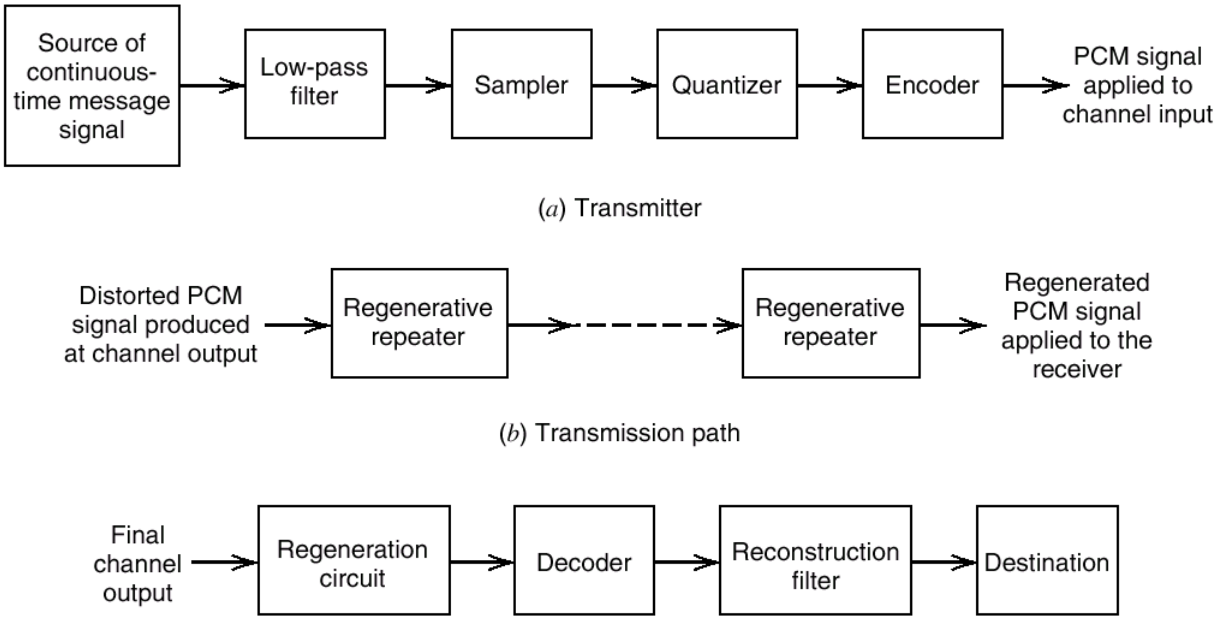
Analog to Digital Conversion

In this chapter we will consider the main steps involved in the transmission of an analog signal using digital means. The following diagram illustrates the basic blocks


Source of continuous-time message signal
Low-pass filter
Sampler
Quantizer
Encoder
PCM signal applied to channel input
Distorted PCM signal produced at channel output
Regenerative repeater
Regenerative repeater
Regenerated PCM signal applied to the receiver
Final channel output
Regeneration circuit
Decoder
Reconstruction filter
Destination

(*a*) Transmitter

(*b*) Transmission path

(*c*) Receiver

Three steps are involved in the conversion of an analog signal into a digital binary signal

Sampling: The sampling process converts a continuous time continuous amplitude signal into a continuous amplitude discrete time signal.

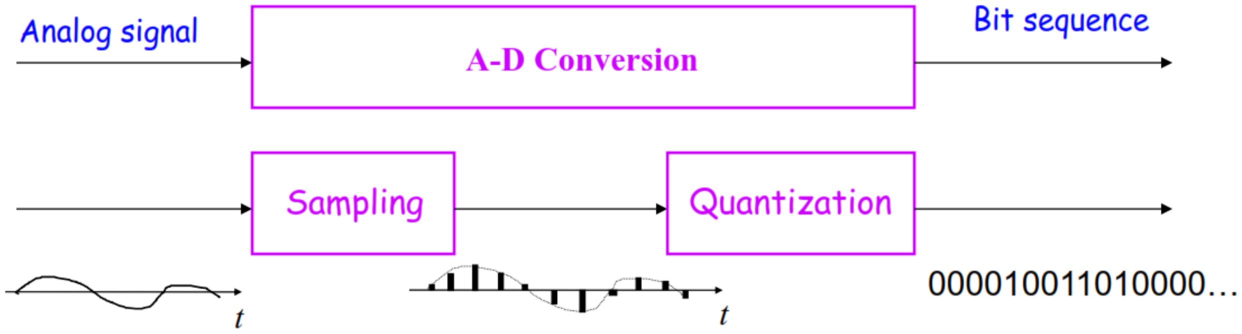
Quantization: A quantizer converts the continuous amplitude samples into discrete amplitude samples. The possible amplitude levels belong to a finite set S = $\{m_1, m_2, \ldots, m_L\}$. This is a finite and countable set of size L.

Binary Encoding: Each quantizer output level is mapped into a sequence of n binary digits. Usually, L and n are related by

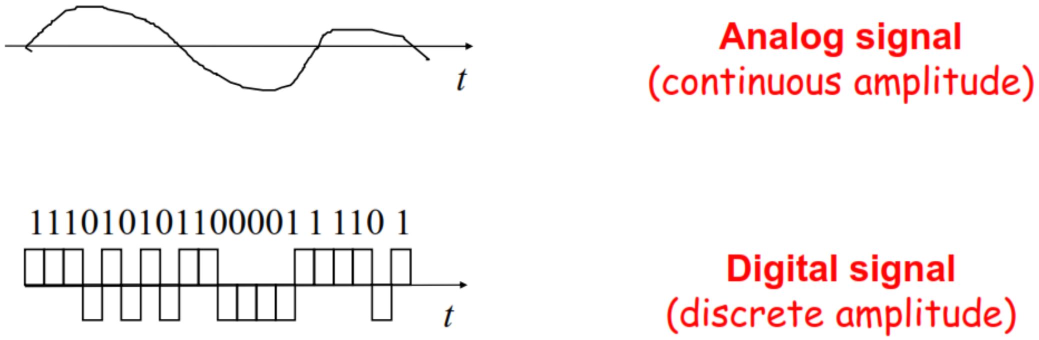
$$L = 2^n$$

Since there are L quantization levels, then we need L binary representations (codewords) for each level.

The conversion of an analog signal into a digital one is demonstrated in the figure



In this figure, we show typical analog and digital signal waveforms



Analog vs Digital: Advantages and Disadvantages

An analog signal can take an infinite variety of shapes. The distortion caused by noise cannot be removed by amplification or filtering.

Advantages of Digital Transmission

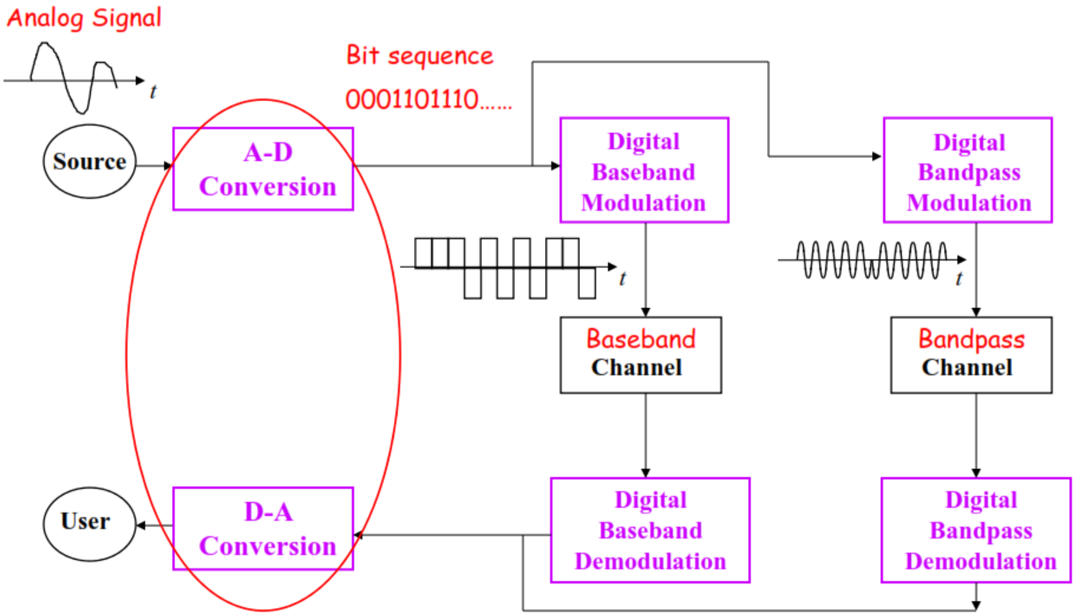
- Digital signals are more immune to channel noise by using channel coding techniques where error correction can be implemented (ideally, perfect decoding is possible by virtue of Shannon channel coding theorem).
- Digital signals belong to a finite set of possible waveforms. Repeaters along the transmission path can identify the transmitted digital waveform logically (0 or 1) and regenerates a noise free pulse sequence.
- Digital signals derived from all types of analog sources can be represented using a uniform format.
- Digital signals are easier to process by using microprocessors and VLSI
- Digital systems are more flexible to implement and allow for implementation of sophisticated functions and control.
- Digital signals make use of digital signal processing techniques (encryption, error control coding,…). This will enhance security of the transmitted signal.
- Digital circuits are less subject to distortion and interference
- Digital circuits are more reliable and less expensive than analog circuits

Few Disadvantages of Digital Transmission

- Heavy signal processing

- Synchronization is crucial
- Transmission bandwidth is large
- When the S/N ratio drops below a certain value, the quality of service can change suddenly from very good to very bad

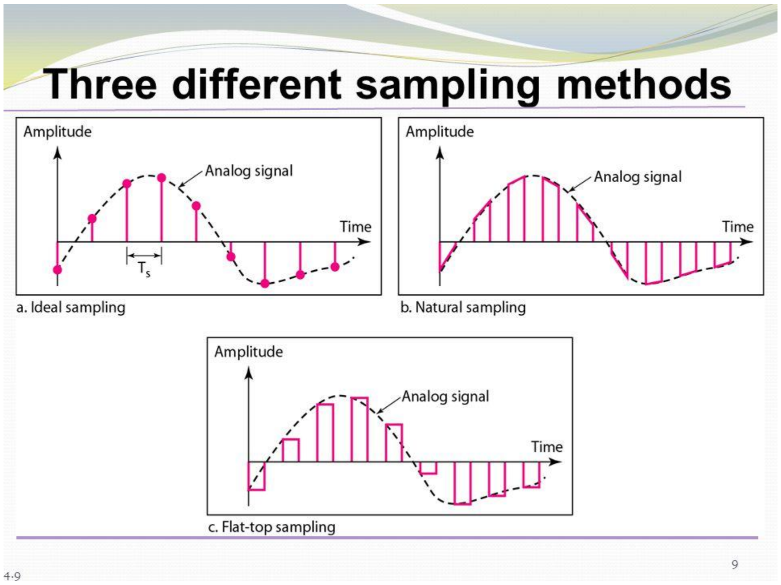
The digital communication system, which we will consider in the remainder of the course, is shown in the figure below:


Analog Signal
t
Bit sequence
0001101110……
Source
A-D Conversion
Digital Baseband Modulation
Digital Bandpass Modulation
t
t
Baseband Channel
Bandpass Channel
User
D-A Conversion
Digital Baseband Demodulation
Digital Bandpass Demodulation

Sampling

- A signal is characterized by its frequency content, which is described by the Fourier transform.
- Recall from the signal analysis part the relationship of the Fourier transform to the power spectral density and the signal bandwidth.
- The rate, at which the signal must be sampled, is very much dependent on the bandwidth of the signal.
- We will consider three methods for sampling a signal

Three Types of Sampling



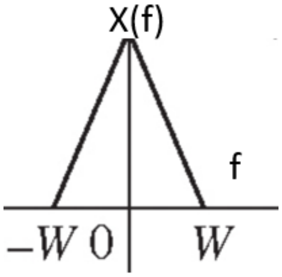
Three different sampling methods

a. Ideal sampling

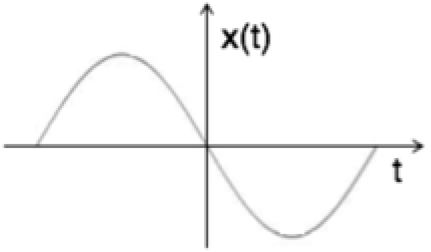
b. Natural sampling

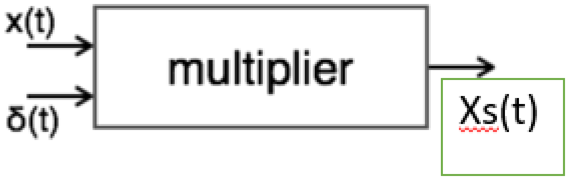
c. Flat-top sampling

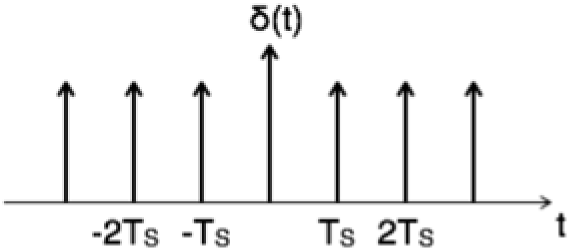
Ideal Sampling: The message, x(t), with Fourier transform X(f), which is assumed to be band-limited to W Hz, is multiplied by a periodic sequence of ideal impulses with period T_s to produce the sampled signal $x_s(t)$.

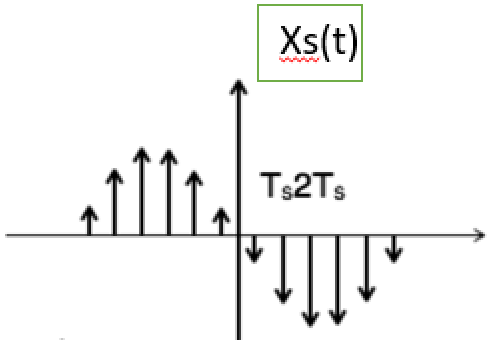


This operation is illustrated in the schematic diagram







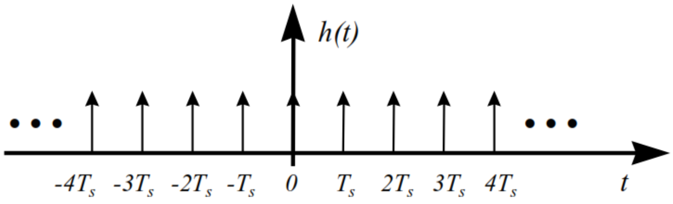


Mathematically, the sampled version, $x_s(t)$, of signal $x(t)$ is:

$$x_s(t) = h(t)\cdot x(t) \Leftrightarrow X_s(f) = H(f) * X(f),$$

$$h(t) = \sum_{k=-\infty}^{\infty}\delta\left(t - kT_s\right) = \frac{1}{T_s}\sum_{k=-\infty}^{\infty} e^{j2\pi k\frac{t}{T_s}}$$

Sampling function



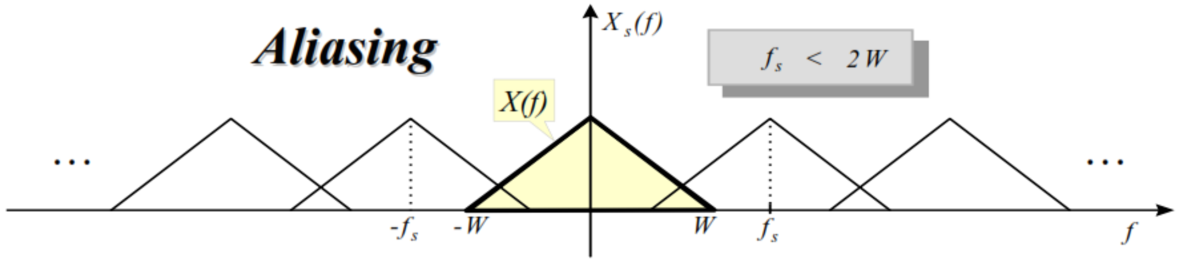
Note that $h(t)$ is a periodic function and, as such, can be represented by a complex Fourier series. Taking the Fourier transform, we get

$$H(f) = \Im\left\{\frac{1}{T_s}\sum_{K=-\infty}^{\infty} e^{j2\frac{\pi kt}{T_s}}\right\} = \frac{1}{T_s}\sum_{k=-\infty}^{\infty}\delta\left(f - \frac{k}{T_s}\right).$$

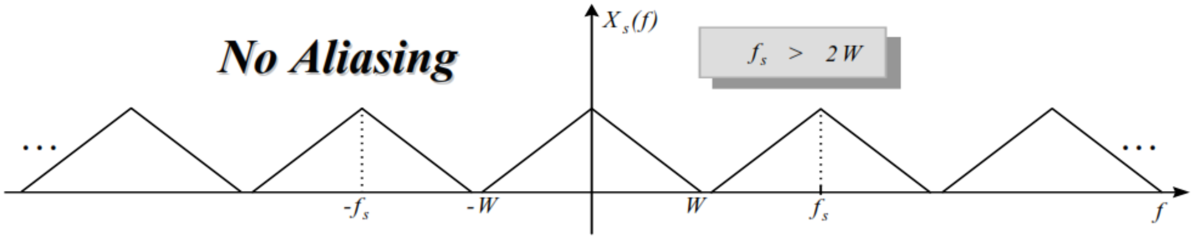
Then:

$$X_s(f) = H(f) * X(f) = \frac{1}{T_s}\sum_{K=-\infty}^{\infty} X\left(f - \frac{k}{T_s}\right).$$

Depending on the sampling frequency f_s, the spectrum of the sampled signal $X_s(f)$ may take any of the forms:



When $f_s < 2W$, using a low pass filter with bandwidth W to recover the message x(t) results in a distortion in the recovered signal. This type of distortion is called "**Aliasing**".



