UNIT 3

BASEBAND TRANSMISSION

Properties of Line codes- Power Spectral Density of Unipolar / Polar RZ & NRZ – Bipolar NRZ - Manchester- ISI – Nyquist criterion for distortionless transmission – Pulse shaping – Correlative coding - Mary schemes – Eye pattern – Equalization

3.1 LINE CODES

In the baseband digital transmitting system, the signal emitted by the source (analog or digital) is passed through three basic blocks of the transmitter; formatter block, source coder block and channel coder block. The channel coded data is already in digital symbol form. So, they can be represented by various dc levels. However, these dc levels cannot be directly transmitted through the channel. A channel has a frequency response and noise contamination which is not suitable for dc transmission. So, the symbols are sent as analog waveforms. First, the symbol digits are mapped to a particular waveform before transmission. In baseband, line code are used to map the digits to a pulse waveform. The name came from voice telephony field. If the symbol set is non-binary, the waveform set is called **M-ary line code**. If the binary line code has a 0 level and a single non-zero voltage level, then the signal is called **unipolar**. On the other hand, if the levels are one positive and one negative, then the code is a **bipolar**.

3.1.1 Classification of Line Codes

A digital circuit designer knows that a logical 1 corresponds to a particular voltage level, say 5 V, and a logical 0 to some other voltage level, say 2.5 V. Similarly a tape record manufacturer uses a convention that for a logical 1, a change in level takes place, whereas for a logical 0, no change takes place. A communication engineer should know that both of these are line codes; the line code used in digital logic circuits is called NRZ-L and that in magnetic tape recording NRZ-M. All the line codes are shown together in Fig. 3.1 for comparison.

3.1.1.1 NRZ (Non Return To Zero)

The NRZ is the most commonly used line code. A digital signal occurs only for a bit interval, say T_b. If the waveform stays at any non-zero level for the whole interval *Tb,* then it is called **non return to zero** (NRZ) waveform. It can be of three types: NRZ-L (L for level), NRZ-M (M for Mark) and NRZ-S(S for Space).

In NRZ-L, a binary 1 is represented by one voltage level and a binary zero by another voltage level. There is a change in level whenever the data changes from a one to a zero or from a zero to a one. NRZ-L is widely used in digital logic circuits.

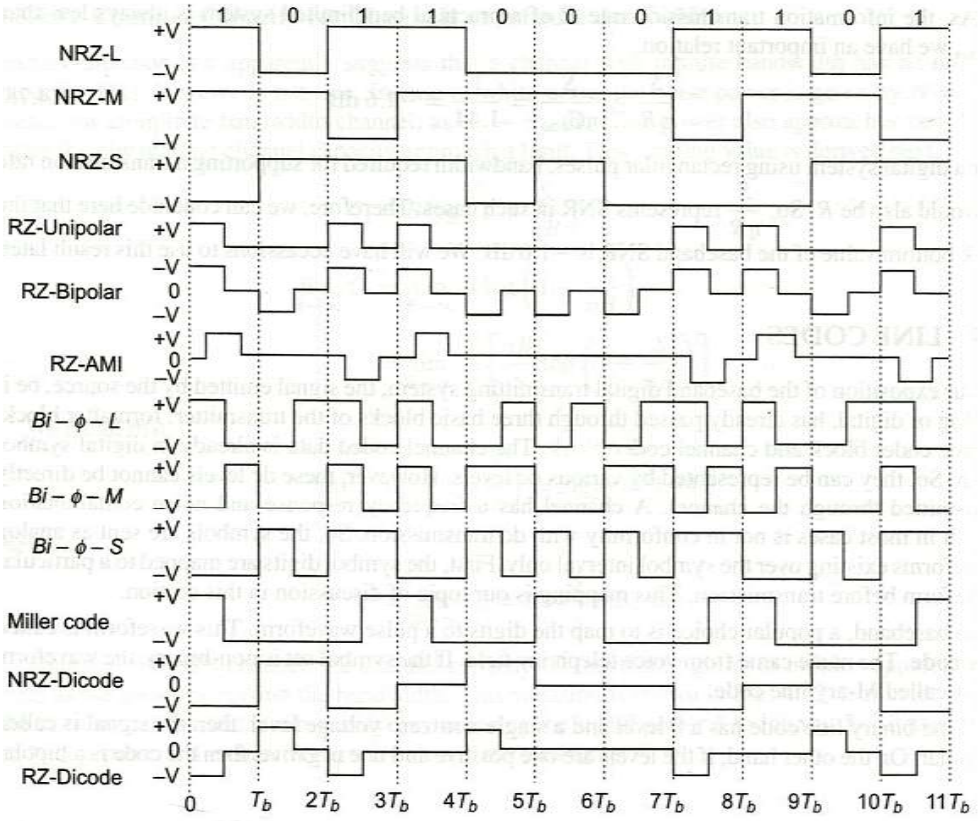


Fig. 3.1 ***Line Codes***

In NRZ-M, the one, or mark, is represented by a change in level, and the zero, or space, by no change in level. This often is called *differential encoding.* NRZ-M is mainly used in magnetic tape recording.

NRZ-S is the complement of NRZ-M. A one is represented by no change in level, and a zero by a change in level.

3.1.1.2 RZ (Return to zero)

In case of Return to zero (RZ) waveform, the signal comes back to zero level after a portion (usually half) of the bit interval T_b. The RZ waveforms consist of unipolar-RZ, bipolar-RZ and RZ-AMI.

In unipolar-RZ, a one is represented by a half-bit-wide pulse and a zero is represented by the absence of a pulse.

With bipolar-RZ, the ones and zeros are represented by opposite-level pulses that are usually one-half bit wide. There is a pulse present in each bit interval.

RZ-AMI (Alternate mark inversion) represents ones by equal amplitude but alternating pulses. The zeros are represented by the absence of pulses.

RZ line codes are widely used in baseband data transmission and magnetic recording. RZ-AMI is used in telephone signaling.

3.1.1.3 Phase Encoded

In Phase-encoded scheme, the time position of the occurrence or transition of a pulse waveform is utilised to distinguish between different logic levels. This group consists of $bi-\phi-L$ (bi-phase-level), $bi-\phi-M$ (hi-phase-mark), $bi-\phi-S$ (bi-phase-space) and Miller coding.

$Bi-\phi-L$ is popularly called **Manchester coding**. A one is represented by a half bit wide pulse positioned during the first half of the bit interval whereas a zero is represented by a half bit wide pulse positioned during the second half of the bit interval.

With $bi-\phi-M$, a transition occurs at the beginning of every bit interval. A one is represented by a second transition one-half bit interval later. A zero is represented by no second transition.

$Bi-\phi-S$ is complementary to $bi-\phi-M$, a transition at the beginning of every bit interval. A one is represented by no second transition and a zero is represented by a second transition one-half bit interval later.

With Miller coding, a one is represented by a transition at the midpoint of the bit interval and a zero by no transition unless it is followed by another zero. In this case a transition is placed at the end of the bit interval of the first zero.

Phase-encoding schemes are used in magnetic recording systems, optical communications and satellite telemetry links.

3.1.1.4 Multilevel Binary

In Multilevel Binary, a binary waveform is transmitted by means of more than two levels. Usually for binary zero one level is fixed, the level corresponding to logic 1 alternates between other two levels. Bipolar RZ and RZ-AMI, described already, belong to this group. The other codes of this group are dicode and duobinary.

With dicode-NRZ, the one-to-zero or zero-to-one transition changes the pulse polarity. When there is no data transition, a zero level is sent.

With dicode-RZ, the one-to-zero or zero-to-one transition produces a half-duration polarity change; otherwise, a zero level is sent.

3. 1.2 Properties of a Line Code

The characteristics of each line code vary which makes a particular code suitable for a specific application. In choosing a line code, some of the following features are to be examined:

1. **DC Component**: Eliminating the dc energy from the signal's power spectrum enables the transmitter to be ac coupled. Magnetic recording systems or systems using transformer coupling, are less sensitive to very low frequency signal components. Thus low-frequency information may be lost, if the presence of dc or near-dc spectral components is significant in the code itself.
2. **Self-Synchronization:** Any digital communication system requires bit synchronization. Additionally coherent detectors require carrier synchronization. Some line codes have inherent synchronization features without requiring extra overhead. This feature helps in the recovery of the clock transmitter, in the receiver. For example, Manchester code has a transition at the middle of every bit interval irrespective of whether a one or a zero is being sent. This assured transition provides a clocking signal at the bit level.
3. **Error Detection**: Some codes, such as duobinary, provide the means of detecting data errors without introducing additional error-detection bits into the data sequence.
4. **Bandwidth Compression**: Some codes, such as multilevel codes, increase the efficiency of bandwidth utilization by allowing a reduction in required bandwidth for a given data rate. Thus more information is transmitted per unit bandwidth.
5. **Differential Encoding**: This technique is useful because it allows the polarity of differentially encoded waveforms to be inverted without affecting the data detection. In communication systems where waveforms sometimes experience inversion, this is of great advantage.
6. **Noise Immunity**: For same transmitted energy, some codes produce lesser bit detection errors than others in the presence of noise. For example, the NRZ waveforms have better noise performance than the RZ type.
7. **Spectral Compatibility with Channel:** One feature of spectrum matching is dc coupling. Also, the transmission bandwidth of the code must be sufficiently small compared to the channel bandwidth so that lSI problem is avoided. In cases where channel characteristic varies over different frequency bands, a line code with a similar power spectral density would be preferred over others, as the previous case is more spectrally adapted to the channel.
8. **Transparency:** A line code should be so designed that the receiver does not go out of synchronization for any sequence of data symbols. If a code is not transparent for some sequence of symbols, the clock is-lost.

3.1.3 Power Spectra of Line Codes

A line coder is the last block in the transmitter of a baseband system before the signal steps into the 'channel'. So, it is important to choose a code which is most favorable in terms of both its spectral occupancy and detection error performance.

A line code, in general, is represented by,

$$s(t) = \sum_{n=-\infty}^{\infty} a_n f(t - nT_s)$$

where a_n represents channel coded random data, *f(t)* is the symbol pulse shape which is deterministic in nature and T_s is the symbol duration. The power spectral density (PSD) of any line code signal can be written as,

$$S_s(f) = \frac{1}{T_s}|F(f)|^2 S_A(f) \qquad --(1)$$

where $F(f)$ is the Fourier transform of the deterministic pulse $f(t)$ and $S_A(f)$ is the PSD of the data sequence a_n. $S_A(f)$ is defined as

$$S_A(f) = \sum_{m=-\infty}^{\infty} R_A(m) e^{-j2\pi f m T_s}$$

If the digital data a_n is complex-valued, then its autocorrelation function $R_A(m)$ is given by

$$R_A(m) = \frac{1}{2} E[a_n a_{n+m}^*]$$

In cases where a_n is real valued, the autocorrelation function is given by a simpler formula,

$$R_A(m) = \sum_{i=1}^{I} (a_n a_{n+m})_i P_i$$

The spectrum of a line code depends on two things:

- the deterministic pulse shape used
- the statistical property of the data

3.1.3.1 Unipolar NRZ

For this code, a_n can take either values $+A$ or $0V$. It is assumed that both the levels A and 0 are equally likely. Also data sequence a_n is assumed to be independent.

For $m = 0$, the probable values of $a_n a_n$ are A^2 and 0. In this case $I = 2$. Since the data are random, probability (of getting A^2) = $1/2$ and probability (of getting 0) = $1/2$.

$$R(0) = \sum_{i=1}^{2} (a_n a_n)_i P_i = A^2 \times \frac{1}{2} + 0 \times \frac{1}{2} = \frac{A^2}{2}$$

For $m \neq 0$, the possibilities of $a_n a_{n+m}$ and the corresponding probabilities are

$$A \text{ x } A \rightarrow \frac{1}{4}$$

$$A \text{ x } 0 \rightarrow \frac{1}{4}$$

$$0 \text{ x } A \rightarrow \frac{1}{4}$$

$$0 \text{ x } 0 \rightarrow \frac{1}{4}$$

$$So, \qquad R(m) = \sum_{i=1}^{4} (a_n a_{n+m})_i P_i = A^2 \times \frac{1}{4} + 3\left\{0 \times \frac{1}{4}\right\} = \frac{A^2}{4}$$

Now, the autocorrelation of an unipolar NRZ waveform,

$$R_{uni\ NRZ}(m) = \begin{cases} \frac{A^2}{2}, & m = 0 \\ \frac{A^2}{4}, & m \neq 0 \end{cases} \qquad --(2)$$

For rectangular NRZ pulse shapes,

$$f(t) = \Pi\left(\frac{t}{T_b}\right) \qquad F(f) = T_b sinc\ (\pi f T_b)$$

For binary line codes, the symbol duration T_s and the bit interval *Tb* are identical. So, the PSD for this line code can be obtained by putting T_b in place of T_s in the PSD expression of Eq. (1).

$$S_{unipolar\ NRZ}(f) = \frac{T_b^2 sinc^2(\pi f T_b)}{T_b}\left[\frac{A^2}{2} + \sum_{m=-\infty, m\neq 0}^{\infty} \frac{A^2}{4} e^{-j2\pi f m T_b}\right]$$

$$= \frac{A^2 T_b}{4}\ sinc^2(\pi f T_b)\left[1 + \sum_{m=-\infty}^{\infty} e^{-j2\pi f m T_b}\right]$$

An impulse train can be represented in terms of its Fourier series with the help of the Poisson sum formula:

$$\sum_{n=-\infty}^{\infty} \delta(t - nT_b) = \sum_{n=-\infty}^{\infty} C_n e^{-jn\frac{2\pi}{T_b}t}$$

where $C_n = 1/Tb$ for all *n*. In the frequency domain,

$$T_b \sum_{m=-\infty}^{\infty} e^{-j2\pi f m T_b} = \sum_{m=-\infty}^{\infty} \delta\left(f - \frac{m}{T_b}\right)$$

Putting this in the PSD equation,

$$S_{unipolar\ NRZ}(f) = \frac{A^2 T_b}{4}\, sinc^2(\pi f T_b)\left[1 + \frac{1}{T_b}\sum_{m=-\infty}^{\infty} \delta\left(f - \frac{m}{T_b}\right)\right]$$

At $f = \frac{1}{T_b}$, $sinc(\pi\, f\, T_b) = sinc(\pi m) = 0$ except at $m = 0$. So, the PSD expression becomes,

$$S_{unipolar\ NRZ}(f) = \frac{A^2 T_b}{4}\, sinc^2(\pi f T_b)\left[1 + \frac{1}{T_b}\delta(f)\right]$$

Choose A in such a way that normalized average power P_n of the unipolar NRZ signal becomes one. Assume that the bit pattern is the worst case sequence 1010.... for normalizing A.

$$P_n = \frac{1}{T_0}\int_0^{T_0} s^2(t)\, dt = \frac{1}{2T_b}\int_0^{T_0} A^2\, dt = \frac{A^2}{2}$$

Putting $P_n = 1$, we get $A = \sqrt{2}$. Hence, the normalized Power spectral density of the unipolar NRZ waveform is given by,

$$\overline{S_{unipolar\ NRZ}(f)} = \frac{T_b}{4}\, sinc^2(\pi f T_b)\left[1 + \frac{1}{T_b}\delta(f)\right]$$

The PSD is plotted in Fig. 3.2.

