1. Signals

1. **Signals - statistics, probability and noise**

2. **ADC and DAC**

Textbook: [Smith, ch. 2 and 3]

1. Signals - statistics, Probability and Noise

Statistics and **probability** - to characterize **signals** and the **processes** that generate them.

1.1 Terminology

A ***signal*** is a function of how **one variable** is related to **another variable**.

Example.
A common type of signal in analog electronics is a *voltage* that varies with *time*.

Signal **variables**

- **Amplitude** - the **vertical axis** may represent voltage, light intensity, sound pressure, or a number of other parameters.
- **Time** is the most common parameter to appear on the horizontal axis of acquired signals; however, other parameters are used in specific applications.

Signal **domain**

- A signal that uses **time** as the independent variable is said to be in the **time domain**. Similarly:
- in the **frequency domain**,
- in the **spatial domain** (distance is a measure of space).

Continuous and **digitized** signals

- Since both parameters can assume a continuous range of values, we call this a **continuous signal**.
- Passing a continuous signal through an **analog-to-digital converter** forces each of the two variable to be ***quantized***.

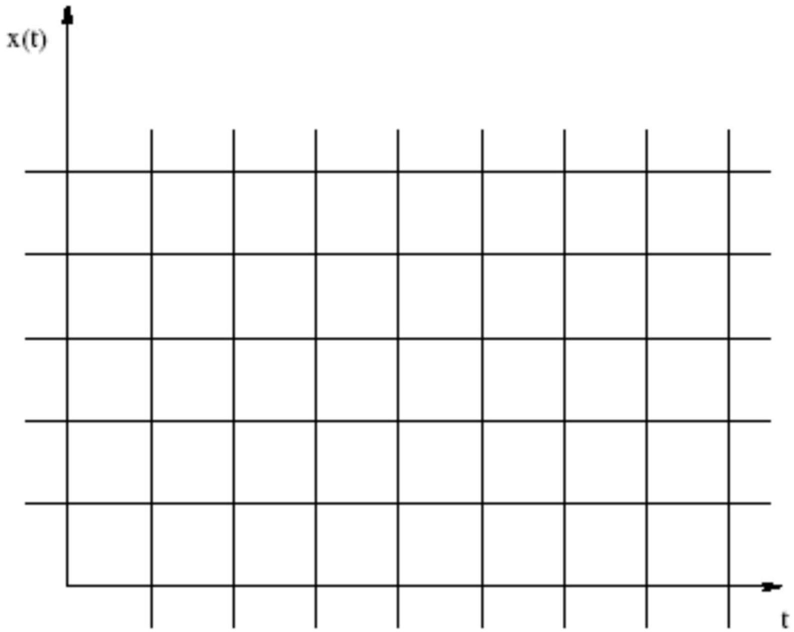
  Signals formed from parameters that are quantized are said to be **discrete signals** or **digitized signals**.

For the most part, continuous signals exist **in nature**, while discrete signals exist **inside computers**.

Signal classification with respect to continuous or discrete amplitude and time:

CA-CT ⇒ fully „**continuous**" signals (exist in nature)
DA-CT ⇒ ?
CA-DT ⇒ **analog** signals in discrete time (in electronic devices)
DA-DT ⇒ **digitized signals** (on digital devices)

Our interest is mainly in DT properties.



The DA-DT space.

Discrete-time (DT) signal **representation**

The variable, ***N***, is widely used to represent the total **number of samples** in a signal. For example, for the signals in Fig. 1: $N = 9$.

To keep the data organized, each sample is assigned its **index** i.

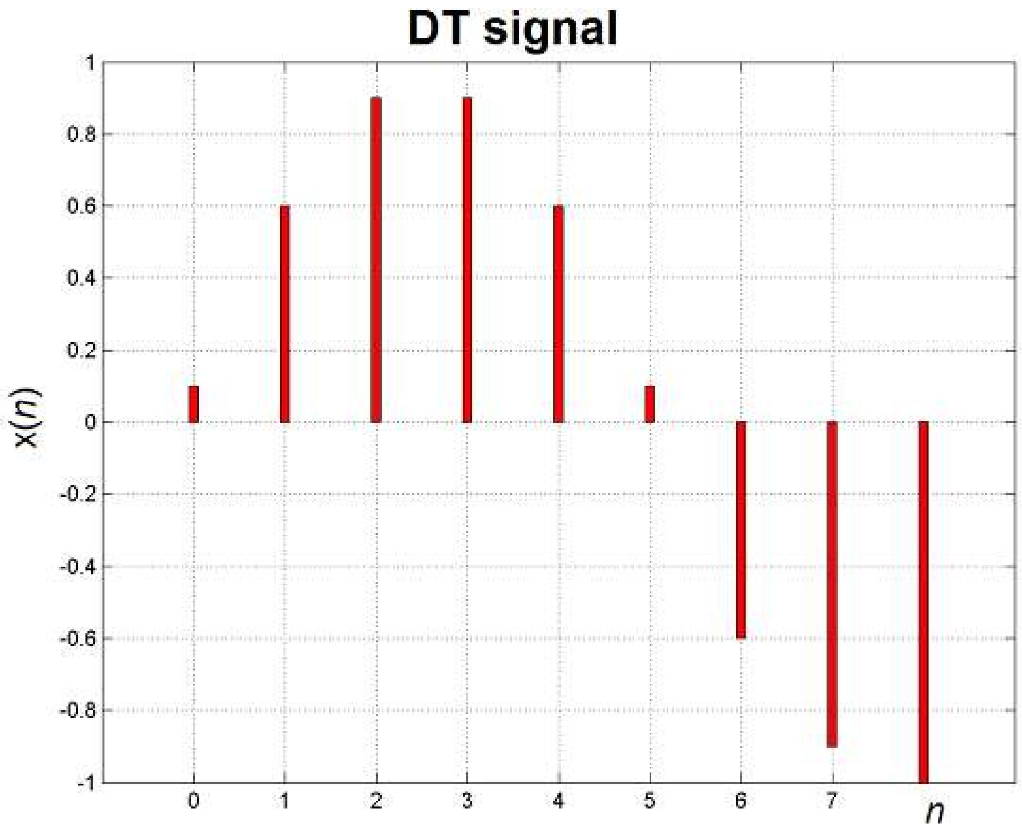


Fig. 1. A DT signal. $x[n] = \{x_i \mid i = 0, 1, ..., N\text{-}1\}$

Mean and Standard Deviation

The **mean**, indicated by **μ**, is the expected average value of a signal.

$$\mu = \frac{1}{N}\sum_{i=0}^{N-1} x_i \tag{1-1}$$

Let μ be the mean found from Eq. 1-1, *N* is the number of samples. Then σ is the **standard deviation**:

$$\sigma^2 = \frac{1}{N}\sum_{i=0}^{N-1} (x_i - \mu)^2 \tag{1-2}$$

The term, σ^2, is given the name **variance**.

DC and AC

- In electronics, the ***mean*** is commonly called the **DC** (direct current) value. **AC** (alternating current) refers to how the **signal fluctuates** around the mean.

- The above expression (1-2) describes how far in average the sample *deviates* (differs) from the mean. The variance, σ^2, represents the **energy of this fluctuation**.

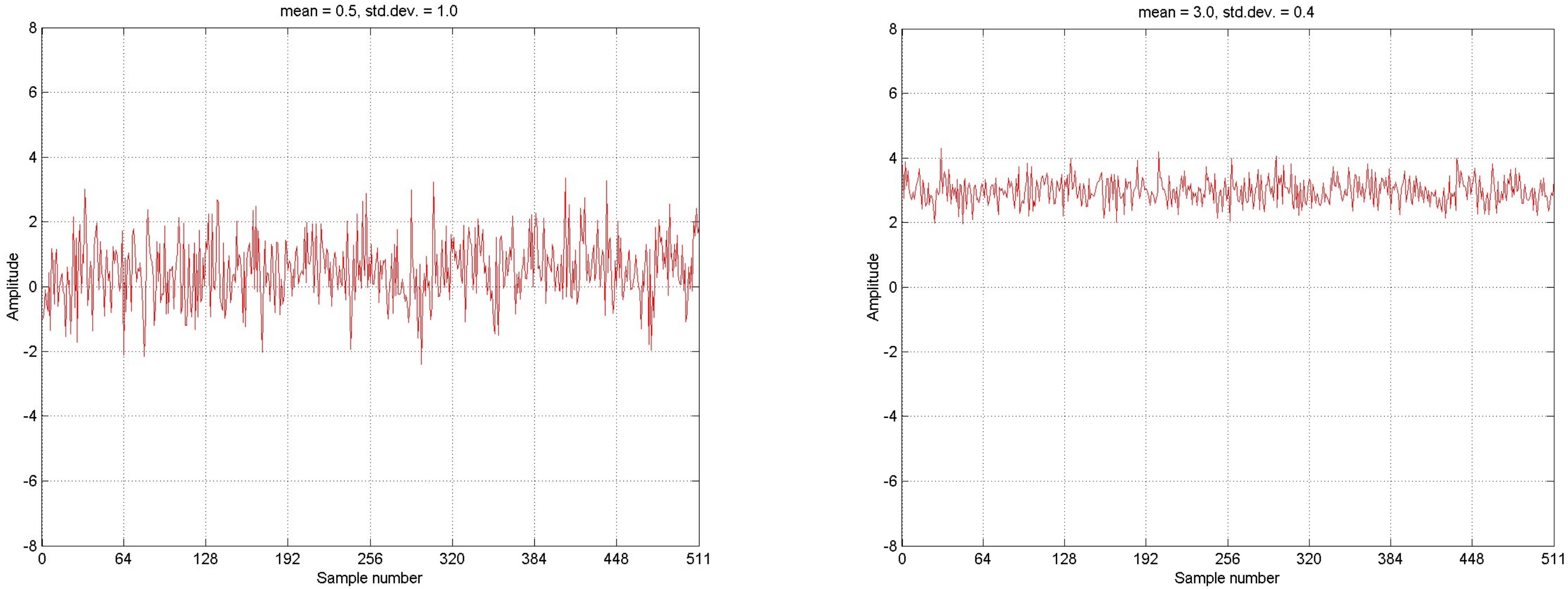


Fig. 2. Two digitized signals with different means and standard deviations.