(a)

If f_s>2W, the original signal *x(t)* can be obtained from $x_s(t)$ ***through simple low-pass filtering***. In the frequency domain, we have

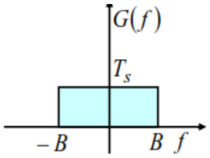
$$X(f) = X_s(f) \cdot G(f),$$

where

$$G(f) = \begin{cases} T_s, & |f| \le B \\ 0, & \text{oherwise.} \end{cases} \quad \text{for} \quad W \le B \le f_s - W.$$

The impulse response of the low-pass filter, *g(t)*, is then

$$g(t) = \Im^{-1}[G(f)] = \int_{-B}^{B} G(f) \cdot e^{j2\pi ft} df = 2BT_s \frac{\sin(2\pi Bt)}{2\pi Bt}.$$



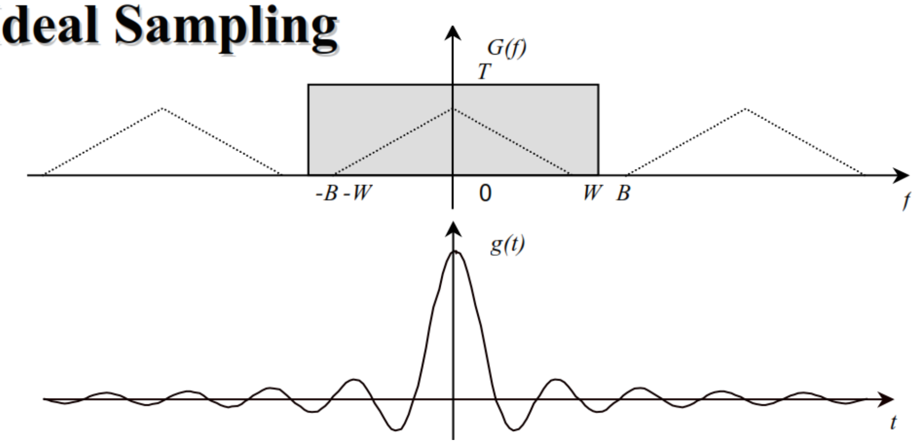
From the convolution property of the Fourier transform we have:

$$x(t) = \int_{-\infty}^{\infty} x_s(a) \cdot g(t-a) da = \sum_k x(kT_s) \cdot \int_{-\infty}^{\infty} \delta(a - kT_s) g(t-a) da = \sum_k x(kT_s) \cdot g(t - kT_s).$$

Thus, we have the following *interpolation formula*

$$\boxed{x(t) = \sum_k x(kT_s) \cdot g(t - kT_s)}$$

Ideal Sampling



The Sampling Theorem:

A bandlimited signal with no spectral components above W $\mathrm{H_z}$ can be recovered uniquely from its samples taken every T_s seconds, provided that

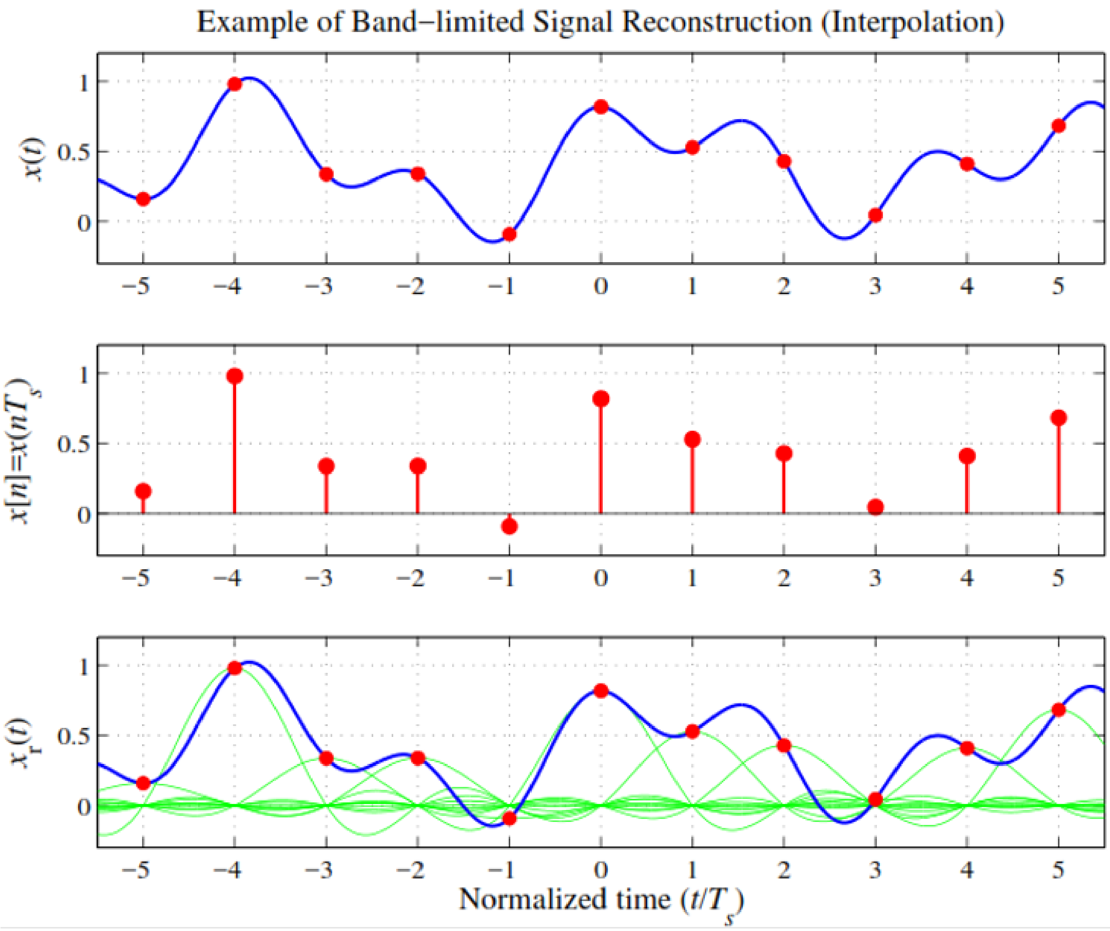
$$T_s \le \frac{1}{2W}, \quad \text{or, equivalently,} \quad f_s \ge 2W.$$ ← ***Nyquist Rate***

Extraction of *x(t)* from its samples can be done by passing the sampled signal through a low-pass filter. Mathematically, *x(t)* can be expressed in terms of its samples by:

$$x(t) = \sum_k x(kT_s)\cdot g(t-kT_s)$$

The sampling frequency $f_s = 2W,$ is called the **Nyquist sampling rate**. It represents the minimum rate at which a signal with bandwidth W must be sampled in order to reconstruct it from its samples without distortion.

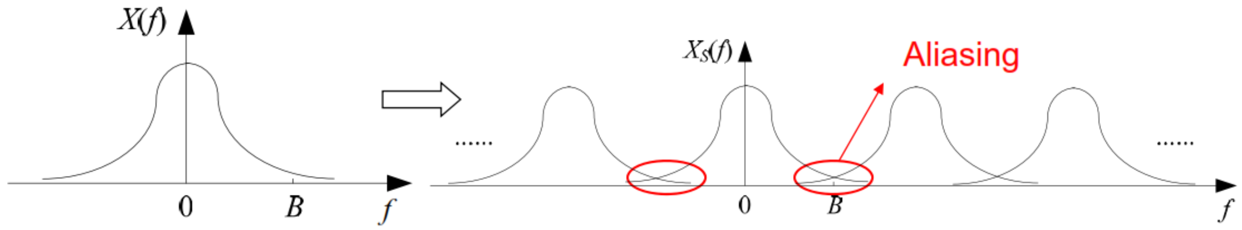
The sampling and reconstruction of signals is shown in the figure below



A

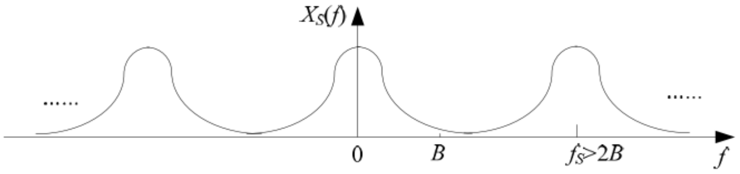
Practical Consideration:

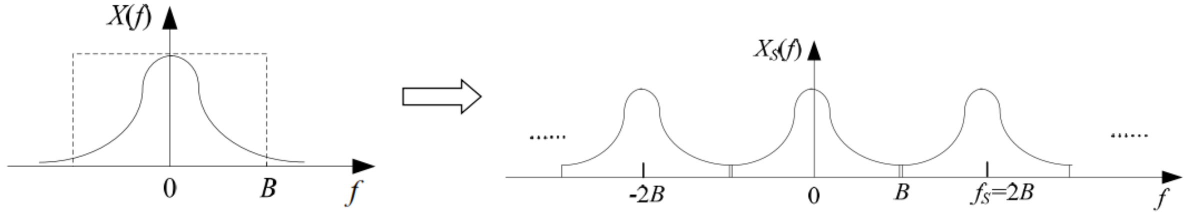
If the signal under consideration is not bandlimited, then we pass it through a pre-filter with bandwidth W so that the truncated signal can be sampled without aliasing. This is illustrated in the following figure:



- Use a sampling rate $f_S > 2B$

- Pre-filtering



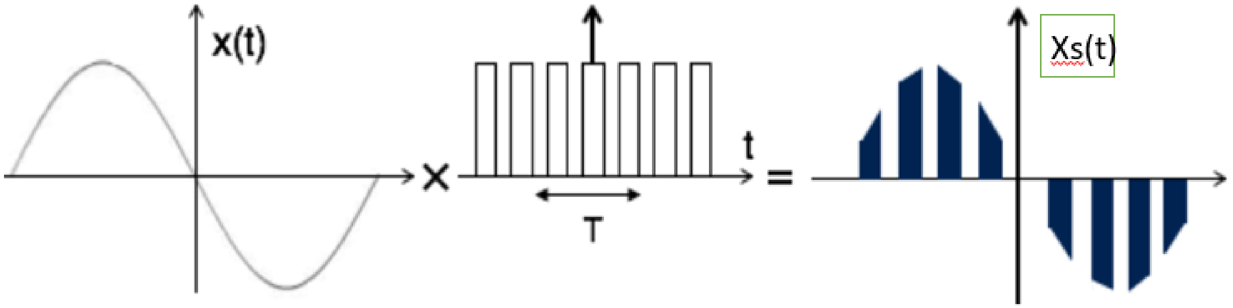


Natural Sampling:

The band-limited signal x(t), with bandwidth W Hz, is multiplied by a periodic sequence of rectangular pulses, $g_p(t)$, with a duty cycle (τ/T_s)

$$x_s(t) = x(t)g_p(t)$$

This sampling method is demonstrated in the next figure



Expanding $g_p(t)$ in Fourier series, we get

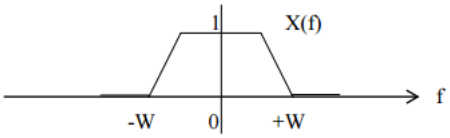
$$x_s(t) = x(t)[C_0 + 2C_1\cos(2\pi f_s t) + 2C_2\cos(2\pi(2f_s)t) + 2C_3\cos(2\pi(3f_s)t) + \cdots]$$

Taking the Fourier transform, and simplifying, we get:

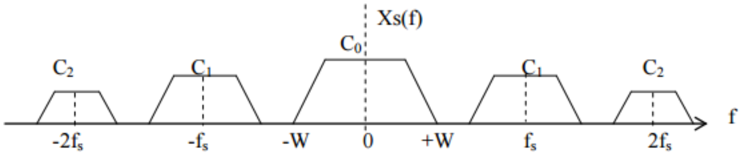
$$X_s(f) = C_0X(f) + C_1X(f - f_s) + C_1X(f + f_s) + C_2X(f - 2f_s) + C_2X(f + 2f_s) + \cdots$$

As can be seen, the spectrum of the naturally sampled signal looks quite similar to that of the ideally sampled signal. The only difference is that the higher order frequency terms are attenuated by a factor resulting from the Fourier series expansion of the periodic pulse train. When $x(t)$ is sampled at a rate higher than or equal to the Nuquist rate, it can be recovered by passing $x_s(t)$ through a low pass filter with bandwidth W. The filter output is

$$\boldsymbol{y(t) = C_0x(t)}$$



Message Signal Spectrum



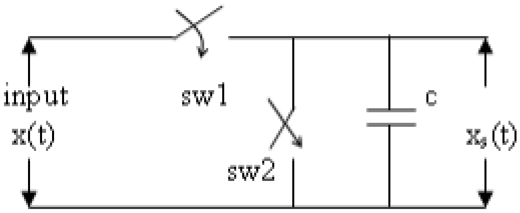
Sampled Signal Spectrum (fs > 2W)

Flat-Topped Sampling (Sample and Hold)

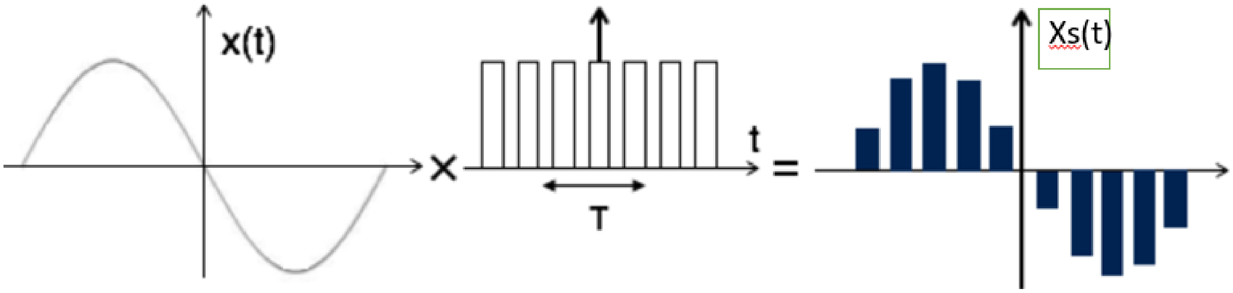
Let p(t) be a basic unit amplitude pulse defined as:

$$p(t) = \begin{cases} 1 & 0 < t < \tau, \\ 0 & otherwise \end{cases}$$

In flat-topped sampling, the sampler generates a sequence of equally spaced rectangular pulses whose amplitudes are proportional to the message signal $x(t)$ at the sampling times $x(kT_s)$. A circuit that performs flat-topped sampling is shown below:



This sampling technique is modeled in the next figure



The sampled signal is represented as

$$x_s(t) = \sum_{-\infty}^{\infty} x(kT_s)p(t - kT_s)$$

Using the identity, $\boldsymbol{p(t) * \delta(t - kT_s) = p(t - kT_s)}$, $x_s(t)$ can be expressed as

$$x_s(t) = p(t) * \sum_{-\infty}^{\infty} x(kT_s)\delta(t - kT_s)$$

Taking the Fourier transform and recognizing that the second term corresponds to an ideally sampled sequence, we get

$$X_s(f) = P(f)\frac{1}{T_s}\sum_{-\infty}^{\infty} X(f - kf_s)$$

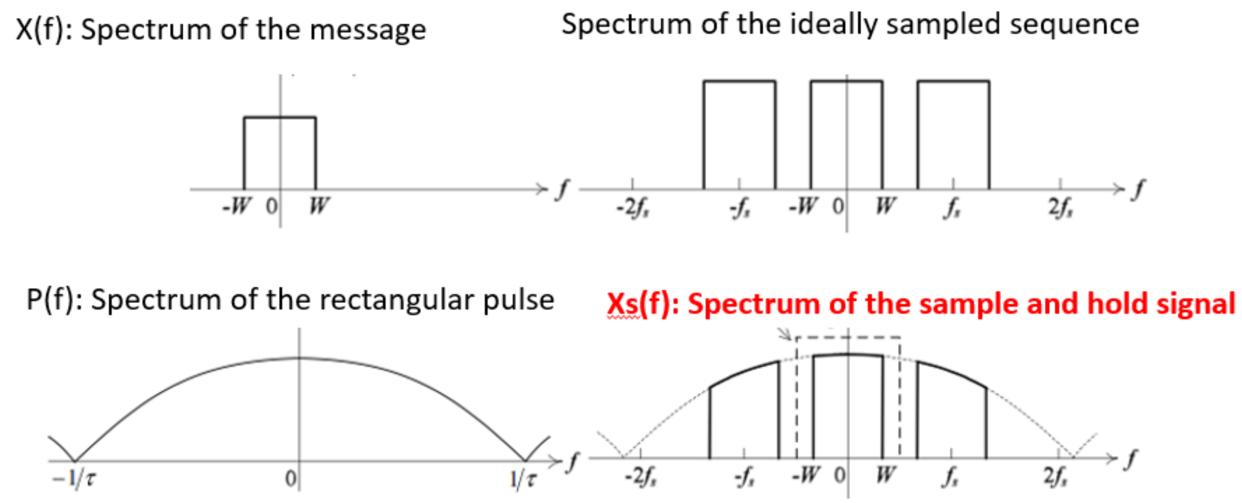
where $P(f) = \tau sinc(f\tau)e^{-j\pi f\tau}$ is the Fourier transform of $p(t)$.

