ECE 317

Digital Signal Processing I (DSP I)

Lecture 4

Prof. Mojtaba Soltanalian

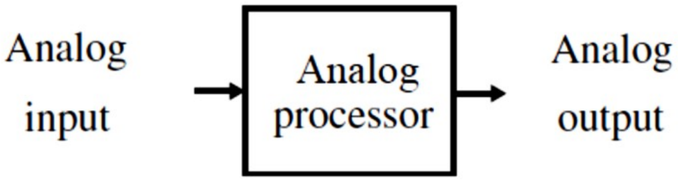
* Based on Slides from Prof. Rashid Ansari.

Topics of last and today's lectures

- **Last class:**
  - DT signal representation
  - Basic DT signals
  - Complex numbers
- **Today's class:**
  - Two paradigms for signal processing
  - CT vs DT signal processing

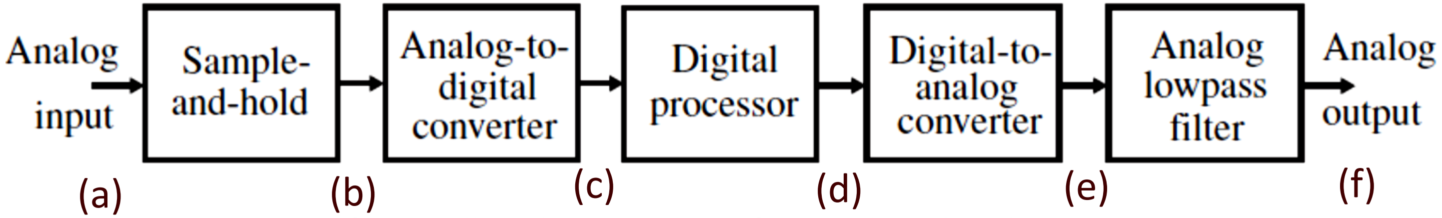
Two paradigms for signal processing

- Analog (CT) signal processing paradigm



Scheme for the analog processing of an analog signal.

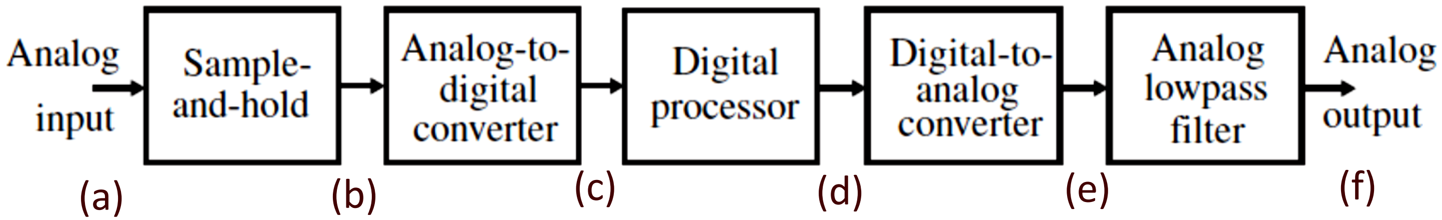
- Digital (DT) signal processing paradigm



Scheme for the digital processing of an analog signal.

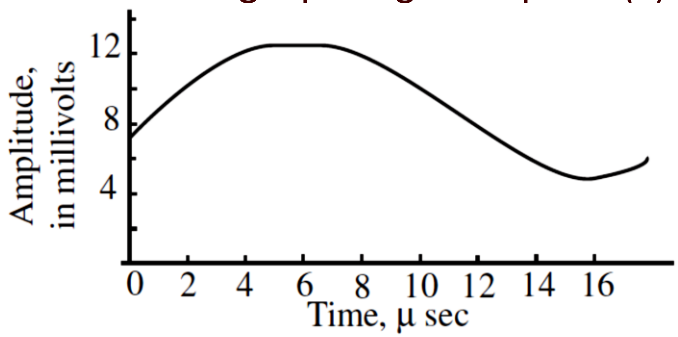
Let us examine the signals at points (a)-(f)

Look at the analog input to S&H unit and corresponding output



Scheme for the digital processing of an analog signal.