RMS

The **rms (root-mean-square)** value is frequently used in electronics. The standard deviation only measures the AC portion of a signal, while the **rms** value measures **both the AC and DC components**.

$$rms = \sqrt{\frac{1}{N}\sum_{i=0}^{N-1} x_i^2} \qquad (1\text{-}3)$$

Peak-to-peak ratio

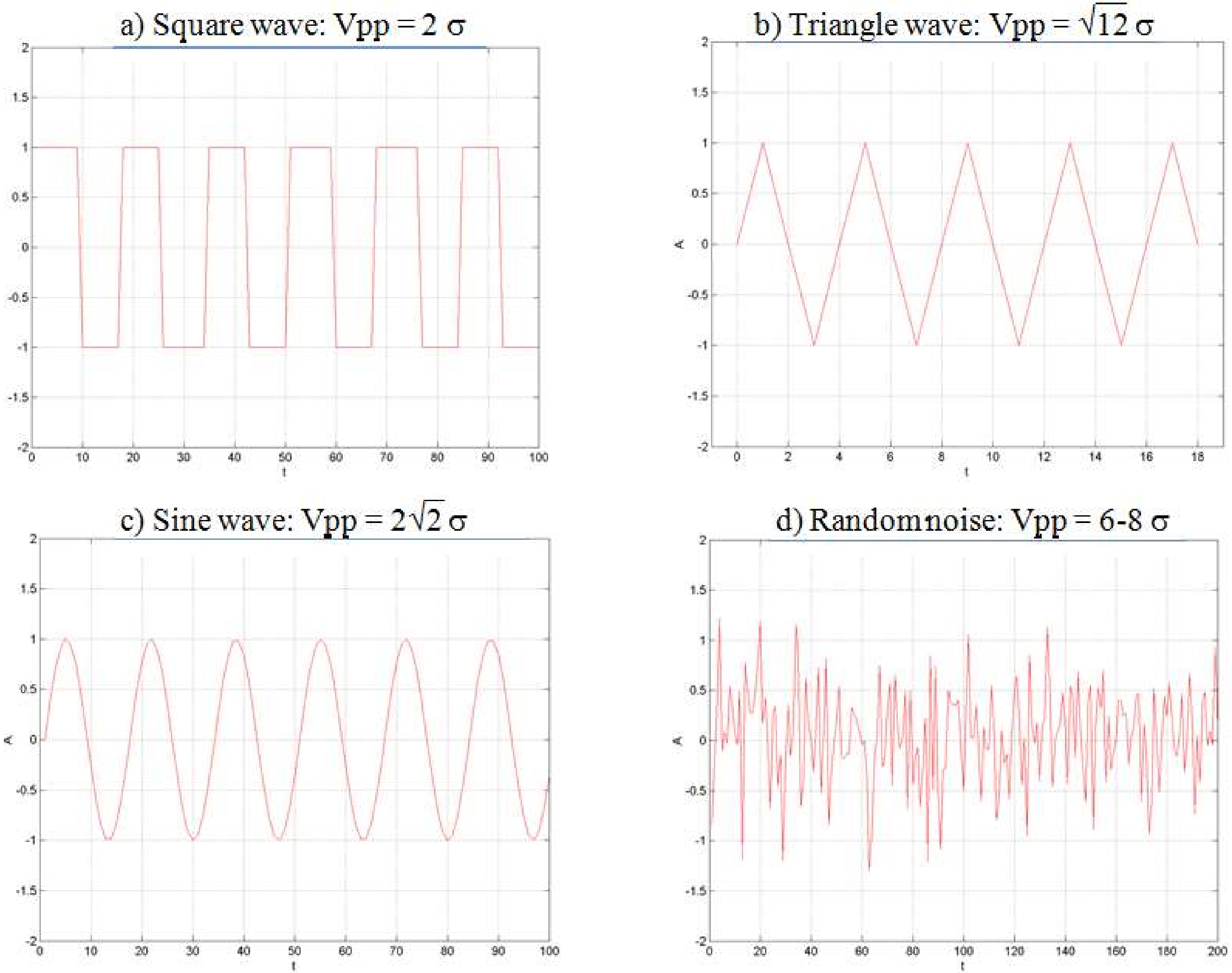


Fig. 3. Ratio of the peak-to-peak amplitude to standard deviation for some signals.

Signal-to-noise-ratio (SNR)

In some situations, the ***mean*** describes what is being measured, while the ***standard deviation*** represents noise and other interference. In these cases, the standard deviation is not important in itself, but only in *comparison* to the mean.

$$\boxed{\mathrm{SNR} = \frac{\mu}{\sigma}} \qquad (1\text{-}5)$$

For a **general SNR** one considers two signals:

1. an original source signal $\{ x_i \}$ and
2. a corresponding signal, which is the original one but corrupted by noise: $\{x_i'\}$

The difference signal represents the noise $\{n_i\}$, where:

$$n_i = x_i - x_i'.$$

Then the **signal-to-noise-ratio** is defined as (E means "*expected value*"):

$$\boxed{\mathrm{SNR} = 10 \log_{10} \frac{E\{x_i^2\}}{E\{n_i^2\}} \quad [\mathrm{dB}]} \qquad (1\text{-}6)$$

Better data (i.e. lower noise) means a higher value of the signal-to-noise ratio.

1.2 Signal vs. Underlying Process

- **Statistics** is interpreting ***numerical data***, such as **acquired signals**.
- **Probability** is used to understand the **processes** that generate signals.

Although they are closely related, a key question is to see the distinction between the **acquired signal** and the **underlying process**.

Example

Imagine creating a 1000 point signal by flipping a coin 1000 times: on heads the value is one and on tails the sample is set to zero.

- The ***process*** that created this signal has a mean of 0.5, determined by the relative probability of each possible outcome: 50% heads, 50% tails.
- However, it is unlikely that the actual 1000 point ***signal*** will have a mean of exactly 0.5. Random chance will make the number of ones and zeros slightly different each time the signal is generated.
- The ***probabilities*** of the underlying process are **constant**, but the ***statistics*** of the acquired signal **change** when the experiment is repeated.

This random irregularity found in actual data is called:

- **statistical variation,**
  statistical fluctuation, or
  statistical noise.

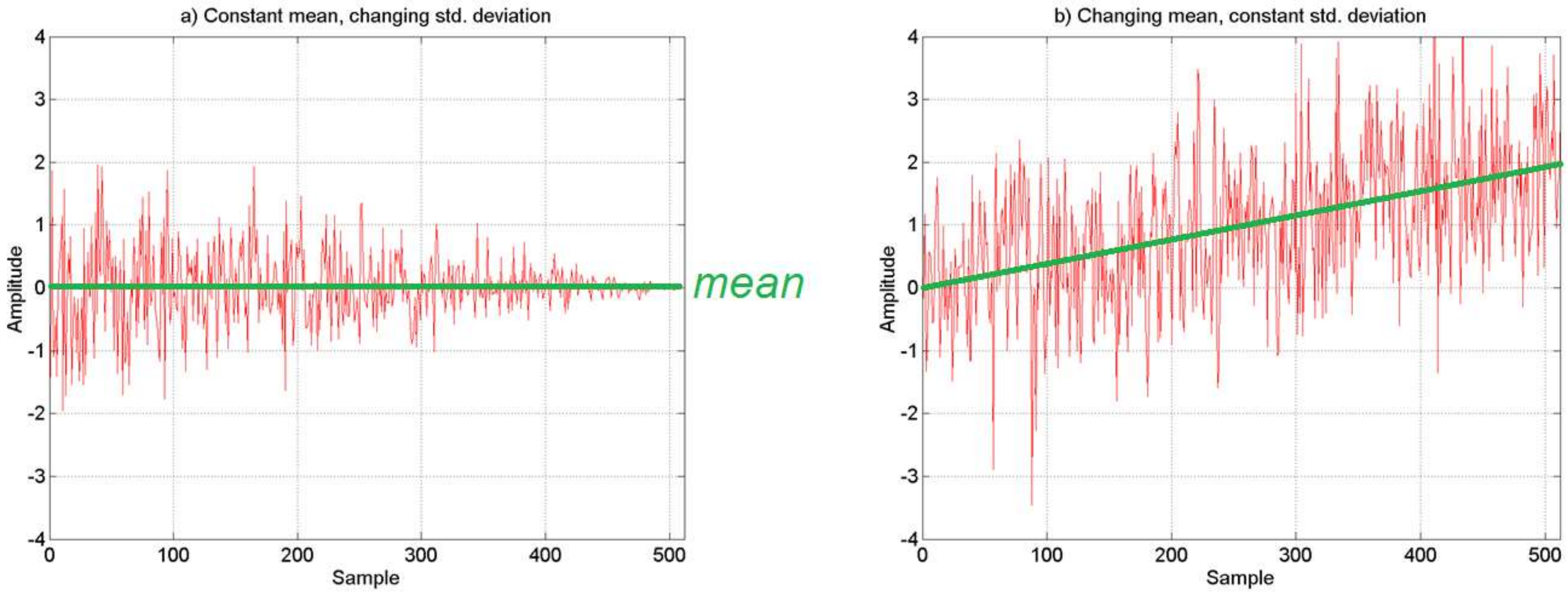
In particular, for **random signals**, the typical **error between the mean** of the N points, and the **mean of the underlying process**, is given by:

$$\boxed{\mathrm{error} = \frac{\sigma}{\sqrt{N}}} \tag{1-7}$$

- The larger the value of N, the smaller the expected error will become.
- The error becomes zero as N approaches infinity.

Nonstationary processes: processes that change their characteristics in time.

Common **problem** with **nonstationary signals**: the slowly changing ***mean interferes*** with the calculation of the *standard deviation*.



(a) (b)

Fig. 4. In (a), the mean is constant (= 0) while standard deviation changes from 1 to 0. The calculated σ of the entire signal is 0.57. (b) The slowly changing ***mean interferes*** with the calculation of the *standard deviation*. In (b), the standard deviation remains a constant value of 1, while the mean changes from a value of 0 to 2. The calculated standard deviation of the entire signal is 1.19.

This error can be nearly eliminated by **breaking the signal** into short sections, and calculating the statistics for each section individually.

1.3 The Histogram, Pmf and Pdf

Histogram

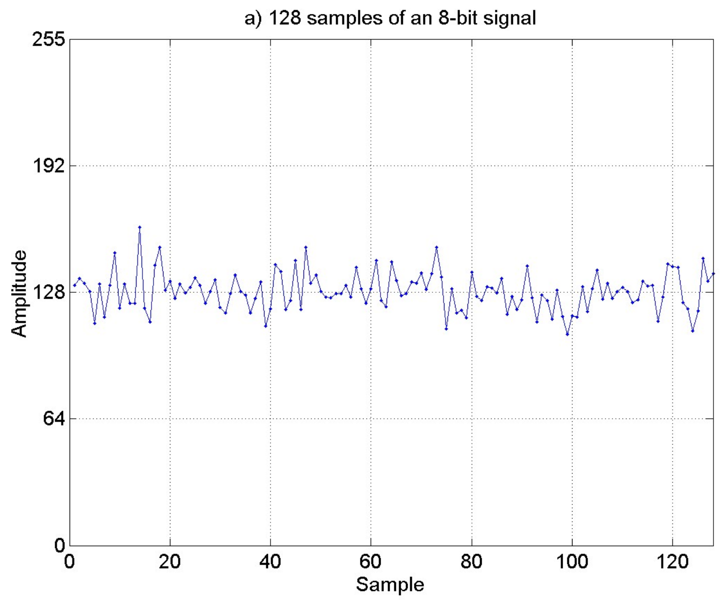
The **histogram** displays the *number of samples* that have each of *possible amplitude values*.

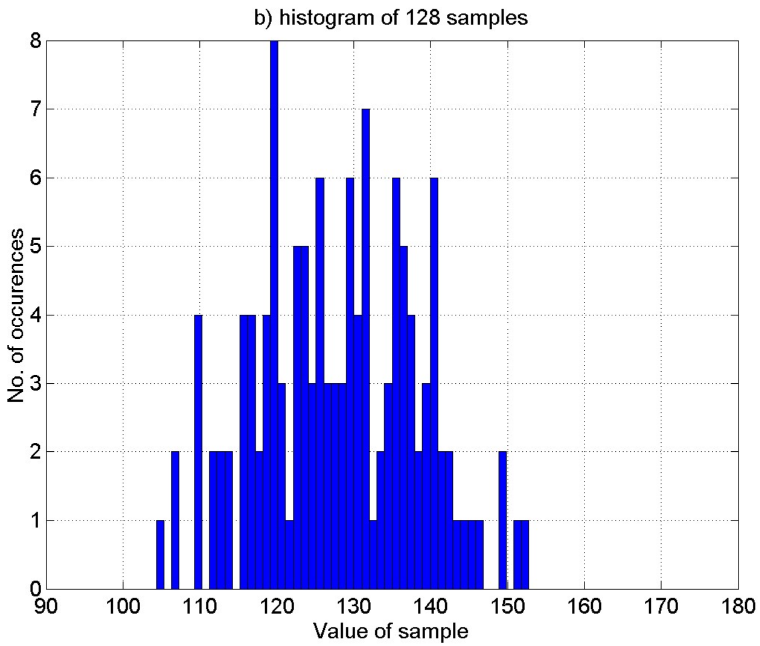
The sum of all of the values in the histogram is equal to the number of points in the signal:

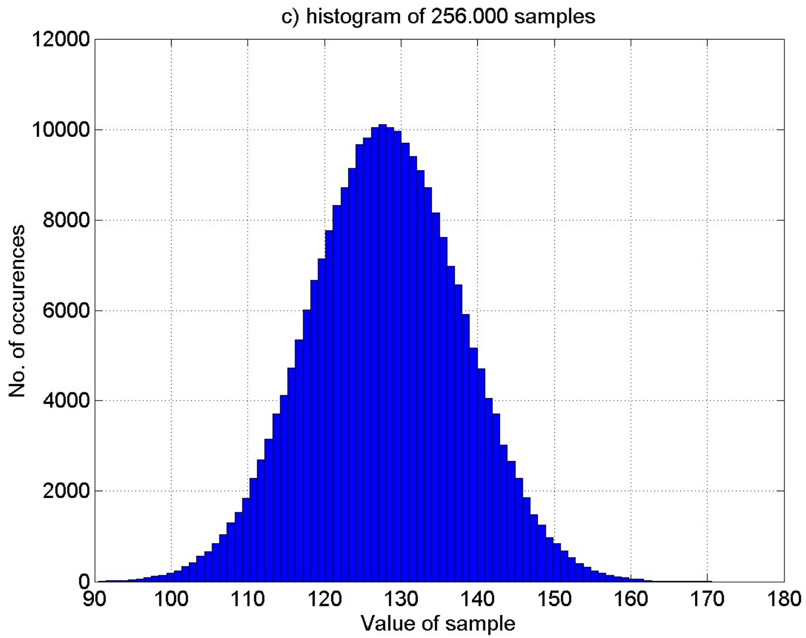
$$N = \sum_{i=0}^{M-1} H_i \tag{1-8}$$

H_i are the histogram values (specified along the Y-axis in the histogram), N is the number of points in the signal, and M is the number of possible amplitude values (along X-axis in the histogram).

Fig. 5. Figure (a) shows 128 samples from a very long signal, with each sample being an integer between 0 and 255. Figures (b) and (c) show histograms using 128 and 256,000 samples from the signal, respectively. →


a) 128 samples of an 8-bit signal
Amplitude
255
192
128
64
0
0
20
40
60
80
100
120
Sample


b) histogram of 128 samples
No. of occurences
8
7
6
5
4
3
2
1
0
90
100
110
120
130
140
150
160
170
180
Value of sample


c) histogram of 256.000 samples
No. of occurences
12000
10000
8000
6000
4000
2000
0
90
100
110
120
130
140
150
160
170
180
Value of sample

The role of histogram

The histogram can be used to **efficiently calculate** the mean and standard deviation of **very large** data sets :

$$\mu_X = \frac{1}{N} \sum_{i=0}^{M-1} x[i] \cdot H_i \qquad (1\text{-}9)$$

$$\sigma_X^{\ 2} = \frac{1}{N} \sum_{i=0}^{M-1} (x[i] - \mu_X)^2 \cdot H_i \qquad (1\text{-}10)$$

Pmf

The ***histogram*** is what is formed from an **acquired (digital) signal**.

The normalized histogram, i.e. $\boldsymbol{H}' = \left[\frac{H_i}{N}; i = 1, .., M-1\right]$, with $N \rightarrow \infty$, is called the **probability mass function (pmf)**.

The **pmf** describes the ***probability*** that a certain value will be generated.

The **histogram** and **pmf** are used with **discrete data**, such as DT signals.

Pdf

The **probability density function (pdf)** (or the **probability distribution function)** is for continuous signals the same as **pmf** is for DT signals.

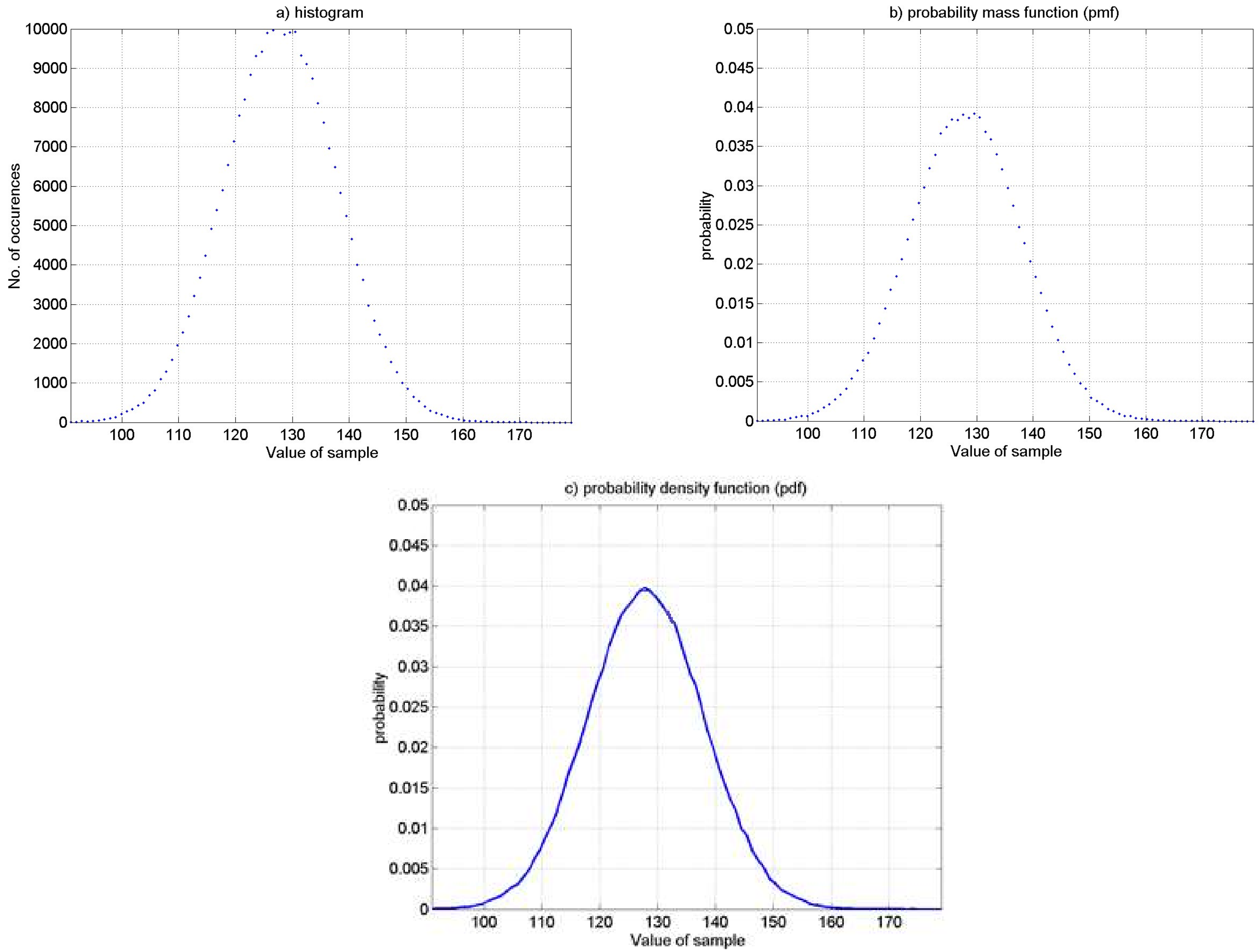


Fig. 6. The relationship between (a) the histogram, (b) the pmf, and (c) the pdf.

The **amplitude** of the three curves on fig. 6 satisfies:
(a) the **sum of the values** in the histogram is equal to the number of samples in the signal;
(b) the **sum of the values** in the pmf is equal to one, and
(c) the **area under** the pdf curve is equal to one.

The vertical axis of the pdf is expressed in units of **probability density**.
Example
A pdf of 0.03 at 120.5 *does not* mean that 120.5 will occur 3% of the time.
In fact, the probability of the continuous signal being exactly 120.5 is nearly zero.

To calculate a ***probability*** a *range* of values is needed. Probabilities are assigned to such realizations of X that take a value from some interval, e.g. for an interval [A,B] we have:

$$P(A \leq X \leq B) = \int_A^B p_X(x)dx \tag{1-12}$$

The **cumulative distribution function (cdf)**, $F_X(x) = p_X(X \leq x)$, is:

$$F_X(x) = \int_{-\infty}^{x} f(z)dz \tag{1-13}$$

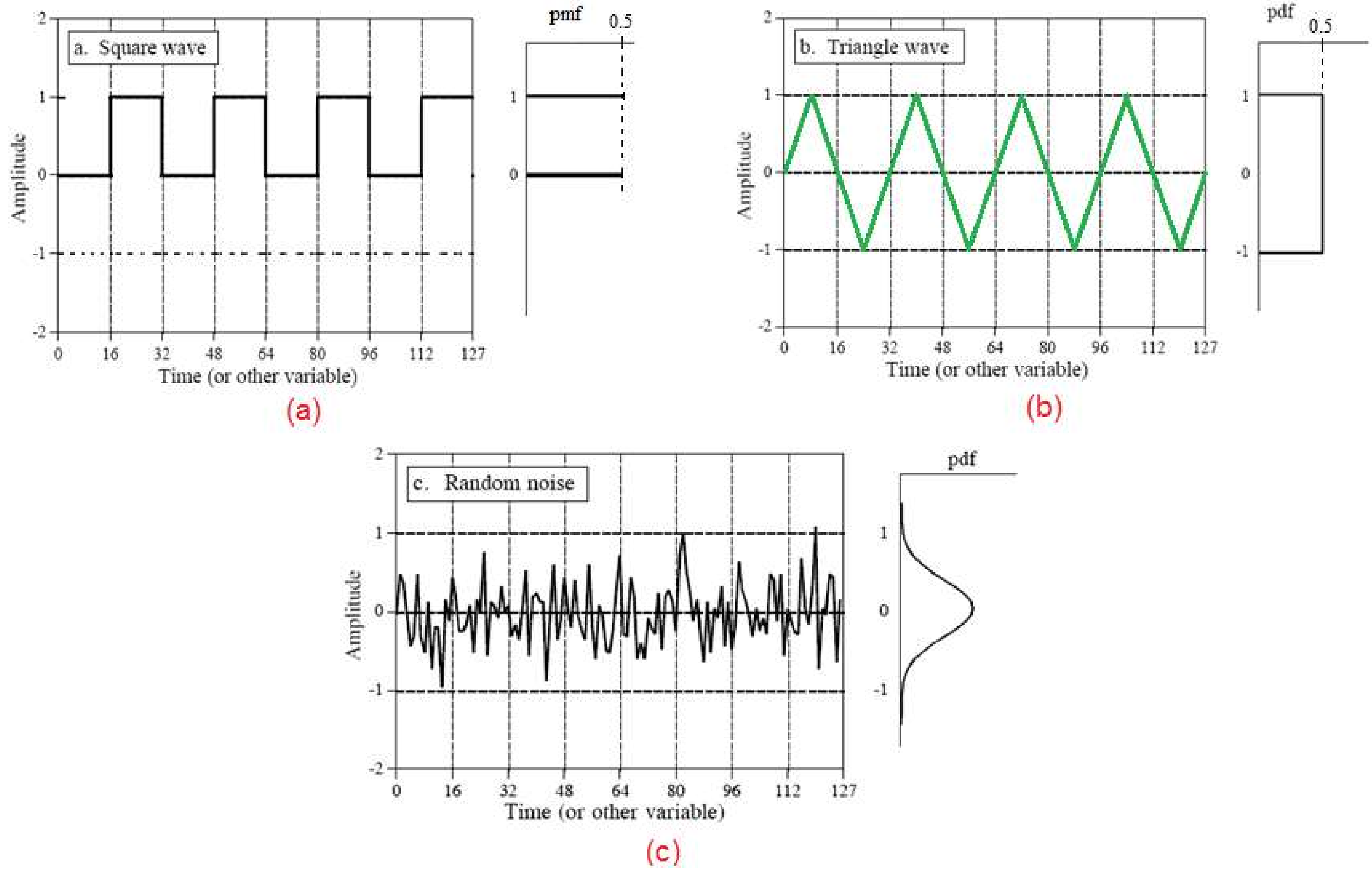
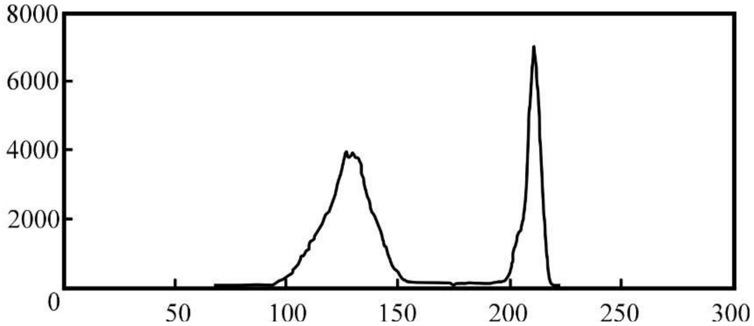


Fig. 7. Three common waveforms and their probability distribution functions: (a) **binary** pmf, (b) constant value - **uniform** distribution, (c) bell shaped curve known as a **Gaussian**.

