Sampling and quantization

The previous two chapters have reviewed and discussed the various characteristics of signals along with methods of describing and representing them. This chapter begins the discussion of transmitting the signals or messages using a digital communication system. Though one can easily visualize messages produced by sources that are inherently digital in nature, witness text messaging via the keyboard or keypad, two of the most common message sources, audio and video, are analog, i.e., they produce continuous time signals. To make them amenable for digital transmission it is first required to transform the analog information source into digital symbols which are compatible with digital processing and transmission.

The first step in this transformation process is to discretize the time axis, which involves sampling the continuous time signal at discrete values of time. The sampling process, primarily how many samples per second are needed to exactly represent the signal, practical sampling schemes, and how to reconstruct, at the receiver, the analog message from the samples is considered first. This is followed by a brief discussion of three pulse modulation techniques, a sort of half-way house between the analog modulation methods of AM and FM and the various digital modulation–demodulation methods which are the focus of the rest of the text.

Though time has been discretized by the sampling process the sample values are still analog, i.e., they are continuous variables. To represent the sample value by a digital symbol chosen from a finite set necessitates the choice of a discrete set of amplitudes to represent the continuous range of possible amplitudes. This process is known as quantization and unlike discretization of the time axis, it results in a distortion of the original signal since it is a many-to-one mapping. The measure of this distortion is commonly expressed by the signal power to quantization noise power ratio, SNR_q. Various approaches to quantization and the resultant SNR_q are the major focus of this chapter.

The final step is to map (or encode) the quantized signal sample into a string of digital, typically binary, symbols, commonly known as pulse-code modulation (PCM). The complete process of analog-to-digital (A/D) conversion is a special, but important, case of source coding.[1]

4.1 Sampling of continuous-time signals

The first operation in A/D conversion is the sampling process. Both the theoretical and practical implementations of this process are studied in this section.

4.1.1 Ideal (or impulse) sampling

The sampling process converts an analog waveform into a sequence of discrete samples that are usually spaced ***uniformly*** in time. This process can be mathematically described as in Figure 4.1(a). Here the analog waveform $m(t)$ is multiplied by a periodic train of unit ***impulse*** functions $s(t)$ to produced the sampled waveform $m_s(t)$. The expression for $s(t)$ is as follows:

$$s(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT_s). \tag{4.1}$$

Thus the sampled waveform $m_s(t)$ can be expressed as

$$\begin{aligned} m_s(t) &= m(t)s(t) \\ &= \sum_{n=-\infty}^{\infty} m(t)\delta(t - nT_s) = \sum_{n=-\infty}^{\infty} m(nT_s)\delta(t - nT_s). \end{aligned} \tag{4.2}$$

The parameter T_s in (4.1) and (4.2) is the period of the impulse train, also referred to as the ***sampling period***. The inverse of the sampling period, $f_s = 1/T_s$, is called the sampling frequency or ***sampling rate***. Figures 4.1(b)–(d) graphically illustrate the ideal sampling process. It is intuitive that the higher the sampling rate is, the more accurate the representation of $m(t)$ by $m_s(t)$ is. However, to achieve a high efficiency, it is desired to use as low a sampling rate as possible. Thus an important question is: what is the minimum sampling rate for the sampled version $m_s(t)$ to exactly represent the original analog signal $m(t)$? For the family of bandlimited signals, this question is answered by the ***sampling theorem***, which is derived next.

Consider the Fourier transform of the sampled waveform $m_s(t)$. Since $m_s(t)$ is the product of $m(t)$ and $s(t)$, the Fourier transform of $m_s(t)$ is the ***convolution*** of the Fourier transforms of $m(t)$ and $s(t)$. Recall that the Fourier transform of an impulse train is another impulse train, where the values of the periods of the two trains are reciprocally related to one another. The Fourier transform of $s(t)$ is given by

$$S(f) = \frac{1}{T_s} \sum_{n=-\infty}^{\infty} \delta(f - nf_s). \tag{4.3}$$

Also note that convolution with an impulse function simply shifts the original function as follows:

$$X(f) * \delta(f - f_0) = X(f - f_0). \tag{4.4}$$

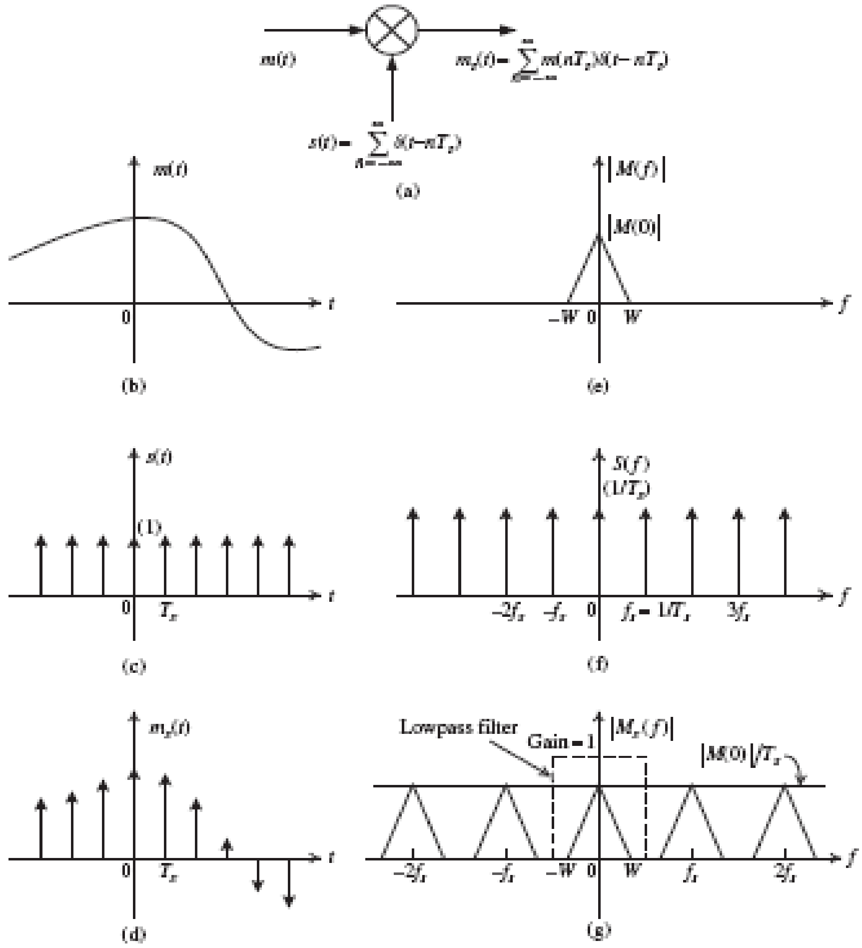


Fig. 4.1 Ideal sampling process: (a) mathematical model, (b) analog signal, (c) impulse train, (d) sampled version of the analog signal, (e) spectrum of bandlimited signal, (f) spectrum of the impulse train, (g) spectrum of the sampled waveform.

From the above equations, the transform of the sampled waveform can be written as

$$M_s(f) = M(f) * S(f) = M(f) * \left[\frac{1}{T_s}\sum_{n=-\infty}^{\infty}\delta(f - nf_s)\right]$$

$$= \frac{1}{T_s}\sum_{n=-\infty}^{\infty} M(f - nf_s). \qquad (4.5)$$

Equation (4.5) shows that the spectrum of the sampled waveform consists of an infinite number of scaled and shifted copies of the spectrum of the original signal $m(t)$. More precisely, the spectrum $M(f)$ is scaled by $1/T_s$ and periodically repeated every f_s. It should be noted that the relation in (4.5) holds for any continuous-time signal $m(t)$, even if it is not bandlimited, or of finite energy.

However, for the bandlimited waveform $m(t)$ with bandwidth limited to W hertz, a generic Fourier transform of the sampled signal $m_s(t)$ is illustrated in Figure 4.1(e). The triangular shape chosen for the magnitude spectrum of $m(t)$ is only for ease of illustration. In general, $M(f)$ can be of arbitrary shape, as long as it is confined to $[-W, W]$. Since within the original bandwidth (around zero frequency) the spectrum of the sampled waveform is the same as that of the original signal (except for a scaling factor $1/T_s$), it suggests that the original waveform $m(t)$ can be completely recovered from $m_s(t)$ by an ideal lowpass filter (LPF) of bandwidth W as shown in Figure 4.1(g). However, a closer investigation of Figure 4.1(g) reveals that this is only possible if the sampling rate f_s is high enough that there is no overlap among the copies of $M(f)$ in the spectrum of $m_s(t)$. It is easy to see that the condition for no overlapping of the copies of $M(f)$ is $f_s \geq 2W$, therefore the minimum sampling rate is $f_s = 2W$. When the sampling rate $f_s < 2W$ (undersampling), then the copies of $M(f)$ overlap in the frequency domain and it is not possible to recover the original signal $m(t)$ by filtering. The distortion of the recovered signal due to undersampling is referred to as *aliasing*.

It has been shown using the frequency domain that the original continuous signal $m(t)$ can be completely recovered from the sampled signal $m_s(t)$. Next we wish to show how to reconstruct the continuous signal $m(t)$ from its sampled values $m(nT_s)$, $n = 0, \pm 1, \pm 2, \ldots$. To this end, write the Fourier transform of $m_s(t)$ as follows:

$$\begin{aligned} M_s(f) = \mathcal{F}\{m_s(t)\} &= \sum_{n=-\infty}^{\infty} m(nT_s)\mathcal{F}\{\delta(t - nT_s)\} \\ &= \sum_{n=-\infty}^{\infty} m(nT_s)e^{-j2\pi nfT_s}. \end{aligned} \tag{4.6}$$

Since $M(f) = M_s(f)/f_s$, for $-W \leq f \leq W$, one can write

$$M(f) = \frac{1}{f_s}\sum_{n=-\infty}^{\infty} m(nT_s)e^{-j2\pi nfT_s}, \quad -W \leq f \leq W. \tag{4.7}$$

The signal $m(t)$ is the inverse Fourier transform of $M(f)$ and it can be found as follows:

$$\begin{aligned} m(t) = \mathcal{F}^{-1}\{M(f)\} &= \int_{-\infty}^{\infty} M(f)e^{j2\pi ft}df \\ &= \int_{-W}^{W} \frac{1}{f_s}\sum_{n=-\infty}^{\infty} m(nT_s)e^{-j2\pi nfT_s}e^{j2\pi ft}df \\ &= \frac{1}{f_s}\sum_{n=-\infty}^{\infty} m(nT_s)\int_{-W}^{W} e^{j2\pi f(t-nT_s)}df \\ &= \sum_{n=-\infty}^{\infty} m(nT_s)\frac{\sin[2\pi W(t - nT_s)]}{\pi f_s(t - nT_s)} \end{aligned}$$

$$= \sum_{n=-\infty}^{\infty} m\left(\frac{n}{2W}\right) \frac{\sin(2\pi Wt - n\pi)}{(2\pi Wt - n\pi)} \quad (4.8)$$

$$= \sum_{n=-\infty}^{\infty} m\left(\frac{n}{2W}\right) \operatorname{sinc}(2Wt - n), \quad (4.9)$$

where, to arrive at the last two equations, the minimum sampling rate $f_s = 2W$ has been used and $\operatorname{sinc}(x) = \sin(\pi x)/(\pi x)$.

Equation (4.9) provides an *interpolation formula* for the construction of the original signal $m(t)$ from its sampled values $m(n/2W)$. The sinc function $\operatorname{sinc}(2Wt)$ plays the role of an *interpolating function* (also known as the sampling function). In essence, each sample is multiplied by a delayed version of the interpolating function and all the resulting waveforms are added up to obtain the original signal.

Now the sampling theorem can be stated as follows.