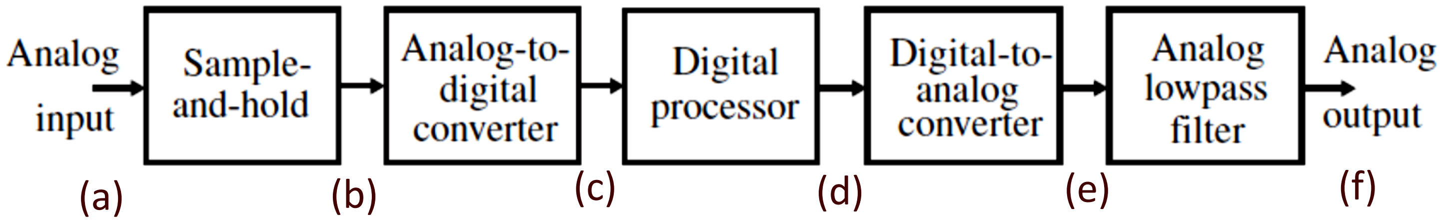
Analog input signal at point (a)



(a)

What is the domain of the analog input signal at (a)? What is the co-domain of the signal?

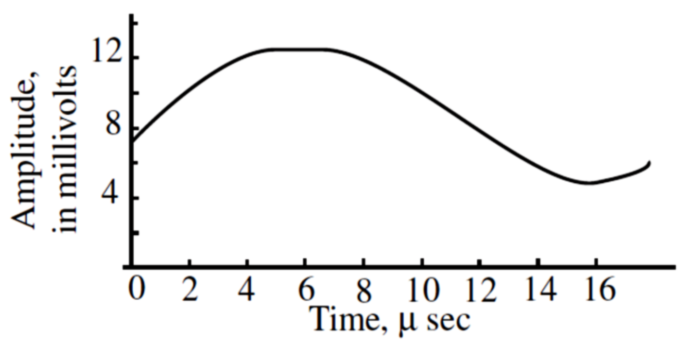
Given the analog input to S&H unit, examine the corresponding output



Scheme for the digital processing of an analog signal.

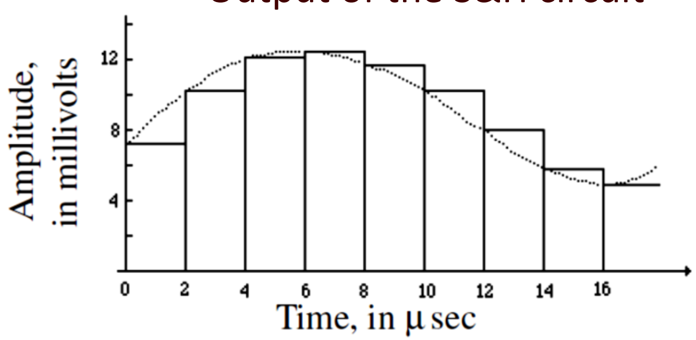
Let us sample the analog input signal (a) at $f_s = 500$ kHz and hold the value until the next sampling instant. What is the sampling period T?

Analog input signal



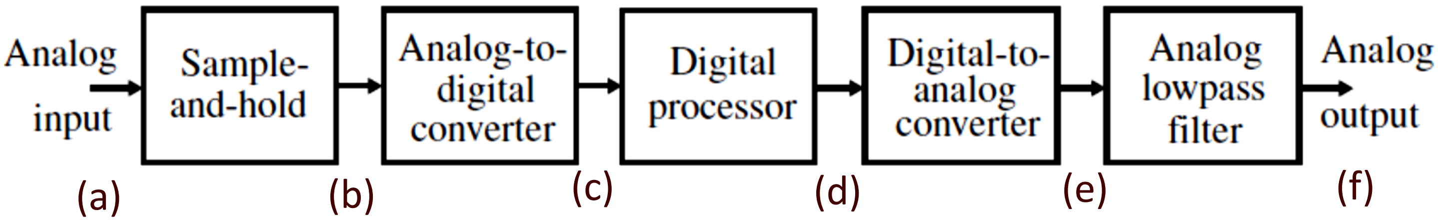
(a)