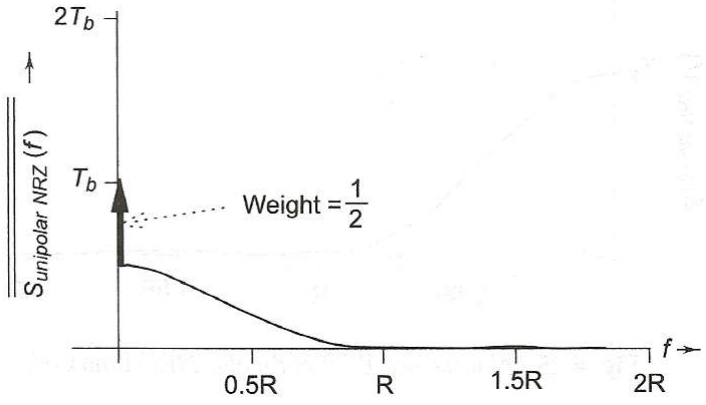


Fig. 3.2 ***Normalized PSD of Unipolar NRZ Line Code***

As seen from the plot, the unipolar NRZ contains large power in its dc component. Usually the dc component does not carry any useful information, so the power in it gets wasted. If this line code is used, then the channel should be able to pass dc which requires dc coupled circuits. Another important feature is that the impulse is present only at dc, so clock cannot be extracted

from this code. For a pulse energy of E_p the probability of error for this code is given by $P_e = Q\left(\sqrt{\frac{E_p}{\eta}}\right)$.

The advantages of an unipolar NRZ waveform,

- It is very simple to generate.
- This code requires only one power supply.

Standard TTL and CMOS circuits can be used to implement unipolar NRZ line code circuitry.

3.1.3.2 Bipolar NRZ

Possible values for data element are: $+A\ or\ -A\ Volts$. It is assumed that both these levels are equally likely.

$$R(0) = \sum_{i=1}^{2} (a_n a_n)_i P_i = A^2 \times \frac{1}{2} + (-A)^2 \times \frac{1}{2} = A^2$$

$$R(m, m \neq 0) = \sum_{i=1}^{4} (a_n a_{n+m})_i P_i$$

$$= A^2 \times \frac{1}{4} + (-A)(A) \times \frac{1}{4} + (A)(-A) \times \frac{1}{4} + (-A)(-A) \times \frac{1}{4} = 0$$

For rectangular NRZ pulse shapes, the PSD expression becomes,

$$S_{bipolar\ NRZ}(f) = A^2 T_b\ sinc^2(\pi f T_b)$$

Normalized average power for the worst case sequence of a bipolar NRZ signal is

$$P_n = \frac{1}{2T_b}\left[\int_0^{T_b} A^2\, dt + \int_{T_b}^{2T_b} (-A)^2\, dt\right] = A^2$$

Putting $P_n = 1$, we get $A = 1$. Hence, the normalised power spectral density of the bipolar NRZ waveform is given by,

$$\overline{\overline{S_{bipolar\ NRZ}(f)}} = T_b\ sinc^2(\pi f T_b)$$

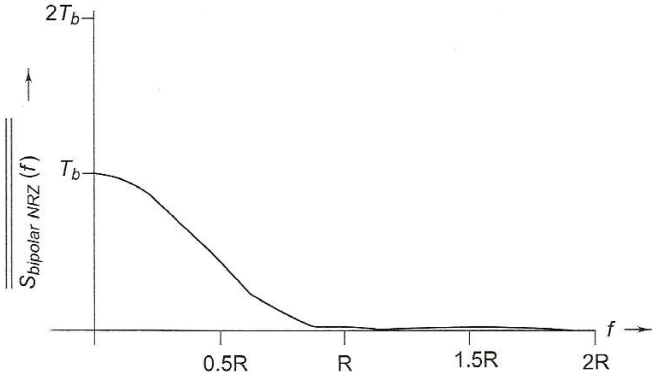


Fig. 3.3 ***Normalized PSD of Bipolar NRZ Line Code***

As seen from the plot of PSD in Fig. 3.3, a bipolar NRZ code also contains large power in the dc component and there is no possibility of clock extraction from this code. The error probability of this code is the lowest among all line codes $P_e = Q\left(\sqrt{\frac{2E_p}{\eta}}\right)$.

The drawback of bipolar NRZ waveform is the requirement of two distinct power supplies. Nowadays lCs are available that generate dual supply voltage from a single power supply.

Example 3.1 The RS-232 serial port on a personal computer is transmitting 2.4 kbps data using a bipolar NRZ line code that has a peak value of 12 V. Assume that binary 1's and O's are equally likely to occur. Compute the P.S.D. for this RS-232 signal.

$$S_{bipolar\ NRZ}(f) = A^2 T_b\ sinc^2(\pi f T_b)$$

$$Given, \qquad T_b = \frac{1}{R_b} = \frac{1}{2400} sec$$

$$S_{bipolar\ NRZ}(f) = 12^2 . \frac{1}{2400}\ sinc^2\left(\pi \frac{f}{2400}\right)$$

$$= 0.06\ sinc^2\left(\pi \frac{f}{2400}\right)$$

3.1.3.3 Unipolar RZ

Due to the unipolar nature of the data sequence, the autocorrelation function of this code is same as that of the unipolar NRZ as shown in Eq. (2).

RZ signaling changes the Fourier transform of the pulse shape. In this case, the pulse duration is $\frac{T_b}{2}$. So, $F(f)$ becomes,

$$F(f) = \frac{T_b}{2} sinc\left(\pi f \frac{T_b}{2}\right)$$

and hence, the unnormalized PSD of unipolar RZ waveform takes the following form

$$S_{unipolar\ RZ}(f) = \frac{T_b^2}{4T_b} sinc^2\left(\pi f \frac{T_b}{2}\right)\left[\frac{A^2}{4} + \sum_{m=-\infty}^{\infty} \frac{A^2}{4}\frac{1}{T_b}\,\delta\left(f - \frac{m}{T_b}\right)\right]$$

$$= \frac{A^2 T_b}{16} sinc^2\left(\pi f \frac{T_b}{2}\right)\left[1 + \frac{1}{T_b}\sum_{m=-\infty}^{\infty} \delta\left(f - \frac{m}{T_b}\right)\right]$$

Normalized average power for the worst case sequence of a unipolar RZ signal is

$$P_n = \frac{1}{2T_b}\int_0^{T_b/2} A^2\, dt = \frac{A^2}{4}$$

Putting $P_n = 1$, we get $A = 2$. Hence, the normalized power spectral density of the unipolar RZ waveform is given by,

$$\overline{\overline{S_{unipolar\ RZ}(f)}} = \frac{T_b}{4} sinc^2\left(\pi f \frac{T_b}{2}\right)\left[1 + \frac{1}{T_b}\sum_{m=-\infty}^{\infty} \delta\left(f - \frac{m}{T_b}\right)\right]$$

As seen from the plot of PSD in Fig. 3.4, unipolar RZ code contains reduced power in the dc and low frequency components. The first null bandwidth is 2R which is twice of NRZ case, since the pulse duration of RZ is $T_b/2$. The probability of error for this code is $P_e = Q\left(\sqrt{\frac{2E_p}{\eta}}\right)$.

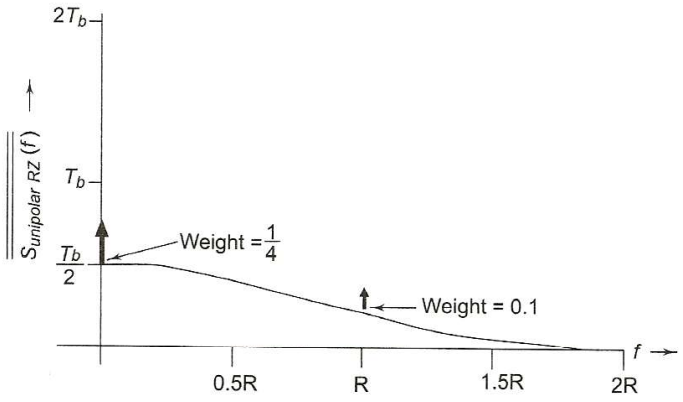


Fig. 3.4 ***Normalized PSD of Unipolar* RZ *Line Code***

So, to maintain same P_e, 3 dB more power is needed in unipolar signaling compared to the power needed in bipolar signaling.

The advantage of this code is the presence of a discrete impulse train at every clock frequency. Thus this code has self-clock extracting capability.

3.1.3.4 RZ AMI

This is an example of a multilevel binary code. Possible values for data element a_n are: $+A, -A$ and $0V$. 1 is represented by alternating $+A and -A$. 0 is represented by 0. For $m = 0$, the possibilities of $a_n a_n$ are A^2 and 0 with probability 1/2. So, $R(0) = \frac{A^2}{2}$. For m = 1, possible data sequence and the corresponding probabilities are

$$\pm A \text{ x } \mp A \rightarrow \frac{1}{4}$$

$$\pm A \text{ x } 0 \rightarrow \frac{1}{4}$$

$$0 \text{ x } \pm A \rightarrow \frac{1}{4}$$

$$0 \text{ x } 0 \rightarrow \frac{1}{4}$$

$$So, \quad R(1) = \frac{-A^2}{4}$$

For m>1, possibilities are

$$1\ 00 \rightarrow A \text{ x } 0 \rightarrow \frac{1}{8}$$
$$1\ 01 \rightarrow A \text{ x } -A \rightarrow \frac{1}{8}$$
$$1\ 10 \rightarrow A \text{ x } 0 \rightarrow \frac{1}{8}$$
$$1\ 11 \rightarrow A \text{ x } A \rightarrow \frac{1}{8}$$
$$0\ 00 \rightarrow 0 \text{ x } 0 \rightarrow \frac{1}{8}$$
$$0\ 01 \rightarrow 0 \text{ x } \pm A \rightarrow \frac{1}{8}$$
$$0\ 10 \rightarrow 0 \text{ x } 0 \rightarrow \frac{1}{8}$$
$$0\ 11 \rightarrow 0 \text{ x } \pm A \rightarrow \frac{1}{8}$$
$$So, \ R(m > 1) = \frac{-A^2}{8} + \frac{A^2}{8} = 0$$

Hence, the autocorrelation of an AMI RZ code is

$$R_{AMI\,RZ}(m) = \begin{cases} \frac{A^2}{2}, & m = 0 \\ \frac{-A^2}{4} & |m| = 1 \\ 0 & |m| > 1 \end{cases}$$

So, the unnormalised PSD of AMI RZ waveform becomes

$$S_{AMI\,RZ}(f) = \frac{T_b^2}{4T_b} sinc^2\left(\pi f \frac{T_b}{2}\right)\left[\frac{A^2}{2} - \frac{A^2}{4}e^{-j2\pi f T_b} - \frac{A^2}{4}e^{j2\pi f T_b}\right]$$

$$= \frac{A^2 T_b}{4} sinc^2\left(\pi f \frac{T_b}{2}\right)\sin^2(\pi f T_b)$$

Normalised average power of the worst case sequence of a AMI RZ code is

$$P_n = \frac{1}{2T_b}\int_0^{T_b/2} (\pm A)^2\, dt = \frac{A^2}{4}$$

Putting $P_n = 1$, we get $A = 2$. Hence, the normalised power spectral density of the AMI RZ waveform is given by,

$$\overline{\overline{S_{AMI\,RZ}(f)}} = T_b sinc^2\left(\pi f \frac{T_b}{2}\right)\sin^2(\pi f T_b)$$

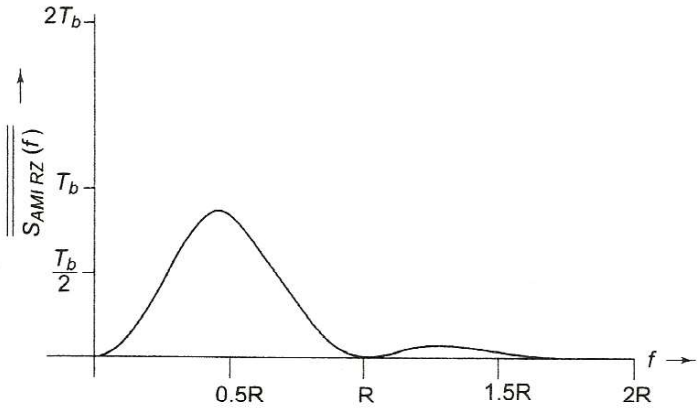


Fig. 3.5 ***Normalised PSD of AMI* RZ *Line Code***

The following features are seen from the PSD of AMI RZ code shown in Fig. 3.5

- Null at dc. *So,* this code is most efficient in ac coupling.
- Due to alternating A's, single error can be easily detected.