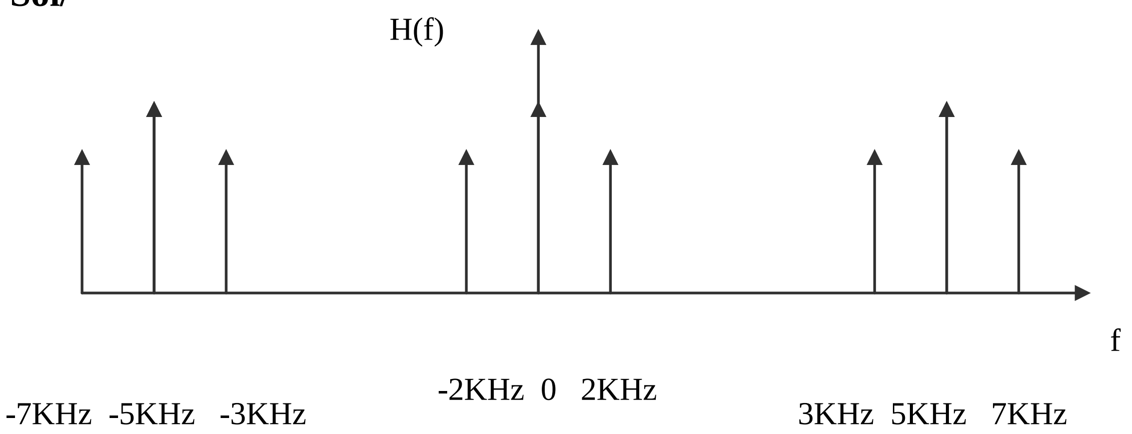
Theorem 4.1 (sampling theorem) *A signal having no frequency components above W hertz is completely described by specifying the values of the signal at periodic time instants that are separated by at most 1/2W seconds.*

The theorem stated in terms of the sampling rate, $f_s \geq 2W$, is known as the *Nyquist criterion*. The sampling rate $f_s = 2W$ is called the *Nyquist rate* with the reciprocal called the *Nyquist interval*.

The sampling process considered so far is known as *ideal sampling* because it involves ideal impulse functions. Obviously, ideal sampling is not practical. In the next two sections two practical methods of implementing sampling of continuous-time signals are introduced.

Example/ an analog signal $f(t) = 1 + \cos(4000\pi t)$ is sample by $f_s = 5000$ Hz draw the sampling signal spectrum. Calculate min. sampling frequency.

Sol/



$f_s min = 2 * f_m$

$f_s min = 2 * 2000 = 4000$ Hz

4.2 Pulse modulation

Recall that in analog (continuous-wave) modulation, some parameter of a sinusoidal carrier $A_c \cos(2\pi f_c t + \theta)$, such as the amplitude A_c, the frequency f_c, or the phase θ, is varied *continuously* in accordance with the message signal $m(t)$. Similarly, in pulse modulation, some parameter of a *pulse train* is varied in accordance with the sample values of a message signal.

Pulse-amplitude modulation (PAM) is the simplest and most basic form of analog pulse modulation. In PAM, the *amplitudes* of regularly spaced pulses are varied in proportion to the corresponding sample values of a continuous message signal. In general, the pulses can be of some appropriate shape. In the simplest case, when the pulse is rectangular, then the PAM signal is identical to the signal produced by flat-top sampling described in Section 4.1.3.

It should be noted that PAM transmission does not improve the noise performance over baseband modulation (which is the transmission of the original continuous signal). The main (perhaps the only) advantage of PAM is that it allows multiplexing, i.e., the sharing of the same transmission media by different sources (or users). This is because a PAM signal only occurs in slots of time, leaving the idle time for the transmission of other PAM signals. However, this advantage comes at the expense of a larger transmission bandwidth, as can be seen from Figures 4.5(a) and 4.5(d).

It is well known that in analog FM, bandwidth can be traded for noise performance. As mentioned before, PAM signals require a larger transmission bandwidth without any improvement in noise performance. This suggests that there should be better pulse modulations than PAM in terms of noise performance. Two such forms of pulse modulation are:

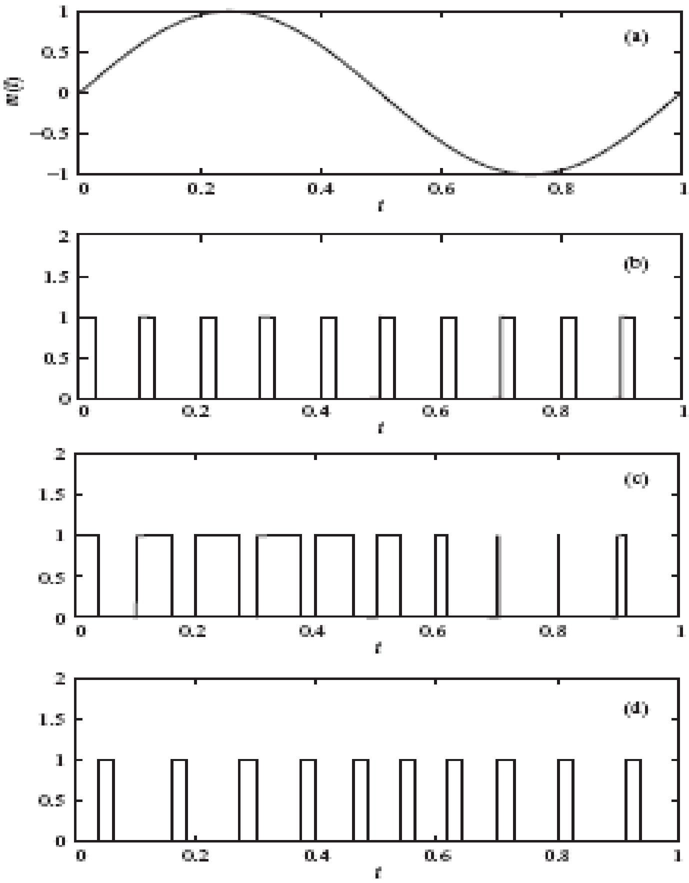
- Pulse-width modulation (PWM): in PWM, the samples of the message signal are used to vary the *width* of the individual pulses in the pulse train.
- Pulse-position modulation (PPM): in PPM, the *position* of a pulse relative to its original time of occurrence is varied in accordance with the sample values of the message.

Examples of PWM and PPM waveforms are shown in Figure 4.6 for a sinusoidal message.

Note that in PWM, long pulses (corresponding to large sample values) expend considerable power, while bearing no additional information. In fact, if only time transitions are preserved, then PWM becomes PPM. Accordingly, PPM is a more power-efficient form of pulse modulation than PWM.

Regarding the noise performance of PWM and PPM systems, since the transmitted information (the sample values) is contained in the relative positions of the modulated pulses, the additive noise, which mainly introduces amplitude distortion, has much less effect. As a consequence, both PWM and PPM systems have better noise performance than PAM.

Pulse modulation techniques, however, are still analog modulation. For digital communications of an analog source, one needs to proceed to the next step, i.e., quantization.



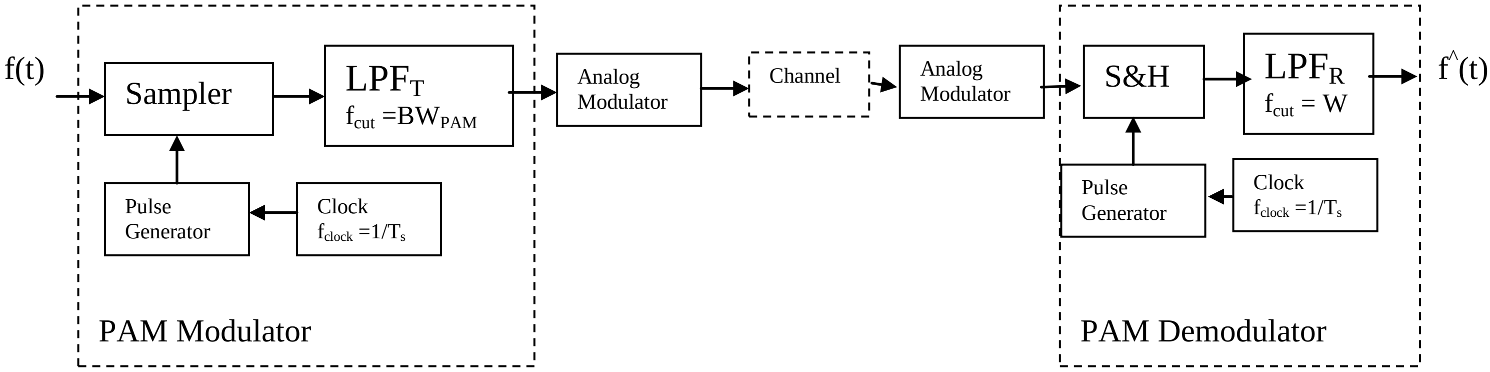
6 Examples of PWM and PPM: (a) a sinusoidal message, (b) pulse carrier, (c) PWM waveform, (d) PPM waveform.

- PAM Modulator and Demodulator

The PAM modulator that satisfy the sampling condition have transmitted bandwidth

$$BW_{PAM} = \frac{1}{2T_S}$$

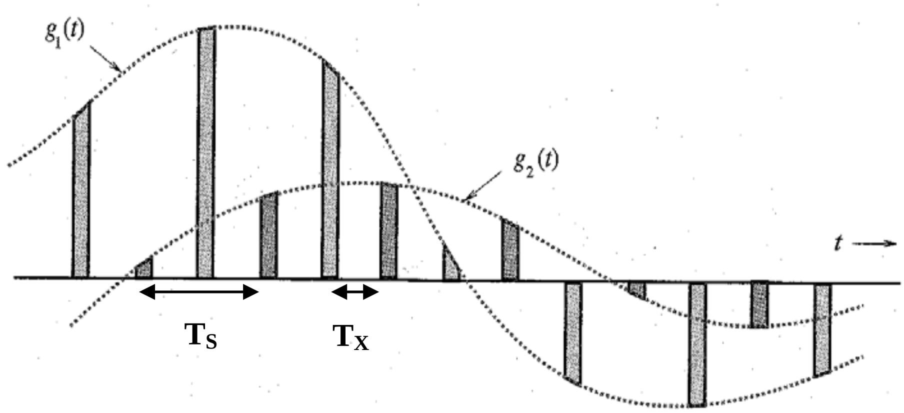
The modulator and demodulator of PAM shown below



The transmit LPF_T use to minimize the transmit signal when the receive LPF_R used to reconstruct the received signal. The analog modulator can by AM, DSBSC, SSBSC, PM, or FM.

- Time Division Multiplexing (TDM)

One advantage of using pulse modulation is that it permits the simultaneous transmission of several signals on a time-sharing basis—**time-division multiplexing (TDM)**. Because a pulse-modulated signal occupies only a part of the channel time, we can transmit several pulse-modulated signals on the same channel by interweaving them. Figure 6.9 shows the TDM of two PAM signals. In this manner we can multiplex several signals on the same channel by reducing pulse widths.