Output of the S&H circuit



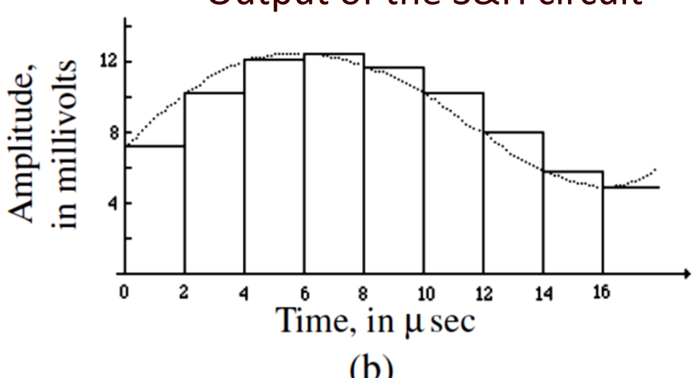
(b)

What are the possible values of the output of the S&H circuit?



Scheme for the digital processing of an analog signal.

Output of the S&H circuit

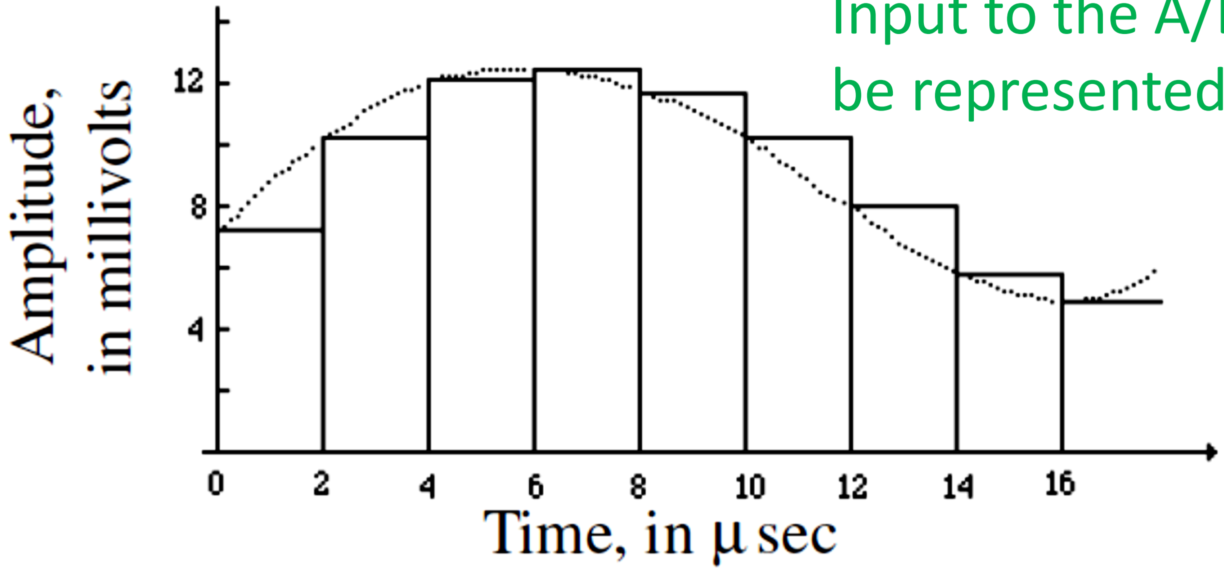


Output of the S&H circuit is the input to the A/D converter

What is the domain of the S&H output signal at (b)? What is the co-domain of the signal?