- Clock can be extracted by converting AMI RZ to unipolar RZ.
- The probability of error for this code is $P_e = \frac{3}{2} Q\left(\sqrt{\frac{2E_p}{\eta}}\right)$, So, this code is most prone to error.
- To maintain same P_e, 3 dB more power is required than other codes.
- AMI codes are not transparent, i.e. a string of zeroes may cause a loss in the clock signal. This is prevented by using high density bipolar n (HDBn) signaling where a string of more than n consecutive zeroes is replaced by a 'filling' sequence that contains marking pulses.
- Receiver needs to distinguish between three levels viz, $A, -A$ and 0 for AMI line coded signals.

3.1.3.5 Manchester NRZ

The pulse shape of Manchester NRZ code can be written as,

$$F(t) = \prod\left(\frac{t + \frac{T_b}{4}}{\frac{T_b}{2}}\right) - \prod\left(\frac{t - \frac{T_b}{4}}{\frac{T_b}{2}}\right)$$

So, its Fourier Transform is

$$F(f) = \frac{T_b}{2}\left[\frac{\sin(\pi f T_b/2)}{\pi f T_b/2}\right] e^{j\omega T_b/4} - \frac{T_b}{2}\left[\frac{\sin(\pi f T_b/2)}{\pi f T_b/2}\right] e^{-j\omega T_b/4}$$

$$= jT_b \,\mathrm{sinc}(\pi f T_b/2)\,\mathrm{sinc}(\omega T_b/4)$$

On a bit by bit basis, a Manchester code is a bipolar NRZ signal. So its autocorrelation is given by

$$R_{Manchester\ NRZ}(m) = \begin{cases} A^2 & m = 0 \\ 0 & m \neq 0 \end{cases}$$

So, the unnormalized PSD of Manchester NRZ waveform becomes

$$S_{Manchester\ NRZ}(f) = A^2 T_b sinc^2\left(\pi f \frac{T_b}{2}\right) \sin^2\left(\pi f \frac{T_b}{2}\right)$$

For normalized average power, A becomes 1.

The following features are seen from the PSD of Manchester NRZ code in Fig. 3.6.

- Null at dc.
- Due to alternating A's, single error can be easily detected.
- The probability of error for this code is $P_e = Q\left(\sqrt{\frac{2E_p}{\eta}}\right)$.

- The code is transparent i.e. string of zeroes will not cause loss of clock.
- Null bandwidth is twice that in the bipolar case.

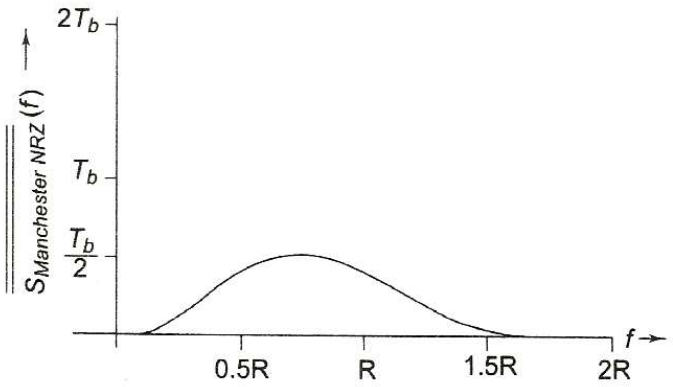


Fig. 3.6 ***Normalized PSD of Manchester NRZ Line Code***

The error performance of various line codes are shown in Fig. 3.7 for easy comparison.

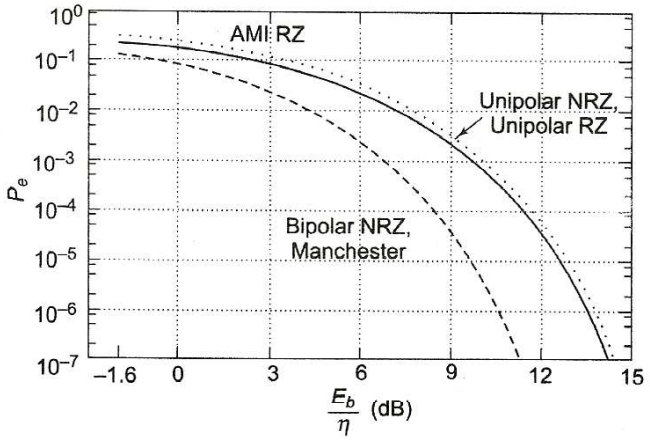


Fig. 3.7 ***Error Probability versus Received Power of Different Line Codes***

Example 3.2 Let a line code $s(t)$ is represented as,

$$s(t) = \sum_{n=-\infty}^{\infty} a_n f(t - nT_s)$$

where, $f(t) = \prod\left(\frac{t}{T_b}\right)$

and $a_k = g_0 A_k + g_1 A_{k-1}$

Here, A_k is a random variable with two equiprobable outcomes $\pm A$ such that

$$E\left[A_k A_{k+m} = \begin{cases} A^2 & m = 0 \\ 0 & m \neq 0 \end{cases}\right]$$

Find the power spectral density of this line code.

Table 3.1 $m = 0$ *and* $m = \pm 1$

A_{k-1}	A_k	A_{k+1}	a_k	a_{k+1}	$a_k\, a_{k+1}$	*Probability*
A	A	A	$A(g_0+g_1)$	$A(g_0+g_1)$	$A^2(g_0+g_1)^2$	1/8
A	A	$-A$	$A(g_0+g_1)$	$A(-g_0+g_1)$	$-A^2(g_0^2-g_1^2)$	1/8
A	$-A$	A	$A(-g_0+g_1)$	$A(g_0-g_1)$	$-A^2(g_0-g_1)^2$	1/8
A	$-A$	$-A$	$A(-g_0+g_1)$	$-A(g_0+g_1)$	$A^2(g_0^2-g_1^2)$	1/8
$-A$	A	A	$A(g_0-g_1)$	$A(g_0+g_1)$	$A^2(g_0^2-g_1^2)$	1/8
$-A$	A	$-A$	$A(g_0-g_1)$	$A(-g_0+g_1)$	$-A^2(g_0-g_1)^2$	1/8
$-A$	$-A$	A	$-A(g_0+g_1)$	$A(g_0-g_1)$	$-A^2(g_0^2-g_1^2)$	1/8
$-A$	$-A$	$-A$	$-A(g_0+g_1)$	$-A(g_0+g_1)$	$A^2(g_0+g_1)^2$	1/8

For $m = 0$, the distribution of the random variable a_k can be written by looking Table 3.1

$$E[a_k a_k] = \frac{4}{8}[A^2(g_0+g_1)^2] + \frac{4}{8}[A^2(g_0-g_1)^2] = A^2(g_0^2+g_1^2)$$

For $m = 1$, the distribution is shown in Table 3.1. From the table, we can write,

$$E[a_k a_{k+1}] = \frac{A^2}{8}[2(g_0+g_1)^2 - 2(g_0-g_1)^2] = A^2 g_0 g_1$$

For $m > 1$, the distribution of $a_k a_{k+m}$ is identical for all m as a_k depends only on the present and past one sample value of A_k. From Table 3.2,

$$E[a_k a_{k+1}] = 0$$

Table 3.2 $m > 1$

A_{k-1}	A_k	A_{k+m-1}	A_{k+m}	a_k	a_{k+m}	$a_k a_{k+m}$	Probability
$-A$	$-A$	$-A$	$-A$	$-A(g_0+g_1)$	$-A(g_0+g_1)$	$A^2(g_0+g_1)^2$	1/8
$-A$	$-A$	$-A$	A	$-A(g_0+g_1)$	$A(g_0-g_1)$	$-A^2(g_0^2-g_1^2)$	1/8
$-A$	$-A$	A	$-A$	$-A(g_0+g_1)$	$-A(g_0-g_1)$	$A^2(g_0^2-g_1^2)$	1/8
$-A$	$-A$	A	A	$-A(g_0+g_1)$	$A(g_0+g_1)$	$-A^2(g_0+g_1)^2$	1/8
$-A$	A	$-A$	$-A$	$A(g_0-g_1)$	$-A(g_0+g_1)$	$-A^2(g_0^2-g_1^2)$	1/8
$-A$	A	$-A$	A	$A(g_0-g_1)$	$A(g_0-g_1)$	$A^2(g_0-g_1)^2$	1/8
$-A$	A	A	$-A$	$A(g_0-g_1)$	$-A(g_0-g_1)$	$-A^2(g_0-g_1)^2$	1/8
$-A$	A	A	A	$A(g_0-g_1)$	$A(g_0+g_1)$	$A^2(g_0^2-g_1^2)$	1/8
A	$-A$	$-A$	$-A$	$-A(g_0-g_1)$	$-A(g_0+g_1)$	$A^2(g_0^2-g_1^2)$	1/8
A	$-A$	$-A$	A	$-A(g_0-g_1)$	$A(g_0-g_1)$	$-A^2(g_0-g_1)^2$	1/8
A	$-A$	A	$-A$	$-A(g_0-g_1)$	$-A(g_0-g_1)$	$A^2(g_0-g_1)^2$	1/8
A	$-A$	A	A	$-A(g_0-g_1)$	$A(g_0+g_1)$	$-A^2(g_0^2-g_1^2)$	1/8
A	A	$-A$	$-A$	$A(g_0+g_1)$	$-A(g_0+g_1)$	$-A^2(g_0+g_1)^2$	1/8
A	A	$-A$	A	$A(g_0+g_1)$	$A(g_0-g_1)$	$A^2(g_0^2-g_1^2)$	1/8
A	A	A	$-A$	$A(g_0+g_1)$	$-A(g_0-g_1)$	$-A^2(g_0^2-g_1^2)$	1/8
A	A	A	A	$A(g_0+g_1)$	$A(g_0+g_1)$	$A^2(g_0+g_1)^2$	1/8

So, the autocorrelation becomes

$$R[m] = \begin{cases} A^2(g_0^2+g_1^2), & m=0 \\ g_0 g_1 A^2, & m=\pm 1 \\ 0, & otherwise \end{cases}$$

For rectangular pulse shapes,

$$f(t) = \prod\left(\frac{t}{T_b}\right)$$

$$F(f) = T_b sinc(\pi f T_b)$$

Hence, the PSD of this line code is

$$S(f) = \frac{|F(f)|^2}{T_b}\sum_{m=-\infty}^{\infty} R[m]\, e^{-j2\pi f m T}$$

$$= A^2 T_b sinc^2(\pi f T_b)[g_0^2+g_1^2+2g_0 g_1 \cos(2\pi f T_b)]$$

3.2 INTERSYMBOL INTERFERENCE

Fig. 3.8 shows the basic elements of a *baseband binary PAM system.* The input signal consists of a binary data sequence $\{b_k\}$ with a bit duration of T_b seconds.

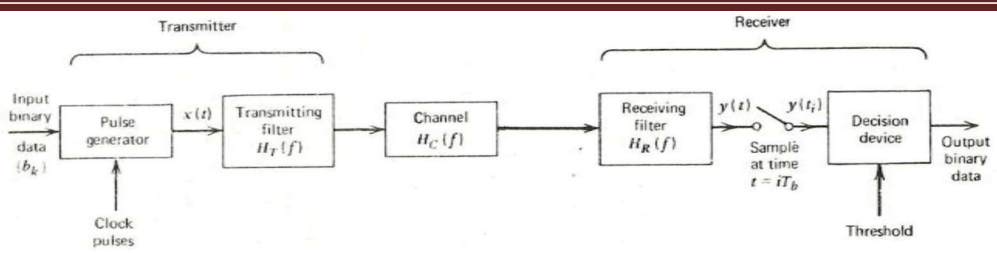


Fig. 3.8 Baseband Binary data transmission system

This sequence is applied to a pulse generator, producing the discrete PAM signal

$$x(t) = \sum_{k=\infty}^{\infty} a_k v(t - kT_b)$$

where $v(t)$ denotes the basic pulse (shaping pulse), normalized such that $v(0) = 1$. The coefficient a_k, depends on the input data and the type of format used.

$$a_k = \begin{cases} +a, & if\ b_k = 1 \\ -a, & if\ b_k = 0 \end{cases}$$

The signal $x(t)$ is then passed through the transmitting filter. The transmitting filter combines all the necessary transmitting circuits and systems. The combined transfer function of the transmitting filter is $H_T(f)$. The signal is then passed through the channel having the transfer function $H_C(f)$.The channel delivers the signal to the reciving filter. It consists of all the necessary receiving circuits and systems. The combined transfer function of the receiving filter is $H_R(f)$ The output of the receiving filter is $y(t)$. This $y(t)$ is noisy replica of the transmitted signal $x(t)$

The signal $y(t)$ is sampled synchrnously with transmitter. The sampling instants at the transmitter. The sampled instants are sysnchronous to the clock pulses at the transmitter. The sampled signal $y(t_i)$ is then given to the decision device. The decision device compares the input signal with threshold $'\lambda'$. Then the decsion is taken as follows:

$If\ y(t_i) > \lambda$ select symbol '1'

$If\ y(t_i) \leq \lambda$ select symbol '0'

The receiving filter output can be written as,

$$y(t) = \mu \sum_{k=\infty}^{\infty} a_k p(t - kT_b)$$

where μ is a scaling factor and the pulse $p(t)$ is *normalized* such that

$$p(0) = 1 \qquad --(1)$$

In fig 3.8 $a_k v(t)$ is the signal applied to the input of cascade of transmitting filter, channel and receiving filter. The output of this cascade connection is $\mu a_k p(t)$

Let the fourier transform of $v(t)$ be $V(f)$ and that of $p(t)$ be $P(f)$. Then in frequency domain,

$$\mu\, P(f) = V(f)H_T(f)H_C(f)H_R(f)$$

Note that the normalization of $p(t)$ as in eqn (1) means that the total area under the curve of $P(f)$ equals unity

The receiving filter output $y(t)$ is sampled at time $t_i = iT_b$ (with i takes integer values),

$$y(t_i) = \mu \sum_{k=\infty}^{\infty} a_k p(iT_b - kT_b), \qquad i = 0, \pm 1, \pm 2, \ldots$$

$$= \mu a_i p(0) + \mu \sum_{\substack{k=\infty \\ k \neq i}}^{\infty} a_k p(iT_b - kT_b)$$

$$= \mu a_i + \mu \sum_{\substack{k=\infty \\ k \neq i}}^{\infty} a_k p(iT_b - kT_b) \qquad \because p(0) = 1 \qquad --(2)$$

In Eq. (2), the first term μa_i is produced by the i^{th} transmitted bit. The second term represents the residual effect of all other transmitted bits on the decoding of the i^{th} bit; this residual effect is called *intersymbol interference* (lSI).