T_S : sampling time

T_X : TDM sampling time

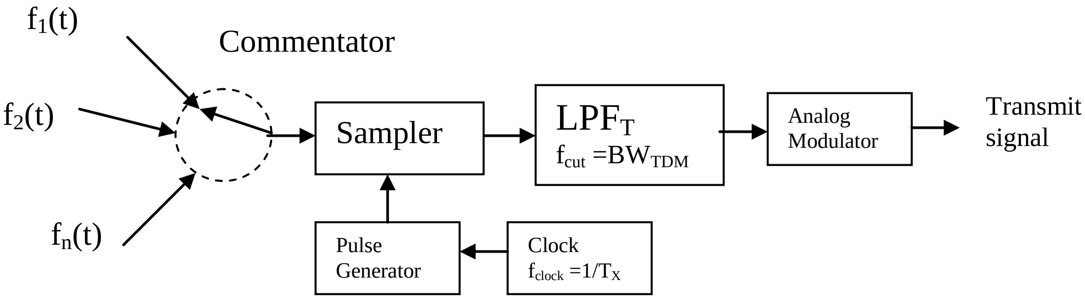
$$T_X = \frac{T_S}{n}$$

n: number of multiplexed channels. The bandwidth and the number of sample transmitted by sec for the TDM transmit signal are

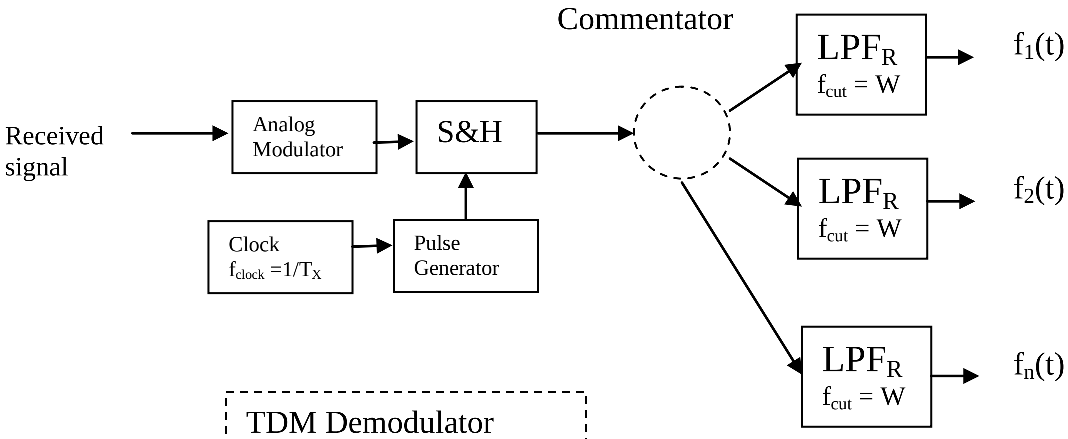
$$BW_{TDM} = \frac{1}{2T_X} \qquad (Hz)$$

$$Sampling\ \ Rate = \frac{1}{T_X} \qquad (Sample/\sec)$$

The TDM modulator and demodulator shown in figure below



TDM Modulator (for n channel)



TDM Demodulator

Example / Ten low-pass signal each band limited to 4KHz are to be multiplex in Time by sampling frequency 10KHz . calculate

SOL/

1- min clock frequency of TDM system

$$f_{clock} = \frac{1}{T_x} = n\ f_s$$

$$f_{clock} = 10 \times 10000 = 100KHz$$

2- what is the min cutoff frequency of the transmitted LPF_T

$$f_{Cut-off\,(T)} = BW_{TDM} = \frac{1}{2T_x} = \frac{n\ f_s}{2}$$

$$f_{Cut-off\,(T)} = \frac{10 \times 10000}{2} = 50KHz$$

3- what is the min and max. (Range) cutoff frequency of the received LPF_R

$$f_{Cut-off\,(R)\min} = W = 4KHz$$

$$f_{Cut-off\,(R)Max} = f_s / 2 = 5KHz$$

4- What is the total system pulse rate

$$Rate = \frac{1}{T_x} = n\ \ f_s$$

$$Rate = 10 \times 10000 = 100\ \ KPulse/\sec$$

Sheet 3

Q1) 30 low-pass signal each band limited to 3.3 KHz are to be multiplex in Time by sampling frequency 10KHz . calculate min clock frequency of TDM system, total system pulse rate and Bandwidth.

Q2) 10 low-pass signal each band limited to W KHz are to be multiplex in Time by min. sampling frequency if the transmitted ideal LPF cut-off is 100 KHz calculate W.

Q3) The three signal $f_1(t)$, $f_2(t)$, and $f_3(t)$ is sampled by f_s= 4 Hz draw the TDM signal from 0 sec to 1.5 sec. Then design TDM modulator and demodulator.(Hint assume the pulse duration neglected)

$$f_1(t) = -2t \qquad 0 \le t < 1$$

$$f_2(t) = 3 \qquad 0 \le t < 2$$

$$f_3(t) = e^{-3t} \qquad 0 \le t$$

4.3 Quantization

In all the sampling processes described in the previous section, the sampled signals are discrete in time but still continuous in amplitude. To obtain a fully digital representation of a continuous signal, two further operations are needed: *quantization* of the amplitude of the sampled signal and *encoding* of the quantized values, as illustrated in Figure 4.7. This section discusses quantization.

Fig. 4.7 Quantization and encoding operations.

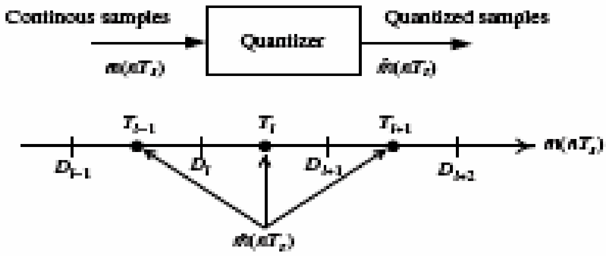


Fig. 4.8 Description of a memoryless quantizer.

By definition, amplitude quantization is the process of transforming the sample amplitude $m(nT_s)$ of a message signal $m(t)$ at time $t = nT_s$ into a discrete amplitude $\hat{m}(nT_s)$ taken from a *finite* set of possible amplitudes. Clearly, if the finite set of amplitudes is chosen such that the spacing between two adjacent amplitude levels is sufficiently small, then the approximated (or quantized) signal, $\hat{m}(nT_s)$, can be made practically indistinguishable from the continuous sampled signal, $m(nT_s)$. Nevertheless, unlike the sampling process, there is always a loss of information associated with the quantization process, no matter how finely one may choose the finite set of the amplitudes for quantization. This implies that it is not possible to *completely* recover the sampled signal from the quantized signal.

In this section, we shall assume that the quantization process is *memoryless* and *instantaneous*, meaning that the quantization of sample value at time $t = nT_s$ is independent of earlier or later samples. With this assumption, the quantization process can be described as in Figure 4.8. Let the amplitude range of the continuous signal be partitioned into L intervals, where the lth interval, denoted by $\mathcal{I}_l$, is determined by the *decision levels* (also called the *threshold levels*) D_l and D_{l+1}:

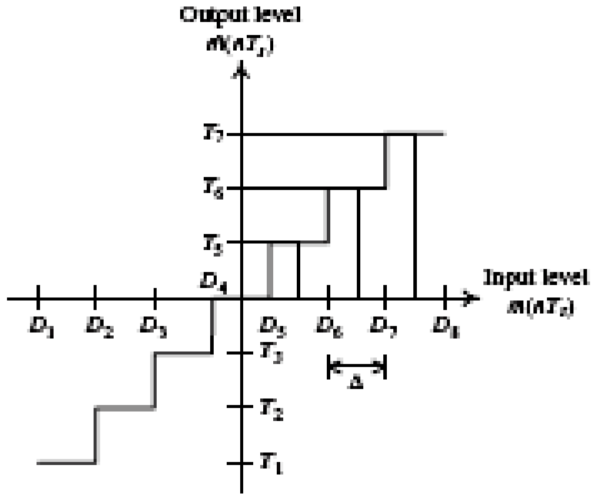
$$\mathcal{I}_l : \{D_l < m \le D_{l+1}\}, \quad l = 1, \ldots, L. \tag{4.18}$$

Then the quantizer represents all the signal amplitudes in the interval $\mathcal{I}_l$ by some amplitude $T_l \in \mathcal{I}_l$ referred to as the *target level* (also known as the *representation level* or *reconstruction level*). The spacing between two adjacent decision levels is called the *step-size*. If the step-size is the same for each interval, then the quantizer is called a *uniform quantizer*, otherwise the quantizer is nonuniform. The uniform quantizer is the simplest and most practical one. Besides having equal decision intervals, the target level is chosen to lie in the middle of the interval.

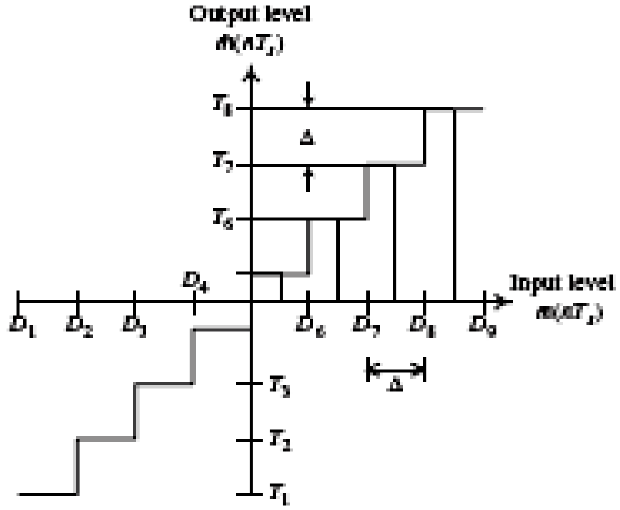
4.3.1 Uniform quantizer

From the description of the quantizer, it follows that the input–output characteristic of the quantizer (or *quantizer characteristic*) is a staircase function. Figures 4.9(a) and 4.9(b) display two uniform quantizer characteristics, called *midtread* and *midrise*. As can be seen from these figures, the classification whether a characteristic is midtread or midrise depends on whether the origin lies in the middle of a tread, or a rise of the staircase characteristic. For both characteristics, the decision levels are equally spaced and the lth target level is the midpoint of the lth interval, i.e.,

$$T_l = \frac{D_l + D_{l+1}}{2}. \quad (4.19)$$



(a)



(b)

Two types of uniform quantization: (a) midtread and (b) midrise.

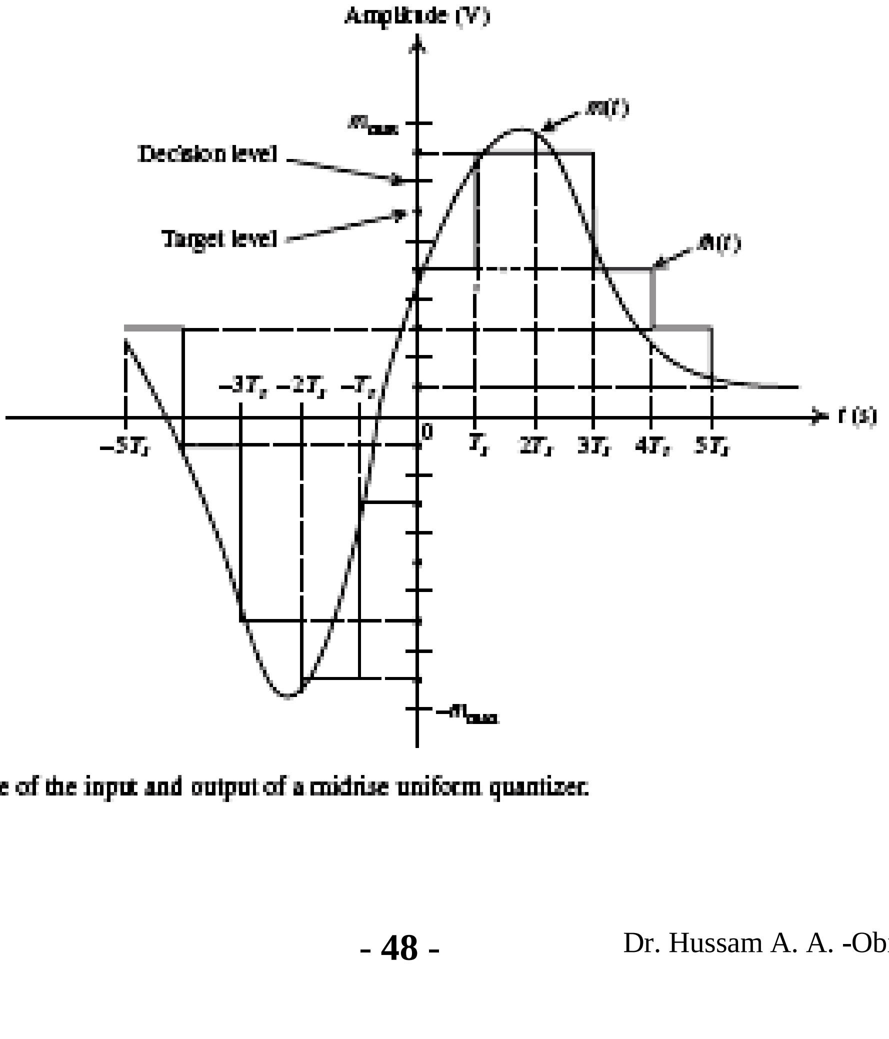
As an example, Figure 4.10 plots the input and output waveforms of a midrise uniform quantizer.