Look at A/D converter (ADC) input-output relation

Assume analog signal is non-negative (0-15 mV) and assume we have a **4-bit** A/D converter (for illustrative purposes)

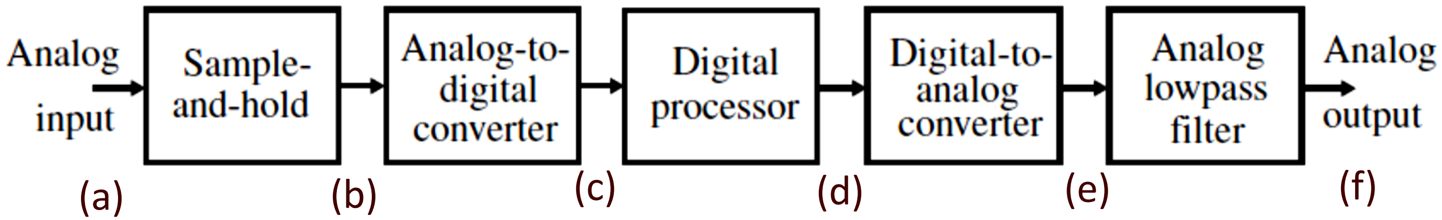


Input to the A/D converter will be represented with 4 bits.

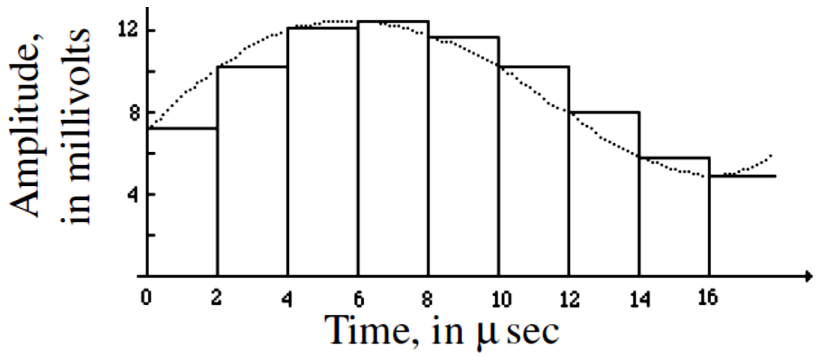
Signal amplitude at time 0 μsec? → Close to 7 mV → Maps to 0111

Signal amplitude at time 2 μsec? → Close to 10 mV → Maps to 1010

Look at A/D converter output

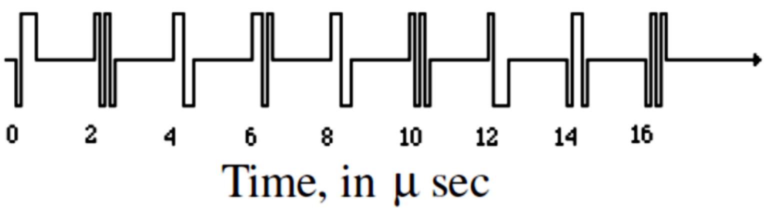


Scheme for the digital processing of an analog signal.



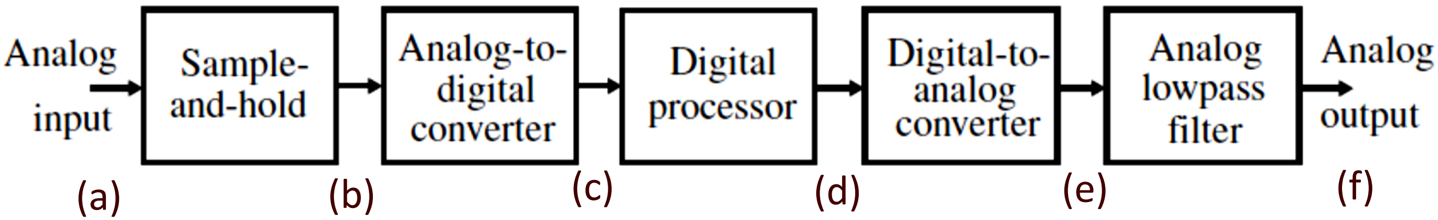
Output of the S/H circuit is input to A/D converter