In physical terms, lSI arises because of imperfections in the overall frequency response of the system. When a short pulse of duration T_b seconds is transmitted through a band-limited system, the frequency components constituting the input pulse are differentially attenuated and differentially delayed by the system. As a result, the pulse appearing at the output of the system is *dispersed* over an interval longer than T_b seconds. Thus, when a sequence of short pulses (representing binary 1s and 0s) are transmitted through the system, one pulse every T_b seconds, the dispersed responses originating from different symbol intervals will interfere with each other, thereby resulting in intersymbol interference.

In the absence of lSI, Eq. (2) becomes,

$$y(t_i) = \mu a_i$$

which shows that, under these conditions, the i^{th} transmitted bit can be decoded correctly. The presence of ISI in the system, however, introduces errors in the decision device at the receiver output. Therefore, in the design of the transmitting and receiving filters, the objective is to

minimize the effects of ISI, and thereby deliver the digital data to the receiver with the smallest error rate.

3.3 NYQUIST CRITERION FOR DISTORTIONLESS BASEBAND BINARY TRANSMISSION

Typically, the transfer function of the channel and the transmitted pulse shape are specified, and the problem is to determine the transfer functions of the transmitting and receiving filters so as to reconstruct the transmitted da~ sequence $\{b_k\}$. The receiver does this by *extracting* and then *decoding* the corresponding sequence of weights, $\{a_k\}$, from the output $y(t)$. Except for a scaling factor, $y(t)$ is determined by the a_k and the received pulse $p(t)$. The *extraction* involves sampling the output $y(t)$ at $t = iT_b$. The *decoding* requires that the weighted pulse contribution $a_k p(iT_b - kT_b)$ for $k = i$ be *free* from lSI due to the overlapping tails of all other weighted pulse contributions represented by $k \neq i$. This, in turn, requires that we *control* the received pulse *p(t),* as shown by

$$p(iT_b - kT_b) = \begin{cases} 1, & i = k \\ 0, & i \neq k \end{cases} \qquad --(3)$$

where, by normalization, $p(0) = 1$. If $p(t)$ satisfies the condition of Eq. (3), the receiver output, given by Eq. (2), simplifies to

$$y(t_i) = \mu a_i$$

which implies zero intersymbol interference. Hence, the Eq. (3) gives perfect reception in the absence of noise.

Let $p(nTb)$, represent the impulses at which $p(t)$ is sampled for decision. These samples are taken at the rate of T_b. Fourier spectrum of these impulses is given as,

$$P_\delta(f) = R_b \sum_{n=-\infty}^{\infty} P(f - nR_b) \qquad --(4)$$

where $R_b = 1/T_b$ is the bit rate. $P_\delta(f)$ is the Fourier transform of an infinite periodic sequence of delta functions of period T_b and whose strengths are weighted by the respective sample values of $p(t)$. That is, $P_\delta(f)$ is given by,

$$P_\delta(f) = R_b \int_{-\infty}^{\infty} \sum_{n=-\infty}^{\infty} [p(mT_b)\delta(t - nT_b)] \exp(-j2\pi f t)\, dt \qquad --(5)$$

Let the integer $m = i - k$. Then, $i = k$ corresponds to $m = 0$, and likewise $i \neq k$ corresponds to $m \neq 0$. Accordingly, imposing the condition of Eq. (3) on the sample values of $p(t)$ in the integral of Eq. (5), we get

$$P_\delta(f) = \int_{-\infty}^{\infty} p(0)\delta(t)\exp(-j2\pi ft)\,dt$$
$$= p(0) \qquad --(6)$$

using shifting property of to delta function. Since $p(0) = 1$, by normalization, from Eqs. (4) and (6) that the condition for zero intersymbol interference is satisfied if

$$\sum_{n=-\infty}^{\infty} P(f - nR_b) = T_b \qquad --(7)$$

Equation (3) formulated in terms of the time function $p(t)$, or equivalently, Eq, (7) formulated in terms of the corresponding frequency function $P(f)$, constitutes the *Nyquist criterion for distortion less baseband transmission* in the absence of noise. It provides a method for constructing band-limited functions to overcome the effects of intersymbol interference. The method depends on sampling the received signal at midpoints of the signaling intervals.

3.3.1 Ideal Solution

A frequency function $P(f)$, occupying the narrowest band, that satisfies Eq. (7) is obtained by permitting only one nonzero component in the series (on the left side of the equation) for each f in the range extending from $-B_0$ *to* B_0, where B_0 denotes half the bit rate:

$$B_0 = \frac{R_b}{2}$$

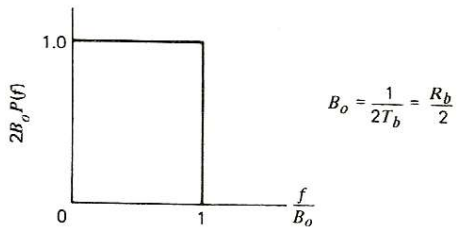
That is, we specify $P(f)$ as

$$P(f) = \frac{1}{2B_0} rect\left(\frac{f}{2B_0}\right)$$

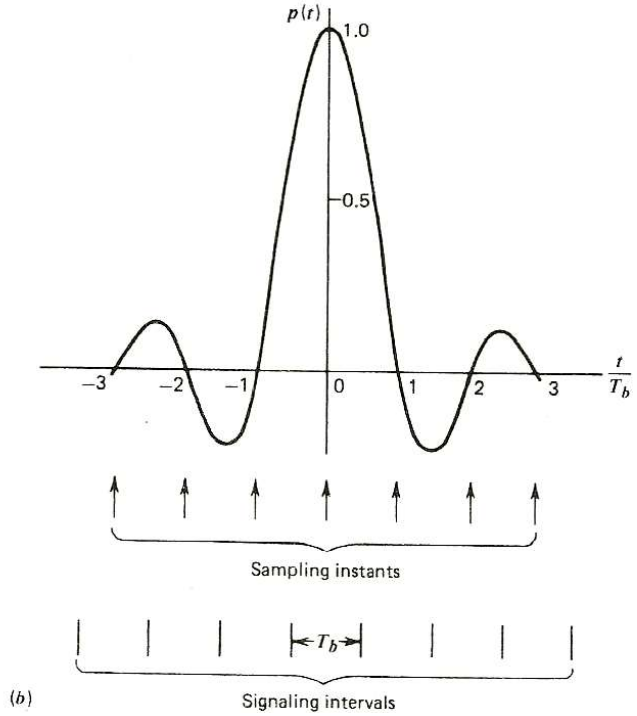
In this solution, no frequencies of absolute value exceeding half the bit rate are needed. Hence, one signal waveform that produces zero intersymbol interference is defined by the *sine function:*

$$p(t) = \frac{\sin(2\pi B_0 t)}{2\pi B_0 t}$$
$$= \text{sinc}(2B_0 t)$$

Figures *3.9a* and *3.9b* show plots of $P(f)$ and $p(t)$, respectively. In Fig. *3.9a,* the frequency function $P(f)$ is shown for positive frequencies only. In Fig. *3.9b* the signaling intervals and the corresponding centered sampling instants are included.



(a)



(b)

Figure 3.9 ***(a)*** **Ideal amplitude response.** ***(b)*** **Ideal basic pulse shape.**

The function $p(t)$ can be regarded as the impulse response of an ideal low-pass filter with passband amplitude response $1/(2B_0)$ and bandwidth B_0. The function $p(t)$ has its peak value at the origin and goes through zero at integer multiples of the bit duration T_s. It is apparent that if the received waveform $y(t)$ is sampled at the instants of time $t = 0, \pm T_b \pm 2T_b, \ldots$, then the pulses defined by $\mu p(t - iT_b)$with arbitrary amplitude μ and index $i = 0, \pm 1, \pm 2, \ldots$, will not interfere with each other.

Although this choice of pulse shape for $p(t)$ achieves economy in bandwidth in that it solves the problem of zero intersymbol interference with the minimum bandwidth possible, there are two practical difficulties that make it an undesirable objective for system design:

1. It requires that the amplitude characteristic of $P(f)$ be flat from $- B_0$ *to* B_0, and zero elsewhere. This is physically unrealizable because of the abrupt transitions at $\pm B_0$.

2. The function $p(t)$ decreases as $1/|t|$for large |t|' resulting in a slow rate of decay. This is caused by the discontinuity of $P(f)$ at $\pm B_0$. Accordingly, there is practically no margin of error in sampling times in the receiver.

To evaluate the effect of this *timing error,* consider the sample of *y(t)* at$t = \Delta t$, where Δt is the timing error. To simplify the analysis we have put the correct sampling time t_i, equal to zero. We thus obtain, in the absence of noise

$$y(\Delta t) = \mu \sum_k a_k p(\Delta t - kT_b)$$

$$= \mu \sum_k a_k \frac{\sin[2\pi B_0(\Delta t - kT_b)]}{2\pi B_0(\Delta t - kT_b)} \quad --(8)$$

Since $2B_0T_b = 1$, *we may rewrite eqn* (8)*as,*

$$y(\Delta t) = \mu \sum_k a_k sinc(2B_0\Delta t - k)$$

$$= \mu a_0 sinc\,(2B_0\Delta t) + \frac{\mu sin(2\pi B_0\Delta t)}{\pi} \sum_{\substack{k \\ k\neq 0}} \frac{(-1)^k a_k}{2B_0\Delta t - k} \quad --(9)$$

The first term on the right side of Eq. (9) defines the desired symbol, whereas the remaining series represents the intersymbol interference caused by the timing error Δt in sampling the output $y(t)$. In certain cases, it is possible for this series to diverge, thereby causing erroneous decisions in the receiver.

3.3.2 Practical Solution

We may overcome the practical difficulties posed by the ideal solution by extending the bandwidth from $B_0 = R_b l2$ to an adjustable value between B_o *and* $2B_o$. In so doing, we permit three components in the series on the left side of Eq. (7) for$|f| \leq B_0$, as shown by

$$P(f) + P(f - 2B_0) + P(f + 2B_0) = \frac{1}{2B_0} \quad -B_0 \leq f \leq B_0 \quad --(10)$$

We may devise several band-limited functions that satisfy Eq. (10). A particular form of $P(f)$ that embodies many desirable features is constructed by a *raised cosine spectrum.* This frequency characteristic consists of a flat portion and a *rolloff* portion that has a sinusoidal form, as follows

$$P(f) = \begin{cases} \dfrac{1}{2B_0}, & |f| < f_1 \\ \dfrac{1}{4B_0}\left\{1 + \cos\left[\dfrac{\pi|f| - f_1}{2B_0 - 2f_1}\right]\right\}, & f_1 \leq |f| < 2B_0 - f_1 \\ 0, & |f| \geq 2B_0 - f_1 \end{cases}$$

The frequency f_1, and bandwidth B_0 are related by

$$\alpha = 1 - \frac{f_1}{B_0}$$

The parameter α is called the *rolloff factor.* For $\alpha = 0$, that is. $f_1 = Bo$, we get the minimum bandwidth solution described earlier.

The frequency response $P(f)$, normalized by multiplying it by $2B_o$, is shown plotted in Fig. *3.10a* for three values of α namely, 0, 0.5, and 1. We see that for $\alpha = 0.5$ or 1, the-rolloff characteristic of $P(f)$ cuts off gradually as compared with an ideal low-pass filter (corresponding to $\alpha = 0$), and it is therefore easier to realize in practice.