Exercise 1-1

The intensity distribution in the image is modelled as a stochastic variable with the pmf $p_X(A \le X \le B)$ and cumulated distribution $F_X(x) = p_X(X \le x)$:



Let the following 4x4 images be given.

a)

0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15

b)

3	3	3	3
7	7	7	7
11	11	11	11
15	15	15	15

(1) Scan an image to a 1D signal.

(2) Obtain the pmf-s of the digital intensity and pdf-s of the expected original analogue signals that best match the statistics of the digital signals.

(3) Compute and compare the means and variances of the pmf-s and pdf-s.

Binned histogram

- A problem occurs when the **number of value levels** each sample can take on is much larger than the **number of samples** in the signal. This is typical for signals represented in *floating point* notation.

The solution to these problems is a technique called **binning**:

- Arbitrarily select the length of the histogram to be some convenient number, such as 1000 points, often called **bins**.
- The value of each bin represent the total number of samples in the signal that have a value within a ***certain range***.

How many bins should be used (Fig. 8):

- Too many bins makes it difficult to estimate the *amplitude* of the underlying pmf – only a few samples fall into each bin, making high statistical noise;
- Too few of bins makes it difficult to estimate the underlying pmf in the *horizontal* direction.

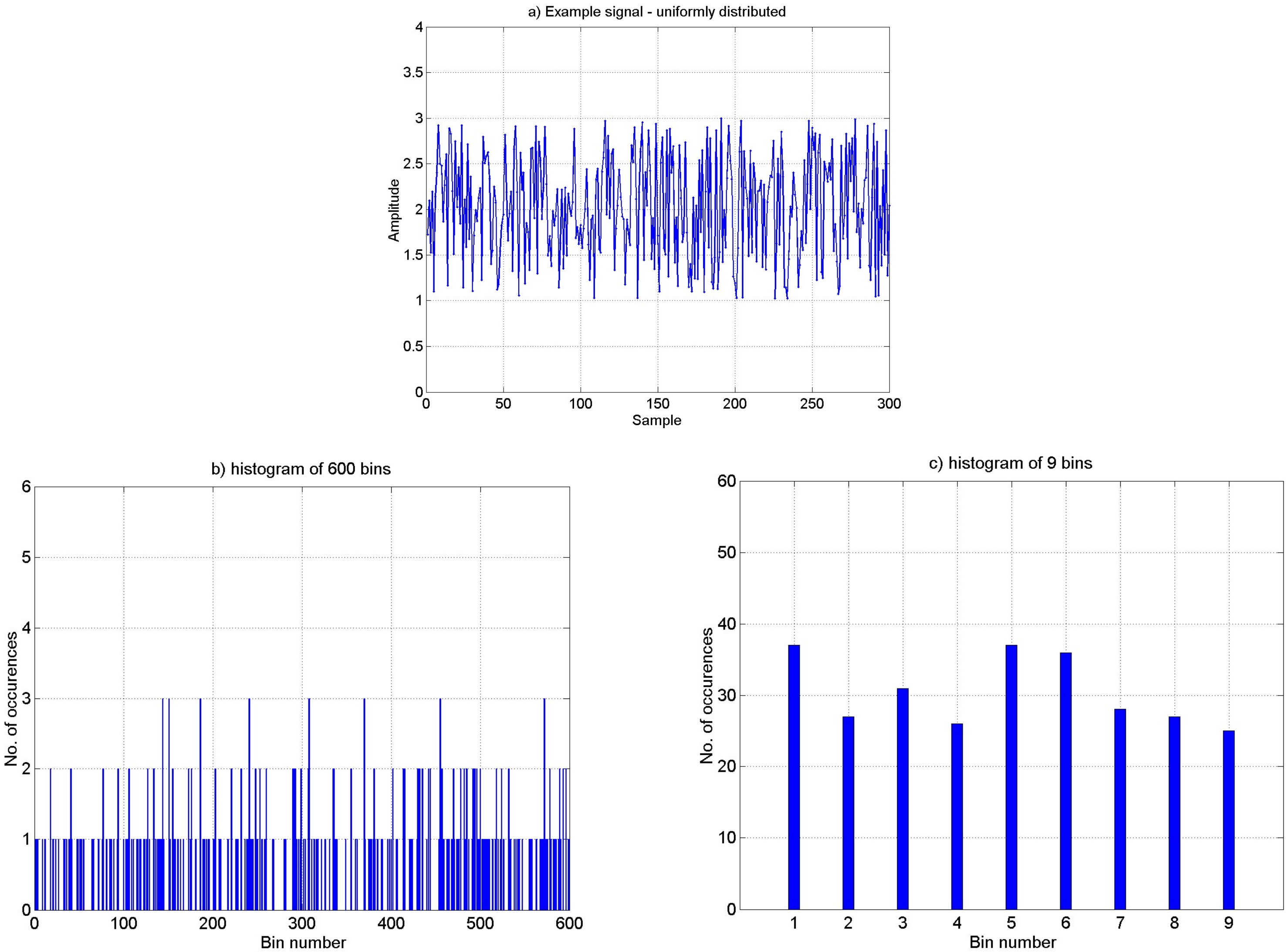


Fig. 8. (a) The signal is 300 samples long, with each sample a floating point number uniformly distributed between 1 and 3. (b) and (c) show binned histograms of this signal, using 600 and 9 bins, respectively:

1.5 The Normal Distribution

Signals formed from random processes usually have a bell shaped pdf. This is called a **normal distribution**, a **Gauss distribution**, or a **Gaussian**.

The **basic shape** of the curve is generated from a *negative squared exponent*:

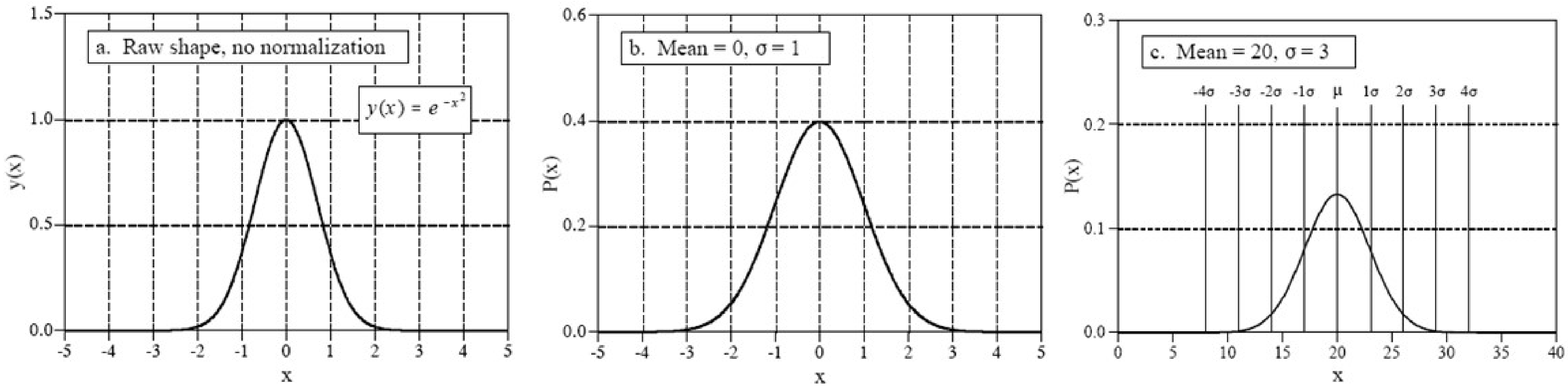
$$y(x) = e^{-x^2} \tag{1-14}$$

This **raw curve** can be converted into the **complete Gaussian** by adding two parameters:

1. an **adjustable mean,** μ, and
2. **standard deviation,** σ.

In addition, the equation must be **normalized** so that the total area under the curve is equal to one:

$$\boxed{P(x) = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-(x-\mu)^2/2\sigma^2}} \tag{1-15}$$



(a) (b) (c)

Fig. 9. Examples of Gaussian curves: (a) a raw curve; (b) and (c) - the complete Gaussian curve for various means and standard deviations.

Bounded appearance of Gaussian

An interesting characteristic of the Gaussian is that the ***tails*** **drop toward zero very rapidly**, much faster than with other common functions such as **decaying exponentials** or **1/*x***.

In practice, the sharp drop of the Gaussian pdf means that **extreme unlimited values** of amplitude **almost never** occur. This results in the waveform having a relatively **bounded appearance** with a peak-to-peak amplitude of about **6-8 σ.**

Samples taken from a normally distributed signal will be:

- within ±1σ of the mean about 68% of the time,
- within ±2σ about 95% of the time, and
- within **±3σ about 99.75%** of the time.

1.6 Digital Noise Generation

The heart of digital noise generation is the **random number generator**.
Let the **function *RND*** returns a new random number **between zero and one** each time the function is called.
Each random number has **equal probability**.

Digital noise with a *Gaussian* pdf

Two basic methods for generating a Gaussian using a random number generator:

1. add twelve random numbers
2. transform two random numbers

1-st method

- Generate and add **twelve numbers** to produce a **sample**, i.e.,

$$X = RND + \ldots + RND.$$

The sum X is from the interval <0, 12>. The mean is 6, and the standard deviation is 1.

- **For each sample** in the signal:

(1) add twelve random numbers,

(2) subtract six to make the mean= zero,

(3) multiply by the standard deviation desired, and

(4) add the desired mean.