As mentioned before, the quantization process always introduces an error. The performance of a quantizer is usually evaluated in terms of its SNR. In what follows, this parameter is derived for the uniform quantizer.

Since we concentrate only on memoryless quantization, we can ignore the time index and simply write m and $\hat{m}$ instead of $m(nT_s)$ and $\hat{m}(nT_s)$ for the input and output of the quantizer respectively. Typically, the input of the quantizer can be modeled as a zero-mean random variable m with some pdf $f_m(m)$. Furthermore, assume that the amplitude range of m is $-m_{max} \leq m \leq m_{max}$, that the uniform quantizer is of midrise type, and that the number of quantization levels is L. Then the quantization step-size is given by

$$\Delta = \frac{2m_{max}}{L}. \tag{4.20}$$

Let $q = m - \hat{m}$ be the error introduced by the quantizer, then $-\Delta/2 \leq q \leq \Delta/2$. If the step-size Δ is sufficiently small (i.e., the number of quantization intervals L is sufficiently large), then it is reasonable to assume that the quantization error q is a *uniform* random variable over the range $[-\Delta/2, \Delta/2]$. The pdf of the random variable q is therefore given by



An example of the input and output of a midrise uniform quantizer.

$$f_q(q) = \begin{cases} 1/\Delta, & -\Delta/2 < q \le \Delta/2 \\ 0, & \text{otherwise} \end{cases} \quad (4.21)$$

Note that with this assumption, the mean of the quantization error is zero, while its variance can be calculated as follows:

$$\sigma_q^2 = \int_{-\Delta/2}^{\Delta/2} q^2 f_q(q) dq = \int_{-\Delta/2}^{\Delta/2} q^2 \left(\frac{1}{\Delta}\right) dq$$
$$= \frac{\Delta^2}{12} = \frac{m_{max}^2}{3L^2}, \quad (4.22)$$

where the last equality follows from (4.20).

Each target level at the output of a quantizer is typically encoded (or represented) in binary form, i.e., a binary string. For convenience, the number of quantization levels is usually chosen to be a power of 2, i.e., $L = 2^R$, where R is the number of bits needed to represent each target level. Substituting $L = 2^R$ into (4.22) one obtains the following expression for the variance of the quantization error:

$$\sigma_q^2 = \frac{m_{max}^2}{3 \times 2^{2R}}. \quad (4.23)$$

Since the message sample m is a zero-mean random variable whose pdf is $f_m(m)$, the average power of the message is equal to the variance of m, i.e., $\sigma_m^2 = \int_{-m_{max}}^{m_{max}} m^2 f_m(m) dm$. Therefore, the SNR_q can be expressed as,

$$SNR_q = \left(\frac{3\sigma_m^2}{m_{max}^2}\right) 2^{2R} \quad (4.24)$$
$$= \frac{3 \times 2^{2R}}{F^2}. \quad (4.25)$$

The parameter F in (4.25) is called the *crest factor* of the message, defined as

$$F = \frac{\text{peak value of the signal}}{\text{RMS value of the signal}} = \frac{m_{max}}{\sigma_m}. \quad (4.26)$$

Equation (4.25) shows that the SNR_q of a uniform quantizer increases *exponentially* with the number of bits per sample R and decreases with the square of the message's crest factor. The message's crest factor is an inherent property of the signal source, while R is a technical specification, i.e., under an engineer's control.

Expressed in decibels, the SNR_q is given by

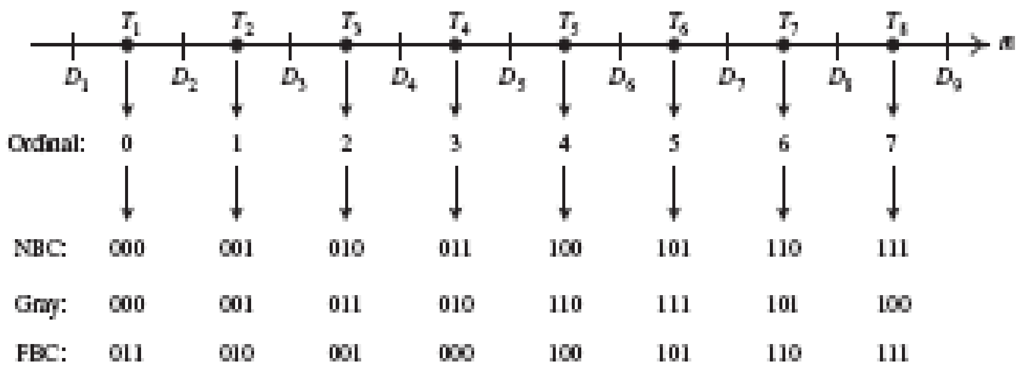
$$10 \log_{10} SNR_q = 6.02R + 10 \log_{10}\left(\frac{\sigma_m^2}{m_{max}^2}\right) + 4.77 \quad (4.27)$$
$$= 6.02R - 20 \log_{10} F + 4.77. \quad (4.28)$$

The above equation, called the *6 decibel rule*, points out a significant performance characteristic of a uniform quantizer: *an additional 6 decibel improvement in* SNR_q *is obtained for each bit added to represent the continuous signal sample.*

4.4 Pulse-code modulation (PCM)

The last block in Figure 4.7 to be discussed is the *encoder*. A PCM signal is obtained from the quantized PAM signal by encoding each quantized sample to a *digital codeword*. If the PAM signals are quantized using L target levels, then in binary PCM each quantized sample is digitally encoded into an R-bit binary codeword, where $R = \lceil \log_2 L \rceil + 1$. The quantizing and encoding operations are usually performed in the same circuit known as an A/D converter. The advantage of having a PCM signal over a quantized PAM signal is that the binary digits of a PCM signal can be transmitted using many efficient modulation schemes compared to the transmission of a PAM signal. The topics of baseband and passband modulation of binary digits are covered in Chapters 6 and 7.

There are several ways to establish a one-to-one correspondence between target levels and the codeword. A convenient method, known as natural binary coding (NBC), is to express the ordinal number of the target level as a binary number as described in Figure 4.19. Another popular mapping method is called *Gray* mapping. Gray mapping is important in the demodulation of the signal because the most likely errors caused by noise involve the erroneous selection of a target level that is adjacent to the transmitted target level. Still another mapping, called foldover binary coding (FBC), is also sometimes encountered.



9 Encoding of quantized levels into PCM codewords, $L = 8$.

- The PCM Bandwidth and Bit Rate

The sampling time T_S is divided into equal space call T_b bit time in this space the bit represented by an electrical shape

$$T_b = \frac{T_S}{R}$$

$$R : number\ of\ bit / sample$$

The transmitted bandwidth and it rate are

$$BW_{PCM} \geq \frac{1}{2T_b} \qquad Hz$$

$$BW_{PCM} \geq \frac{Log_2(L)}{2T_S}$$

$$Bit\ \ Rate = \frac{1}{T_b} = R\ \ f_S \qquad bit/\sec$$

Example/

Common input voltage ranges for commercial A/D converters (i.e., quantizers) are ±1 volt, ±5 volts, ±10 volts. Take an A/D converter with a ±1 volt range and compute the number of levels and the step size if it is (a) 8-bit, (b) 12-bit, and (c) 16-bit.

SOL/

1-

$$L = \frac{2\ \ m_{\max}}{\Delta} = \frac{2*1}{1} = 2\ \ level$$

$$R = \log_2(L) = \frac{\ln(L)}{\ln(2)} = 1\ \ bit$$

2-

$$\Delta = \frac{2\ \ m_{\max}}{L}$$

$$L = 2^R$$

$$m_{\max} = 1\ \ volt$$

$$R = 8 \qquad L = 256$$

$$\Delta = \frac{2\ \ *1}{256} = 0.0078125\ \ volt$$

This solution repeated for other case

Example/

(*PCM representation*) Your USB drive has a capacity of 10^9 bytes (1 gigabyte). You wish to store a digital representation of an information source on the drive. Using a straightforward PCM representation, determine the maximum recording duration for each of the following sources:

(a) 4 kilohertz speech signal, with eight bits per sample.

(b) 22 kilohertz stereo audio signal, with 16 bits per sample in each stereo channel.

(c) 5 megahertz video signal, with 12 bits per sample, and combined with the audio signal in (b).

(d) Digital surveillance video signal, with 1024 × 768 pixels per frame, eight bits per pixel, and one frame per second.

Sol/

a) $f_s = 2f_m$

$f_s = 8000$ Hz

bit Rate = R f_S

bit Rate = 8*8000=64000 bit/sec

MAX. recording Time= total memory size/bit rate

MAX. recording Time=10^9*8/64000 = 125000 sec

= 2083 min and 20sec

= 34 Hour, 43 min, and 20 sec

b) $f_s = 2f_m$

$f_s = 44000$ Hz

bit Rate = R f_S

bit Rate = 16*44000=704000 bit/sec

MAX. recording Time= total memory size/bit rate

MAX. recording Time=10^9*8/704000 = 11363.6 sec

= 189 min and 23.6 sec

= 3 Hour, 9 min, and 23.6 sec

c) $f_s = 2f_m$

f_s = 10 MHz

bit Rate = R f_S

bit Rate = $12*10*10^6$=120 Mbit/sec (Video)

bit Rate = 16*44000=704000 bit/sec (Sound)

total bit/sec= $120*10^6+0.704*10^6=120.704*10^6$ bit/sec

MAX. recording Time= total memory size/bit rate

MAX. recording Time=$10^9*8/(120.704*10^6)$= 66.28 sec

= 1 min, and 6.28 sec

$$Rate = \frac{pixel}{fram} * \frac{bit}{pixel} * \frac{fram}{\sec}$$

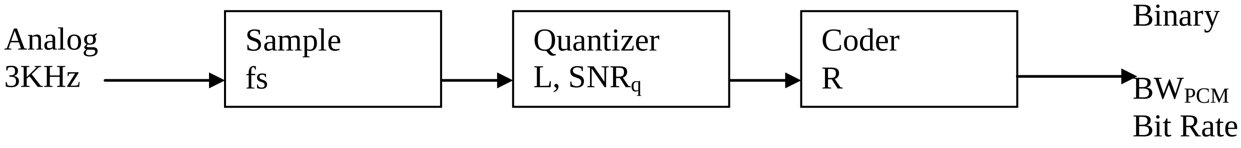
d) $Rate = 1024 * 768 * (8) * 1$

$Rate = 786432 \ bit / \sec$

MAX. recording Time=10^9*8/(786432)= 10172.53 sec

Example/ The PCM modulator have resolution $\pm 1\%$ used to transmit binary data the analog signal have max, frequency 3 kHz and sampling frequency is 2.5 time the analog frequency. Calculate PCM modulator parameters

SOL/



fs = 2.5 f_m

fs = 7.5 KHz

L = 1/(resolution)