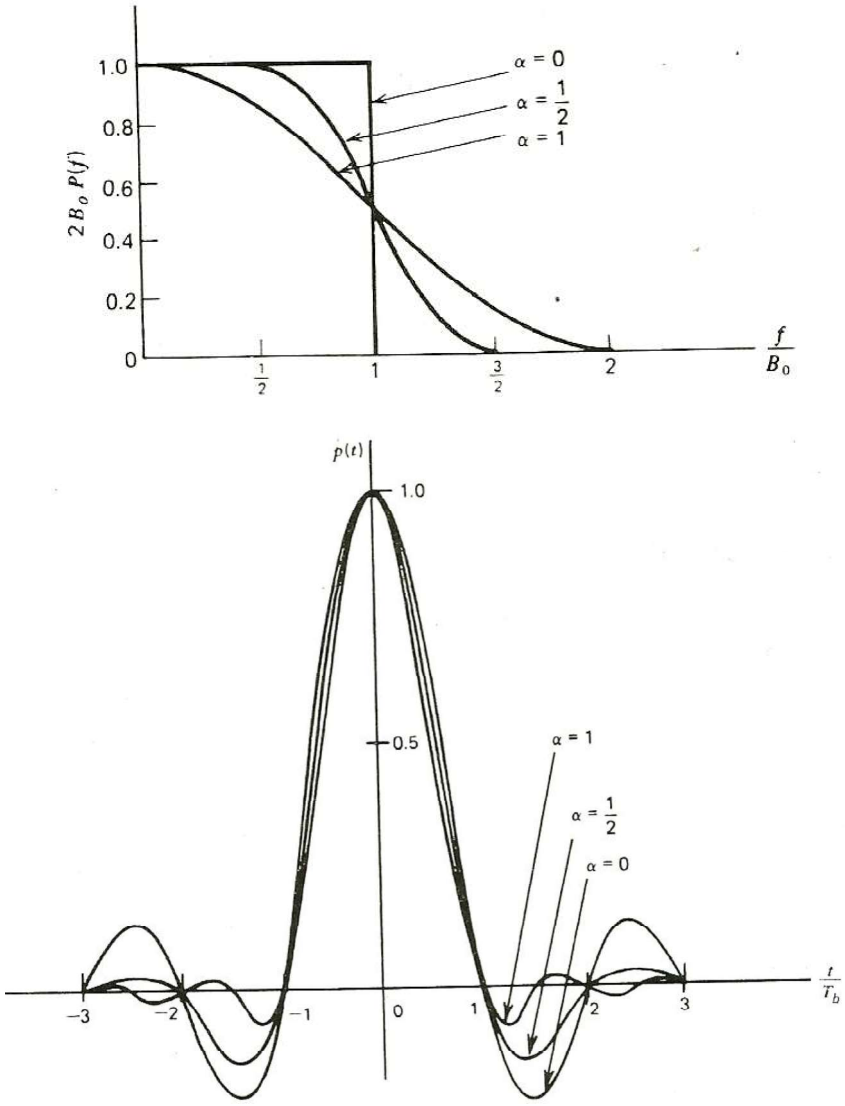


Figure 3.10 Responses for different rolloff factors. (a) Frequency response. *(b)* Time response. Note that *Bo = 1/2h.*

Also the function $P(f)$ exhibits odd symmetry about the cutoff frequency B_o of the ideal low-pass filter. The time response $p(t)$, that is, the inverse Fourier transform of $P(f)$, is defined by

$$p(t) = sinc(2B_0t)\frac{\cos(2\pi\alpha B_0 t)}{1-16\alpha^2 B_0^2 t^2}$$

This function consists of the product of two factors: the factor $sinc(2B_ot)$ associated with the ideal filter, and a second factor that decreases as $1/|t|^2$for large $|t|$. The first factor ensures zero crossings of $p(t)$ at the desired sampling instants of time $t = iT_b$ with i an integer (positive and negative). The second factor reduces the tails of the pulse considerably below that obtained from the ideal low-pass filter, so that the transmission of binary waves using such pulses is relatively insensitive to sampling time errors. In fact, the amount of intersymbol interference resulting from this timing error decreases as the rolloff factor α is increased from zero to unity.

The time response $p(t)$ is shown plotted in Fig. *3.10b* for α = 0,0.5, and 1. For the special case of α = 1, the function $p(t)$ simplifies to

$$p(t) = \frac{sinc(4B_0t)}{1-16B_0^2t^2}$$

This time response exhibits two interesting properties:

1. At $t = \pm T_b/2 = \pm 1I4B_o$, we have $p(t) = 0.5$; that is, the pulse width measured at half amplitude is exactly equal to the bit duration T_b.

2. There are zero crossings at $t = \pm 3T_b/2, \pm 5T_b/2, \ldots$ in addition to the usual zero crossings at the sampling times $t = \pm Ti; = 2Tb, \ldots$.

These two properties are particularly useful in generating a timing signal from the received signal for the purpose of synchronization. However, this requires the use of a transmission bandwidth double that required for the ideal case corresponding to $\alpha = 0$.

3.4 CORRELATIVE CODING

If intersymbol interference is added to the transmitted signal in a controlled manner, it is possible to achieve a bit rate of $2B_0$ bits per second in a channel of bandwidth B_o hertz. Such schemes are called **correlative coding or partial-response signaling schemes**. Thus correlative coding helps in achieving the theoretical maximum signaling rate of $2B_o$ bits per second in a bandwidth of B_0 hertz, as postulated by Nyquist, using realizable and perturbation-tolerant filters.

3.4.1 Duobinary Signaling

In **duobinary signaling** "duo" implies doubling of the transmission capacity of a straight binary system. Consider a binary input sequence $\{bk\}$ consisting of uncorrelated binary digits each having duration T_b seconds, with symbol 1 represented by a pulse of amplitude $+1$ $volt$, and

symbol 0 by a pulse of amplitude $-1\ volt$. When this sequence is applied to a **duobinary encoder**, it is converted into a **three-level output**, namely, $-2, 0, and\ +2\ volts$. This scheme is illustrated in Fig. 3.11.

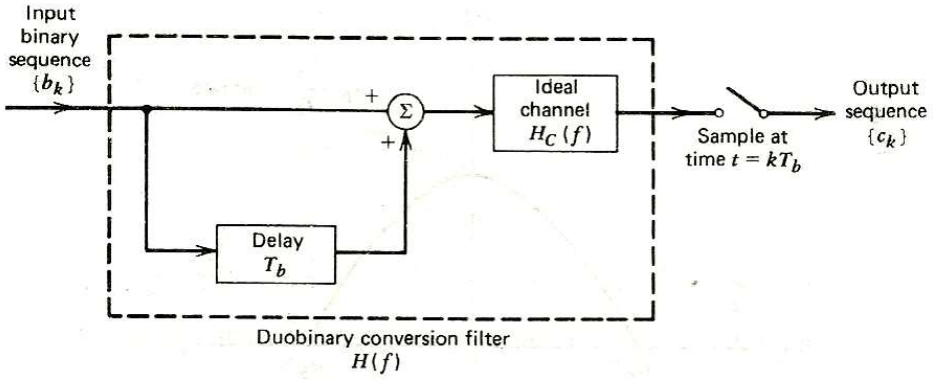


Figure 3.11 Duobinary signaling scheme

The binary sequence $\{bk\}$ is first passed through a simple filter involving a single delay element. For every unit impulse applied to the input of this filter, we get two unit impulses spaced T_b seconds at the filter output. Thus the digit c_k at the duobinary coder output is the sum of the present binary digit b_k, and its previous value b_{k-1}, as shown by

$$c_k = b_k + b_{k-1} \qquad --(1)$$

In Eq. (1) the input sequence $\{bk\}$ of uncorrelated binary digits is transformed into a sequence $\{c_k\}$ of correlated digits. This correlation between the adjacent transmitted levels is viewed as introducing intersymbol interference into the transmitted signal in an artificial manner. An ideal delay element, producing a delay of T_b seconds, has the transfer function $exp(-j2\pi f\ T_b)$, so that the transfer function of simple filter shown in fig. 3.11 is $1 + exp(-j2\pi f\ T_b)$. Hence, the overall transfer function of this filter connected in cascade with the ideal channel $H_C(f)$ is

$$H(f) = H_C(f)[1 + \exp(-j2\pi fT_b)]$$
$$= H_C(f)[\exp(j\pi fT_b) + \exp(-j\pi fT_b)]\exp(-j\pi fT_b)$$
$$= 2H_C(f)\cos(\pi fT_b)\exp(-j\pi fT_b)$$

For an ideal channel of bandwidth $B_o = R_b/2$, we have

$$H_C(f) = \begin{cases} 1 & |f| \le \frac{R_b}{2} \\ 0 & otherwise \end{cases} \qquad --(2)$$

Thus the overall frequency response has the form of a half-cycle cosine function, as shown by

$$H(f) = \begin{cases} 2\cos(\pi fT_b)\exp(-j\pi fT_b) & |f| \le \frac{R_b}{2} \\ 0 & otherwise \end{cases}$$

for which the amplitude response and phase response are as shown in Fig. *3.12a* and Fig. *3.12b,* respectively. An advantage of this frequency response is that it can be easily approximated in practice.

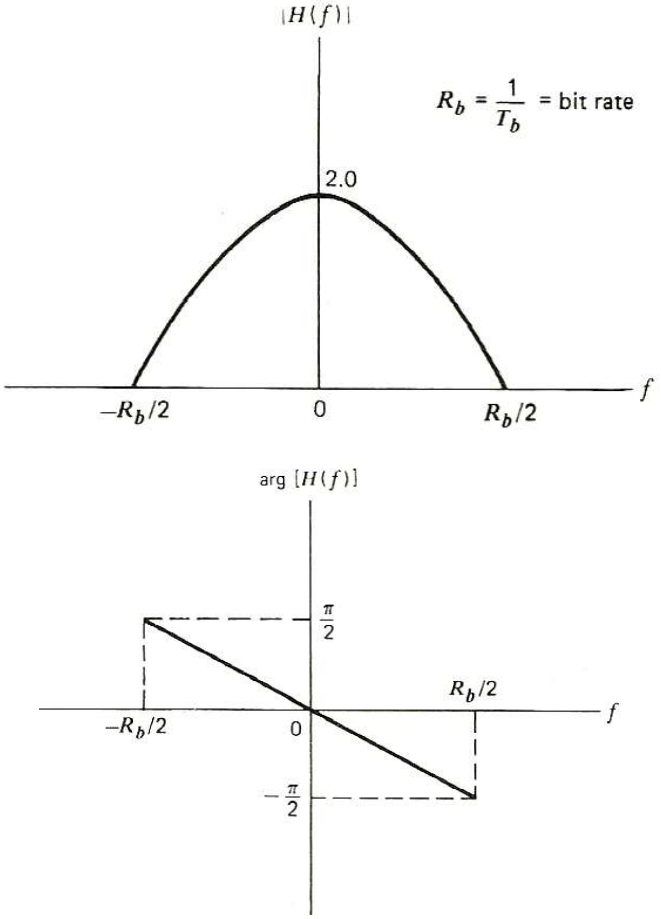


Figure 3.12 Frequency response of duobinary conversion filter. ***(a)*** **Amplitude response.** ***(b)*** **Phase response**

The corresponding value of the impulse response consists of two sine pulses, time-displaced by T_b seconds, as shown by (except for a scaling factor) which is shown plotted in Fig. 3.13

$$h(t) = \frac{\sin(\pi t/T_b)}{\pi t/T_b} + \frac{\sin(\pi(t-T_b)/T_b)}{\pi(t-T_b)/T_b}$$

$$= \frac{\sin(\pi t/T_b)}{\pi t/T_b} - \frac{\sin(\pi t/T_b)}{\pi(t-T_b)/T_b}$$

$$= \frac{T_b^2 \sin(\pi t/T_b)}{\pi t(T_b - t)}$$

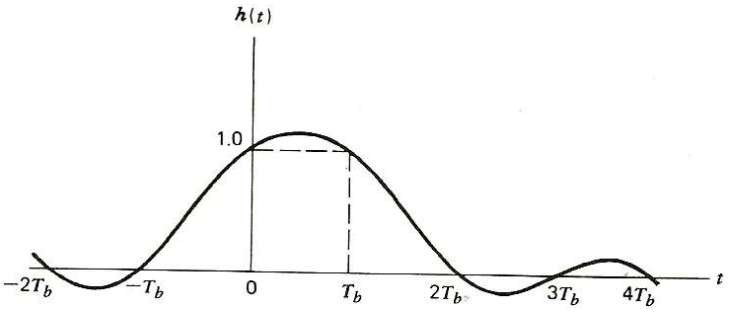


Figure 3.13 Impulse response of duobinary conversion filter.

The overall impulse response $h(t)$ has only *two* distinguishable values at the sampling instants.

The original data $\{b_k\}$ can be detected from the duobinary-coded sequence $\{c_k\}$ by subtracting the previous decoded binary digit from the currently received digit c_k in accordance with Eq. (1). Let $\hat{b}_k$ represent the *estimate* of the original binary digit b_k as received by the receiver at time t equal to kT_b, we have

$$\hat{b}_k = c_k - \hat{b}_{k-1}$$

If c_k is received without error and if also the previous estimate $\hat{b}_{k-1}$ at time $t = (k - 1)T_b$ corresponds to a correct decision, then the current estimate b_k, will also be correct. The technique of using a stored estimate of the previous symbol is called *decision feedback.*

The detection procedure is an inverse of the operation of the simple filter at the transmitter. However, a drawback of this detection process is that once errors are made, they tend to propagate. This is due to the fact that a decision on the current binary digit b_k, depends on the correctness of the decision made on the previous binary digit b_{k-1}.

A practical means of avoiding this error propagation is to use **precoding** before the duobinary coding, as shown in Fig. 3.14.

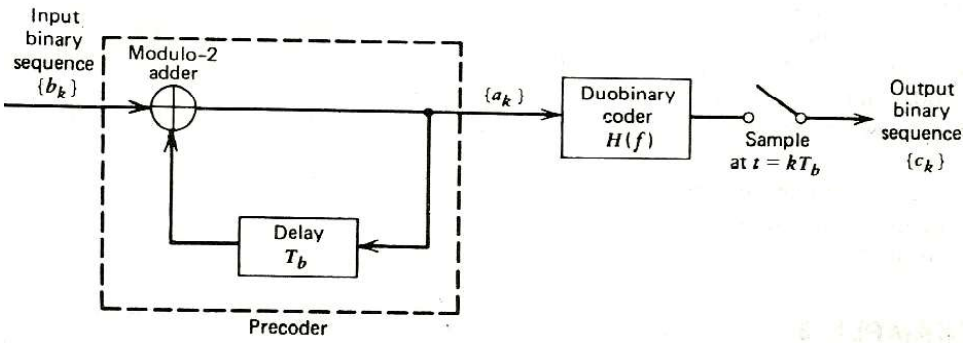


Figure 3.14 A precoded duobinary scheme. Details of the duobinary coder are given in Fig. 3.11.