0111 1010

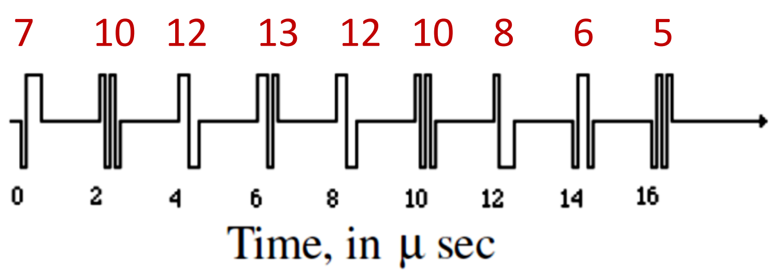


Output of the A/D Converter at point (c)

Digital signal **input** to digital signal processor



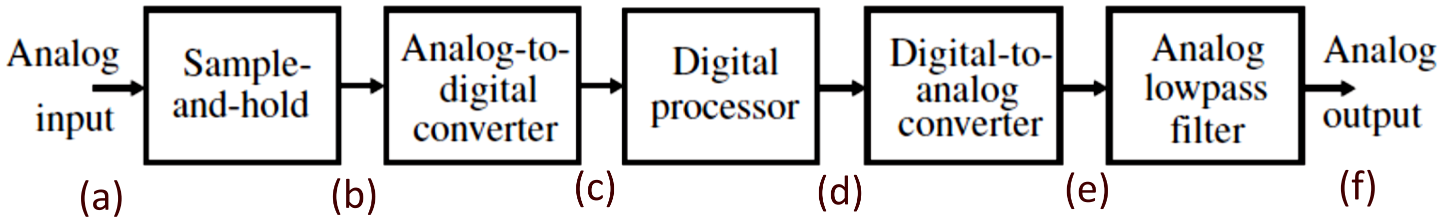
Scheme for the digital processing of an analog signal.



Output of the A/D Converter is stored in memory as a digital signal with suitably assumed time indexing

$x[n]$:	$\underline{7}$	10	12	13	12	10	8	6	5
n:	0	1	2	3	4	5	6	7	8

Illustrative digital signal output of digital signal processor



Scheme for the digital processing of an analog signal.

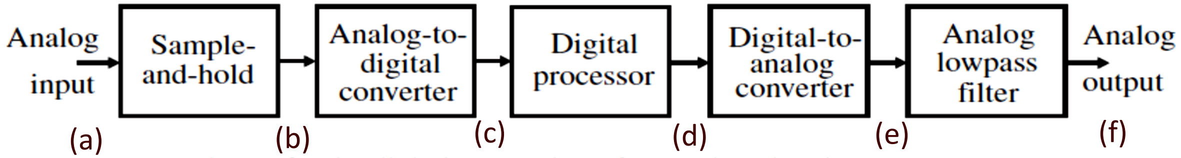
Input to digital signal processor at point (c)

$x[n]$:	$\underline{7}$	10	12	13	12	10	8	6	5
n:	0	1	2	3	4	5	6	7	8

Let us assume that the output of the digital signal processor, at point (d), is

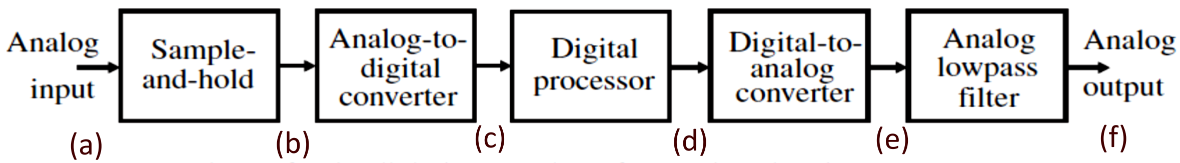
$y[n]$:	$\underline{12}$	13	9	4	5	10	14	12	11
n:	0	1	2	3	4	5	6	7	8