Here we observe that the spectrum of the flat-topped sampled signal corresponds to the spectrum of the ideally sampled signal multiplied by the Fourier transform of the rectangular pulse ($\tau sinc(f\tau)$).

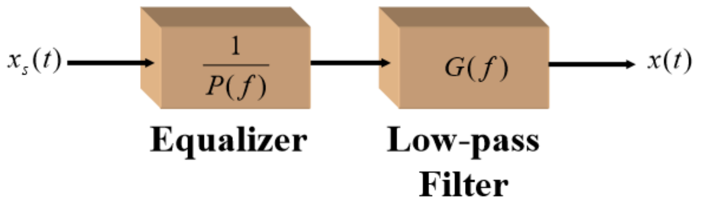
When $x_s(t)$ is passed through a low pass filter with bandwidth W, the output is

$$\boldsymbol{Y(f) = \frac{1}{T_s}P(f)\, X(f)}$$

The spectrum of the sample and hold signal is demonstrated in the next figure



A distortion-free signal can be obtained by using an equalizing filter whose transfer function is the reciprocal of that of the unit pulse; i.e., $H_E(f) = 1/P(f)$. The process of equalization is demonstrated in the figure below.



Reconstruction is possible but an equalizer may be needed

Analog Pulse Modulation

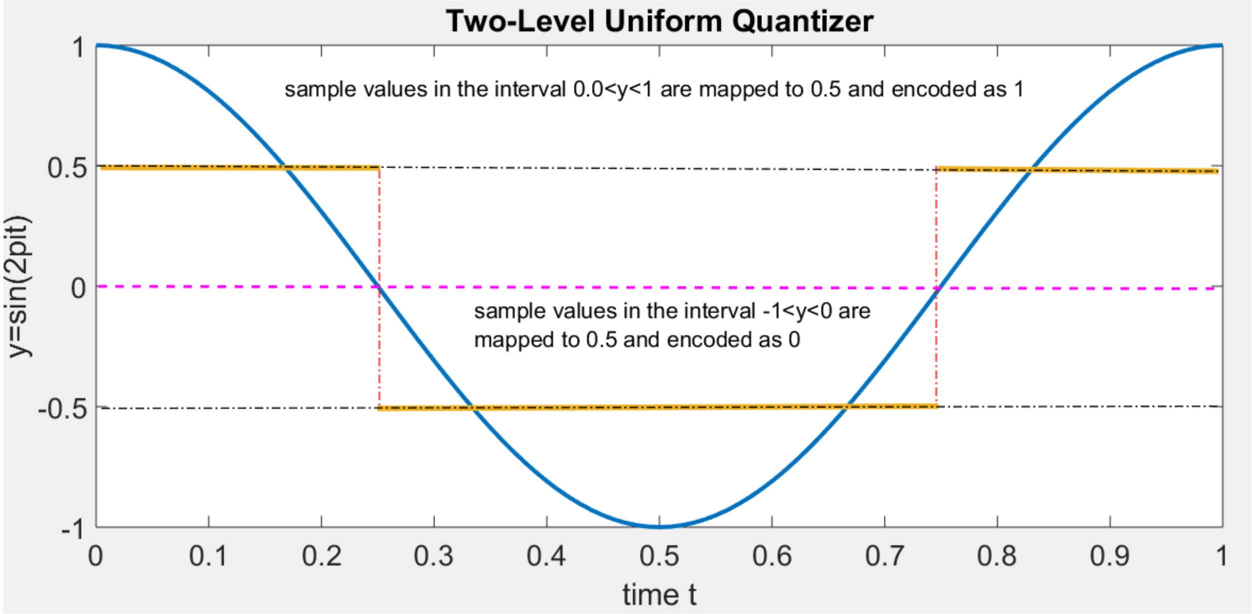
To be given to ENEE 3306 students from another set of course slides.

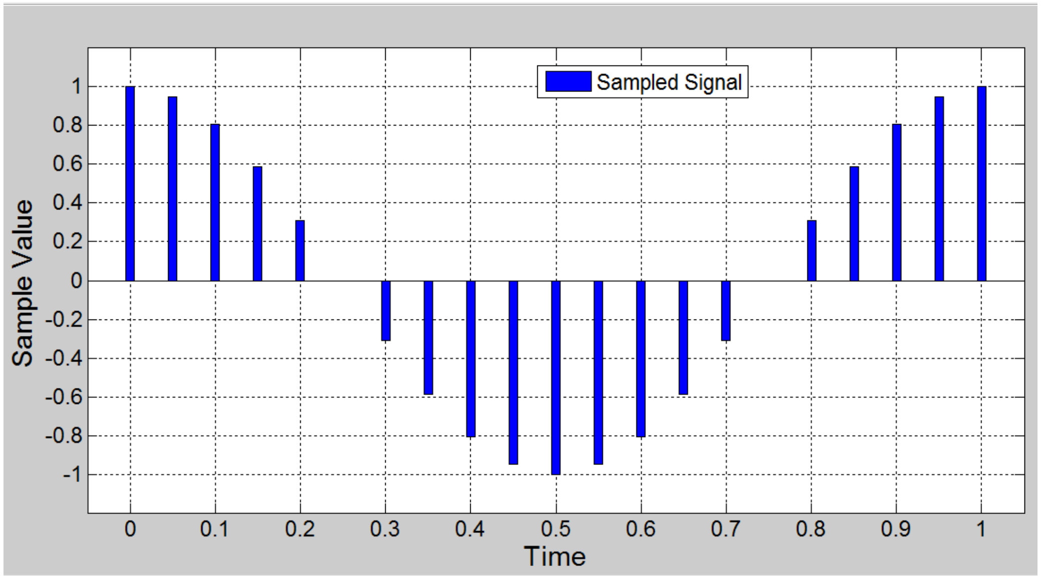
Quantization

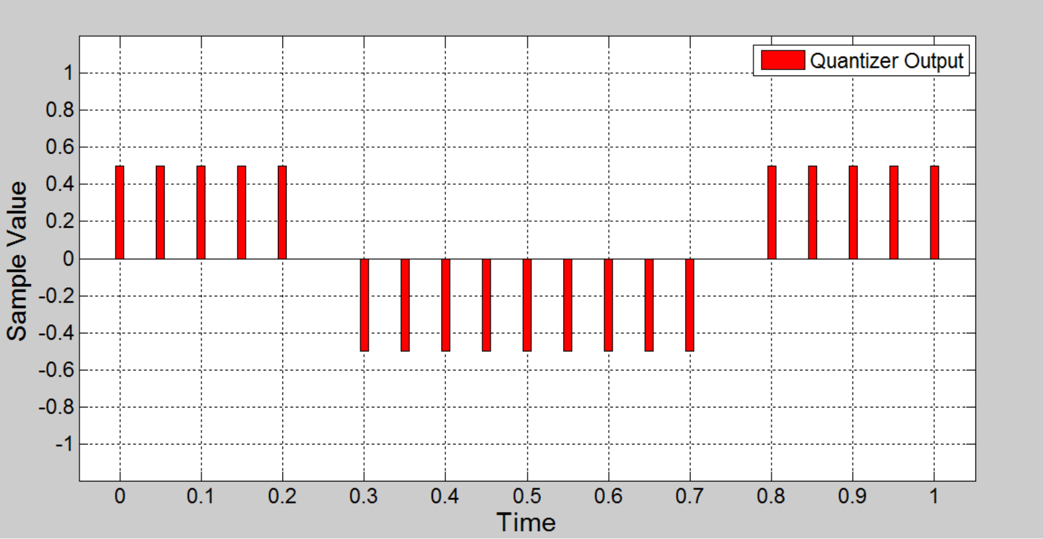
A motivation Example on a One-bit Quantizer: The signal $\mathrm{x}(\mathrm{t}) = \cos(2\pi t)$ is uniformly sampled at a rate of 20 samples per second. The samples are applied to a sign detector, whose input-output characteristic is defined as:

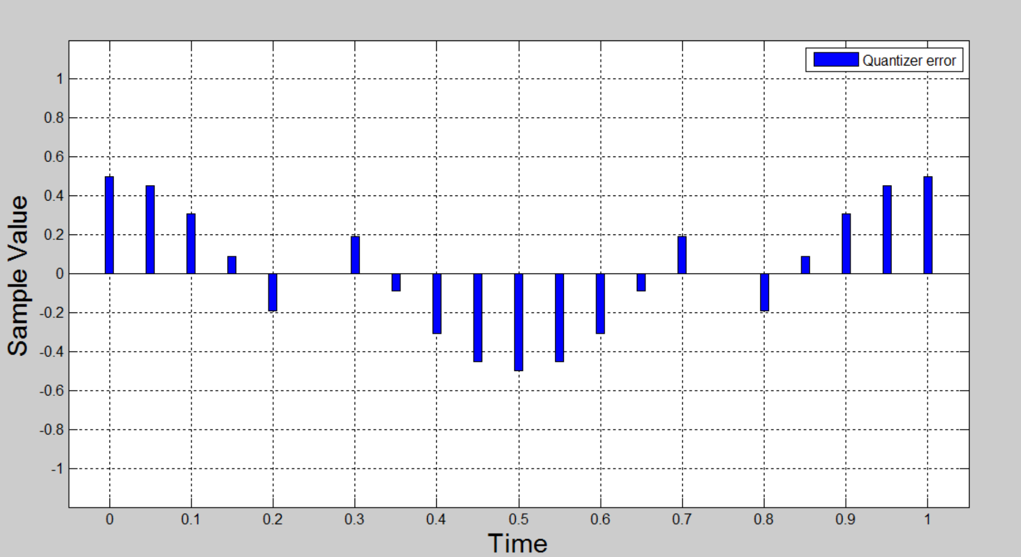
$$y(t) = \begin{cases} 0.5, & 0 < x < 1 \\ -0.5, & -1 < x < 0 \end{cases}$$

The next figures depict the quantizer operation, the input samples to the sign detector, the quantized output, and the quantization error defined as $e = (x - y)$. This is an example of what we will call a one-bit quantizer. Here, the input samples are mapped either to +0.5 or to -0.5 depending on their sign.




Sampled Signal
Sample Value
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Time

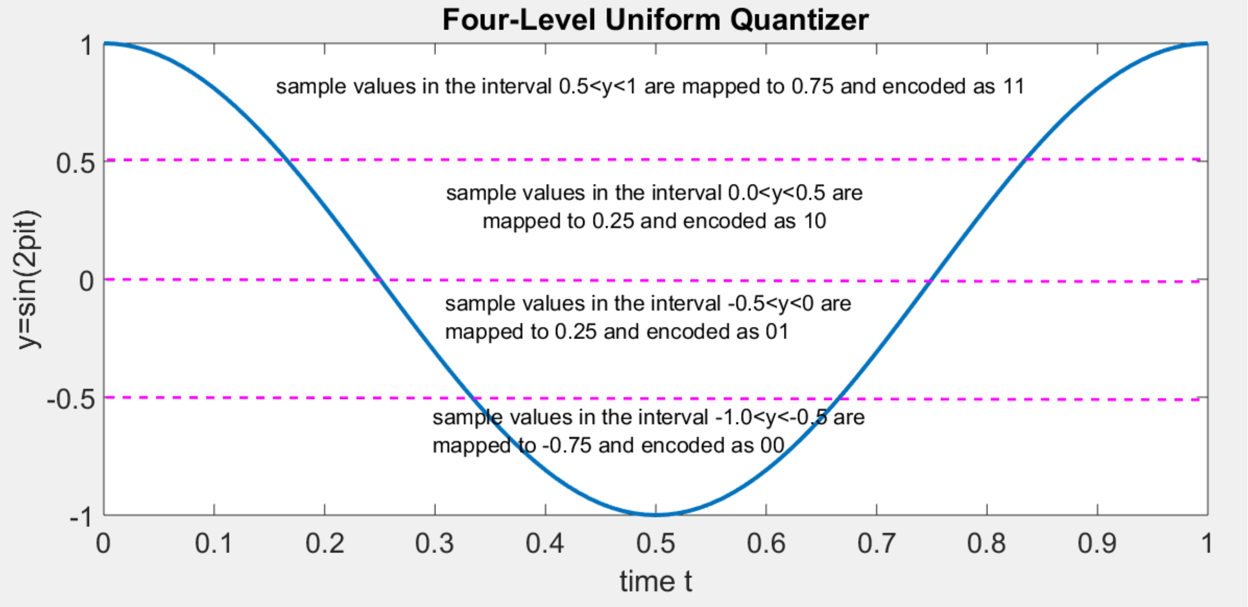

Quantizer Output
Sample Value
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Time

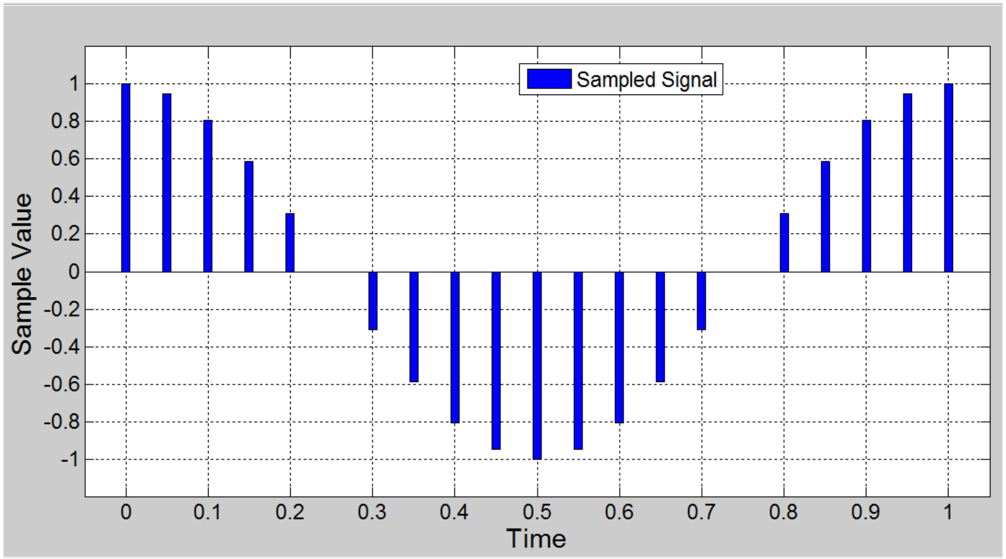

Quantizer error
Sample Value
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Time

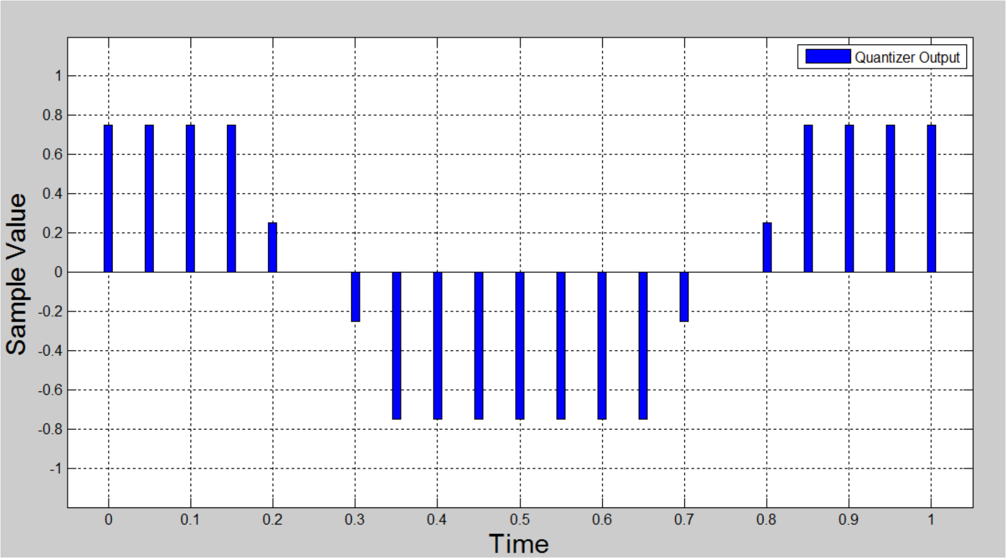
Another Motivating Example on a Two-bit Quantizer: The signal $\mathrm{x(t)} = \cos(2\pi t)$ is sampled uniformly at a rate of 20 samples per second. The samples are applied to a four-level uniform quantizer with input-output characteristic

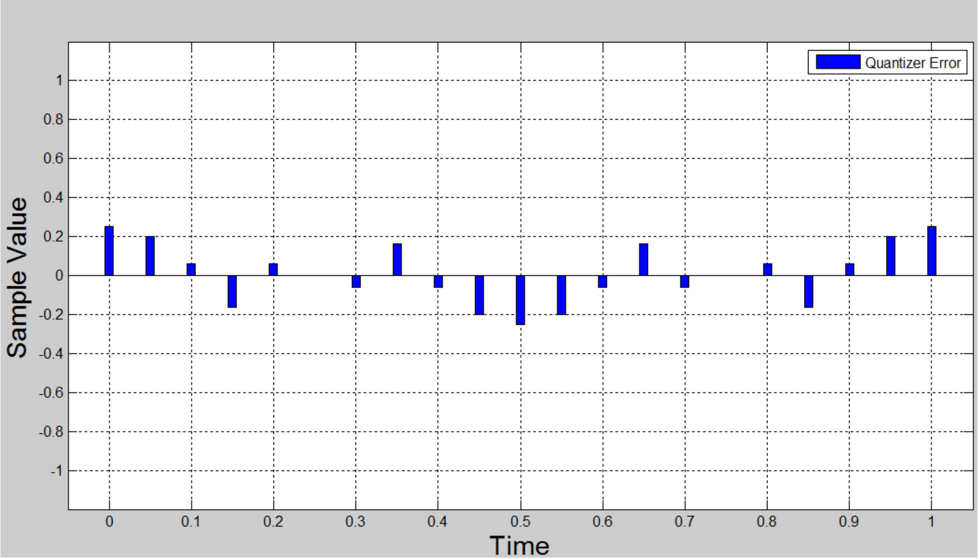
$$y(t) = \begin{cases} 0.75, & 0.5 < x < 1 \\ 0.25, & 0 < x < 0.5 \\ -0.25, & -0.5 < x < 0 \\ -0.75, & -1 < x < -0.5 \end{cases}$$

The next figures depict the input signal, the quantization regions, the sample values, the quantized samples and the error defined as $e = (x - y)$.