The precoding operation performed on the input binary sequence $\{b_k\}$ converts it into another binary sequence $\{a_k\}$ defined by

$$a_k = b_k + a_{k-1} \quad modulo - 2 \qquad --(3)$$

Module-2 addition is equivalent to the exclusive-OR (X-OR) operation. Here, if the inputs of an X-OR gate are different, then the output is 1; otherwise, the output is a 0. The resulting precoder output $\{a_k\}$ is next applied to the duobinary coder, thereby producing the sequence $\{c_k\}$ as follows:

$$c_k = a_k + a_{k-1} \qquad --(4)$$

Assume that symbol 1 at the precoder output in Fig. 3.14 is represented by $+1\ volt$ and symbol 0 by $-1\ volt$. Therefore, from Eqs. (3) and (4), we find that

$$c_k = \begin{cases} \pm 2\ volts, & if\ b_k\ is\ represented\ by\ symbol\ 0 \\ 0\ volts, & if\ b_k\ is\ represented\ by\ symbol\ 1 \end{cases} \qquad --(5)$$

From Eq. (5), we deduce the following decision rule for detecting the original input binary sequence $\{b_k\}$ from $\{C_k\}$:

$$b_k = \begin{cases} symbol\ 0 & if\ |c_k| > 1volt \\ symbol\ 1 & if\ |c_k| < 1volt \end{cases} \qquad --(6)$$

According to Eq. (6), the detector consists of a rectifier, the output of which is compared to a threshold of 1 volt, and thus the original binary sequence $\{b_k\}$ is detected.

A block diagram of the detector is shown in Fig. 3.15.

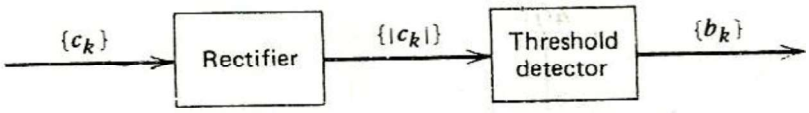


Figure 3.15 Detector for recovering original binary sequence from the precoded duobinary coder output.

In this detector only the knowledge of present input sample is required. Hence, error propagation cannot occur in the detector of Fig. 3.15

3.4.2 Modified Duobinary Technique

The **modified duobinary technique** involves a correlation span of two binary digits. This is achieved by subtracting input binary digits spaced $2T_b$ seconds, as indicated in the block diagram of Fig. 3.16.

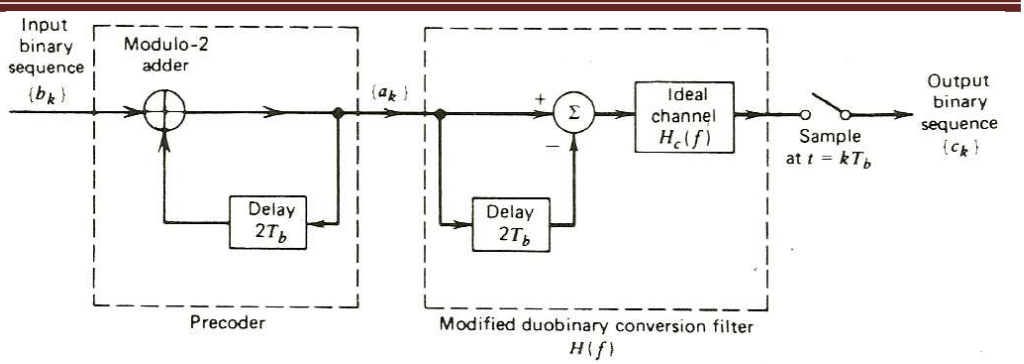


Figure 3.16 Modified duobinary signaling scheme.

The output of the modified duobinary conversion filter is related to the sequence $\{a_k\}$ at its input as follows:

$$c_k = a_k - a_{k-2}$$

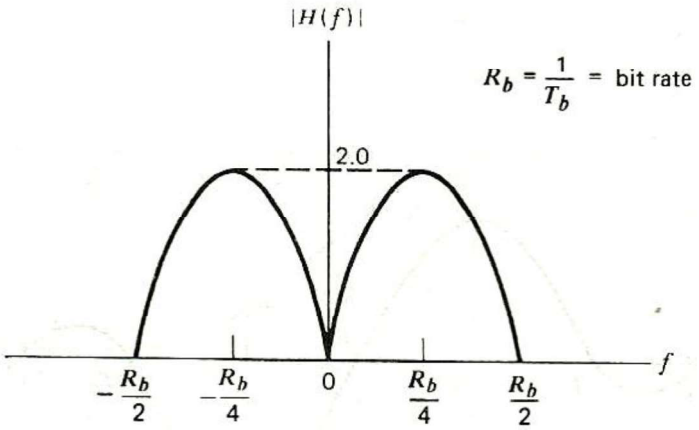
Here a three-level signal is generated. If $a_k = \pm 1\ volt$, c_k takes on one of three values: $2, 0, and\ -2\ volts$. The overall transfer function of the tapped-delay-line filter connected in cascade with the ideal channel, as in Fig. 3.16, is given by

$$H(f) = H_C(f)[1 - \exp(-j4\pi f T_b)]$$
$$= 2jH_C(f)\sin(2\pi f T_b)\exp(-j2\pi f T_b)$$

where $H_C(f)$ is as defined in Eq. (2). Then the overall frequency response will be in the form of a half-cycle sine function, as shown by

$$H(f) = \begin{cases} 2j\sin(2\pi f T_b)\exp(-j2\pi f T_b) & |f| \le R_b/2 \\ 0 & elsewhere \end{cases}$$

The corresponding amplitude response and phase response of the modified duobinary-coder are shown in Fig. *3.16a* and Fig. 3.16b, respectively.



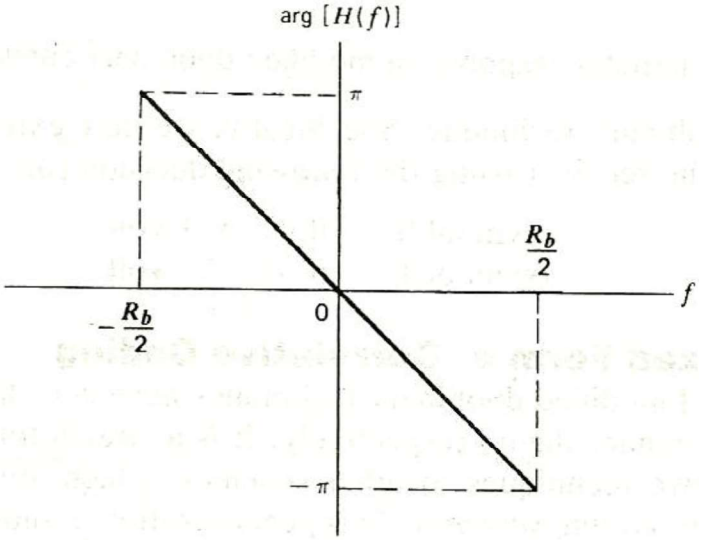


Figure 3.16 Frequency response of modified duobinary conversion filter. *(a)* Amplitude response. *(b)* Phase response.

Note that the phase response in Fig. 3.16b does not include the constant 90^o phase shift due to the multiplying factor j. In modified duobinary coder the output has no dc component. This property is important since many communication channels cannot transmit a dc component.

The impulse response of the modified duobinary coder consists of two sine pulses that are time-displaced by $2T_b$ seconds, as shown by (except for a scaling factor)

$$h(t) = \frac{\sin(\pi t/T_b)}{\pi t/T_b} + \frac{\sin(\pi(t-2T_b)/T_b)}{\pi(t-2T_b)/T_b}$$

$$= \frac{\sin(\pi t/T_b)}{\pi t/T_b} - \frac{\sin(\pi t/T_b)}{\pi(t-2T_b)/T_b}$$

$$= \frac{2T_b^2 \sin(\pi t/T_b)}{\pi t(2T_b - t)}$$

This impulse response is plotted in Fig. 3.17, which shows that it has *three* distinguishable levels at the sampling instants.

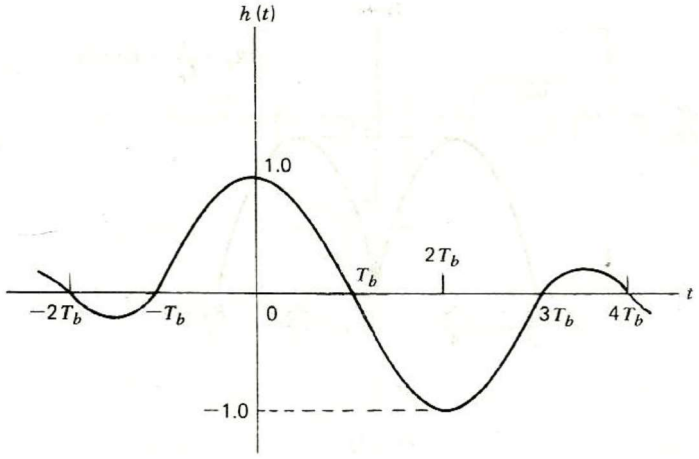


Figure 3.17 Impulse response of modified duobinary conversion filter.

In order to eliminate the error propagation in the modified duobinary system, we use a precoding procedure similar to that of the duobinary case. A rnodulo-2 logical addition is applied on signals before the generation of the modified duobinary signal, as shown by (see Fig. 3.16)

$$a_k = b_k + a_{k-2} \quad modulo - 2$$

where $\{b_k\}$ is the input binary sequence and $\{a_k\}$ is the sequence at the precoder output. Note that rnodulo-2 addition and rnodulo-2 subtraction are the same. The sequence $\{a_k\}$ thus produced is then applied to the modified duobinary conversion filter.

In the case of Fig 3.16, the output digit $\{c_k\}$ equals $0, +2, or\ -2\ volt$. Also we find that b_k can be extracted from c_k by ignoring the polarity of c_k, same as duobinary technique. The original sequence $\{b_k\}$ can be extracted at the receiver using the following decision rule:

$$b_k = \begin{cases} symbol\ 0 & if\ |c_k| > 1volt \\ symbol\ 1 & if\ |c_k| < 1volt \end{cases}$$

3.4.3 Generalized Form of Correlative Coding

The duobinary and modified duobinary techniques have correlation spans of 1 binary digit and 2 binary digits, respectively. The above 2 techniques are generalized to other schemes, which are collectively as **correlative coding schemes**. This generalization is shown in Fig. 3.18, where $H_C(f)$ is defined in Eq. (2).

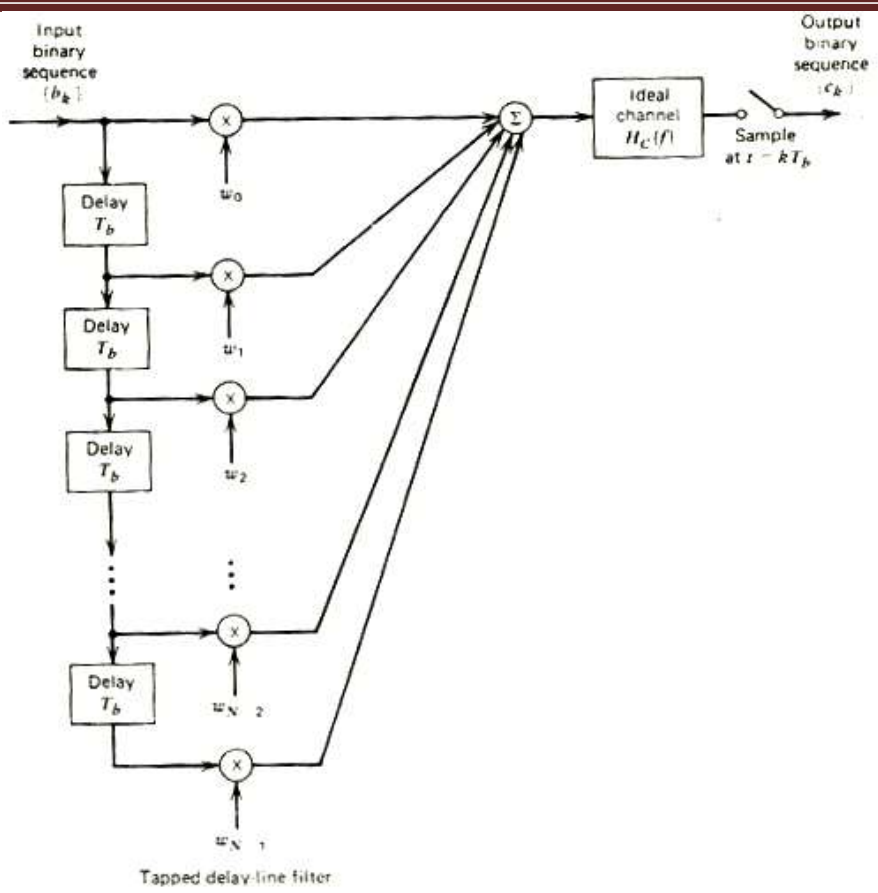


Figure 3.18 Generalized correlative coding scheme

It involves the use of a tapped-delay-line filter with tap weights $w_0, w_1, \ldots, w_{N-1}$. Specifically, a correlative sample c_k is obtained from a superposition of N successive input sample values b_k, as shown by

$$c_k = \sum_{n=0}^{N-1} w_n b_{k-n}$$

Thus, by choosing various combinations of integer values for the w_n, we obtain different forms of correlative coding schemes to suit different applications.

For example, in the duobinary case, we have

$$w_0 = +1$$
$$w_1 = +1$$

and $w_n = 0$ for $n \geq 2$.

In the modified duobinary case, we have

$$w_0 = +1$$
$$w_1 = 0$$

$$w_2 = -1$$

and $w_n = 0$ for $n \geq 3$.