DSP with analog input: Signals at different points

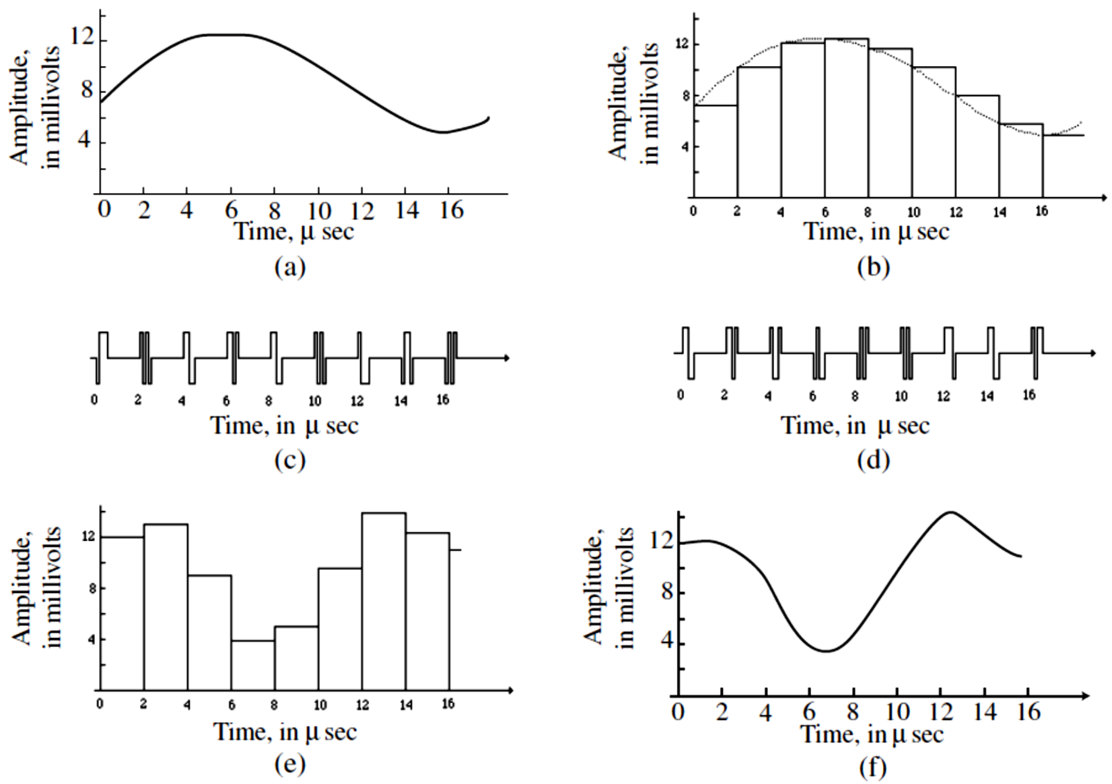


Scheme for the digital processing of an analog signal.

DSP with analog input: Signals at different points



Scheme for the digital processing of an analog signal.



CT vs DT signal processing

- DTSP advantages:
- Allows sophisticated and complex processing. (Compression, image editing, video compression)
- Simple prototyping: Simulation using high-level languages
- Zero-tolerance design: Filter coefficients are fixed and exactly reproducible.
- Adaptive processing
- Achieving exact linear phase, multirate filter banks.
- Storing analog signals is not easy and they degrade over time.

- CTSP advantages:
- Cost of A/D D/A conversion, although this is going down.
- High-frequency applications
- Low-power applications

DT systems

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

A discrete-time system is an entity that maps one or more input signals applied to it and yields one or more output signals.
The entity may be a mathematically defined transformation or an algorithm or an actual harware unit.