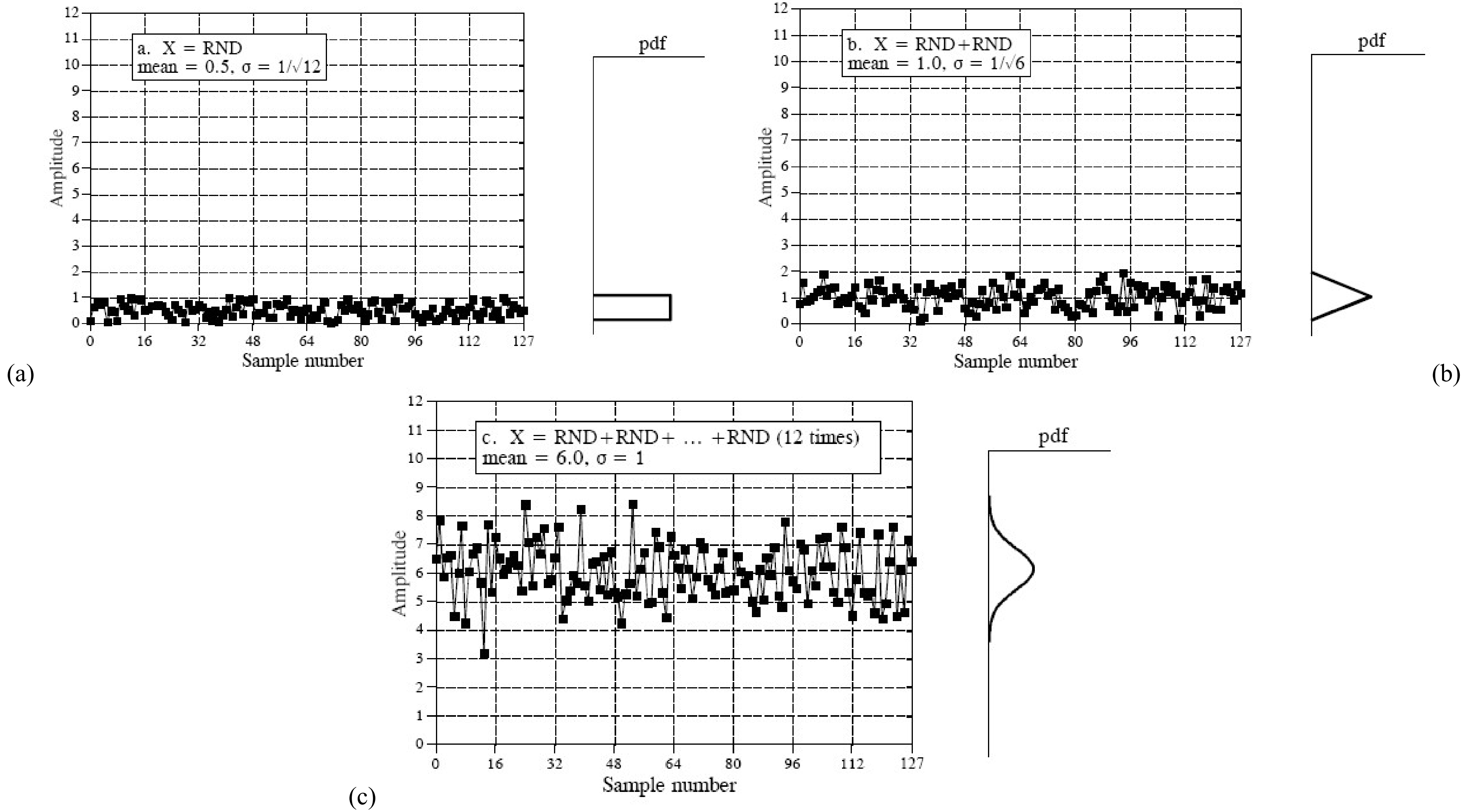


Fig. 11. Converting a uniform distribution to a Gaussian distribution: (a) a signal of 128 samples generated by *RND* - uniformly distributed between zero and one; (b) adding *two* values from the random number generator; (c) adding *twelve* values from the random number generator. The pdf of (c) is very nearly Gaussian, with a mean of *six*, and a standard deviation of *one*.

The **Central Limit Theorem**:
a ***sum*** of ***random*** numbers becomes ***normally distributed*** as more and more of the random numbers are added together.

The Central Limit Theorem ***does not*** **require** the individual random numbers be from any particular distribution, or even that the random numbers be from the *same* distribution.

Why **normally distributed** signals are seen **so widely in nature**?
→Whenever many different random forces are interacting, the resulting pdf becomes a Gaussian.

Second method

- The random number generator is called ***twice***, to obtain R_1 and R_2. A normally distributed random number, X, can then be found as:

$$X = (-2\log R_1)^{1/2}\cos(2\pi R_2) \tag{1.16}$$

X is normally distributed with a mean of zero, and a standard deviation of 1. The log is base *e,* and the cosine is in radians.

- To generate a Gaussian with an arbitrary mean and standard deviation; multiply X by the desired standard deviation, and add the desired mean.

Pseudo-random generator

Random number generators operate by starting with a **seed**, a number between zero and one.

The algorithm that transforms the seed into the random number is often as follows

$$R = (aS + b)\,\mathrm{mod}(c) \tag{1-17}$$

(the quantity $aS+b$ is divided by c, and the remainder is taken as R), where

- S is the seed,
- R is the new random number, and
- a, b, c are appropriately chosen constants.

Most program libraries have a possibility to **reseed** the random number generator, allowing to choose the number first used as the seed.

- A common technique is to use the *time* (from the system's clock) as the seed, thus providing a new sequence each time the program is run.

Precision and Accuracy

Precision and accuracy are terms used to evaluate systems and methods that ***measure*, *estimate***, or ***predict***.

There is some variable you wish to know - the **true value**, or simply, **truth**.

The **measured value** should be as close to the true value as possible.

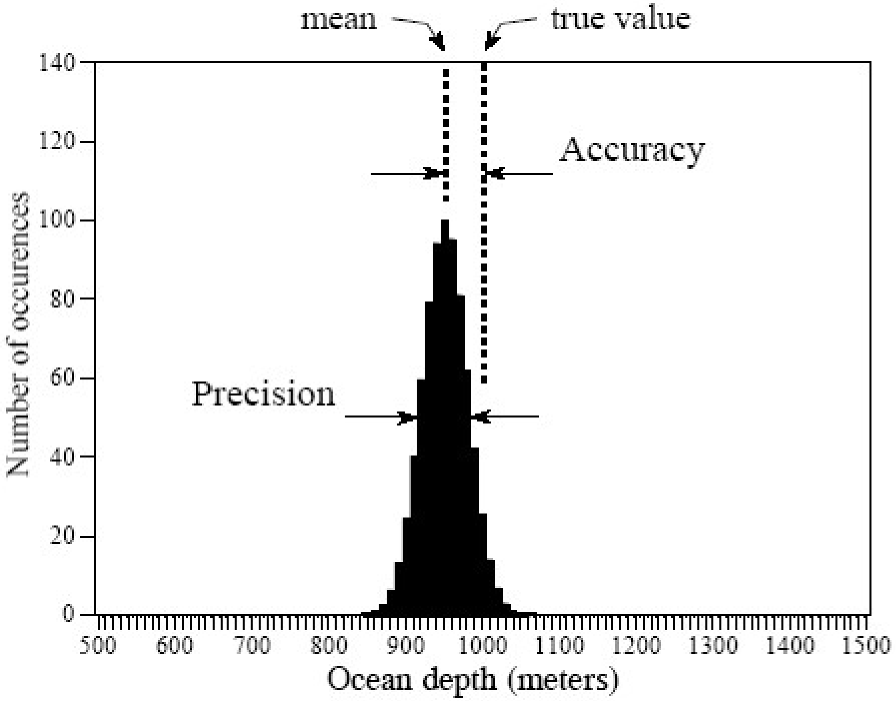


Fig. 12. **Accuracy** is the difference between the true value and the mean of the under-lying process that generates the data. **Precision** is the spread of the values, specified by the standard deviation or the signal-to-noise ratio.

Precision is a measure of **random noise**.

Poor accuracy results from **systematic errors.**

Averaging individual measurements does nothing to improve the accuracy.

Accuracy is usually dependent on how you *calibrate* the system -
accuracy is a measure of ***calibration***.

To know what is the error type, check the following:

1. Will **averaging successive readings** provide a better measurement?
   If yes, call the error **precision**; if no, call it **accuracy**.

2. Will **calibration** correct the error?
   If yes, call it **accuracy**; if no, call it **precision**.

2. ADC and DAC

2.1 Sampling and Quantization

Analog-to-Digital Conversion (ADC) and **Digital-to-Analog Conversion** (DAC) - processes that allow digital computers **to interact** with *everyday signals*.

The **ADC** consists of two steps:

1. the **sample-and-hold (S/H)**, and
2. the **analog-to-digital value converter** (quantization, digitalization).

Sampling converts the *independent variable* (time in this example) from **continuous** to **discrete**.

The output of the sample-and-hold changes only at **periodic intervals**, at which time it is made **identical** to the **instantaneous** value of the input signal.

Quantization converts the *dependent variable* from continuous to discrete.

It produces an integer value between, say, 0 and 4095 for each of the amplitude intervals.

Sampling and quantization **degrade** the signal in **different** ways.

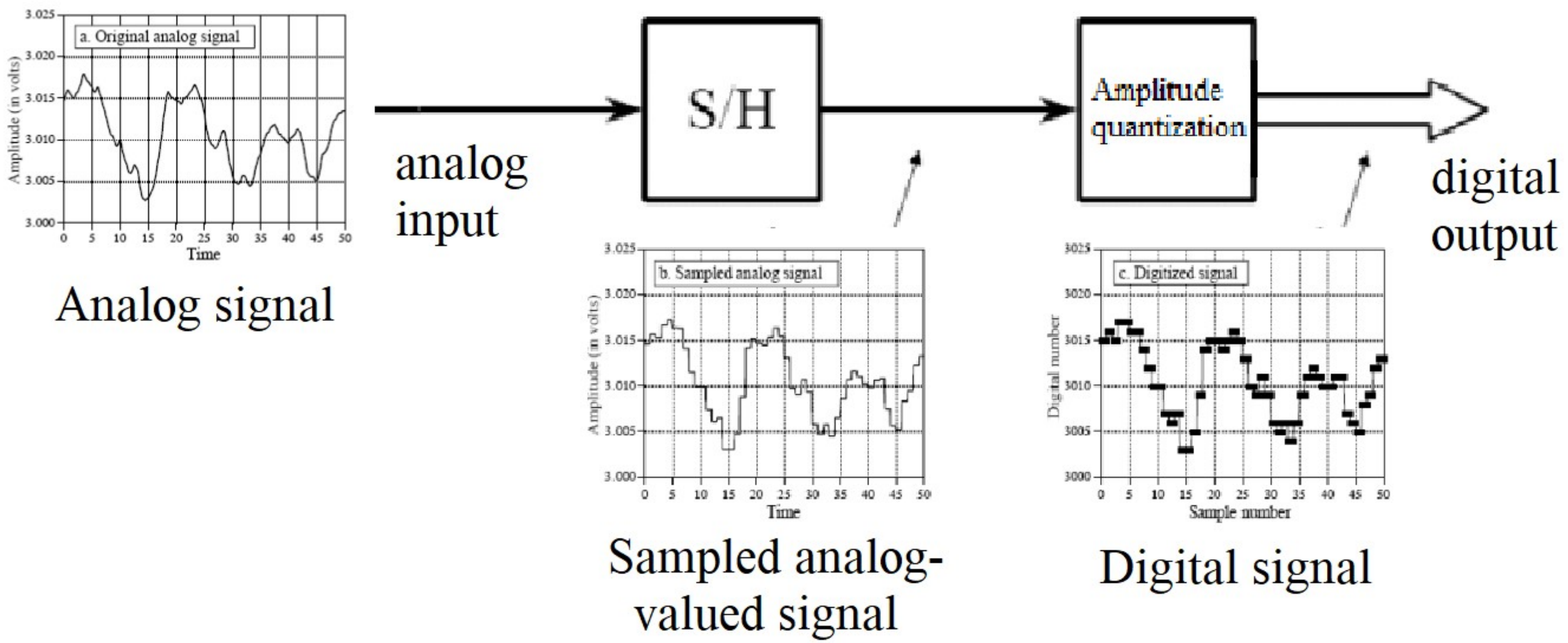
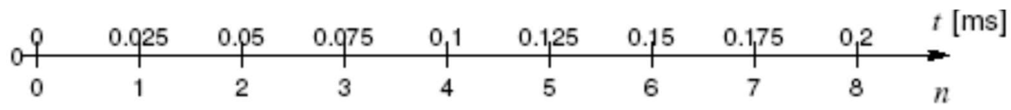


Fig.13. Stage 1 is the **sample-and-hold** (S/H) process - **periodic sampling** takes place. In second stage the quantization process converts the **amplitude** of every sample to the nearest digital level.