3.5 M-ARY MODULATION TECHNIQUES

In an *M-ary signaling scheme,* we send one of M possible signals, $s_1(t), s_2(t), \ldots, s_M(t)$, during each signaling interval of duration T. For almost all applications, the number of possible signals $M = 2^n$, where n is an integer. The symbol duration $T = nT_b$, where T_b is the bit duration. These signals are generated by changing the amplitude, phase, or frequency of a carrier in M discrete steps. M-ary signaling schems are **M-ary ASK, M-ary PSK, and M-ary FSK** digital modulation schemes. The QPSK system is an example of M-ary PSK with $M = 4$

Another way of generating M-ary signals is to combine different methods of modulation into a hybrid form. For example, we can combine discrete changes in both the amplitude and phase of a carrier to produce *M-ary amplitude-phase keying* (APK). A special form of this hybrid modulation is called *M-ary* QAM.

M-ary signaling schemes are preferred over binary signaling schemes for transmitting digital information over band-pass channels when there is a need to maintain bandwidth at increased power. In practice, when the bandwidth of the channel is same as the bandwidth required for transmitting information, binary signaling schemes are used. But, when the bandwidth of the channel is less than the required value, we use M-ary signaling schemes to utilize the channel efficiently.

To illustrate the bandwidth-conservation capability of M-ary signaling schemes, consider the transmission of information consisting of a binary sequence with bit duration T_b. For example, if binary PSK is used we require a bandwidth inversely proportional to T_b. However, if we take blocks of n bits and use an M-ary PSK scheme with $M = 2^n$ and symbol duration $T = nT_b$, the bandwidth required is inversely proportional to $1/nT_b$. This shows that the use of M-ary PSK reduces transmission bandwidth by the factor $n = \log_2 M$ over binary PSK.

3.5.1 M-ary PSK

In M-ary PSK, the phase of the carrier takes one of M possible values, namely, $\theta_i = 2i\pi/M$, where $i = 0, 1, \ldots, M-1$. Accordingly, during each signaling interval of duration T, one of the M possible signals

$$s_i(t) = \sqrt{\frac{2E}{T}} \cos\left(2\pi f_c t + \frac{2\pi i}{M}\right) \qquad i = 0,1, \ldots \ldots, M-1$$

is sent, where E is the signal energy per symbol. The carrier frequency $f_c = n_c/T$ for some fixed integer n_c

Each $s_i(t)$ is expanded in terms of two basis functions ϕ_1 and ϕ_2 defined as

$$\phi_1(t) = \sqrt{\frac{2}{T}}\cos(2\pi f_c t) \qquad 0 \le t \le T$$

$$\phi_2(t) = \sqrt{\frac{2}{T}}\sin(2\pi f_c t) \qquad 0 \le t \le T$$

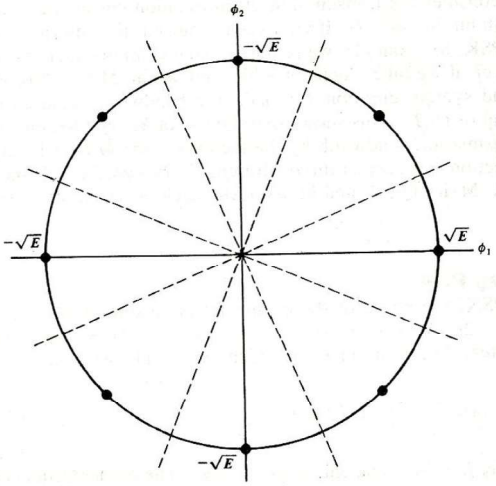


Figure 3.19 Signal constellation for octaphase-shift-keying (i.e., M = 8). The decision boundaries are shown as dashed lines.

Both ϕ_1 and ϕ_2 have unit energy. The signal constellation of M-ary PSK is therefore two-dimensional. The M message points are equally spaced on a circle of radius $\sqrt{E}$ and center at the origin, as illustrated in Fig. 3.19 for *octaphase-shift-keying* (i.e., M = 8). This figure also includes the corresponding decision boundaries indicated by dashed lines.

The optimum receiver for coherent M-ary PSK (assuming perfect synchronization with the transmitter) is shown in block diagram form in Fig. 3.20.

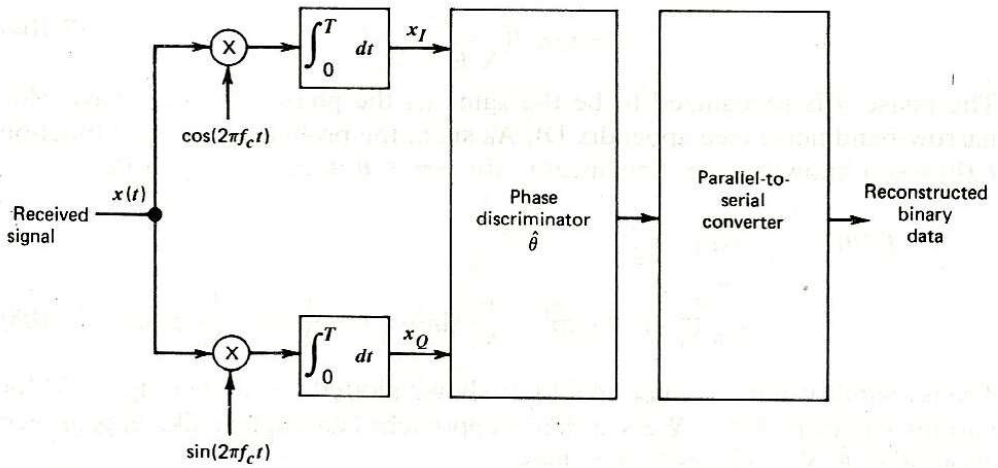


Figure 3.20 Receiver for coherent M-ary PSK.

It includes a pair of correlators with reference signals in phase quadrature. The two correlator outputs, denoted as x_I *and* x_Q, are fed into a **phase discriminator** that first computes the phase estimate

$$\hat{\theta} = \tan^{-1}\left(\frac{x_Q}{x_I}\right)$$

The phase discriminator then selects from the set $\{s_i(t), i = 0,..,M-I\}$ that particular signal whose phase is closest to the estimate $\hat{\theta}$

In the presence of noise, the decision-making process in the phase discriminator is based on the noisy inputs

$$x_I = \sqrt{E}\cos\left(\frac{2\pi i}{M}\right) + w_I \qquad i = 0,1,\ldots\ldots,M-1$$

and

$$x_Q = -\sqrt{E}\sin\left(\frac{2\pi i}{M}\right) + w_Q \qquad i = 0,1,\ldots\ldots,M-1$$

where w_I *and* w_Q are samples of two independent Gaussian random variables W_I *and* W_Q whose mean is zero and common variance equals

$$\sigma^2 = \frac{N_0}{2}$$

In Fig. 3.19, message points exhibit circular symmetry. Also, the random variables W_I *and* $W_Q$have a symmetric probability density function. Thus, in an M-ary PSK system, the average probability of symbol error,P_e is independent of the transmitted signal $s_i(t)$. Therefore, we simplify the calculation of P_e by setting $\theta = 0$, which corresponds to the message point whose coordinates along the $\phi_1(t) -$ *and* $\phi_2(t) -$*axes* are $+\sqrt{E}$ *and* 0, respectively. The decision region related to this message point [i.e., the signal $s_o(t)$] is bounded by the threshold $\hat{\theta} = -\pi M$ below the $\phi_1(t) -$ *axis* and the threshold $\hat{\theta} = \pi M$ above the $\phi_1(t) -$ axis. Therefore the probability of correct reception is

$$P_c = \int_{-\pi/M}^{\pi/M} f_\Theta(\hat{\theta})\, d\hat{\theta}$$

where $f_\theta(\hat{\theta})$ is the probability density function of the random variable Θ whose sample value equals the phase discriminator output θ produced in response to a received signal that consists of the signal $s_o(t)$ plus AWGN. That is,

$$\hat{\theta} = \tan^{-1}\left(\frac{W_Q}{\sqrt{E} + W_I}\right)$$

The phase $\hat{\theta}$ is same as the phase of a sine wave plus narrow-band noise. As such, the probability density function $f_\theta(\hat{\theta})$ has a known value. Specifically, for $-\pi \le \hat{\theta} \le -\pi$, we can write

$$f_\Theta(\hat{\theta}) = \frac{1}{2\pi}\exp\left(-\frac{E}{N_0}\right) + \sqrt{\frac{E}{\pi N_0}}\cos\hat{\theta}\exp\left(-\frac{E}{N_0}\sin^2\hat{\theta}\right)\left[1 - \frac{1}{2}erfc\left(\frac{E}{N_0}\cos\hat{\theta}\right)\right] \quad --(1)$$

The probability density function $f_\theta(\hat{\theta})$ is plotted versus θ shown in Fig. 3.21 for various values of E/N_o. It approaches an impulse-like appearance about θ as E/N_o goes high.

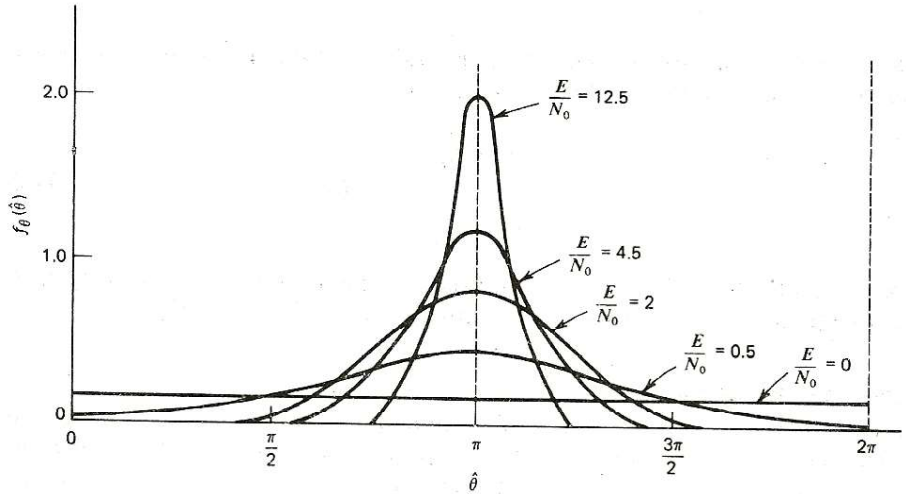


Figure 3.21 Probability density function of phase estimate

A decision error is made if the angle $\hat{\theta}$ falls outside $-\pi/M \le \hat{\theta} \le -\pi/M$. The probability of symbol error is therefore

$$P_e = 1 - P_c$$

$$= 1 - \int_{-\pi/M}^{\pi/M} f_\Theta(\hat{\theta})d\hat{\theta} \quad --(2)$$

The integral in eqn (2) does not reduce to a simple form, except for $M = 2$ and $M = 4$. Hence, for $M > 4$, it must be evaluated by using numerical integration.

However, for large M and high values of E/N_o, we can derive an approximate formula for P_e. For high values of E/N_o and for $\hat{\theta} < \pi/2$, we use the approximation

$$erfc\left(-\frac{E}{N_0}\cos\hat{\theta}\right) \approx \sqrt{\frac{N_0}{\pi E}}\frac{1}{\cos\hat{\theta}}\exp\left(-\frac{E}{N_0}\cos^2\hat{\theta}\right)$$

Hence, using this approximation for the complementary error function in Eq. (1) and simplifying terms, we get

$$f_\Theta(\hat{\theta}) \approx \sqrt{\frac{E}{\pi N_0}}\cos\hat{\theta}\exp\left(-\frac{E}{N_0}\sin^2\hat{\theta}\right) \qquad |\hat{\theta}| < \frac{\pi}{2} \qquad --(3)$$

Thus, substituting Eq. (3) in Eq. (2), we get

$$P_e \approx 1 - \sqrt{\frac{E}{\pi N_0}}\int_{-\pi/M}^{\pi/M}\cos\hat{\theta}\exp\left(-\frac{E}{N_0}\sin^2\hat{\theta}\right) \qquad --(4)$$

Changing the variable of integration from $\hat{\theta}$ to

$$z = \sqrt{\frac{E}{N_0}}\sin\hat{\theta}$$

Rewriting eqn (4)

$$P_e \approx 1 - \frac{2}{\sqrt{\pi}}\int_0^{\sqrt{E/N_0}\sin(\pi/M)}\exp(-z^2)\,dz$$

$$= erfc\left(\sqrt{\frac{E}{N_0}}\sin(\pi/M)\right) \qquad --(5)$$

This is the desired approximate formula for the probability of symbol error for coherent M-ary PSK ($M \geq 4$). The approximation becomes fixed, for fixed *M,* as E/N_o is increased.

Coherent M-ary PSK requires exact knowledge of the carrier frequency and phase for the receiver to be accurately synchronized to the transmitter. When carrier recovery at the receiver is impossible, we can use differential encoding based on the phase difference between successive symbols with some degradation in performance. If the incoming data are encoded by a phase shift rather than by absolute phase, the receiver performs detection by comparing the phase of one symbol with that of the previous symbol, and the need for a coherent reference is thereby eliminated. This procedure is the same as binary DPSK. The exact calculation of probability of symbol error for the differential detection of differential M-ary PSK (commonly referred to as *M-ary* DPSK) is too complicated for $M > 2$. However, for large values of E/N_o and $M \geq 4$, the probability of symbol error is approximately given by,

$$P_e \approx erfc\left(\sqrt{\frac{2E}{N_0}}\sin\left(\frac{\pi}{2M}\right)\right) \qquad M \geq 4 \qquad --(6)$$

Comparing Eqs. (5) and (6), for $M \geq 4$ an M-ary DPSK system attains the same probability of symbol error as the corresponding M-ary PSK system provided that the transmitted energy per symbol is increased by the following factor:

$$k(M) = \frac{\sin^2\left(\frac{\pi}{M}\right)}{2\sin^2\left(\frac{\pi}{2M}\right)} \qquad M \geq 4$$

For example, $k(4) = 1.7$. That is, differential QPSK (which is noncoherent) is approximately $2.3\ dB$ poorer in performance than coherent QPSK.