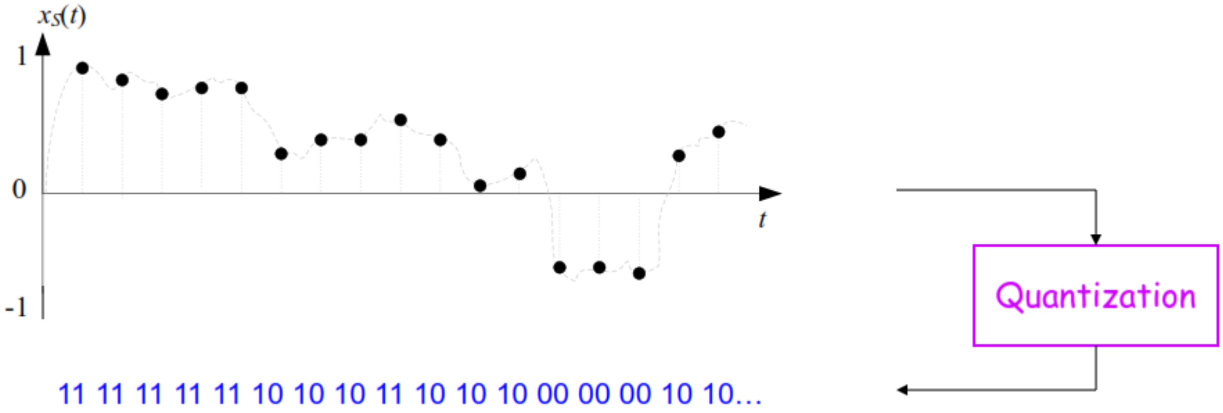



Sampled Signal
Sample Value
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Time


Quantizer Output
Sample Value
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Time


Quantizer Error
Sample Value
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Time

Another example that illustrates the four-level quantization (with a dynamic range -1, 1) and encoding of samples is shown in the figure below:

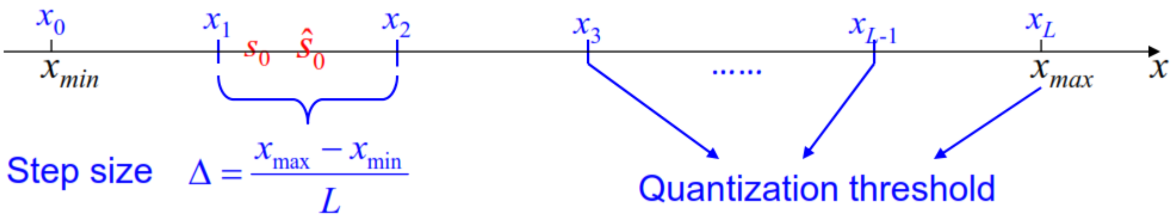


Now, we define quantization in a formal way.

- Quantization is defined as the process of converting the continuous sample amplitude (infinite in number) $x(kT_s)$ of a message signal into a discrete amplitude $\hat{x}(kT_s)$ taken from a finite set of L possible values:

$$\hat{x} = \{\widehat{x_1}, \widehat{x_2}, \dots, \widehat{x_L}\}$$

- When the spacing between the adjacent levels is made small, $\hat{x}(kT_s)$ can be made practically indistinguishable from $x(kT_s)$.



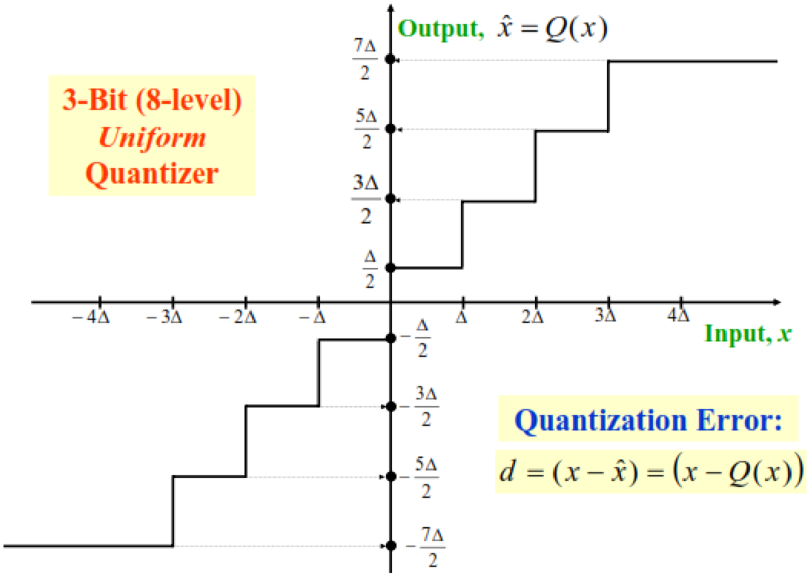
- There is always a loss of information associated with the quantization process. Therefore, it is not possible to completely recover the sampled signal from the quantized signal.
- The amplitude range of the input signal is partitioned into L intervals such that if $x(kT_s) \in R_i$, the quantizer output will be a level $\widehat{x_i} = \{\widehat{x_1}, \widehat{x_2}, \dots, \widehat{x_L}\}$
- The boundary points separating adjacent regions are called decision levels or threshold levels.

- The qauntizer output is called a representation or reconstruction level
- The spacing between representation levels is called the step size.

The Uniform Quantizer

A quantizer is called uniform when the L regions are of equal length Δ and the spacing between representation levels is uniform and equals to Δ.

The input-output characteristic of a uniform quantizer (midrise type) is shown below for L=8.



If the amplitude of the signal $x(t)$ varies between $-x_{max} < x < x_{max}$, then

$$\boldsymbol{\Delta = \frac{2x_{max}}{L}}$$

Exercise:

a. Design an 8-level uniform quantizer with a dynamic range of (-4, +4) V. Here, you need to specify the thresholds and the representation values.
b. How many binary digits are needed to represent the samples
c. Find the representation value and the quantization error when a 1.64 V sample is applied to the quantizer.
d. Find the binary representation corresponding to the sample -2.1 V.

SQNR of a Uniform Scalar Quantizers

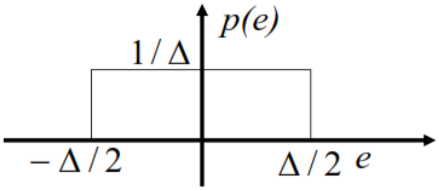
The quantization error per sample is the difference between the input and output of the quantizer, i.e.,

$$e = (x - \hat{x})$$

The maximum error (also referred to as the resolution) = $|\frac{\Delta}{2}|$

When Δ is small, the error, e, is assumed to be a uniform random variable over the interval $-\Delta|2 < e < \Delta|2$.

$$\left[-\frac{\Delta}{2},\frac{\Delta}{2}\right]$$



The average quantization error (distortion) over all samples of the signal is

$$D = E(x - \hat{x})^2 = E(e^2)$$
$$D = \frac{1}{\Delta}\int_{-\Delta/2}^{+\Delta/2}(e)^2\, de$$

$$\mathbf{D = \Delta^2|12}$$

Remark: Note that D depends on the design of the quantizer and not on the signal applied to it, as we will see in the next two examples.

Example: Signal Matches Dynamic Range of a Uniform Quantizer

Let the sinusoidal signal $x(t) = Acos(2\pi f_0 t)$ be applied to a uniform quantizer with a dynamic range $(-A, A)$. We need to find the $SQNR$.

Solution:

The average power, P_x , in $x(t)$ is: $P_x = A^2|2$

The signal power to quantization noise ratio, $SQNR = \frac{A^2/2}{\Delta^2/12}$

Here, $\Delta = 2\mathrm{A}/\mathrm{L}$. If $\mathrm{L} = 2^{\mathrm{n}}$, then the $SQNR$ becomes

$$SQNR = \frac{3}{2}\mathbf{L}^2 = \frac{3}{2}2^{2\mathrm{n}}$$

In dB, the $SQNR$, becomes

$$SQNR = 10log\frac{P_x}{D} = 6.02n + 1.76$$

- $SQNR$ increases exponentially with the number n of bits per sample.
- There is a 6-dB improvement in $SQNR$ for each bit added to represent the the continuous sample values.

Example: Weak Signal Applied to a Uniform Quantizer

Now, let the sinusoidal signal $x(t) = A/2cos(2\pi f_0 t)$ be applied to the same uniform quantizer of the previous example, with a dynamic range $(-A, A)$. We need to find the $SQNR$.

Solution:

$$\boldsymbol{SQNR = \frac{(A/2)^2/2}{\Delta^2/12} = \frac{12}{32}\mathrm{L}^2 = \frac{12}{32}2^{2n}}$$

In dB, the $SQNR$, becomes

$$SQNR = 10log\frac{P_x}{D} = 6.02n - 4.77$$

Remark: If the message signal $x(t)$ is a random signal, with an amplitude probability density function $f_X(x)$ and zero mean $(E(X) = 0)$, then the SQNR is given as

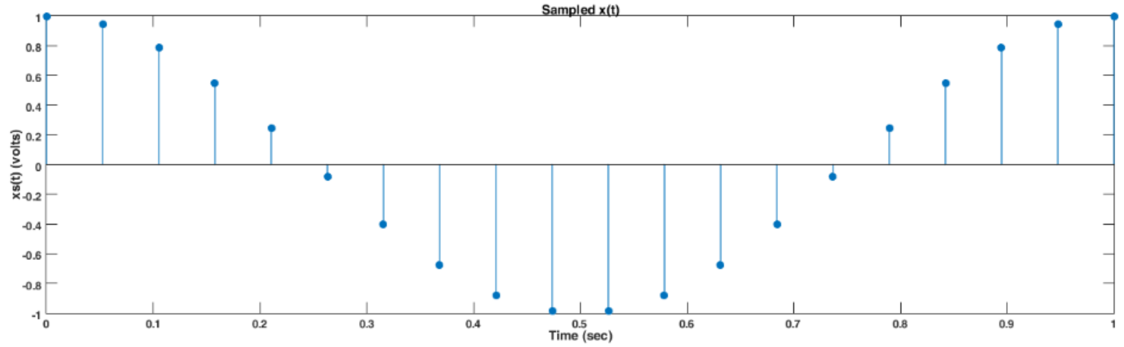
$$\boldsymbol{SQNR = \frac{E(X^2)}{E(x-\hat{x})^2} = \frac{\int_{-\infty}^{\infty} X^2\, f_X(x)dx}{E(x-\hat{x})^2}}$$

Example:

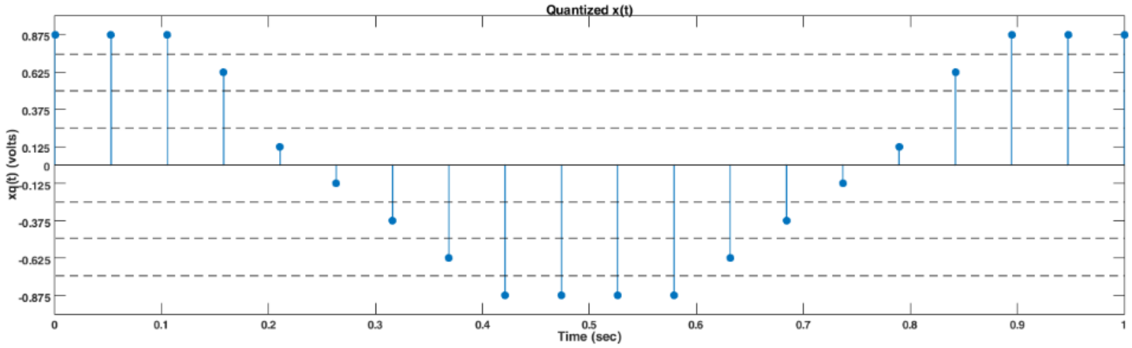
The signal x(t) = cos($2\pi t$) is uniformly sampled at a rate of 20 samples/sec. The samples are applied to an 8-level uniform quantizer with a dynamic range (-1, 1) V and a step size of 0.25 V.

a. Plot the sampled signal over one cycle of the message.
b. Plot the quantizer output over one cycle of the message.
c. Repeat Part b if the signal applied to the quantizer is g(t) = 0.25 cos($2\pi t$).
d. Comment on the results of Parts b and c.