In the case of a single input and a single output single we will represent it as a transformation that acts on input signal $x[n]$ and yields an output $y[n]$.

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

Sequence

This does not mean that $y[n_0]$ is a function of only $x[n_0]$.
It may depend on $x[n_0 \pm 1]$, $x[n_0 \pm 2]$, $x[n_0 \pm 3]$, ...

$y[n_0]$ may be defined in terms of $y[n_0 - 1]$, $y[n_0 - 2]$,... as well as the input samples

Simple DT Systems:
Shift – a basic DT operation

Simple DT Systems: Shift – a basic DT operation

We will begin by looking at simple systems dealing with two basic operations: shifting and scaling sequences.

Shift by n_0 samples: $y[n] = x[n - n_0]$, n_0 integer

Signal is "delayed" if $n_0 > 0$, "advanced" if $n_0 < 0$.

Shift by $n_0 = 1$ is a key unit in creating structures to realize filters. Unit delay

Shifted unit impulse functions: $\delta[n], \delta[n-1], \delta[n-2], \ldots,$ $\delta[n+1], \delta[n+2], \ldots$

- $\delta[n] \quad = [\ldots 0 \;\; 0 \;\; 0 \;\; 0 \;\; \underline{1} \;\; 0 \;\; 0 \;\; 0 \;\; 0 \ldots]$
- $\delta[n-1] = [\ldots 0 \;\; 0 \;\; 0 \;\; 0 \;\; \underline{0} \;\; 1 \;\; 0 \;\; 0 \;\; 0 \ldots]$
- $\delta[n-2] = [\ldots 0 \;\; 0 \;\; 0 \;\; 0 \;\; \underline{0} \;\; 0 \;\; 1 \;\; 0 \;\; 0 \ldots]$
- $\delta[n+1] = [\ldots 0 \;\; 0 \;\; 0 \;\; 1 \;\; \underline{0} \;\; 0 \;\; 0 \;\; 0 \;\; 0 \ldots]$
- $\delta[n+2] = [\ldots 0 \;\; 0 \;\; 1 \;\; 0 \;\; \underline{0} \;\; 0 \;\; 0 \;\; 0 \;\; 0 \ldots]$

Another simple DT system: Scaling – a basic DT operation

Another simple DT system: Scaling – a basic DT operation

We will now look at the basic operation of multiplying the sequence by a scalar. For now we will consider scaling by a real number

Scaling by a real number $a \in \mathbb{R}$: $y[n] = ax[n].$

Scaling shifted unit impulse functions:

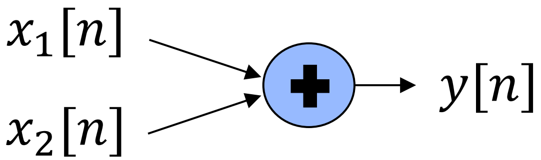
$\delta[n], \delta[n-1], \delta[n-2], \ldots, \delta[n+1], \delta[n+2], \ldots$

- $2\delta[n] = [\ldots 0 \;\; 0 \;\; 0 \;\; 0 \;\; \underline{2} \;\; 0 \;\; 0 \;\; 0 \;\; 0 \ldots]$
- $1.1\delta[n-1] = [\ldots 0 \;\; 0 \;\; 0 \;\; 0 \;\; \underline{0} \;\; 1.1 \;\; 0 \;\; 0 \;\; 0 \ldots]$
- $-\delta[n-2] = [\ldots 0 \;\; 0 \;\; 0 \;\; 0 \;\; \underline{0} \;\; 0 \;\; -1 \;\; 0 \;\; 0 \ldots]$
- $4\delta[n+1] = [\ldots 0 \;\; 0 \;\; 0 \;\; 4 \;\; \underline{0} \;\; 0 \;\; 0 \;\; 0 \;\; 0 \ldots]$
- $3\delta[n+2] = [\ldots 0 \;\; 0 \;\; 3 \;\; 0 \;\; \underline{0} \;\; 0 \;\; 0 \;\; 0 \;\; 0 \ldots]$