L=1/.02 = 50 level

R=log$_2$(L) =ln(L)/ln(2) = 5.648 bit

R = 6 bit/sample

Let the input is sin or cos then $F = \frac{1}{\sqrt{2}}$

SNR$_q$ (dB) = 6.02*R -20*log(F) +4.77= 43.9 dB

$$BW_{PCM} = \frac{1}{2T_b} \quad Hz$$

$$BW_{PCM} = \frac{Log_2(L)}{2T_S} = \frac{R\ f_S}{2} = \frac{6*7500}{2} = 22.5KHz$$

$$Bit\ Rate = \frac{1}{T_b} = R\ f_S = 45\ Kbit/\sec$$

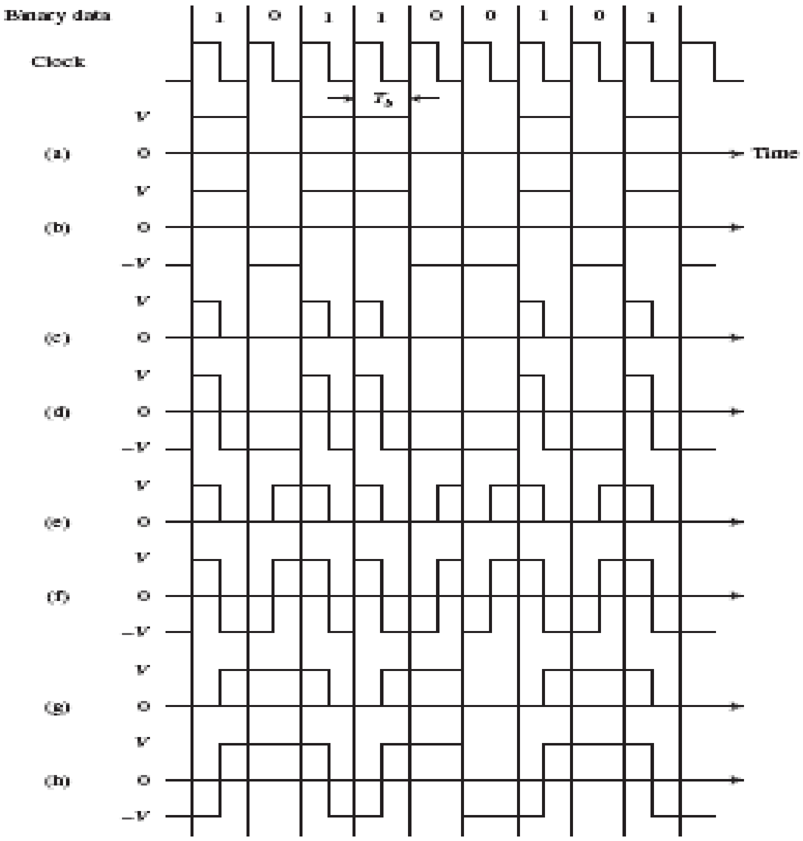
6.2 Baseband signaling schemes

Nonreturn-to-zero (NRZ) code The NRZ code can be regarded as the most basic baseband signaling scheme, since it appears "naturally" in synchronous digital circuits. In NRZ code, the signal alternates between the two voltage levels only when the current bit differs from the previous one.

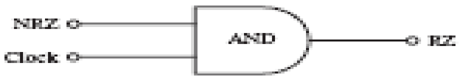
Figure 6.1(a) represents an example of NRZ waveform, where T_b is the bit duration. Note that there is only one polarity in the waveform, hence it is also known as unipolar NRZ waveform. This is the simplest version of NRZ and can easily be generated. However, the DC component of a long random sequence of ones and zeros is nonzero. More precisely, the DC component is $VP[1_T] + 0P[0_T] = VP_2 + 0P_1 = VP_2$. For the common case of equally likely bits, the DC component is one-half of the positive voltage, i.e., $0.5V$ (volts). Therefore it is common to pass the NRZ waveform through a level shifter. The resultant waveform then alternates between $+V$ and $-V$ as shown in Figure 6.1(b). It is called the polar NRZ or NRZ-L waveform, whose DC component is $VP[1_T] + (-V)P[0_T] = V(P_2 - P_1)$. Obviously, the DC component of the NRZ-L waveform is zero if the two bits are equally likely.

Observe that the NRZ-L code produces a transition whenever the current bit in the input sequence differs from the previous one. These transitions can be used for synchronization purposes at the receiver. However, if the transmitted data contain long strings of similar bits, then the timing information is sparse, and regeneration of the clock signal at the receiver can be very difficult.

Return-to-zero (RZ) code The RZ code is similar to the NRZ code except that the information is contained in the first half of the bit interval, while the second half is always at level "zero." An example of an RZ waveform is shown in Figure 6.1(c). Once again the code has a DC component, which is $(0.5V)P[1_T] + 0P[0_T] = (0.5V)P_2$. If $P_1 = P_2 = 0.5$, then the DC component is one-fourth of the positive voltage, i.e., $0.25V$ (volts). Figure 6.2 shows that the RZ code is generated by gating the basic NRZ signal with the transmitter clock.



Various binary signaling formats: (a) NRZ; (b) NRZ-L; (c) RZ; (d) RZ-L; (e) Biϕ, (f) Biϕ-L; (g) Miller, (h) Miller-L.



RZ encoder.

In order to compare RZ fairly with the NRZ-L code in terms of error probability, we shall consider the RZ-L (or bipolar RZ) code where the two levels are $+V$ and $-V$ rather than V and 0. The corresponding waveform of RZ-L code is shown in Figure 6.1(d). The DC component of RZ-L waveform is $-VP_1$, which is $-0.5V$ (volts) if $P_1 = P_2 = 0.5$. Thus the code still has a nonzero DC component.

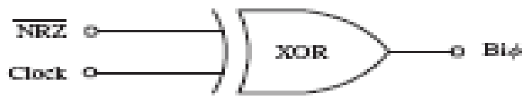
Regarding timing information, note that as opposed to NRZ, with RZ a long string of "1" bits results in transitions from which a clock at the receiver can be regenerated. A string of "0" bits, however, does not have any transitions just as with the NRZ code. For this reason and the fact that it has poor spectral properties and inferior error performance, RZ coding is not used except in some very elementary transmitting and recording equipment.

Biphase (Biϕ) or Manchester code To overcome the poor synchronization capability of NRZ and RZ codes, biphase (Biϕ) coding has been developed. It encodes information in terms of level transitions in the middle of a bit interval. Note that this conversion of bits to electrical waveforms is in a sharp contrast to what done in NRZ and RZ codes, where the information bits are converted to voltage levels. The biphase conversion (or mapping, or encoding) rules are as follows:

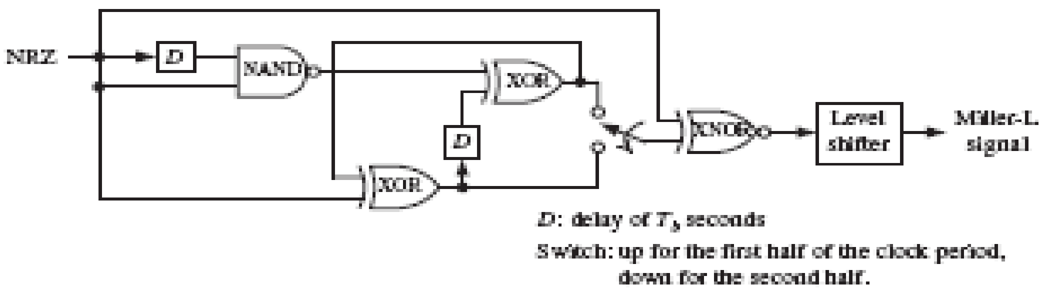
- Bit "1" is encoded as a transition from a high level to a low level occurring in the middle of the bit interval.
- Bit "0" is encoded as a transition from a low level to a high level occurring in the middle of the bit interval.
- An additional "idle" transition may have to be added at the beginning of each bit interval to establish the proper starting level for the information carrying transition.

Examples of Biϕ and Biϕ-L waveforms are shown in Figures 6.1(e) and 6.1(f), respectively. The DC component of the Biϕ signal is evaluated as $(0.5V)P[1_T] + (0.5V)P[0_T] = 0.5V[P_2 + P_1] = 0.5V$, while the DC component of the Biϕ-L signal is obviously 0. Note that the above results for the DC component hold regardless of the *a priori* probabilities of the bits. Therefore the Biϕ-L code does not have a DC component. Figure 6.3 shows that the bi-phase code can be generated with an XOR logic whose inputs are the basic NRZ signal and the transmitter clock.

The Biϕ signal, however, occupies a wider frequency band than the NRZ signal. This is due to the fact that for alternating bits there is one transition per bit interval while for two identical consecutive bits, two transitions occur per bit interval. On the other hand, because there is a predictable transition during every bit interval, the receiver can synchronize on



Biϕ encoder.



Miller-L encoder.

that transition. The Biϕ code is thus known as a self-synchronizing code. It is commonly used in local area networks (LANs), such as the Ethernet.

Miller code This code is an alternative to the Biϕ code. It has at least one transition every two-bit interval and there are never more than two transitions every two-bit interval. It thus provides good synchronization capabilities, while requiring less bandwidth than the Biϕ signal. The encoding rules are:

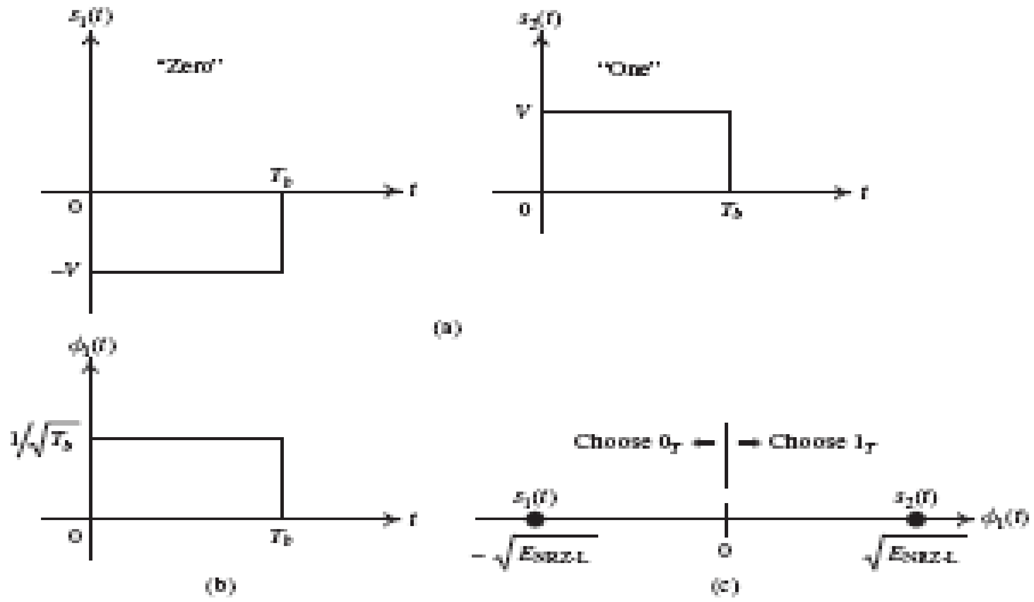
- Bit "1" is encoded by a transition in the middle of the bit interval. Depending on the previous bit this transition may be either upward or downward.
- Bit "0" is encoded by a transition at the beginning of the bit interval if the previous bit is "0". If the previous bit is "1," then there is no transition.

The waveforms for Miller and Miller-level (Miller-L) codes are illustrated in Figures 6.1(g) and 6.1(h), respectively. The Miller-L signal can be generated from the NRZ signal by the circuit shown in Figure 6.4.

6.3 Error performance

To determine the probability of bit error for each of the line codes we shall consider that the transmitted signals are corrupted by zero-mean AWGN noise of spectral strength $N_0/2$ (watts/hertz) and that the two bits, "0" and "1," are *equally likely*. As shown in the previous chapter, the error probability of each line code is readily determined by identifying the elementary signals used for bits "0" and "1" and representing them in the signal space diagram. In all cases, the orthonormal basis set for the signal space can be determined simply by inspection. For each signaling scheme, a voltage swing from $-V$ to V volts is also assumed.