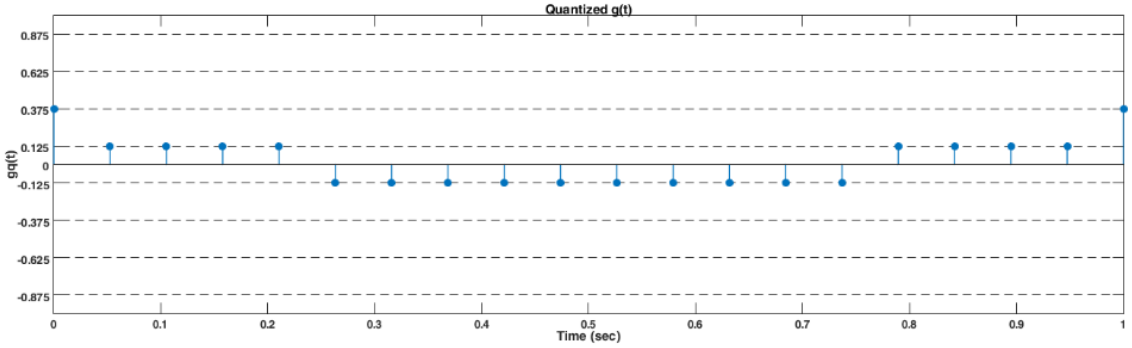
a) Sampled signal



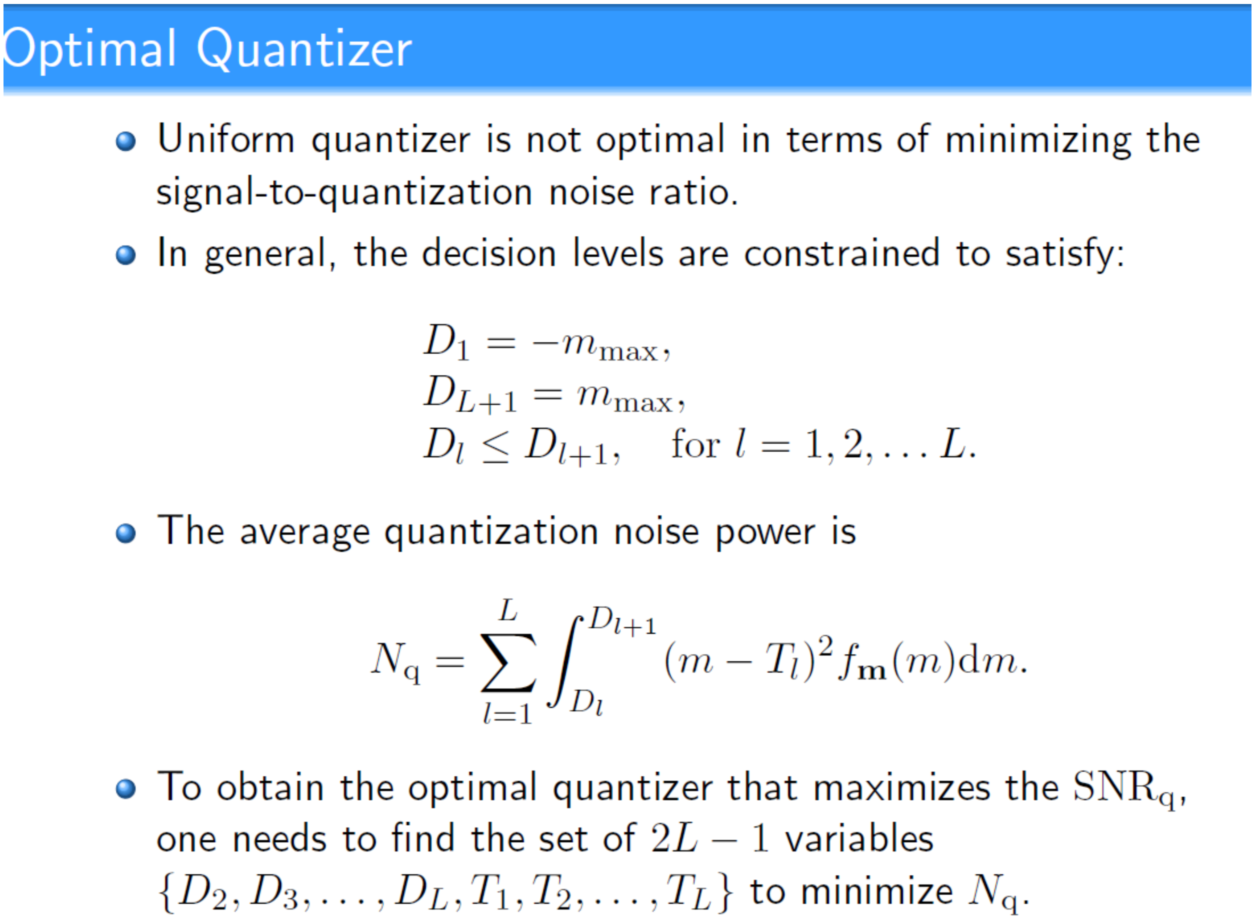
b) Quantized signal



c) Quantized g(t)



The Optimal Quantizer (Not required for ENEE 339)



Optimal Quantizer

- Uniform quantizer is not optimal in terms of minimizing the signal-to-quantization noise ratio.
- In general, the decision levels are constrained to satisfy:

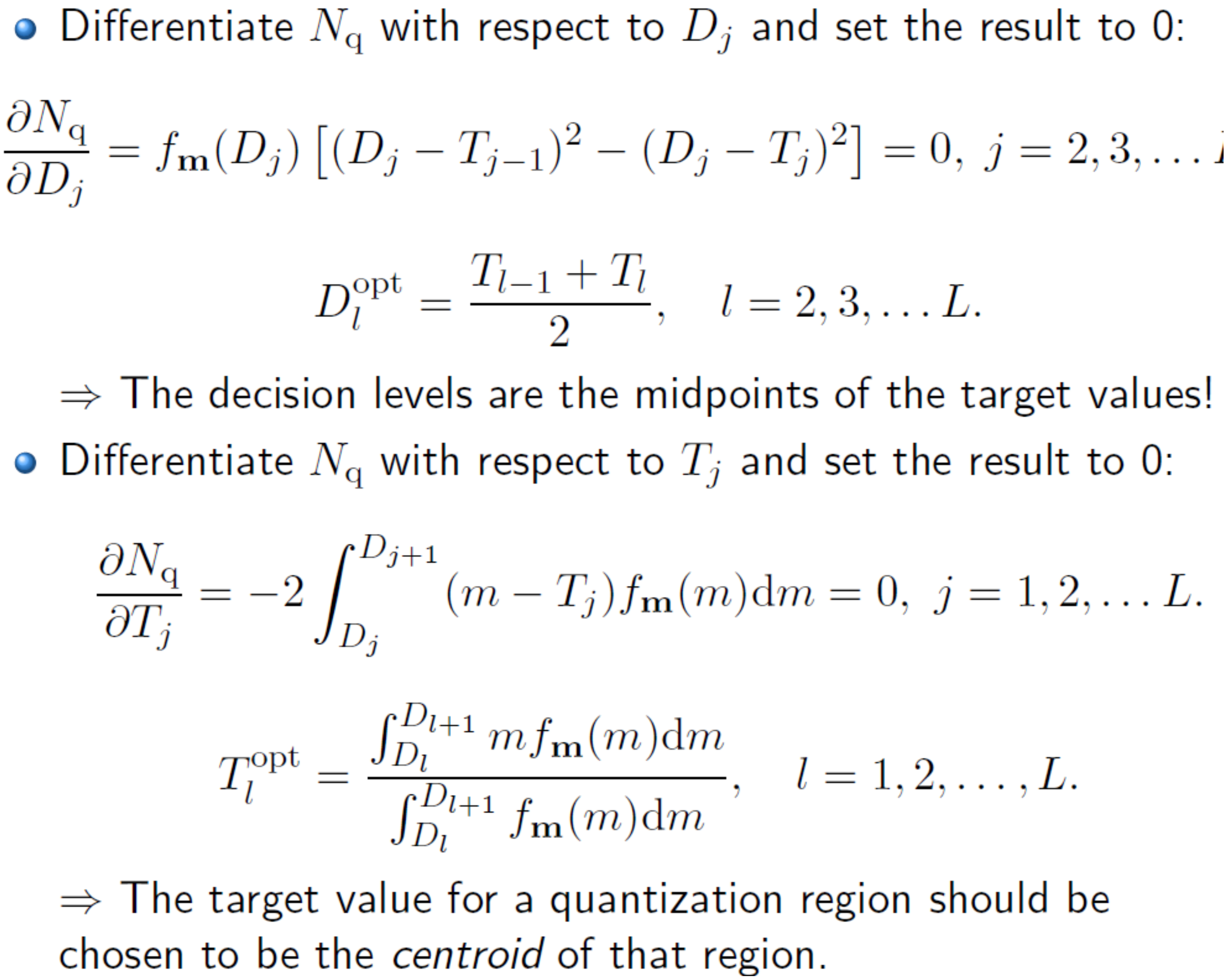
$$\begin{aligned} &D_1 = -m_{\max}, \\ &D_{L+1} = m_{\max}, \\ &D_l \leq D_{l+1}, \quad \text{for } l = 1, 2, \ldots L. \end{aligned}$$

- The average quantization noise power is

$$N_{\mathrm{q}} = \sum_{l=1}^{L} \int_{D_l}^{D_{l+1}} (m - T_l)^2 f_{\mathbf{m}}(m) \mathrm{d}m.$$

- To obtain the optimal quantizer that maximizes the $\mathrm{SNR}_{\mathrm{q}}$, one needs to find the set of $2L - 1$ variables $\{D_2, D_3, \ldots, D_L, T_1, T_2, \ldots, T_L\}$ to minimize N_{q}.

The Optimal Quantizer (continued)



- Differentiate N_{q} with respect to D_j and set the result to 0:

$$\frac{\partial N_{\mathrm{q}}}{\partial D_j} = f_{\mathbf{m}}(D_j)\left[(D_j - T_{j-1})^2 - (D_j - T_j)^2\right] = 0,\ j = 2, 3, \dots$$

$$D_l^{\mathrm{opt}} = \frac{T_{l-1} + T_l}{2}, \quad l = 2, 3, \dots L.$$

⇒ The decision levels are the midpoints of the target values!

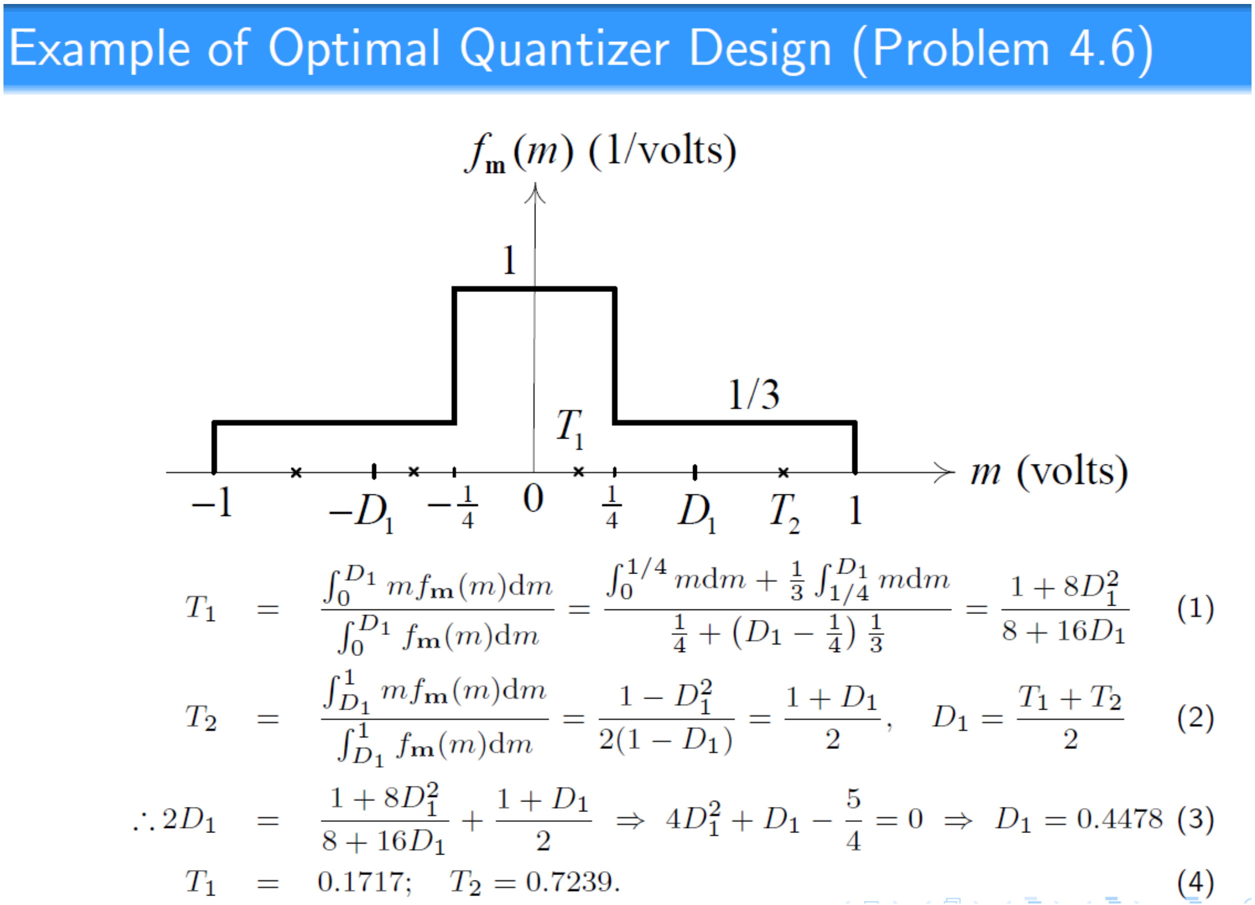
- Differentiate N_{q} with respect to T_j and set the result to 0:

$$\frac{\partial N_{\mathrm{q}}}{\partial T_j} = -2\int_{D_j}^{D_{j+1}} (m - T_j) f_{\mathbf{m}}(m)\mathrm{d}m = 0,\ j = 1, 2, \dots L.$$

$$T_l^{\mathrm{opt}} = \frac{\int_{D_l}^{D_{l+1}} m f_{\mathbf{m}}(m)\mathrm{d}m}{\int_{D_l}^{D_{l+1}} f_{\mathbf{m}}(m)\mathrm{d}m}, \quad l = 1, 2, \dots, L.$$

⇒ The target value for a quantization region should be chosen to be the *centroid* of that region.

Example on the The Optimal Quantizer



Example of Optimal Quantizer Design (Problem 4.6)

$$\begin{aligned}
T_1 &= \frac{\int_0^{D_1} m f_{\mathbf{m}}(m)\mathrm{d}m}{\int_0^{D_1} f_{\mathbf{m}}(m)\mathrm{d}m} = \frac{\int_0^{1/4} m\mathrm{d}m + \frac{1}{3}\int_{1/4}^{D_1} m\mathrm{d}m}{\frac{1}{4} + \left(D_1 - \frac{1}{4}\right)\frac{1}{3}} = \frac{1+8D_1^2}{8+16D_1} && (1)\\
T_2 &= \frac{\int_{D_1}^{1} m f_{\mathbf{m}}(m)\mathrm{d}m}{\int_{D_1}^{1} f_{\mathbf{m}}(m)\mathrm{d}m} = \frac{1-D_1^2}{2(1-D_1)} = \frac{1+D_1}{2}, \quad D_1 = \frac{T_1+T_2}{2} && (2)\\
\therefore 2D_1 &= \frac{1+8D_1^2}{8+16D_1} + \frac{1+D_1}{2} \Rightarrow 4D_1^2 + D_1 - \frac{5}{4} = 0 \Rightarrow D_1 = 0.4478 && (3)\\
T_1 &= 0.1717; \quad T_2 = 0.7239. && (4)
\end{aligned}$$

Lloyd-Max Conditions and Iterative Algorithm

$$D_l^{\text{opt}} = \frac{T_{l-1} + T_l}{2}, \quad (5) \qquad T_l^{\text{opt}} = \frac{\int_{D_l}^{D_{l+1}} m f_{\mathbf{m}}(m)\mathrm{d}m}{\int_{D_l}^{D_{l+1}} f_{\mathbf{m}}(m)\mathrm{d}m}. \quad (6)$$

$$l = 2, 3, \dots L$$

1. Start by specifying an arbitrary set of decision levels (for example the set that results in equal-length regions) and then find the target values using (6).
2. Determine the new decision levels using (5).
3. The two steps are iterated until the parameters do not change significantly from one step to the next.

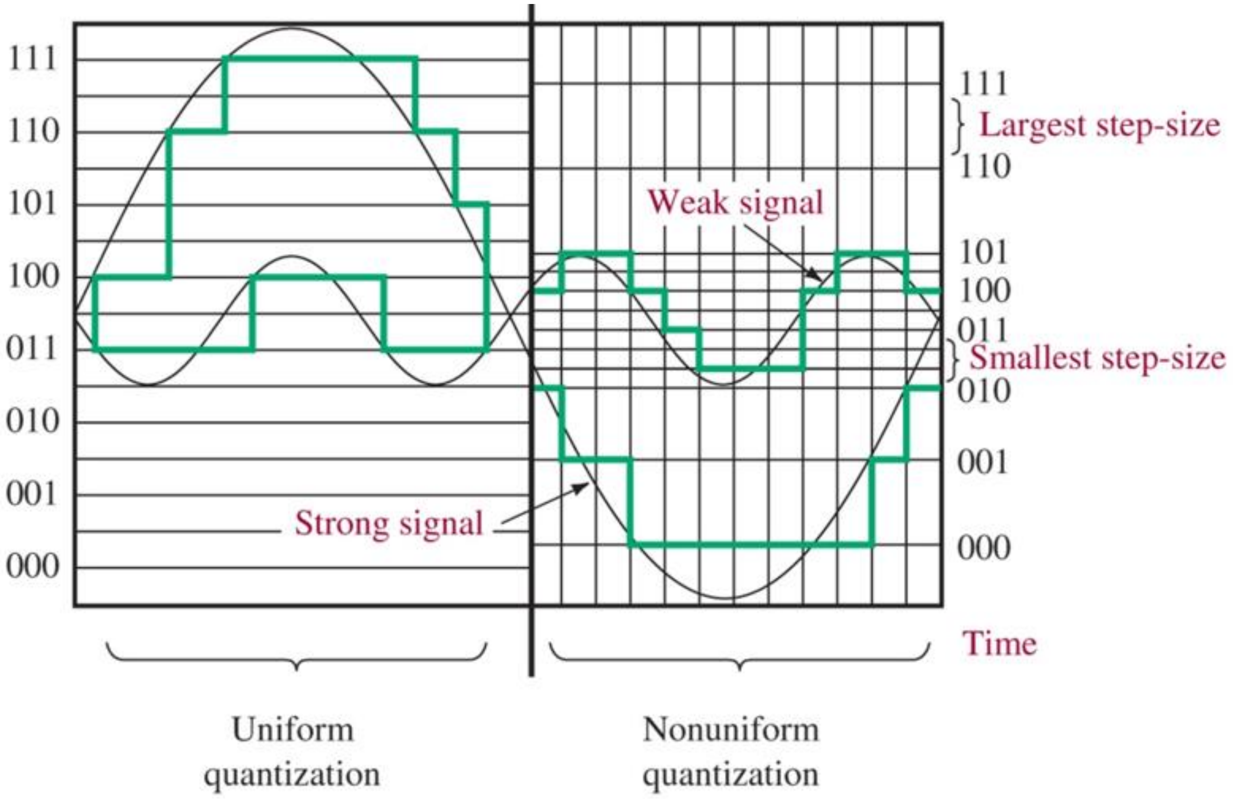
The optimal quantizer needs to know pdf $f_{\mathbf{m}}(m)$ and is designed for a specific $m_{\max}$ $\Rightarrow$ Prefer quantization methods that are robust to source statistics and changes in the signal's power level.