2.2 Periodic sampling



n = t/Ts; Ts = 0.025 [ms]

The definition of ***proper sampling***:

if you can exactly ***reconstruct*** **the analog** signal from the samples, you must have done the **sampling** ***properly*** (the key information has been captured, it can be reversed).

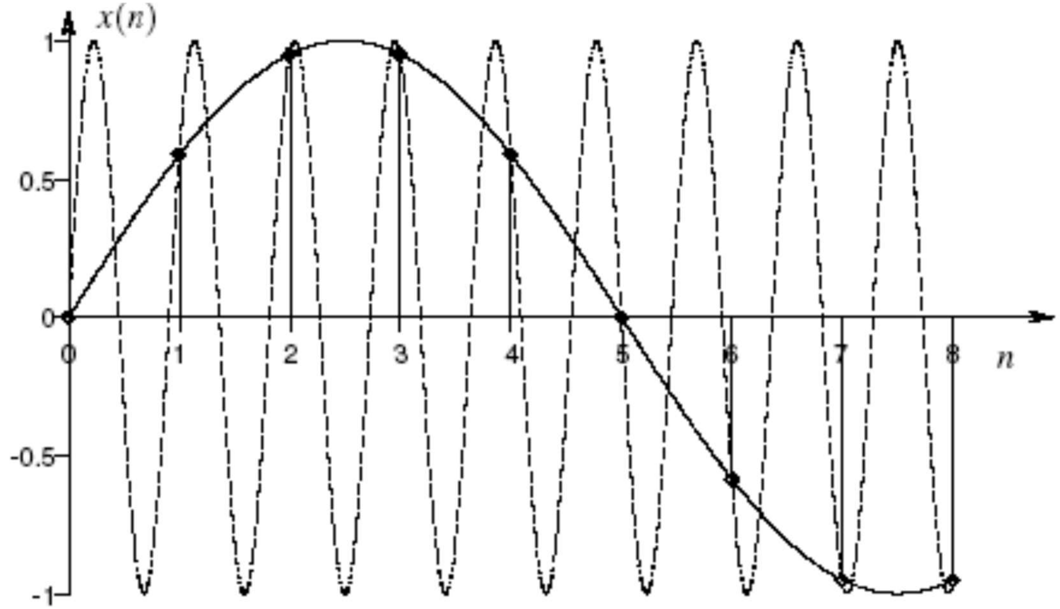


Fig. 14. Example of misinterpretations: we do not know what is between points [0,1,2,...,8]: a) sin(n (1/5) π) or b) sin(n (2+1/5) π) ? We have to **know** which one to choose. This will be explained by the **sampling theorem**.

2.3 Effects of quantization

A single sample in the digitized signal can have a maximum error of $\pm\frac{1}{2}\Delta_{LSB}$, where: *LSB* - **Least Significant Bit**, Δ_{LSB} **-** the distance between adjacent quantization levels.

The digital output is equivalent to the **continuous input** ***plus*** **a quantization error**.

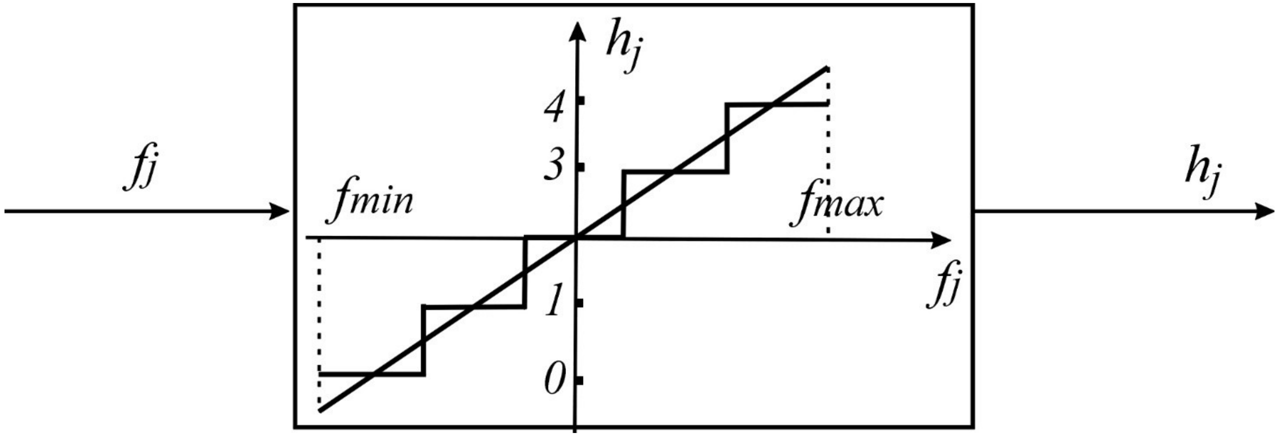
The quantization error appears very much like *random noise*.
In most cases, *quantization* results in nothing more than the addition of a specific amount of *random noise* to the signal.

- The additive noise is uniformly distributed between $\pm\frac{1}{2}\Delta_{LSB}$, has a mean of zero, and a standard deviation of $\boxed{\frac{1}{\sqrt{12}}\Delta_{LSB}}$ ($\approx$ 0.29 Δ_{LSB}).

- The *number of bits* determines the *precision* of the data.

Exercise 1-2

Amplitude digitalization.



Real value f_i ,
discrete value: h_i.

Digitalization error:
$n_i = f_i - h_i$

The signal-to-noise ratio (SNR) expresses the digitalization quality:

$$\mathrm{SNR} = 10 \log_{10} \frac{E(f_i^2)}{E(n_i^2)} \quad [\mathrm{dB}]$$

(A) Estimate the SNR value in case of an analog signal digitalization with *B* bits.
(B) How is the signal-to-noise increasing if the number of bits is increased by 1.

Dithering

Dithering is a common technique for improving the digitization of **slowly varying** signals.

- A small amount of **random noise is added** to the analog signal.
- Even when the original analog signal is changing by less than $\pm\frac{1}{2}\Delta_{LSB}$, the added noise causes the digital output to randomly toggle between adjacent levels.

2.4 The Sampling Theorem

The **sampling theorem** (the *Shannon or Nyquist* sampling theorem):

" a continuous signal can be **properly sampled**, *only if it* **does not contain** *frequency components* **above one-half** *of the* **sampling rate**".

The **Nyquist frequency** **(**or the **Nyquist rate)** means ***one-half the sampling rate***.

If the frequency of the analog sine wave is greater than the **Nyquist frequency** the this results in ***aliasing*** and the original signal cannot be reconstructed from the samples.

The phenomenon of sinusoids changing frequency during sampling is **aliasing**.

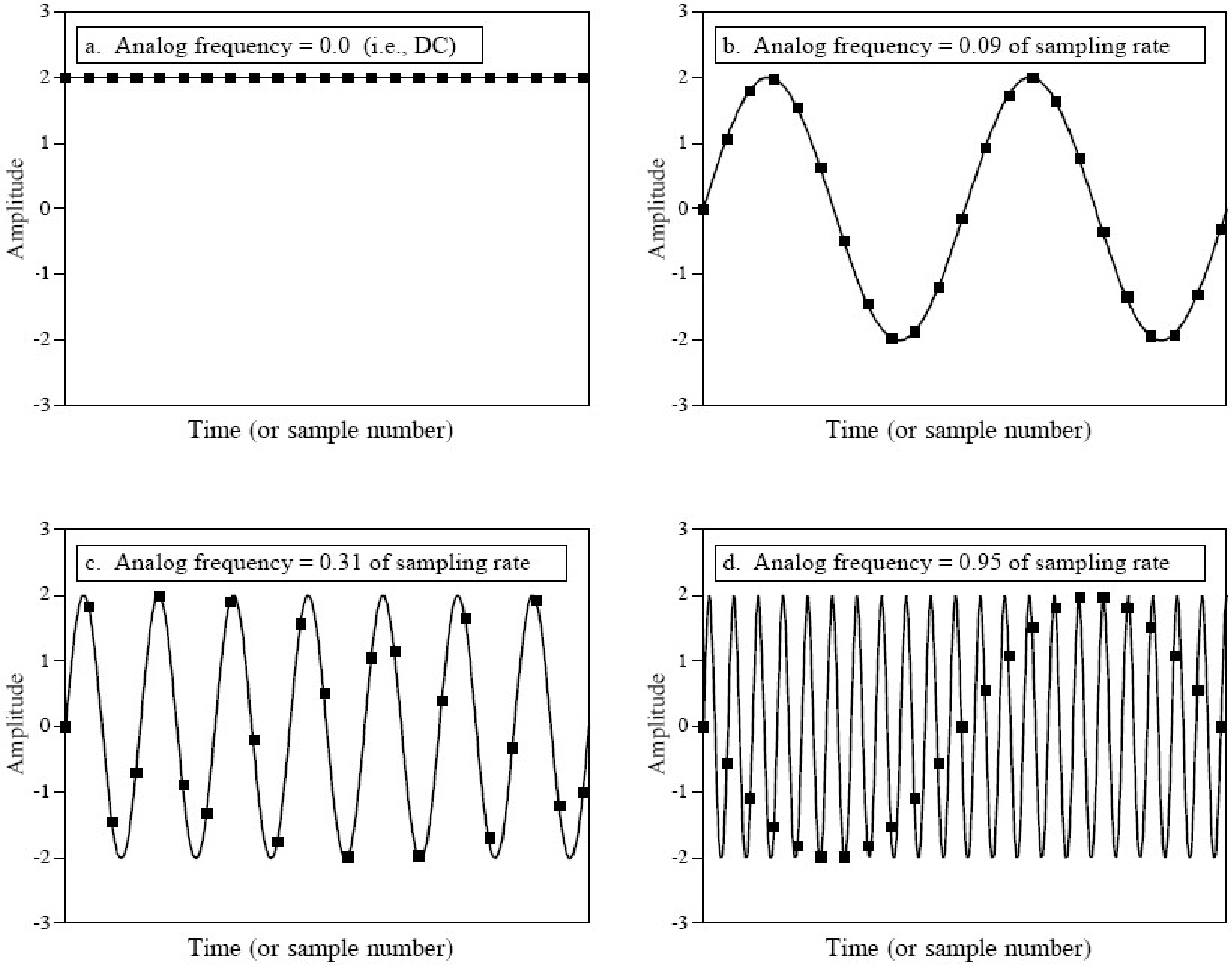


Fig. 16. Illustration of proper (a,b,c) and improper (d) sampling. The original sine wave of 0.95 sampling rate **misrepresents itself** as a sine wave of 0.05 sampling rate in the digital signal.

How frequencies are changed during aliasing?

1. Every continuous frequency **above** the Nyquist rate has a corresponding digital frequency between zero and one-half the sampling rate.

Example

The digital frequency of "$0.2f_s$" could have come from any one of an infinite number of frequencies in the analog signal: 0.2, 0.8, 1.2, 1.8, 2.2,....

2. Aliasing can **change the *phase*** by 180 degrees.

   The zero phase shift occurs for analog frequencies of 0 to 0.5, 1.0 to 1.5, 2.0 to 2.5, etc.

   An inverted phase occurs for analog frequencies of 0.5 to 1.0, 1.5 to 2.0, 2.5 to 3.0, and so on.