NRZ-L code The elementary signals are shown in Figure 6.5(a), where each signal has energy $E_{\text{NRZ-L}} = V^2T_b$ (joules). The single basis function and signal space plot



NRZ-L code: (a) elementary signals; (b) basis function; (c) signal space and decision regions.

are given in Figures 6.5(b) and 6.5(c). Applying (5.105), the probability of bit error is given by

$$P[\text{error}]_{\text{NRZ-L}} = Q\left(\sqrt{2E_{\text{NRZ-L}}/N_0}\right). \tag{6.1}$$

Relationship between Q(x) and erfc(x) Besides the Q-function, another function widely used in error probability calculation is the *complementary error function*, denoted by erfc(.). The erfc function is defined as follows:

$$\text{erfc}(x) = \frac{2}{\sqrt{\pi}}\int_x^{\infty} e^{-\lambda^2}\,d\lambda \tag{5.106}$$

$$= 1 - \text{erf}(x). \tag{5.107}$$

By change of variables, it is not hard to show that the erfc function and the Q function are related by

$$Q(x) = \frac{1}{2}\text{erfc}\left(\frac{x}{\sqrt{2}}\right). \tag{5.108}$$

RZ-L code Figure 6.6 shows the elementary signals, the basis functions, the signal space together with the optimum decision regions of RZ-L signaling. Each signal has energy $E_{RZ\text{-}L} = V^2 T_b = E_{NRZ\text{-}L}$ (joules) and the error probability is given as

$$P[\text{error}]_{RZ\text{-}L} = Q\left(\sqrt{E_{RZ\text{-}L}/N_0}\right). \quad (6.2)$$

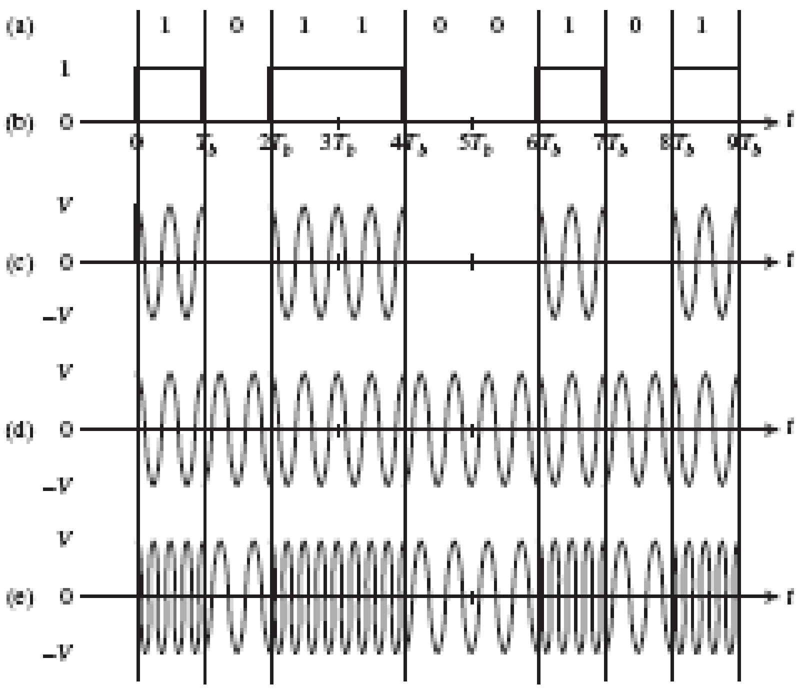
7.2 Binary amplitude-shift keying (BASK)

In BASK a sinusoidal carrier is simply gated on and off by the bit sequence to be transmitted. A typical transmitted waveform would be that shown in Figure 7.1(c), corresponding to the information sequence of Figure 7.1(a). The transmitted signal can be written as

$$s(t) = m(t)c(t), \quad (7.1)$$

where $m(t)$ is the modulating signal (the baseband signal, an NRZ signal) and $c(t) = V\cos(2\pi f_c t)$ is the sinusoidal carrier. The logic "1" and "0" are represented during any bit interval, T_b, by the following signal set:

$$\begin{cases} s_1(t) = 0, & \text{if "}0_T\text{"} \\ s_2(t) = V\cos(2\pi f_c t), & \text{if "}1_T\text{"} \end{cases}, \quad 0 < t \le T_b, \quad (7.2)$$



Binary passband modulation techniques (a) binary data; (b) modulating signal $m(t)$; (c) BASK signal; (d) BPSK signal; (e) BFSK signal.

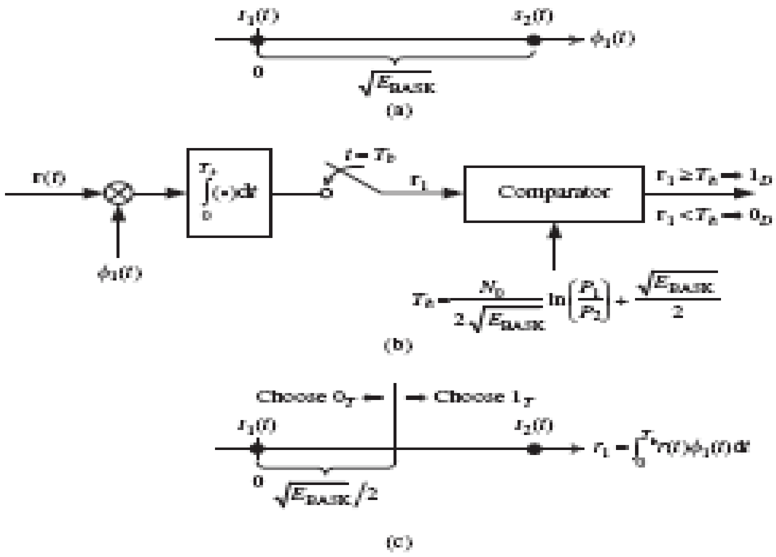
where the carrier frequency is usually chosen such that there is an integer number of cycles over the bit duration T_b, i.e., $f_c = n/T_b$, n an integer. The energy in $s_2(t)$ is $E_{\text{BASK}} = V^2T_b/2$ joules.

The received signal is $r(t) = s_i(t) + w(t)$, where $i = 1$ or 2 depending on the transmitted signal, and $w(t)$ is a zero-mean Gaussian noise process with two-sided PSD $N_0/2$. Only one orthonormal basis function, $\phi_1(t) = s_2(t)/\sqrt{E_{\text{BASK}}}$, is needed to represent the signal set. The signal space plot is shown in Figure 7.2(a). The optimum receiver, i.e., the one with the minimum error probability, is shown in Figure 7.2(b) in the form of a correlation receiver. The threshold, T_h, in Figure 7.2(b) is given by

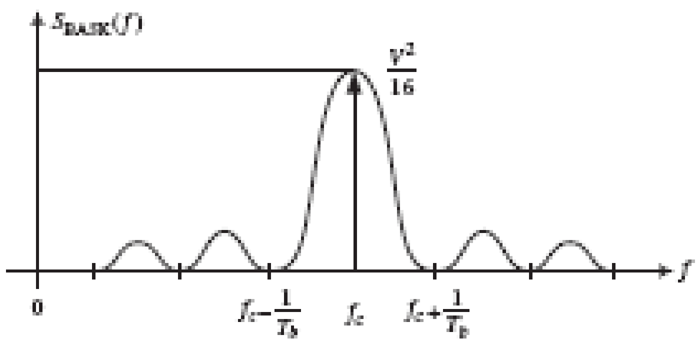
$$T_h = \frac{N_0}{2\sqrt{E_{\text{BASK}}}}\ln\left(\frac{P_1}{P_2}\right) + \frac{\sqrt{E_{\text{BASK}}}}{2}. \tag{7.3}$$

For $P_1 = P_2$, the decision regions are depicted in Figure 7.2(c). The error probability for this case is

$$P[\text{error}]_{\text{BASK}} = Q\left(\sqrt{\frac{E_{\text{BASK}}}{2N_0}}\right). \tag{7.4}$$



BASK signaling scheme: (a) signal space plot; (b) optimum receiver implementation; (c) decision regions.

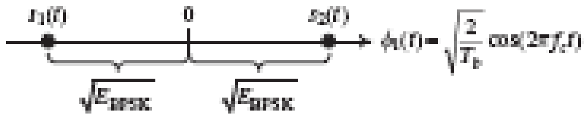


PSD of BASK.

7.3 Binary phase-shift keying (BPSK)

A BPSK signal is generated by amplitude modulating the sinusoidal carrier with a NRZ-L signal of amplitude ±1. The transmitted signal is $s(t) = m(t)c(t)$ (where $m(t)$ is a NRZ-L signal) with a resultant phase that is either 0 or π radians. The waveforms are plotted in Figure 7.1(d). The signal set is given by

$$\begin{cases} s_1(t) = -V\cos(2\pi f_c t), & \text{if "}0_T\text{"} \\ s_2(t) = +V\cos(2\pi f_c t), & \text{if "}1_T\text{"} \end{cases}, \quad 0 < t \le T_b, \tag{7.6}$$



Signal space plot of BPSK.

where $f_c = n/T_b$ for some integer n. Each signal has energy $E_{\mathrm{BPSK}} = V^2T_b/2$. The signal space plot is shown in Figure 7.4, where $\phi_1(t) = s_2(t)/\sqrt{E_{\mathrm{BPSK}}} = \sqrt{2/T_b}\cos(2\pi f_c t)$.

The minimum-error-probability (optimum) receiver projects the received signal $r(t)$ onto $\phi_1(t-(k-1)T_b)$, samples the output of the projection at $t = kT_b$ and compares it to the threshold, which in the special case of $P_1 = P_2$ is equal to zero. The error probability for $P_1 = P_2$ is given by

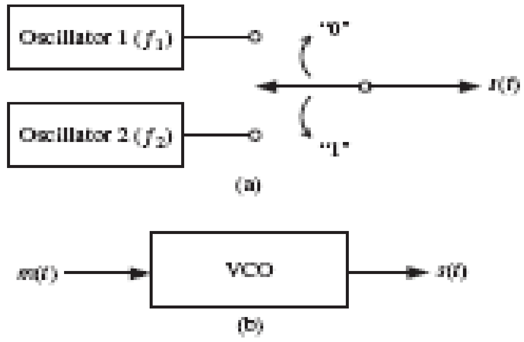
$$P[\mathrm{error}]_{\mathrm{BPSK}} = Q\left(\sqrt{\frac{2E_{\mathrm{BPSK}}}{N_0}}\right). \tag{7.7}$$

7.4 Binary frequency-shift keying (BFSK)

The most basic method of generating BFSK is to gate two oscillators with the modulating signal, as illustrated in Figure 7.5(a).

The elementary signals are

$$\begin{cases} s_1(t) = V\cos(2\pi f_1 t + \theta_1), & \text{if "}0_T\text{"} \\ s_2(t) = V\cos(2\pi f_2 t + \theta_2), & \text{if "}1_T\text{"} \end{cases}, \quad 0 < t \le T_b. \tag{7.9}$$



The two carrier frequencies are chosen to be integer multiples of $1/T_b$, while the two phases θ_1 and θ_2 need not be the same. Furthermore, the frequencies f_1 and f_2 are chosen so that $s_1(t)$ and $s_2(t)$ are *orthogonal* over the interval $[0, T_b]$, i.e.,

$$\int_0^{T_b} s_1(t)s_2(t)\mathrm{d}t = 0. \tag{7.10}$$

To see what this implies about the two frequencies assume, without loss of generality, that $f_2 > f_1$. Substitute for $s_1(t)$, $s_2(t)$ in (7.10) and integrate to obtain

$$\frac{\sin[2\pi(f_2+f_1)T_b + (\theta_2+\theta_1)] - \sin(\theta_2+\theta_1)}{(f_2+f_1)} + \frac{\sin[2\pi(f_2-f_1)T_b + (\theta_2-\theta_1)] - \sin(\theta_2-\theta_1)}{(f_2-f_1)} = 0. \tag{7.11}$$

Equation (7.11) gives the following conditions on f_1 and f_2:

(i) If the two phases are the same, i.e., $\theta_1 = \theta_2$, then

$$f_2 - f_1 = \frac{m}{2T_b}, \quad m = 1, 2, \ldots. \tag{7.12}$$

The minimum frequency separation $(f_2 - f_1)$ for orthogonality occurs when $m = 1$ and is given by

$$(\Delta f)_{\min}^{[\text{coherent}]} = \frac{1}{2T_b}. \tag{7.13}$$

In this case the two sinusoidal carriers are said to be *coherently* orthogonal (coherent because the two phases are the same).

