deterioration; analog messages such as photocopies and films, for example, lose quality at each successive stage of reproduction and have to be transported physically from one distant place to another, often at relatively high cost

disadvantages of DSP

- increased system complexity due to use of A/D and D/A interfaces,
- limited range of frequencies available in practice (affordable rates are gigahertz or less)
- use of more power than is needed for the passive analog circuits

Exercise

find a causal difference equation and initial conditions that approximates the behavior of the second-order differential equation

$$\frac{d^2y(t)}{dt^2} + 3\frac{dy(t)}{dt} + 2y(t) = 1000x(t)$$

with initial conditions $y(0) = 0$ and $\dot{y}(0) = 3$; use the sampling interval $T = 0.05$

References

- B.P. Lathi, *Linear Systems and Signals*, Oxford University Press, chapter 3 (3.1–3.5).
- M. J. Roberts, *Signals and Systems: Analysis Using Transform Methods and MATLAB*, McGraw Hill, chapter 4 (4.3).