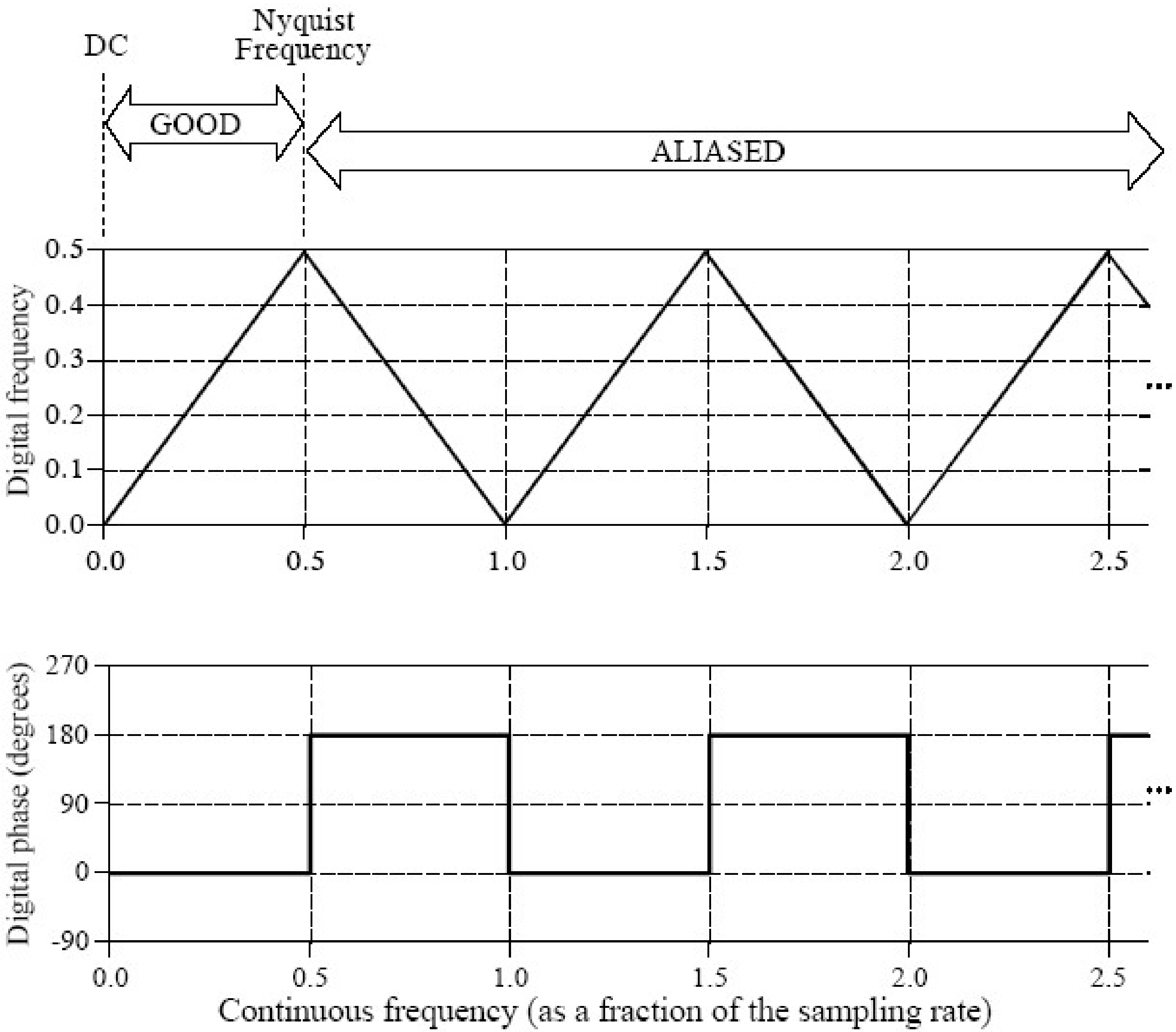


Fig. 17. Conversion of analog frequency into digital frequency during sampling.

Impulse train

a) In the **time domain**, sampling is achieved by multiplying the original signal by an **impulse train** of *unity amplitude* spikes. The frequency spectrum of this unity amplitude impulse train is **of constant unity amplitude**.

b) When two **time domain signals are multiplied**, their frequency spectra are **convolved**. This results in the **original signal's spectrum being duplicated to the location** of multiples of the sampling frequency, f_s, $2f_s$, $3f_s$, $4f_s$, etc.

c) Each multiple of the sampling frequency, f_s, $2f_s$, $3f_s$, $4f_s$, etc., has received a *copy* and a *left-for-right flipped copy* of the original signal's frequency spectrum. The copy is called the **upper sideband**, while the flipped copy is called the **lower sideband**.

The signal in (c) can be transformed back into the signal in (a) by eliminating all frequencies above $\frac{1}{2}f_s$, i.e. an analog low-pass filter will convert the impulse train, (b), back into the original analog signal, (a).

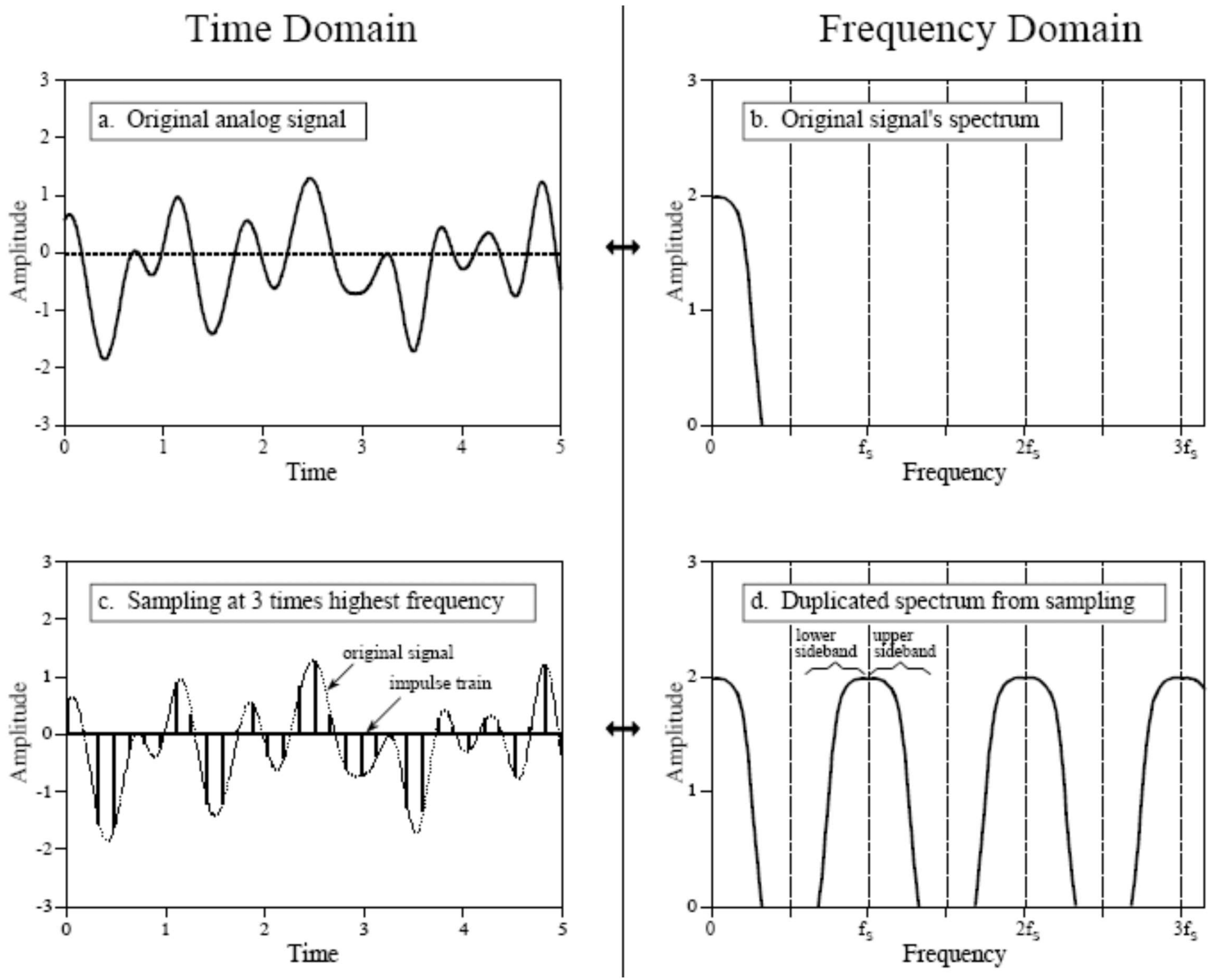


Fig. 18 The sampling theorem.

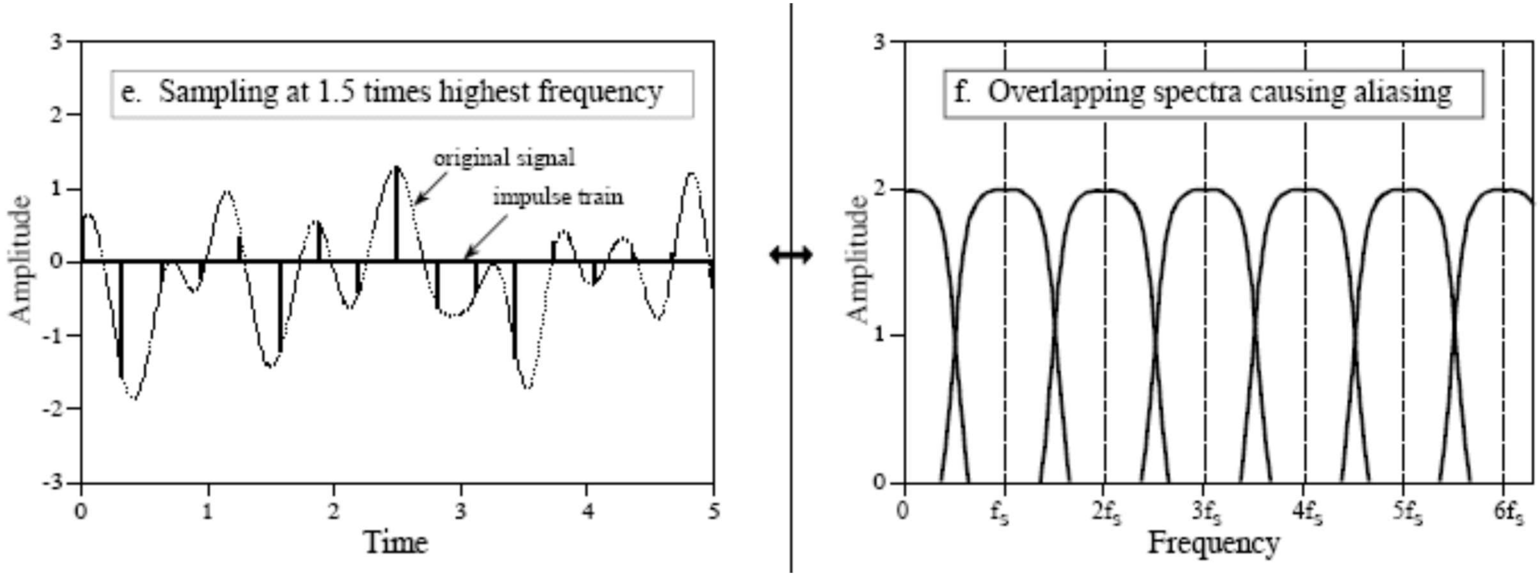


Fig. 18. The sampling theorem: (a) and (b) show an analog signal (in time and frequency resp.) composed of frequency components between 0 and $0.33f_s$; (c) the analog signal is properly sampled (at rate f_s) by converting it to an impulse train; (d) in the frequency domain the spectrum being duplicated into an infinite number of upper and lower sidebands, but disconnected; (e) the analog signal is now sampled at $0.66f_s$; (f) this results in aliasing, indicated by the overlapping sidebands.

2.5 Digital-to-Analog Conversion

In theory, for **digital-to-analog** conversion:

1. pull the samples from memory and convert them into an *impulse train*;
2. pass this impulse train through a low-pass filter, with the cutoff frequency equal to one-half of the sampling rate.

In practice: it is difficult to generate the required narrow pulses in electronics.

Nearly all DACs operate by **holding the last value** until another sample is received (this is called a **zero-order hold**, the DAC equivalent of the sample-and-hold used during ADC).

Other approaches:

- a **first-order hold** is to draw straight lines between the points,
- a **second-order hold** uses parabolas, etc..