(ii) If the two phases are different, i.e., $\theta_1 \neq \theta_2$, then

$$f_2 - f_1 = \frac{m}{T_b}, \quad m = 1, 2, \ldots. \tag{7.14}$$

The minimum frequency separation for this case, called *noncoherent* orthogonality (noncoherent because there is no relationship between the two phases), is

$$(\Delta f)_{\min}^{[\text{noncoherent}]} = \frac{1}{T_b}. \tag{7.15}$$

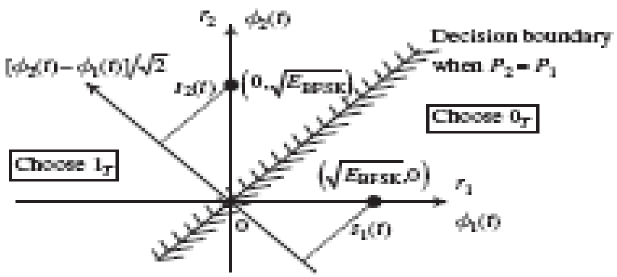
The above shows that relaxing phase synchronization of the two carriers requires a doubling of their *minimum spacing* in order to maintain the orthogonality of the two carriers.

It is also possible to express the signal set in a different way:

$$\begin{cases} s_1(t) = V\cos 2\pi(f_c - f_d)t \\ s_2(t) = V\cos 2\pi(f_c + f_d)t \end{cases}, \quad 0 < t \leq T_b, \tag{7.16}$$

where f_c is the carrier frequency and f_d is the *frequency deviation*. From this viewpoint the transmitted signal is generated by frequency modulating a voltage-controlled oscillator

BFSK is a frequency modulation, i.e., a nonlinear modulation, and one would expect that its PSD, as in the analog FM situation, is more difficult to determine. However, (5.147) applies to BFSK under the provision that the signals are statistically independent from bit interval to bit interval. The PSD of BFSK is therefore given by (again, ignoring the cross product term)



Signal space plot and decision regions of BFSK.

(VCO) with the random binary sequence, $m(t)$. This is illustrated in Figure 7.5(b). It can be verified that the orthogonal condition requires that

$$f_c = n/4T_b, \tag{7.17}$$

$$f_d = \begin{cases} m/4T_b & \text{(coherent orthogonality)} \\ m/2T_b & \text{(noncoherent orthogonality)} \end{cases}, \tag{7.18}$$

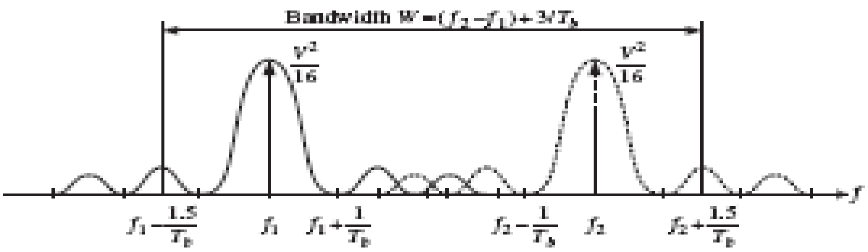
where n and m are positive integers, and $n \gg m$.

With either coherent or noncoherent orthogonality, the energy in each signal of BFSK is given by $E_{BFSK} = V^2 T_b/2$ (joules). Two orthogonal basis functions are required to represent the signal set, namely

$$\phi_1(t) = \frac{s_1(t)}{\sqrt{E_{BFSK}}}, \quad \phi_2(t) = \frac{s_2(t)}{\sqrt{E_{BFSK}}}. \tag{7.19}$$

The signal space plot is shown in Figure 7.6. The optimum receiver projects the received signal along the $[\phi_2(t) - \phi_1(t)]/\sqrt{2}$ axis and compares the projection to a threshold. For the case of $P_1 = P_2$, this threshold is 0 and the decision region is geometrically shown in Figure 7.6. The error probability is given by

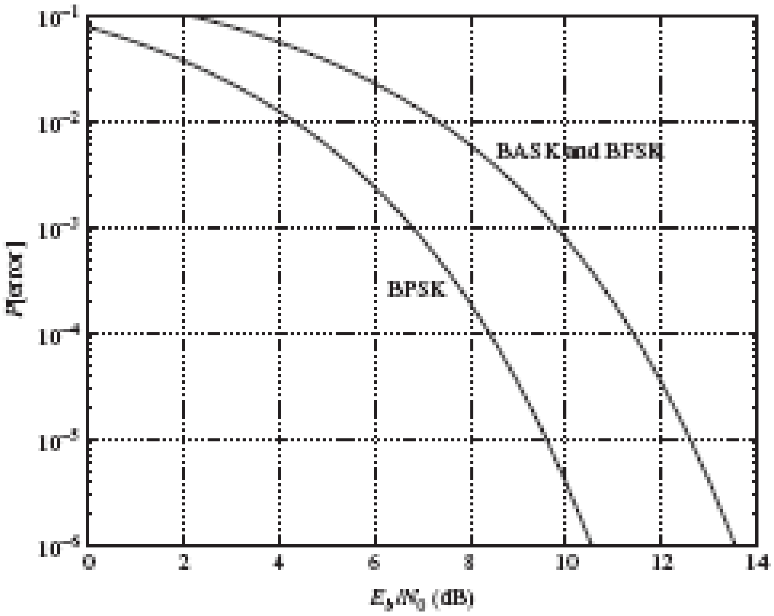
$$P[\text{error}]_{BFSK} = Q\left(\sqrt{\frac{E_{BFSK}}{N_0}}\right). \tag{7.20}$$



Bandwidth approximation for BFSK.

7.5 Performance comparison of BASK, BPSK, and BFSK

To compare the error performance of the three signaling schemes, it is necessary to express the error probabilities in terms of the average *energy per bit*, or E_b. With equally likely information bits "0" and "1" one has the following relationships for different signaling schemes: $E_b = E_{\text{BPSK}}$, $E_b = E_{\text{BASK}}/2$, and $E_b = E_{\text{BFSK}}$. Thus the error performances of different modulation schemes are as follows:



Error performance of binary passband modulation techniques.

$$P[\text{error}]_{\text{BPSK}} = Q\left(\sqrt{\frac{2E_b}{N_0}}\right). \tag{7.22}$$

$$P[\text{error}]_{\text{BASK}} = P[\text{error}]_{\text{BFSK}} = Q\left(\sqrt{\frac{E_b}{N_0}}\right). \tag{7.23}$$

The above shows that BPSK is 3 dB more efficient than BFSK, which has the same performance as BASK. This is shown graphically in Figure 7.8.

In terms of bandwidth, BFSK occupies a larger bandwidth than BPSK and BASK (recall that BPSK and BASK occupy the same bandwidth). Each of the three modulation techniques has a spectrum that decays as $1/f^2$ for frequencies away from the carrier, reflecting the fact that for each modulation the transmitted signal has discontinuities. In the next section, other modulation techniques are introduced which are more spectrally efficient.

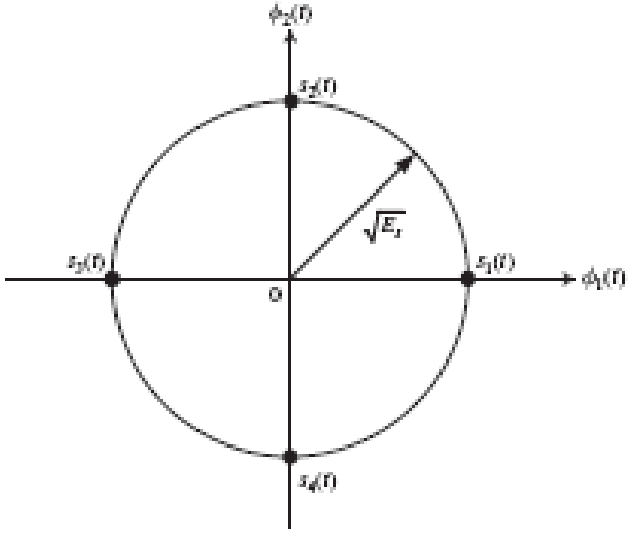
7.6.1 Quadrature phase-shift keying (QPSK)

The basic idea behind QPSK exploits the fact that $\cos(2\pi f_c t)$ and $\sin(2\pi f_c t)$ are orthogonal over the interval $[0, T_b]$ when $f_c = k/T_b$, k integer. Just as in analog modulation, this can be used to transmit two different messages over the same frequency band. To accomplish this the bit stream is taken two bits at a time and mapped into signals as shown in Table 7.1. An example QPSK signal is shown in Figure 7.9. Since each bit occupies T_b seconds, the signals corresponding to the "digits" (or symbols), 00, 01, 11, 10, last for a *symbol duration* of $T_s = 2T_b$ seconds. The symbol signaling rate or what is commonly called the *baud rate* is therefore $r_s = 1/T_s = 1/(2T_b) = r_b/2$ (symbols/second), i.e., *halved*. Since the bandwidth requirement (the PSD will be derived later) is proportional to r_s, it can also be reduced by half for a given bit rate r_b. Conversely, for a fixed bandwidth the bit rate r_b can be doubled.

Though the bit rate has been increased without a corresponding increase in bandwidth it is also necessary to look at what happens to the bit error probability. To accomplish this the signals $s_1(t)$, $s_2(t)$, $s_3(t)$, and $s_4(t)$ are represented, as usual, by an orthonormal basis set. As mentioned, the signals satisfy the following:

$$\int_0^{T_s} s_i^2(t)\mathrm{d}t = \frac{V^2}{2}T_s = V^2 T_b = E_s, \tag{7.24}$$

$$\int_0^{T_s} V\sin(2\pi f_c t)V\cos(2\pi f_c t)\mathrm{d}t = 0. \tag{7.25}$$



Signal space plot of QPSK modulation.

Therefore only two orthonormal functions are needed to represent the four signals, namely,

$$\phi_1(t) = \frac{s_1(t)}{\sqrt{E_s}}, \quad \phi_2(t) = \frac{s_2(t)}{\sqrt{E_s}}. \tag{7.26}$$

For a fair comparison with the error performance of binary modulation schemes considered previously, again it is necessary to express (7.44) in terms of E_b, the *average energy per bit*. Since each signal of QPSK carries two bits and the energy of each signal is $E_s = V^2 T_b$, the average energy per bit is $E_b = E_s/2 = V^2 T_b/2$. Therefore the bit error probability of QPSK with a Gray mapping is

$$P[\text{bit error}] = Q\left(\sqrt{\frac{2E_b}{N_0}}\right). \tag{7.46}$$

which is exactly the same as that of BPSK. This clearly demonstrates the advantage of QPSK over BPSK. With QPSK modulation, the bit rate can be *doubled* without requiring any additional transmission bandwidth or sacrificing the error performance. Interpreted differently, to deliver the same transmission rate at the same bit error performance, using QPSK reduces the transmission bandwidth to half of that required by BPSK.

Example/ The low pass signal of max. frequency 3.3KHz sampled 8 KHz and code by eight bit/ sample transmit by QPSK with E_b/N_0 =6dB. Calculate

1- total transmit bit rate

Transmit bit rate = $(R*f_S)/2$

Transmit bit rate = (8*8000)/2=32 Kbit/sec

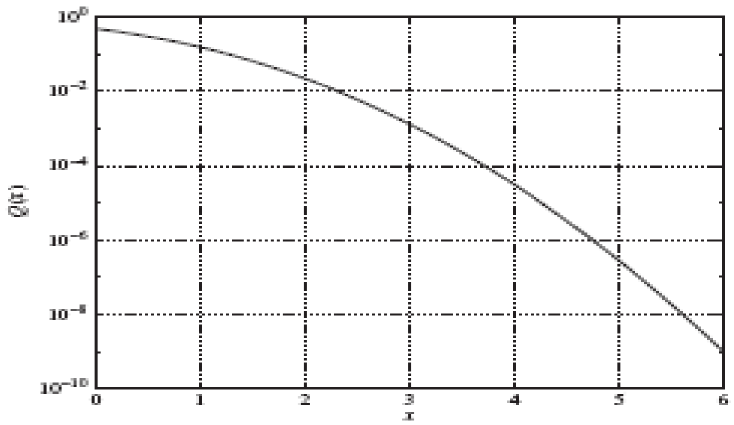
2- probability of error $P_e = Q((2E_b/N_0)^{0.5})$

E_b/N_0 (dB)=10 Log(E_b/N_0)

$E_b/N_0 = 10^{(6/10)}$=3.9811

P_e= Q(2.2817)

$P_e=(1/2)*\text{erfc}(2.2817/2^{0.5})$= 0.0113



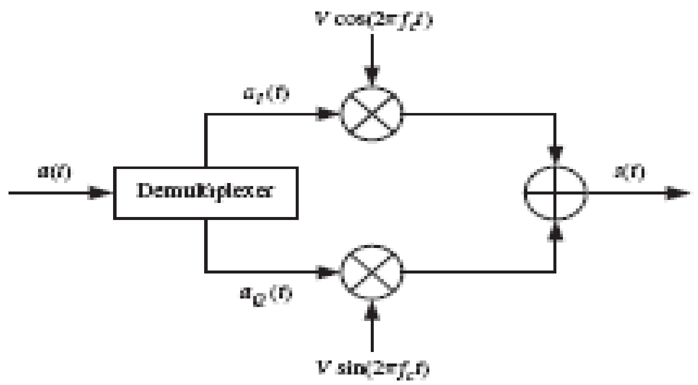
plot of $Q(x)$.