Robust (Non-uniform) Quantization

- We have seen earlier that when a weak signal and a strong signal are applied to the uniform quantizer, the SQNR of the stronger signal is larger than that of the weaker one.
- Maximum SQNR is achieved when the signal strength matches the dynamic range of the quantizer.
- Small amplitudes are more susceptible to quantization noise than large amplitudes.
- The uniform quantizer is easy to build; however, it is not optimal when the input signal is weak most of the time. Here, the signal does not use the entire set of quantization levels available.
- In practical applications, especially speech signals, small signal amplitudes occur more often than large signal amplitudes. This means that for the same signal, small amplitudes will be subject to distortion more than large amplitudes.
- The question that we need to answer is whether it is possible to find a quantizer, which can provide a **SQNR that is somewhat constant and independent of the signal strength and the probability distribution of the input.**

Non-uniform Quantization

A Non-uniform quantizer uses quantization levels of variable spacing, denser at small signal amplitudes, broader at large amplitudes. The next figure shows a weak and a strong signal applied to a uniform and a non-uniform quantizer. As we can see, the non-uniform quantizer represents both signals quite adequately.



Example: Consider the signal, $x(t)$, whose amplitude varies over the range (-4, 4) V.

a. If the signal is applied to a 3-bit uniform quantizer with a dynamic range (-4, 4) V, find the thresholds and representation values. Also, find the representation levels and quantization errors when the sample values given in the table below are applied to the quantizer.
b. The signal is now applied to a non-uniform quantizer with the following design parameters
    1. Threshold values (-4, -0.9882, -0.2353, -0.0471, 0, 0.0471, 0.2353, 0.9882, 4)
    2. Representation values (-1.9922, - 0.4863,- 0.1098, -0.0157 , 0.0157, 0.1098, 0.4863, 1.9922)

c. Find the representation levels and quantization error when the set of samples in the table below are applied to the quantizer of Part b.

Solution:

The design parameters of the uniform quantizer are

Thresholds: -4, -3, -2, -1, 0, 1, 2, 3, 4

Representation Values: -3.5, -2.5, -1.5, -0.5, 0.5, 1.5, 2.5, 3.5

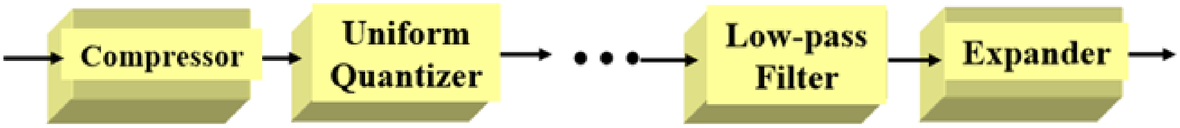
Sample Value	**Uniform Quantizer**		**Non-uniform Quantizer**	
	Output	**% Error**	**Output**	**% Error**
0.01	0.5	$\frac{0.01-0.5}{0.01}100\% = \mathbf{4900\%}$	0.0157	$\frac{0.01-0.0157}{0.01}100\% = \mathbf{57\%}$
0.05	0.5	$\frac{0.05-0.5}{0.05}100\% = \mathbf{900\%}$	0.1098	$\frac{0.05-0.1098}{0.05}100 = \mathbf{119.6\%}$
0.07	0.5	$\frac{0.07-0.5}{0.07}100\% = \mathbf{707\%}$	0.1098	$\frac{0.07-0.1098}{0.07}100\% = \mathbf{56.8\%}$
0.1	0.5	$\frac{0.1-0.5}{0.1}100\% = \mathbf{400\%}$	0.1098	$\frac{0.1-0.1098}{0.1}100\% = \mathbf{9.8\%}$
0.15	0.5	$\frac{0.15-0.5}{0.15}100\% = \mathbf{233\%}$	0.1098	$\frac{0.15-0.1098}{0.15}100\% = \mathbf{26.8\%}$
0.20	0.5	$\frac{0.2-0.5}{0.2}100\% = \mathbf{150\%}$	0.1098	$\frac{0.2-0.1098}{0.2}100\% = \mathbf{45.1\%}$
0.30	0.5	$\frac{0.3-0.5}{0.3}100\% = \mathbf{66.6\%}$	0.4863	$\frac{0.3-0.4863}{0.3}100\% = \mathbf{62.1\%}$
0.35	0.5	$\frac{0.35-0.5}{0.35}100\% = \mathbf{42.8\%}$	0.4863	$\frac{0.35-0.4863}{0.35}100\% = \mathbf{38.9\%}$
0.4	0.5	$\frac{0.4-0.5}{0.4}100\% = \mathbf{25\%}$	0.4862	$\frac{0.4-0.4863}{0.4}100\% = \mathbf{21.5\%}$
0.9	0.5	$\frac{0.9-0.5}{0.9}100\% = \mathbf{44.4\%}$	0.4863	$\frac{0.9-0.4863}{0.9}100\% = \mathbf{45.9\%}$
1.7	1.5	$\frac{1.7-1.5}{1.7}100\% = \mathbf{11.76\%}$	1.9922	$\frac{1.7-1.9922}{1.7}100\% = \mathbf{17.18\%}$
2.9	2.5	$\frac{2.9-2.5}{2.9}100\% = \mathbf{13.79\%}$	1.9922	$\frac{2.9-1.9922}{2.9}100\% = \mathbf{31.3\%}$
3.8	3.5	$\frac{3.8-3.5}{3.8}100\% = \mathbf{7.89\%}$	1.9922	$\frac{3.8-1.9922}{3.8}100\% = \mathbf{47.5\%}$

As we can see from the table, the relative error of small samples for the non-uniform quantizer is much less than that of the uniform one. However, for larger samples, the uniform quantizer performs better. In speech signals, low-level samples have higher probability of occurrence than strong level samples. Therefore, if a speech signal is applied to a non-uniform quantizer, the average quantization noise over all samples will be smaller than that corresponding to the uniform quantizer.

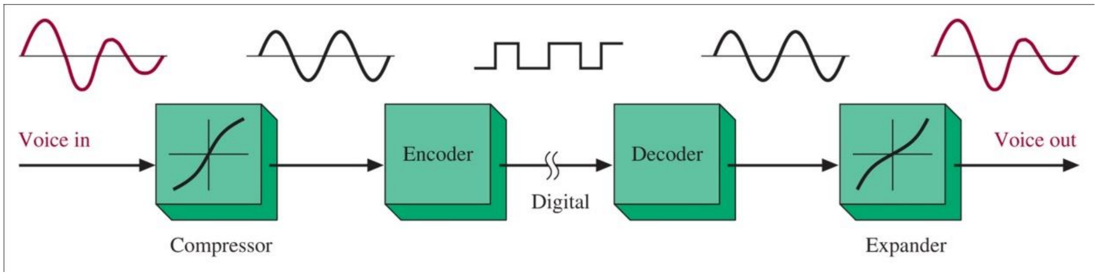
Practical Implementation of a Robust Non-uniform Quantizer:
We will use a type of nonuniform quantizers, called **companding (compansion), t**hat does not require knowledge of the pdf of the signal to be quantized, and yields an almost uniform SQNR over a wide range of signal variations.

Companding:

- The process of pre-distorting the signal at the transmitter is known as (signal) **compression**. At the receiver, this process is reversed to remove distortion and is known as (signal) **expansion.** The two operations together, are typically, referred to as **companding** (or compansion).
- The compressor is a nonlinear operation that amplifies weak signal values more than it amplifies large signal values, thus stretching the signal over more representation levels. This will enhance weak signal levels and improve their SQNR. Large signal levels will suffer more quantization distortion, but the overall effect on the signal is an improvement in SQNR.
- Since the probability of smaller amplitudes is higher than the larger amplitudes, the overall result is an improvement.
- In North America, μ -law companding (with $\mu = 255$) is the standard.
- In summary, companding is performed as follows:
  - Compress the signal using the μ-law. The output is approximately uniformly distributed.
  - Apply the compressed sample to a uniform quantizer
  - Transmit the quantized sample to the receiver.
  - Apply the received sample to the expander. The output is the desired signal value.



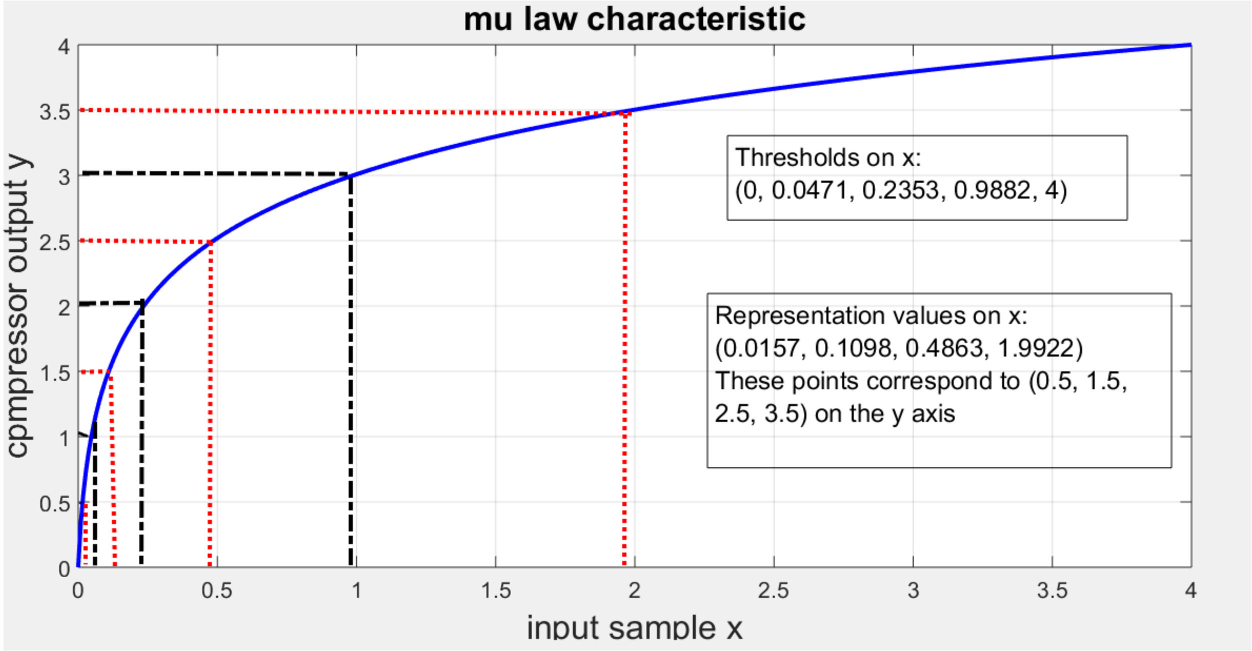
Signal Flow in a Companding System



The input-output characteristic of the compressor is given by

$$y = y_{max} \frac{\ln\left[1 + \mu\left(\frac{|x|}{x_{max}}\right)\right]}{\ln(1 + \mu)} sgn(x)$$

This characteristic, when $\mu = 255$, $x_{max} = y_{max} = 4$, is plotted in the next figure. Also shown in the figure are the uniform quantizer thresholds (on the y-axis) and the corresponding non-uniform quantizer thresholds (on the x-axis).

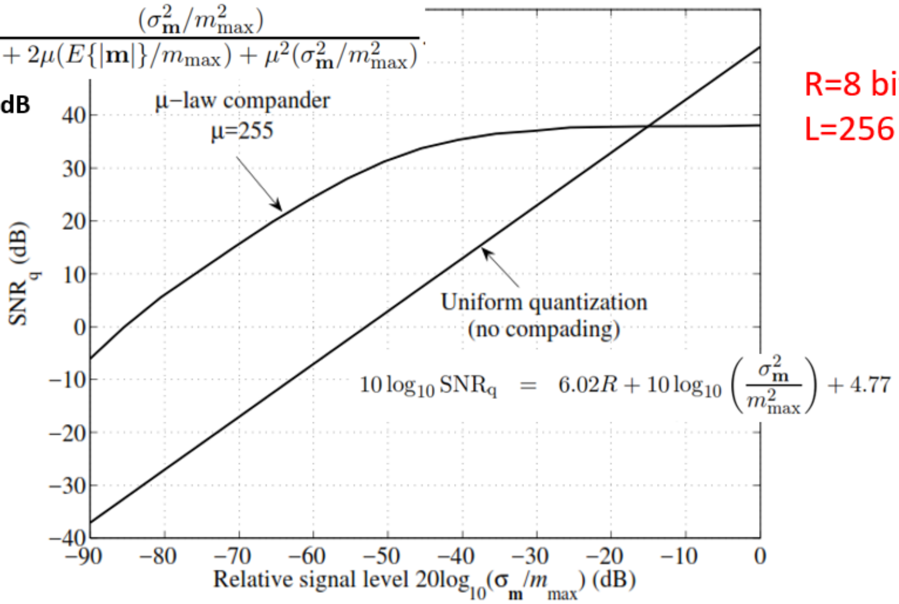


Exercise: Use the μ law characteristic to find the representation values on the x-axis corresponding to the representation values (0.5, 1.5, 2.5, 3.5) on the y-axis.

The next figure shows the NQNR for both an 8-bit uniform and a robust quantizer when a Gaussian-distributed message is applied. As we can observe, the robust quantizer performs better than the uniform for most of signal levels (except when the signal is too strong).

$$\mathrm{SNR_q} = \frac{\sigma^2_{\mathbf{m}}}{N_\mathrm{q}} = \frac{3L^2\mu^2}{\ln^2(1+\mu)} \frac{(\sigma^2_{\mathbf{m}}/m^2_{\max})}{1+2\mu(E\{|\mathbf{m}|\}/m_{\max}) + \mu^2(\sigma^2_{\mathbf{m}}/m^2_{\max})}.$$

$$\boldsymbol{SNR_q = 10log_{10}(SNR_q)}\ \textbf{dB}$$

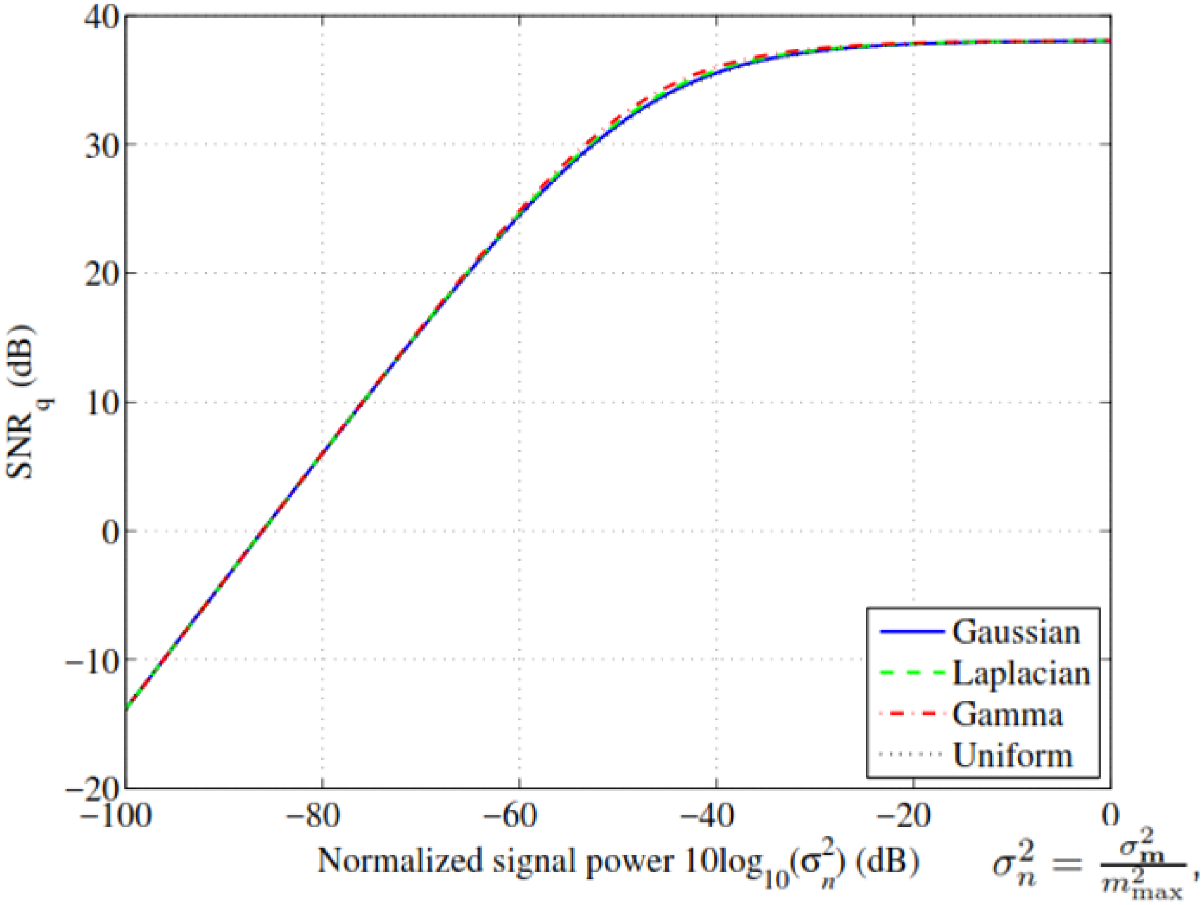


$$10\log_{10}\mathrm{SNR_q} \;=\; 6.02R + 10\log_{10}\left(\frac{\sigma^2_{\mathbf{m}}}{m^2_{\max}}\right) + 4.77$$