Zero-order hold

In the **frequency domain**, the zero-order hold results in the spectrum of the impulse train being ***multiplied*** by the curve given by the equation:

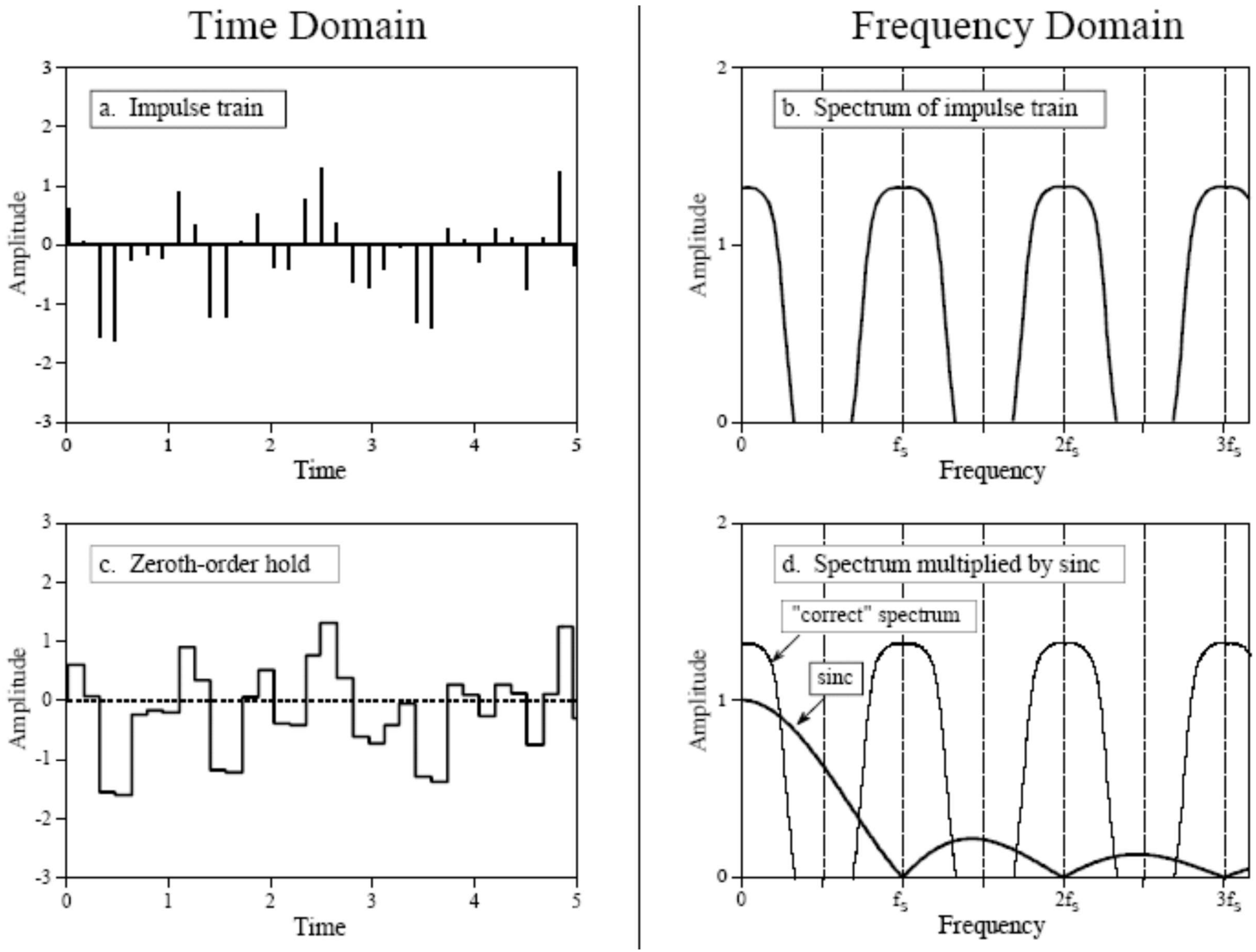
$$H(f)=\left|\frac{\sin(\pi f/f_s)}{\pi f/f_s}\right| \tag{1-18}$$

The sampling frequency is represented by f_S. For $f=0$, $H(f)=1$
This is of the general form:

$$\frac{\sin(x)}{x}\ , \tag{1-19}$$

called the **sinc function** or **sinc(x)**.

The zero-order hold can be understood as the **convolution** of the **impulse train** with a **rectangular pulse**, having a **width equal to the sampling** period.
This results in the frequency domain in the signal’s spectrum ***multiplication*** by the **sinc function**, i.e **the Fourier transform** of the rectangular pulse.


Time Domain
Frequency Domain
a. Impulse train
b. Spectrum of impulse train
c. Zeroth-order hold
d. Spectrum multiplied by sinc
"correct" spectrum
sinc
Amplitude
Time
Frequency
0
1
2
3
4
5
-1
-2
-3
f_s
2f_s
3f_s

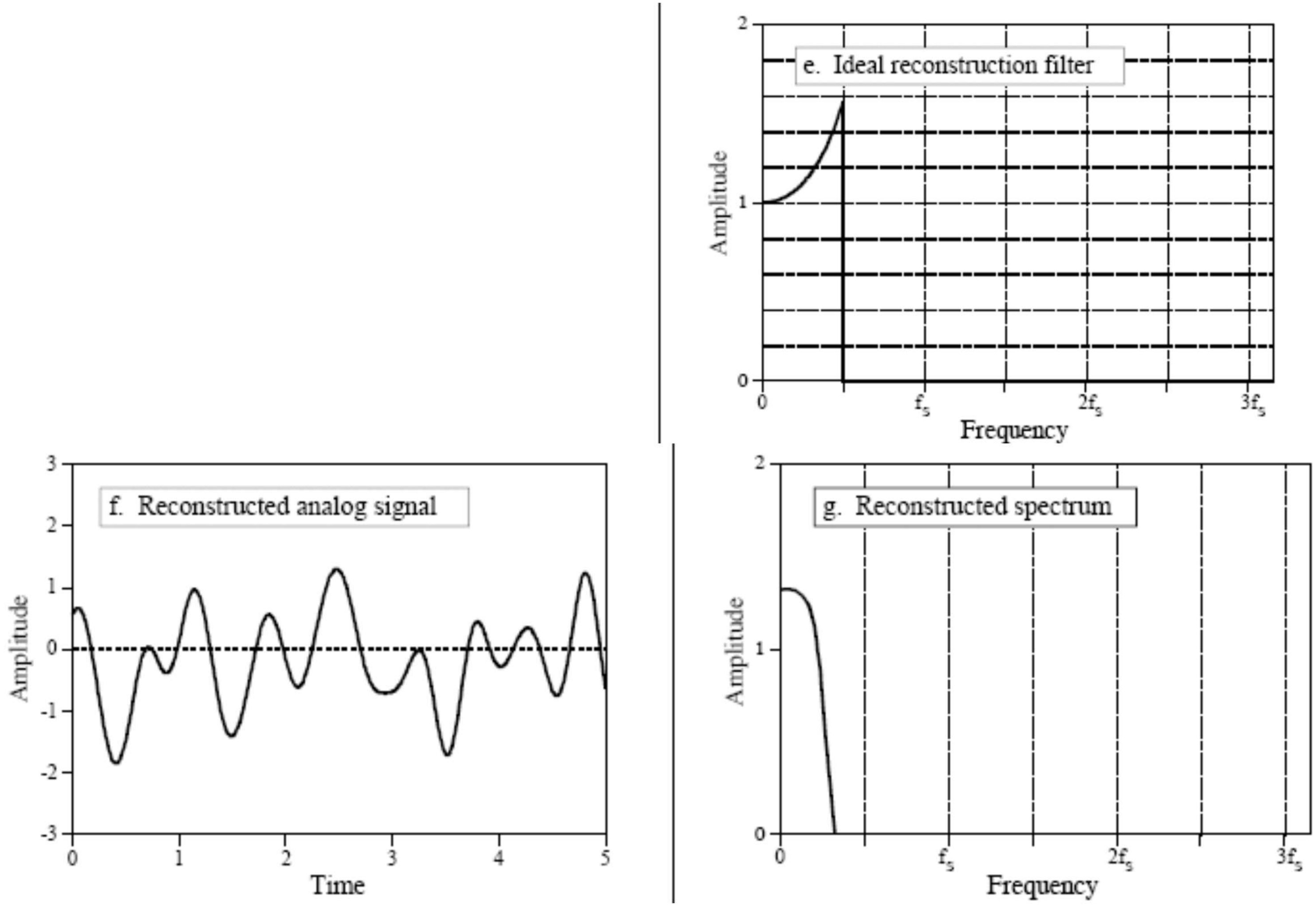


Fig. 19. Analysis of **digital-to-analog** conversion: (a) the digital data as an impulse train and (b) its spectrum; (c) zero-order hold waveform and (d) its spectrum; (e) a low-pass filter to remove frequencies above the Nyquist rate, *and* to correct for the sinc; (f) reconstructed signal and (g) its spectrum.

2.6 Analog Filters for Data Conversion

- The **antialias filter**: the input signal is processed with an electronic low-pass filter to remove all frequencies above the Nyquist frequency - to prevent aliasing during sampling.
- The reconstruction filter: the digitized signal is passed through a digital-to-analog converter and another low-pass filter set to the Nyquist frequency.

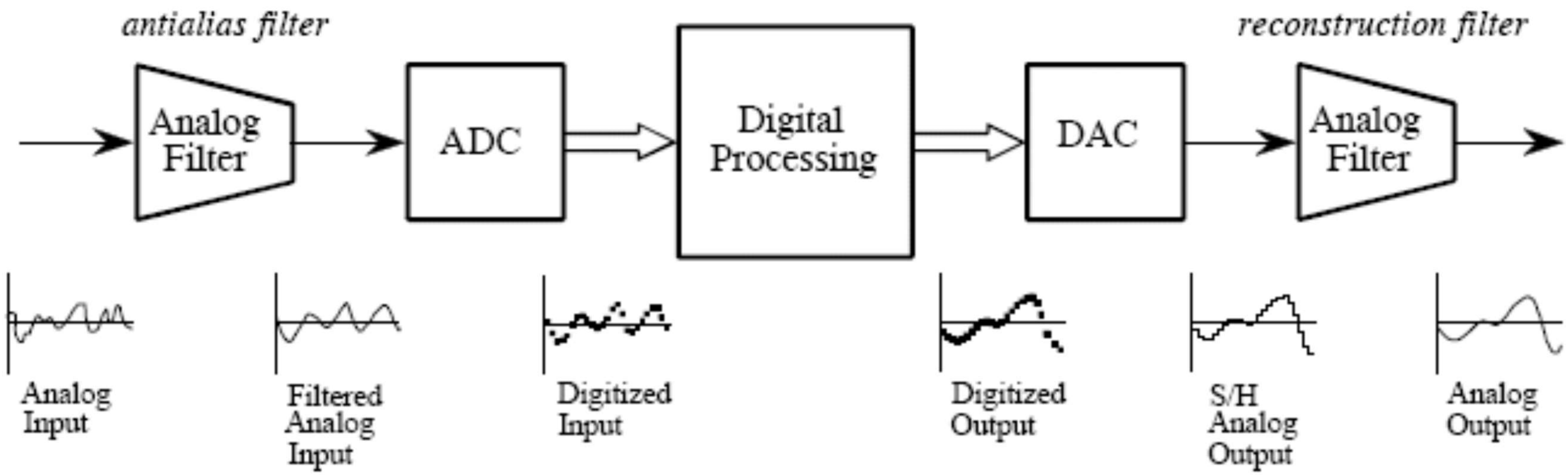


Fig. 20. Analog electronic filters used to comply with the sampling theorem.

Popular analog filter types:

- **Chebyshev**,
- **Butterworth**,
- **Bessel (Thompson)**

Each of these is designed to optimize a different performance parameter.