7.6.2 An alternative representation of QPSK

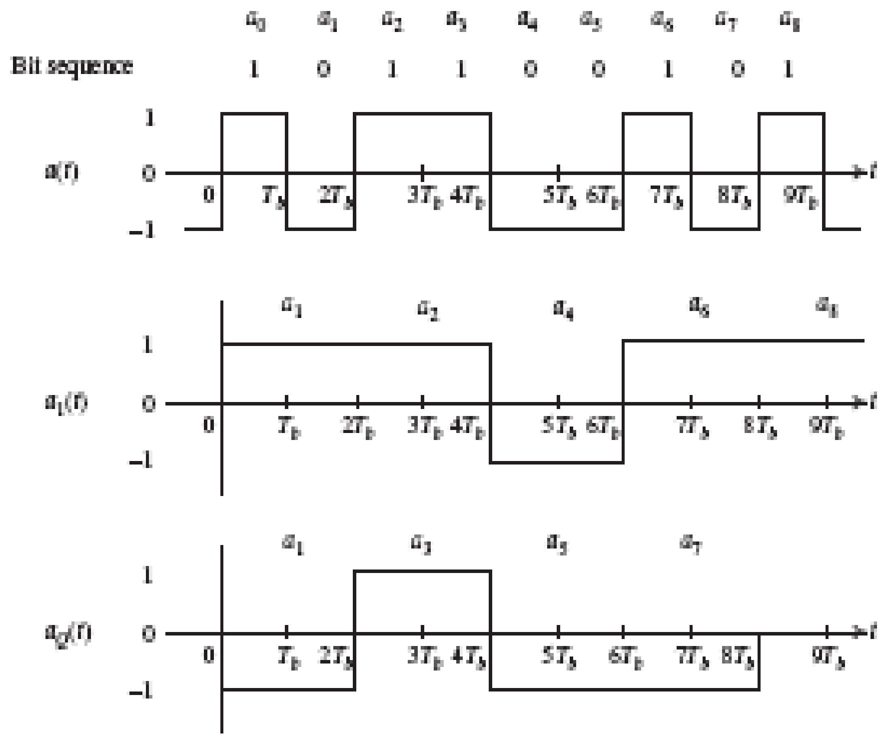
In this representation, the information bit stream is first converted to an NRZ-L waveform $a(t)$ with ± 1 levels. The waveform $a(t)$ is then demultiplexed into even, $a_I(t)$, and odd, $a_Q(t)$, bit streams (waveforms) where I and Q are mnemonics for inphase and quadrature, respectively. The individual bits in each stream occupy $T_s = 2T_b$ seconds and modulate the inphase carrier, $V\cos(2\pi f_c t)$, and quadrature carrier, $V\sin(2\pi f_c t)$, respectively. A block diagram of such a QPSK modulator is illustrated in Figure 7.15 and examples of various waveforms are shown in Figure 7.16.

The transmitted signal is

$$s(t) = a_I(t)V\cos(2\pi f_c t) + a_Q(t)V\sin(2\pi f_c t). \tag{7.47}$$



A different block diagram of a QPSK modulator.



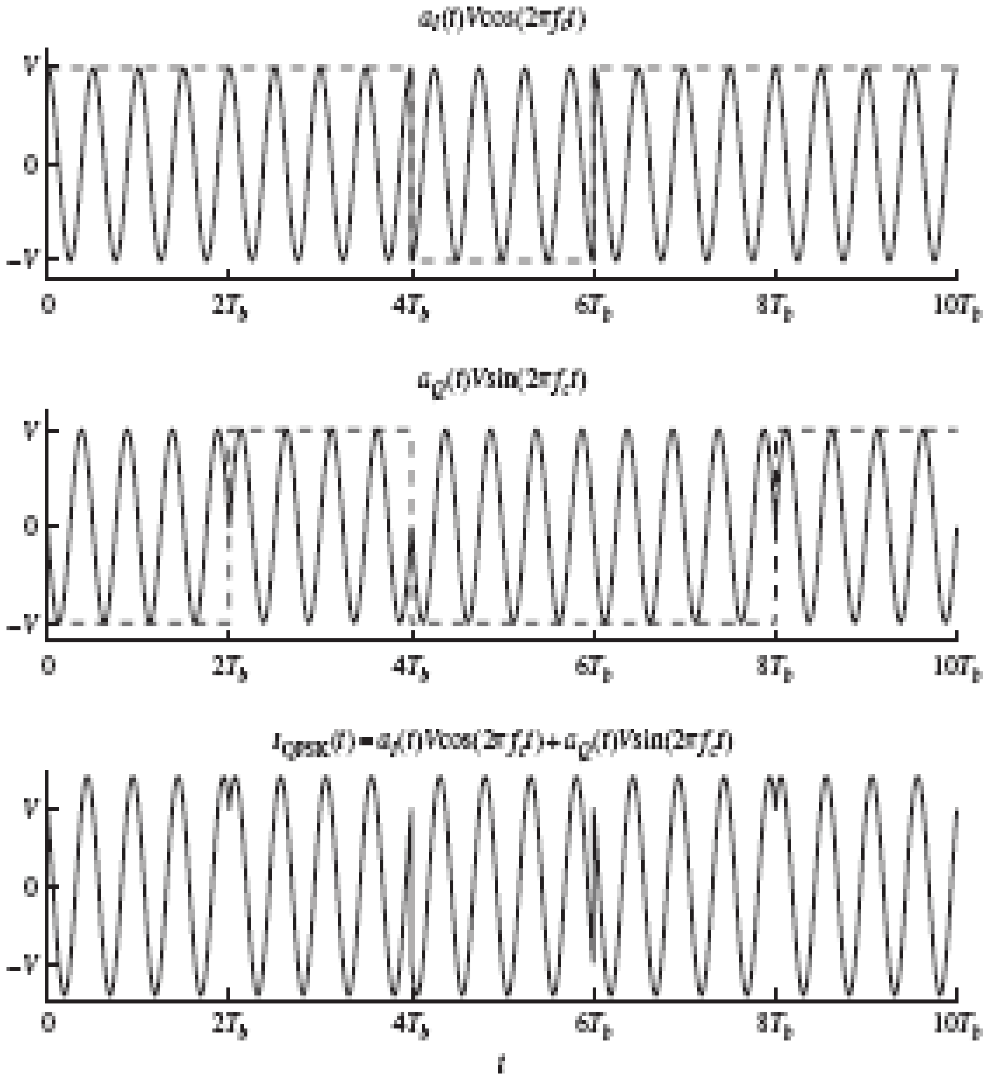
Examples of $a_I(t)$ and $a_Q(t)$ in QPSK modulation.

which can be rewritten as

$$s(t) = \sqrt{a_I^2(t) + a_Q^2(t)}V\cos\left(2\pi f_c t - \tan^{-1}\left(\frac{a_Q(t)}{a_I(t)}\right)\right)$$
$$= \sqrt{2}V\cos[2\pi f_c t - \theta(t)], \qquad (7.48)$$

where the phase $\theta(t)$ is determined as follows:

$$\theta(t) = \begin{cases} \pi/4, & \text{if } a_I = +1, a_Q = +1 \text{ (bits are 11)} \\ -\pi/4, & \text{if } a_I = +1, a_Q = -1 \text{ (bits are 10)} \\ 3\pi/4, & \text{if } a_I = -1, a_Q = +1 \text{ (bits are 01)} \\ -3\pi/4, & \text{if } a_I = -1, a_Q = -1 \text{ (bits are 00)} \end{cases}. \qquad (7.49)$$



An example of a QPSK signal, viewed as a sum of inphase and quadrature components.

6.6 Differential modulation

The basic concept behind differential modulation or coding[3] is that the signal transmitted in one bit interval is relative to the one transmitted in the previous interval. The actual transmitted signal thus depends on the present information bit and the previously transmitted bit. As a concrete example, take NRZ-L. Rather than mapping "1" to level $+V$ and "0" to level $-V$ irrespective of the previous signal, we change the modulation rule to:

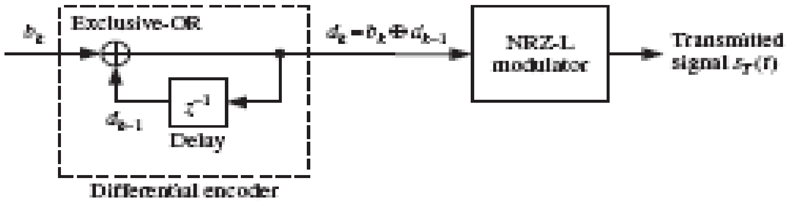
- If the present bit is a "1," then transmit a level that is opposite to that of the previous interval.
- If the present bit is a "0," then stay at the same level.

Put simply, a "1" means a level change, a "0" no change. The transmitted signal in any interval is relative to that of the previous interval. More importantly, if there is a polarity reversal this relativity is still preserved, i.e., two consecutive signals at the same level imply a "0" and a change of level implies a "1."

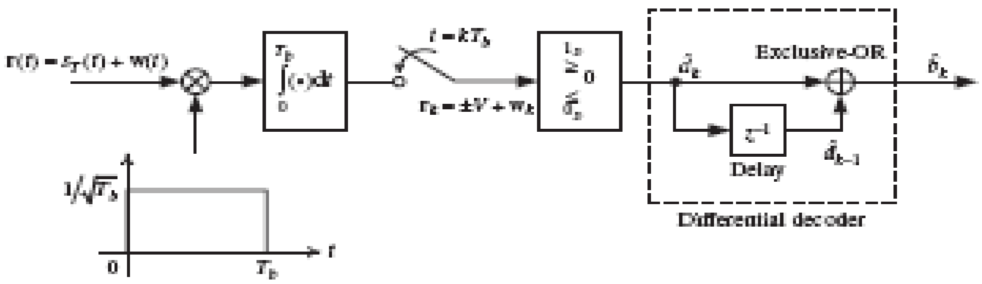
Though described at the modulator output level, the modulation rule can be implemented by first differentially encoding the information bits, followed by an NRZ-L modulation. This is shown in Figure 6.21.

If $b_k = 1$ then $d_k = \overline{d}_{k-1}$, implying a level change, and if $b_k = 0$, then $d_k = d_{k-1}$, which means no level change. Note that because of the memory one must initialize the shift register circuit, say with a logic 0.

To demodulate the received signal one first determines d_k with minimum error probability. Call this estimate $\hat{d}_k$. To recover an estimate of b_k, which is the bit of interest,



1 Differential NRZ-L modulation.



2 Demodulation of differential NRZ-L modulation.

note that $d_k = b_k \oplus d_{k-1}$ and adding modulo-2 d_{k-1} to both sides results in $d_k \oplus d_{k-1} = b_k \oplus d_{k-1} \oplus d_{k-1} = b_k$, since $d_{k-1} \oplus d_{k-1} = 0$ and $b_k \oplus 0 = b_k$. This is what is done at the receiver, except we do not have d_{k-1} but the next best thing, $\hat{d}_{k-1}$. A block diagram of the demodulation is shown in Figure 6.22.