R=8 bits
L=256

(Robustness against input signal level)

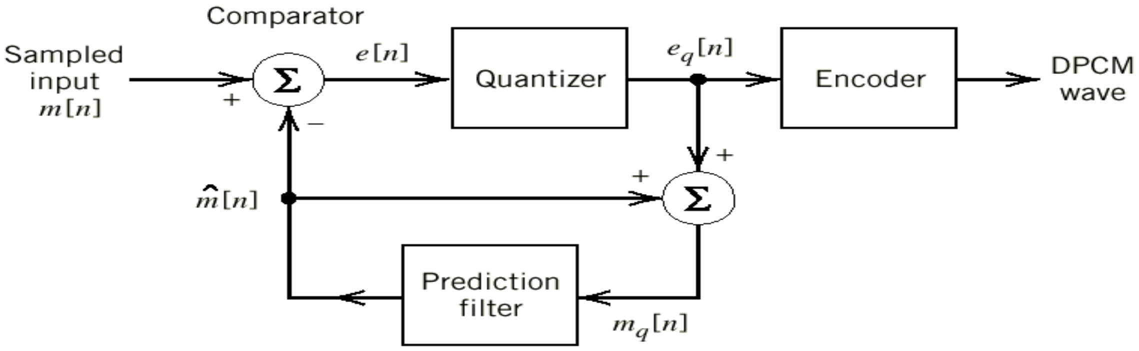
The μ-law quantizer is also robust against the actual probability density function of the input signal, as is illustrated in the next figure. In this figure, the SQNR of an -8-level μ-law quantizer is plotted versus signal power for a number of input pdf's.



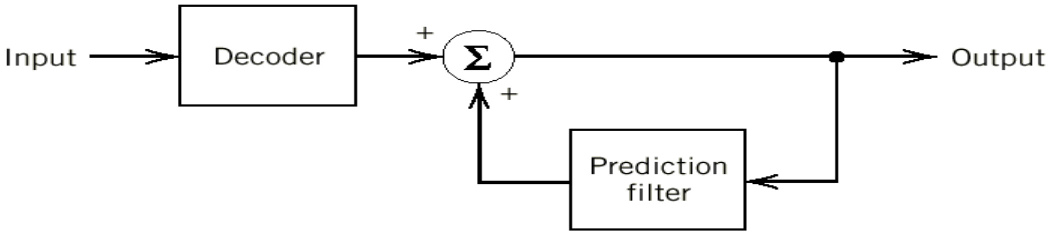
(Robustness against input signal probability distribution)

Differential Pulse Code Modulation (DPCM)

- The quantizers, that we studied so far, are memoryless, in the sense that quantization is done on a sample-by-sample basis. Each sample is quantized and encoded into n binary digits, regardless of any correlation with other samples.
- A ***differential pulse-code modulation (DPCM)*** *quantizer* quantizes the difference between a sample and a predicted value of that sample. Here, correlation between successive samples is utilized.
- The prediction is based, in general, on past m samples of the signal. If successive samples are highly correlated, the predictor output will be very close to the next sample value, and hence the prediction error will be small.
- An error with a small variance further means that **fewer bits (r<n) are needed to represent the error**.
- At the receiver, a predictor similar to the one used at the transmitter is used to reconstruct the original waveform



(a)



(b)

Linear Prediction Filter

It is a discrete-time, finite-duration impulse response filter (FIR), which consists of three blocks:

1. Set of p (p: prediction order) unit-delay elements (z^{-1})

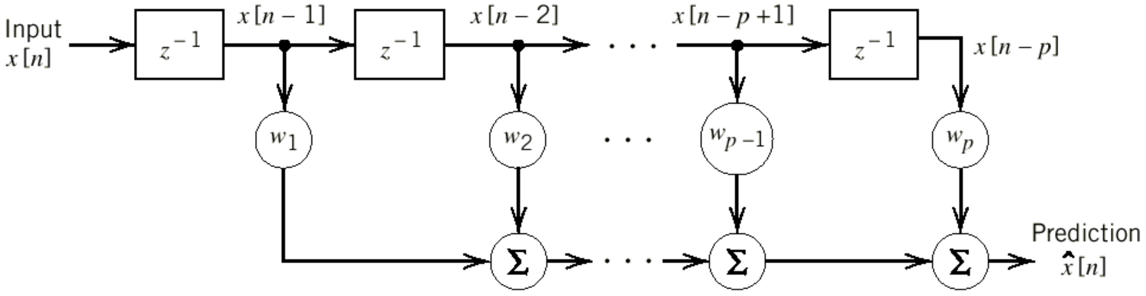
2. Set of multipliers with coefficients $w_1, w_2, \ldots w_p$

3. Set of adders ($\sum$)

This filter expresses the predicted value of the sample at time (nT_s) as a linear combination of the past p samples of the signal.

$$\hat{x}(n) = w_1 x(n-1) + w_2 x(n-2) + \cdots + w_p x(n-p)$$

The coefficients w_1, w_2, …, w_p are chosen so as to minimize the mean square error $E(x(n) - \hat{x}(n))^2$.



Let

$$\epsilon = E((x(n) - \hat{x}(n))^2; \qquad \text{prediction error}$$

Substituting $\hat{x}(n) = w_1 x(n-1) + w_2 x(n-2) + \cdots + w_p x(n-p)$, the prediction error becomes:

$$\epsilon = E((x(n) - w_1 x(n-1) + w_2 x(n-2) + \cdots + w_p x(n-p))^2$$

Expanding ϵ and taking expectation of all terms, we get:

$$\epsilon = E(x(n)^2) - 2\sum_{i=1}^{p} w_i E[x(n)x(n-i)] + \sum_{i=1}^{p}\sum_{j=1}^{p} w_i w_j \; E[x(n-i)x(n-j)]$$

Recognize that: $\boldsymbol{R_x(i) = E[x(n)x(n-i)]}$ is the autocorrelation function of $x(t)$.

Differentiating ϵ with respect to w_i, setting the derivative to zero, and solving, we get (assuming p=3)

$$\begin{bmatrix} R_x(0) & R_x(1) & R_x(2) \\ R_x(-1) & R_x(0) & R_x(1) \\ R_x(-2) & R_x(-1) & R_x(0) \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} = \begin{bmatrix} R_x(1) \\ R_x(2) \\ R_x(3) \end{bmatrix}$$

If p=1, the above equation reduces to

$$R_x(0) = w_1 R_x(1)$$

Note that: $R_x(-1) = R_x(1)$, $R_x(-2) = R_x(2)$, $R_x(1) = R_x(T_s)$, $R_x(2) = R_x(2T_s)$, $R_x(3) = R_x(3T_s)$.

Delta Modulation (DM)

Delta modulation (DM), is a special case of the DPCM where the order of the prediction filter p=1. Here, the quantized previous sample represents the prediction of the current sample. In this scheme, the system transmits the sign of the difference between the current and previous samples. When the input is larger than the prediction, the transmitter sends digit 1 and when the input is smaller than the prediction it sends digit 0. Therefore, the system requires one bit.

Let $$m[n] = m(nT_s), n = 0, \pm 1, \pm 2, \ldots$$

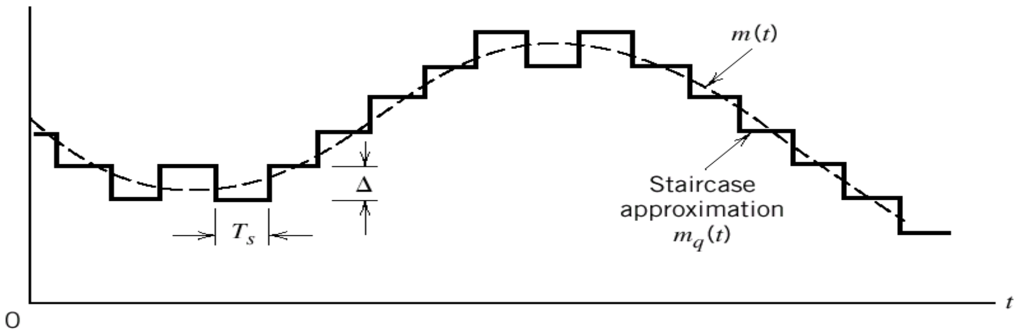
where, T_s is the sampling period and $m(nT_s)$ is a sample of $m(t)$. The error signal is

$$e[n] = m[n] - m_q[n-1]; \qquad \text{(this is an analog signal)}$$

$$e_q[n] = \Delta sgn(e[n]); \qquad \text{(this is the quantized error)}$$

$$m_q[n] = m_q[n-1] + e_q[n]; \quad \text{(the predicted signal at time n)}$$

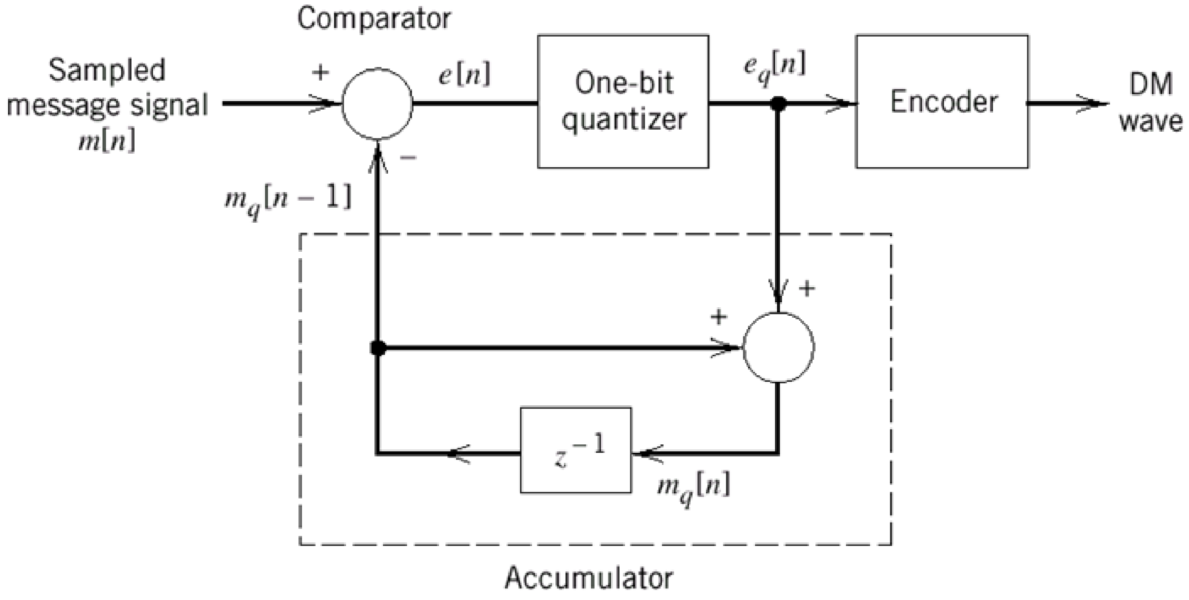
where, $m_q[n]$ is the quantized message value, $e_q[n]$ is the quantized value of the error $e[n]$, and Δ is the step size.

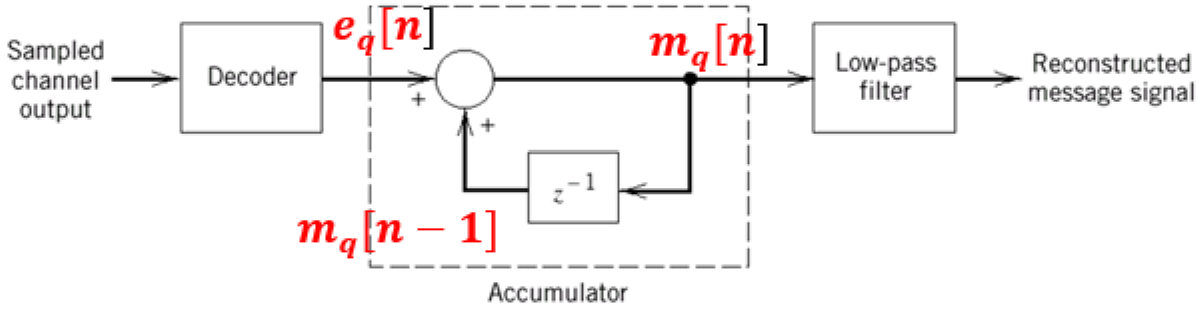


(a)

Binary sequence at modulator output 0 0 1 0 1 1 1 1 1 0 1 0 0 0 0 0 0

(b)



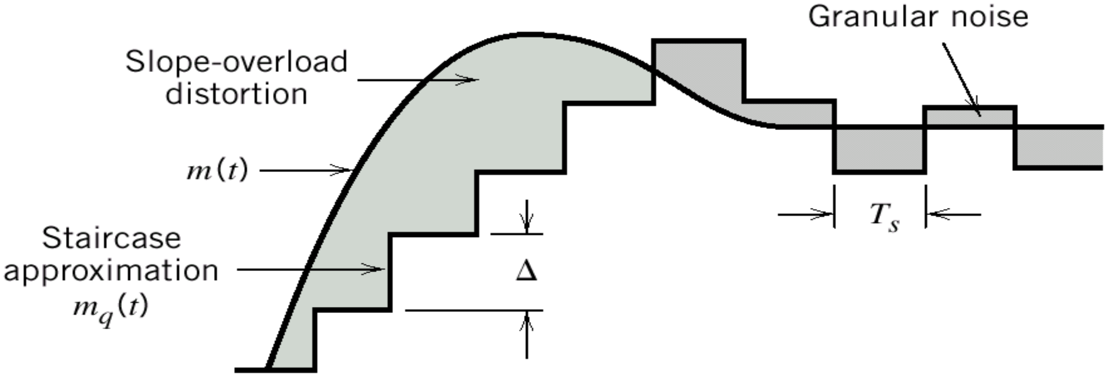


The receiver consists of an accumulator followed by a low pass filter. It regenerates $m_q[n]$ using the basic relationship

$$\boldsymbol{m_q[n] = m_q[n-1] + e_q[n] = \sum_{i=1}^{n} e_q[i]}$$

Slope overload distortion and granular noise

The modulator consists of a comparator, a quantizer, and an accumulator. The output of the accumulator is $m_q[n]$



- Slope overload distortion is due to the fact that the staircase approximation $m_q[n]$ can't follow closely the actual curve of the message signal $m(t)$. In contrast to slope-overload distortion, granular noise occurs when Δ is too large relative to the local slope characteristics of $m(t)$. Granular noise is similar to quantization noise in PCM.
- It seems that a large Δ is needed for rapid variations of $m(t)$ to reduce the slope-overload distortion and a small Δ is needed for slowly varying $m(t)$ to reduce the granular noise. The optimum Δ can only be a compromise between the two cases.
- To satisfy both cases, an adaptive DM is needed, where the step size Δ can be adjusted in accordance with the slope of the input signal $m(t)$. **This type of modulation will be encountered in lab ENEE 4103.**

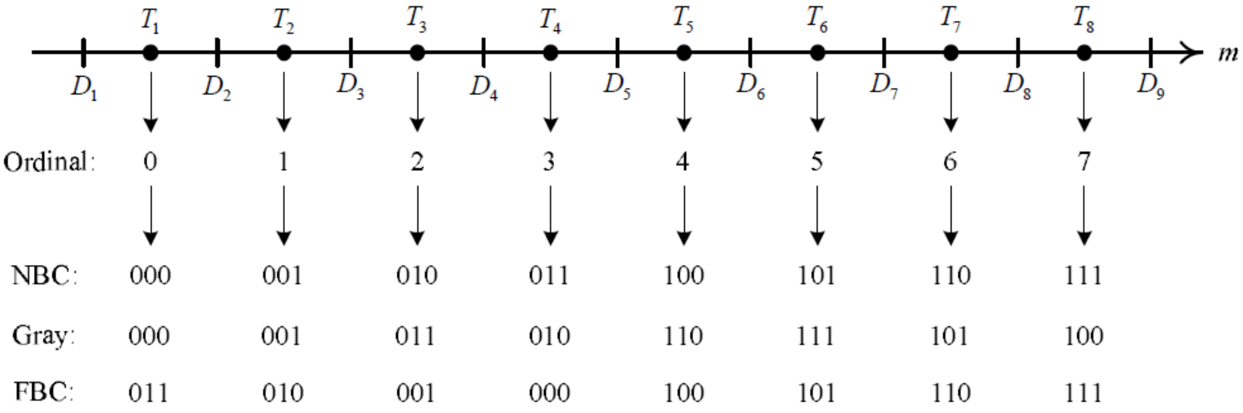
To avoid slope overload, we require that

$$\boldsymbol{\frac{\Delta}{T_s} \geq \max\left|\frac{dm(t)}{dt}\right|}$$

Binary Encoding

It is the assignment of binary digits for each one of the quantized values. There are several mapping encoding schemes. In the table below, we show the mapping for an 8-level quantizer for three types