Example of 2-input 1-output DT system: Summation of Sequences

Example of 2-input 1-output DT system: Summation of Sequences

A new sequence can be obtained by summing sequences $x_1[n], x_2[n]$



$$y[n] = x_1[n] + x_2[n].$$

One of the signals may be a shifted version of the other.

$$y[n] = x[n] + x[n - n_0].$$

An adder is a key unit in creating structures to realize filters.

Adding sequences can be extended to multiple sequences.

This notion is used in representing a DT signal as the sum of multiple basic signals.

We will look at the representation of a general DT signal as the sum of weighted shifted impulse sequences.

Representation of DT signals using shifted impulse sequences

Representation of DT signals using shifted impulse sequences

Consider a DT signal:

$x[n] = [\ldots 0 \quad 0 \quad 3 \quad 4 \quad \underline{2} \; 1.1 \quad -1 \quad 0 \quad 0 \ldots].$

We can represent $x[n]$ as the sum of the following weighted (and shifted) impulse sequences:

- $2\delta[n] \qquad\quad = [\ldots 0 \quad 0 \quad 0 \quad 0 \quad \underline{2} \quad 0 \quad 0 \quad 0 \quad 0 \ldots]$
- $1.1\delta[n-1] = [\ldots 0 \quad 0 \quad 0 \quad 0 \quad \underline{0} \; 1.1 \quad 0 \quad 0 \quad 0 \ldots]$
- $-\delta[n-2] \; = [\ldots 0 \quad 0 \quad 0 \quad 0 \quad \underline{0} \quad 0 \quad -1 \quad 0 \quad 0 \ldots]$
- $4\delta[n+1] \quad = [\ldots 0 \quad 0 \quad 0 \quad 4 \quad \underline{0} \quad 0 \quad 0 \quad 0 \quad 0 \ldots]$
- $3\delta[n+2] \quad = [\ldots 0 \quad 0 \quad 3 \quad 0 \quad \underline{0} \quad 0 \quad 0 \quad 0 \quad 0 \ldots]$

$x[n] = 3\delta[n+2] + 4\delta[n+1] + 2\delta[n] + 1.1\delta[n-1] - \delta[n-2].$ ← Generalize this

$x[n] = x[-2]\delta[n+2] + x[-1]\delta[n+1] + x[0]\delta[n] + x[1]\delta[n-1] + x[2]\delta[n-2].$

Further generalize this to an infinite-duration sequence

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

Another example of 2-input 1-output system: Product of two sequences

Another example of 2-input 1-output system: Product of two sequences

Multiplying sequences $x_1[n], x_2[n]$

$$y[n] = x_1[n]x_2[n].$$

This operation is used in modulation where
(i) one of the signals is a modulating signal of the form $e^{jn\omega_0}$ or $\cos n\omega_0$ and
(ii) the other signal is the modulated signal whose spectrum is shifted in the frequency domain as we will see later.

Other examples of D-T systems

Other examples of D-T systems

Combining scaling and shifting: Moving averager using $2M + 1$ samples

$$y[n] = \frac{1}{2M+1} \sum_{k=-M}^{M} x[n-k]$$

General weighted moving averager:
$y[n] = \frac{1}{N_2 - N_1 + 1} \sum_{k=N_1}^{N_2} w_k x[n-k]$

Squarer:

$$y[n] = x^2[n].$$

ReLU (Rectified Linear Unit) - used in machine learning algorithms:

$$y[n] = \max(0, x[n]).$$

System Properties
(To provide structure on systems)

System Properties (To provide structure on systems)

1. Linearity
2. Shift-Invariance
3. Memoryless (System)
4. Causality
5. Stability