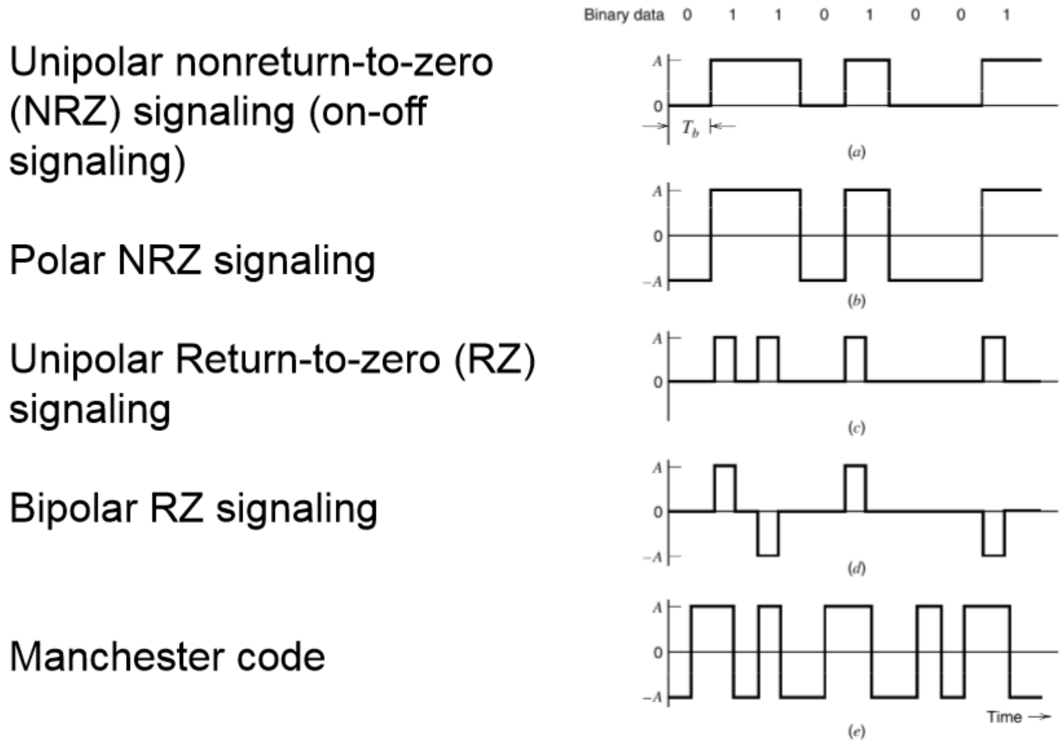
Natural Binary Bits are assigned in an ascending order	**Gray Coding** Each two levels differ by one digit	**Folded Binary** Left digit is reserved for the sign and the other two digits start from the center
111	100	111
110	101	110
101	111	101
100	110	100
011	010	000
010	011	001
001	001	010
000	000	011



Ordinal:	0	1	2	3	4	5	6	7
NBC:	000	001	010	011	100	101	110	111
Gray:	000	001	011	010	110	111	101	100
FBC:	011	010	001	000	100	101	110	111

Line Encoding

- The assignment of pulses (an electrical signal) to the binary digits that come out of the PCM or DPCM system.
- Line coding encodes the bit stream for transmission through a line, or a cable.
- It is used for communications between the CPU and peripherals, and for short-distance baseband communications, such as the Ethernet.

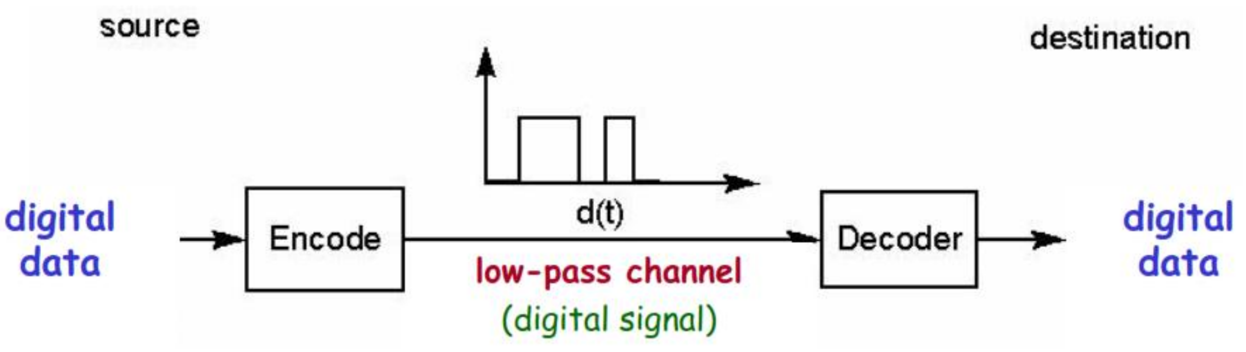

Binary data 0 1 1 0 1 0 0 1
Unipolar nonreturn-to-zero (NRZ) signaling (on-off signaling)
A
0
T_b
(a)
Polar NRZ signaling
A
0
−A
(b)
Unipolar Return-to-zero (RZ) signaling
A
0
(c)
Bipolar RZ signaling
A
0
−A
(d)
Manchester code
A
0
−A
Time
(e)

Concluding Remarks on a Speech Signal:

- PCM: The voice signal is sampled at 8 kHz (bandwidth is taken at 3.4 KHz but sampled at a rate slightly higher than the Nyquist rate), quantized into 256 levels (8 bits). Thus, a telephone PCM signal requires 64 kbps.
- DPCM (differential PCM): quantize the difference between consecutive samples; can save 8 to 16 kbps. ADPCM (Adaptive DPCM) can go further down to 32 kbps.
- Delta modulation: 1-bit DPCM with oversampling; has even lower symbol rate (e.g., 24 kbps).

Line Encoding

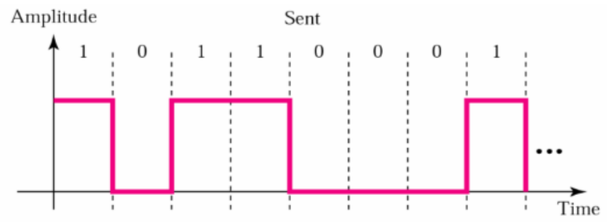
- The assignment of pulses (an electrical signal) to the binary digits that come out of the PCM or DPCM system.
- Line coding encodes the bit stream for transmission through a line, or a cable.
- It is used for communications between the CPU and peripherals, and for short-distance baseband communications, such as the Ethernet.

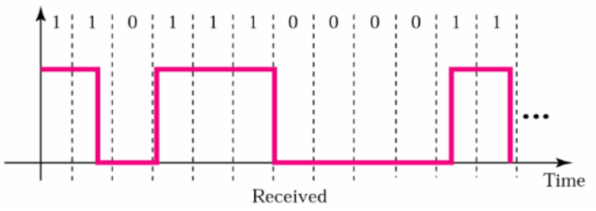


Two Design Considerations

- **DC Component in Line Coding:** Some line coding schemes have a residual (DC) component, which is generally undesirable.
    - Transformers do not allow passage of DC component.
    - DC component ⇒ extra energy – useless!

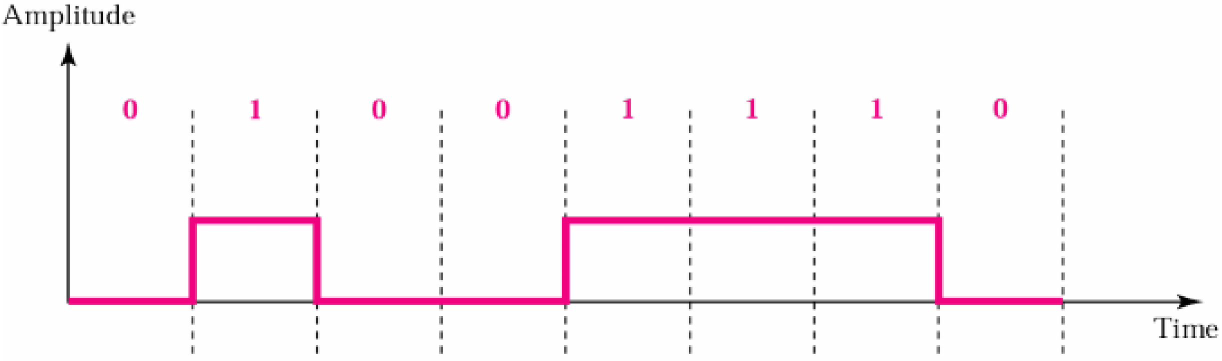
- **Self-Synchronization (clocking):** To correctly interpret signal received from sender, receiver's bit interval must exactly correspond to sender's bit intervals
    - If receiver clock is faster/slower, bit intervals not matched ⇒ receiver misinterprets signal
    - Self-synchronizing digital signals include timing information in itself, to indicate the beginning & end of each pulse





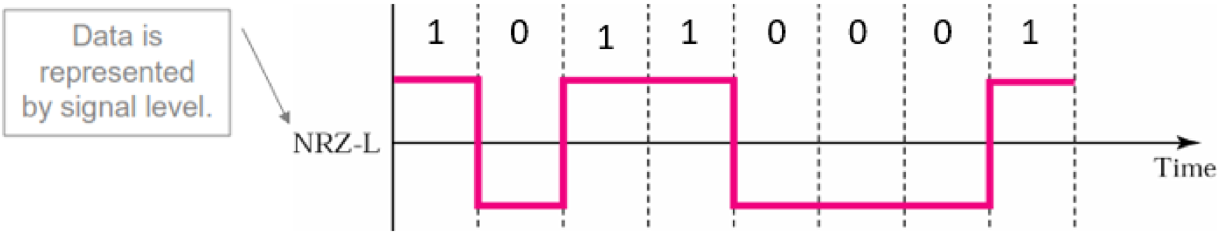
Unipolar Line Coding (Unipolar non-return to zero)

- Uses only one non-zero and one zero voltage level to represent binary digits 1 and 0
- Simple to implement, but obsolete due to two main problems:
  - Presence of a DC component.
  - Lack of synchronization for long series of 1-s or 0-s



Polar Line Coding: Polar non-return to zero

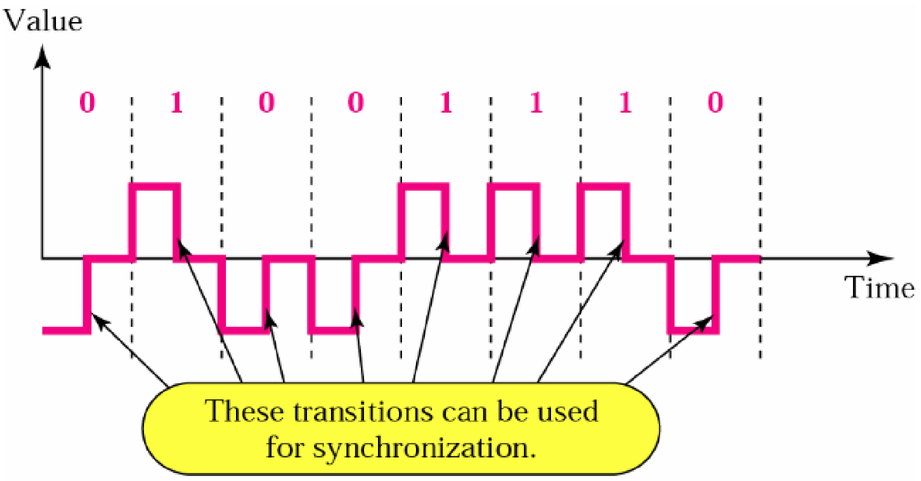
- Uses two non-zero voltage level to represent digits 1 and digit 0. +ve for 1 and –ve for 0
- No DC component is present
- Poor synchronization for long series of 1-s or 0-s


Data is represented by signal level.

Polar Non-return to Zero

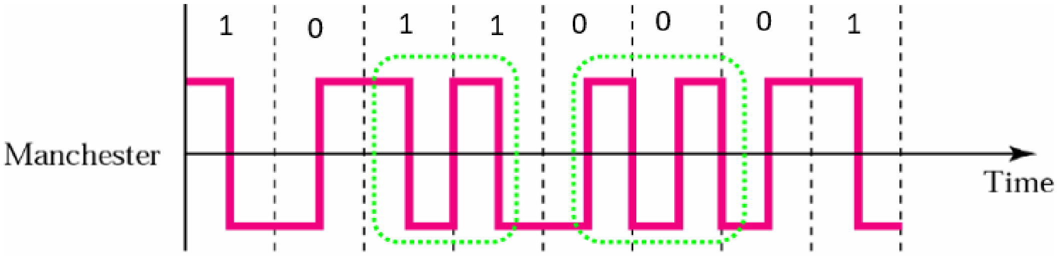
Polar Line Coding: Polar return to zero

- Uses two non-zero voltage level to represent digits 1 and digit 0. +ve for 1 and –ve for 0. **Must** return to zero halfway through each bit interval.
- No DC component is present.
- Perfect synchronization for long series of 1-s or 0-s.
- Twice the bandwidth required for polar non-return to zero, $B.W \propto \frac{1}{pulse\ width}$.



Manchester Line Coding

- Inversion at the middle of each bit interval is used for both synchronization and bit representation
- Digit 0 ⇒ neg-to-pos transition, 1 ⇒ pos-to-neg transition
- Perfect synchronization for long series of 1-s or 0-s
- There is always transition at the middle of the bit, and maybe one transition at the end of each bit.
- Fine for alternating sequences of bits (10101), but wastes bandwidth for long runs of 1-s or 0-s.
- Used by IEEE 802.3 (Ethernet).
- No DC component is present.
- Twice the bandwidth required for polar non-return to zero. Two pulses are used to represent one bit.

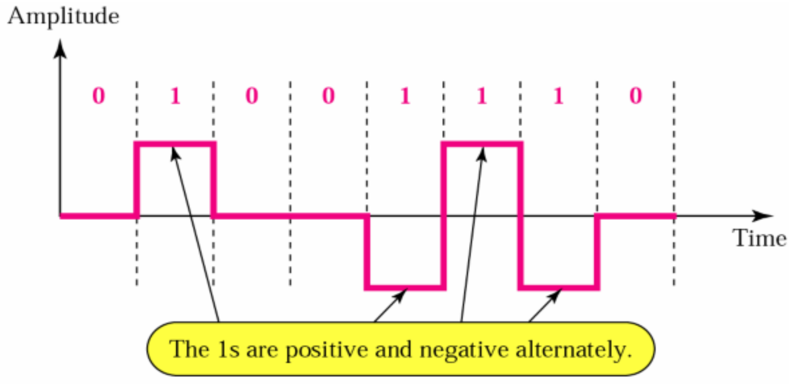


Manchester Encoding

Bipolar Line Coding

- Uses two non-zero and zero voltage level for representation of two data levels.
- **0 = zero level; 1 = alternating positive and negative level.**

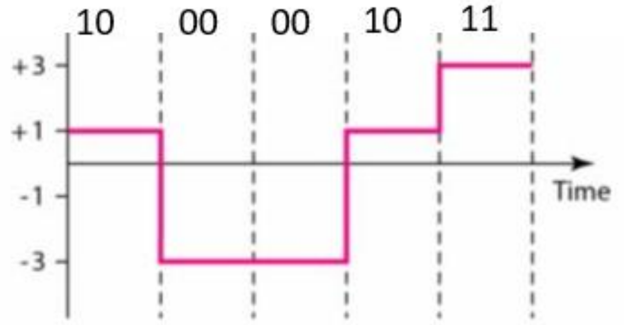
- If first bit 1 is represented by a positive amplitude, second will be represented by negative amplitude, third by positive, etc.
- Less bandwidth required than with Manchester coding (for any sequence of bits).
- Loss of synchronization is possible for long runs of 0-s.
- No DC component is present.



2B1Q (2 Bipolar to 1Quaternary) Line Coding

- Data patterns of size 2 bits are encoded as one signal element belonging to a four-level signal.
- Data is sent two time faster than with polar non-return to zero.
- Receiver has to discern 4 different thresholds

Binary Input	Output Voltage
00	-3
01	-1
10	1
11	3


10
00
00
10
11
+3
+1
-1
-3
Time