3.5.2 M-ary QAM

In an M-ary PSK system, in-phase and quadrature components of the modulated signal are interrelated in such a way that the envelope is constrained to remain constant. This constraint marked itself in a circular constellation for the message points. However, if this constraint is removed a new modulation scheme called *M-ary quadrature amplitude modulation* (QAM) is obtained. In this modulation scheme, the carrier experiences amplitude as well as phase modulation.

The signal constellation for M-ary QAM consists of a square lattice of message points, as illustrated in Fig. 3.22 for $M = 16$.

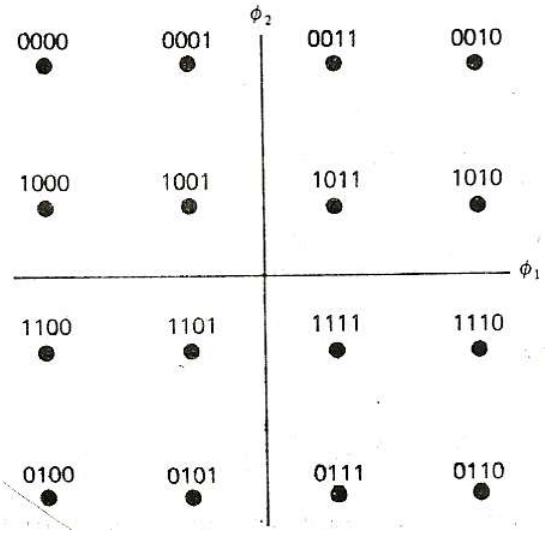


Figure 3.22 Signal-constellation of M-ary QAM for M = 16

The corresponding signal constellations for the in-phase and quadrature components of the amplitude-phase modulated wave are shown in Figs. *3.23a* and *3.23b,* respectively.

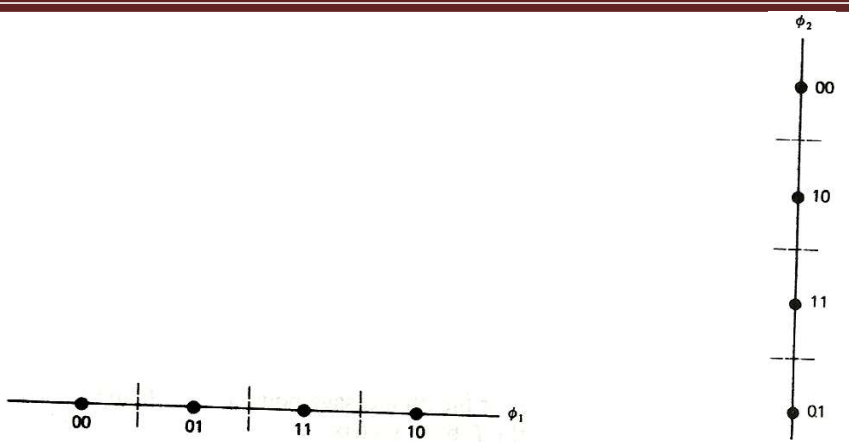


Fig. 3.23 Decomposition of signal constellation of M-ary QAM (for $M = 16$) into two signal-space diagrams for *(a)* in-phase comment $\phi_1(t)$, and *(b)* quadrature comment $\phi_1(t)$.

In general, an M-ary QAM scheme transmits $M = L^2$ independent symbols over the same channel bandwidth. The general form of M-ary QAM is defined by the transmitted signal

$$s_i(t) = \sqrt{\frac{2E_0}{T}} a_i \cos(2\pi f_c t) + \sqrt{\frac{2E_0}{T}} b_i \sin(2\pi f_c t) \qquad 0 \le t \le T$$

where E_o is the energy of the signal with the lowest amplitude, and a_i *and* b_i are a pair of independent integers chosen according to the location of the message point. The signal $s_i(t)$ consists of two phase-quadrature carriers, each of which is modulated by a set of discrete amplitudes; hence, it is called as "quadrature amplitude modulation." (QAM). The signal $s_i(t)$ can be expanded in terms of a pair of basis functions

$$\phi_1(t) = \sqrt{\frac{2}{T}} \cos(2\pi f_c t) \qquad 0 \le t \le T$$

and

$$\phi_2(t) = \sqrt{\frac{2}{T}} \sin(2\pi f_c t) \qquad 0 \le t \le T$$

The coordinates of the i^{th} message, point are $a_i\sqrt{E}$ *and* $b_i\sqrt{E_o}$, where (a_i, b_i) is an element of the *L-by-L* matrix:

$$\{a_i,b_i\} = \begin{bmatrix} (-L+1,L-1) & (-L+3,L-1) & \dots & (L-1,L-1) \\ (-L+1,L-3) & (-L+3,L-3) & \dots & (L-1,L-3) \\ \vdots & \vdots & & \vdots \\ (-L+1,-L+1) & (-L+3,-L+1) & \dots & (L-1,-L+1) \end{bmatrix}$$

where

$$L = \sqrt{M} \qquad --(1)$$

For example, for the 16-QAM whose signal constellation is shown in Fig. 3.22, where $L = 4$, we have the matrix

$$\{a_i,b_i\} = \begin{bmatrix} (-3,3) & (-1,3) & (1,3) & (3,3) \\ (-3,1) & (-1,1) & (1,1) & (3,1) \\ (-3,-1) & (-1,-1) & (1,-1) & (3,-1) \\ (-3,-3) & (-1,-3) & (1,-3) & (3,-3) \end{bmatrix}$$

To calculate the probability of symbol error for M-ary QAM, following steps are followed,

1. Since the in-phase and quadrature components of M-ary QAM are independent, the probability of correct detection for such a scheme can be written as

   $$P_c = (1 - P_e')^2$$

   where P_e' is the probability of symbol error for both component.
2. The signal constellation for the in-phase or quadrature component has geometry similar to that for discrete pulse-amplitude modulation (PAM) with a corresponding number of amplitude levels. Therefore we write

   $$P_e' = \left(1 - \frac{1}{L}\right) erfc\left(\sqrt{\frac{E_0}{N_0}}\right) \qquad --(2)$$

   where L is the square root of M.
3. The probability of symbol error for M-ary QAM is given by

   $$\begin{aligned} P_e &= 1 - P_c \\ &= 1 - (1 - P_e')^2 \\ &\approx 2P_e' \qquad --(3) \end{aligned}$$

   where it is assumed that P_e' is small compared to unity. Putting Eqn.(1) and eqn (2) in Eq, (3), we find that the probability of symbol error for M-ary, QAM is given by

   $$P_e \approx 2\left(1 - \frac{1}{\sqrt{M}}\right) erfc\left(\sqrt{\frac{E_0}{N_0}}\right) \qquad --(4)$$

The transmitted energy in M-ary QAM is variable so, its instantaneous value depends on the particular symbol transmitted. Thus, P_e can be expressed in terms of the *average* value of the

transmitted energy rather than E_o. Assuming that the L amplitude levels of the in-phase or quadrature component are equally likely, we have

$$E_{av} = 2\left[\frac{2E_0}{L}\sum_{i=1}^{L/2}(2i-1)^2\right] \qquad --(5)$$

where the multiplying factor of 2 is for the equal contributions by the in-phase and quadrature components. Summing the series in Eq. (5), we get

$$E_{av} = \frac{2(L^2-1)E_0}{3}$$

$$= \frac{2(M-1)E_0}{3}$$

Accordingly, we can rewrite Eq. (4) in terms of E_{av} as

$$P_e \approx 2\left(1-\frac{1}{\sqrt{M}}\right)erfc\left(\sqrt{\frac{2E_{av}}{2(M-1)N_0}}\right)$$

which is the desired result.

In the special case is M = 4. The signal constellation for this value of M is same as that for QPSK (shown in Fig. 3.24).

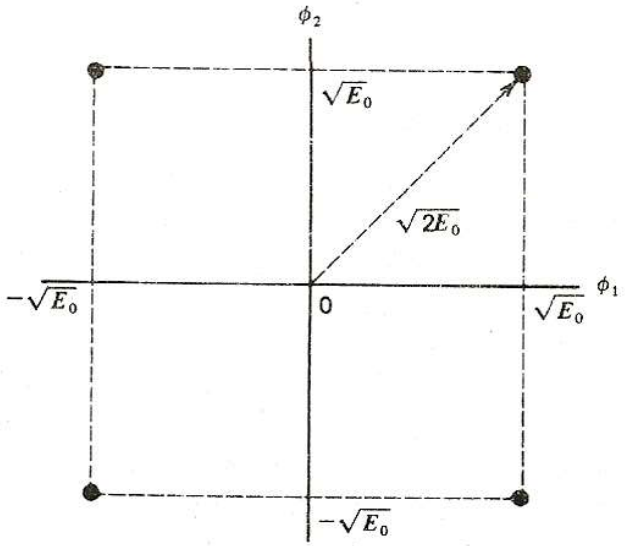
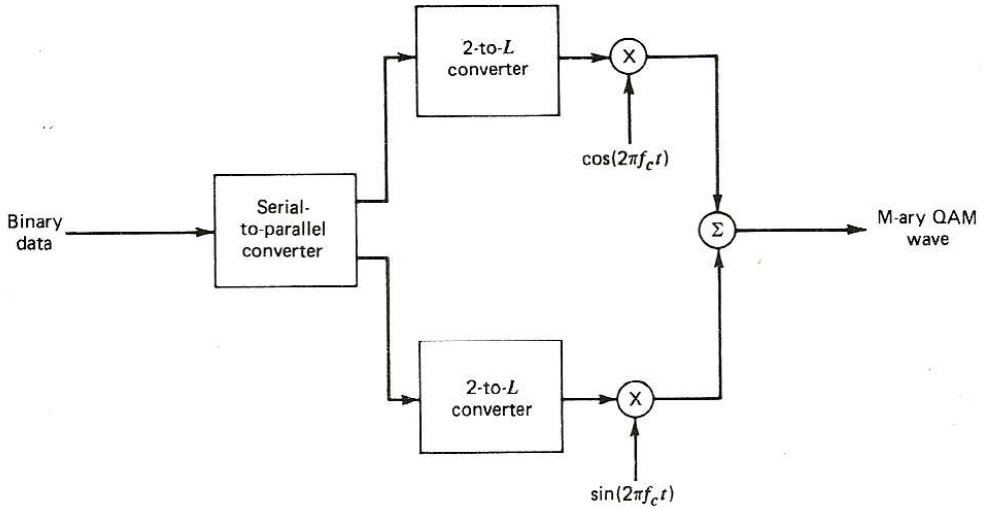
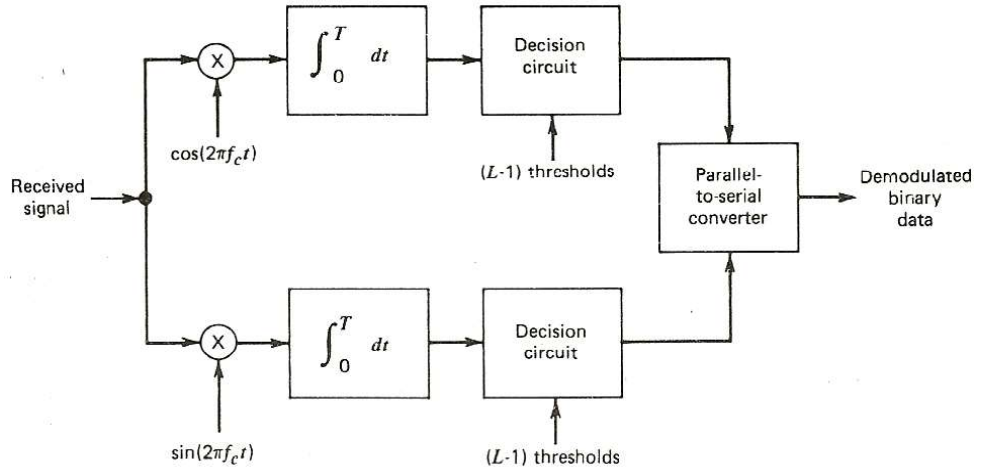


Figure 3.24 Signal constellation for the special case of M-ary QAM for M = 4

Figure *3.25a* shows the block diagram of an M-ary QAM transmitter.



(a)



(b)

Figure 3.25 Block diagrams of M-ary QAM system. *(a)* Transmitter. *(b)* Receiver.

The **serial-to-parallel converter** accepts a binary sequence at a bit rate $R_b = 1/T_b$ and produces two parallel binary sequences whose bit rates are $R_b/2$ each. The **2-to- L level converters**, where $L = \sqrt{M}$, generate polar L-level signals in response to the respective in-phase and quadrature channel inputs. Quadrature-carrier multiplexing of the two polar L-level signals so generated produces the desired M-ary QAM signal.

Figure *3.25b* shows the block diagram of the corresponding receiver. Decoding of each baseband channel is done at the output of the decision circuit. The decision circuit compares the L-level signals against L - 1 decision thresholds. The detected two binary sequences are then combined in the parallel-to-serial converter to reproduce the original binary sequence.

3.5.3 M-ary FSK

In an M-ary FSK scheme, the transmitted signals are defined by

$$s_i(t) = \sqrt{\frac{2E}{T}}\cos\left[\frac{\pi}{T}(n_c + i)t\right] \qquad 0 \le t \le T \qquad --(1)$$

where i = 1, 2, ... , *M,* and the carrier frequency $f_c = n_c/2T$ for some fixed integer *n.,* The transmitted signals are of equal duration T and have equal energy E. Since the individual signal frequencies are separated by $1/2T$ hertz, the signals in Eq. (1) are orthogonal; that is

$$\int_0^T s_i(t)s_j(t)dt = 0 \qquad i \neq j$$

For coherent M-ary FSK, the optimum receiver consists of a bank of M correlators or matched filters, with the signals in Eq. (1). At the sampling times $t = kT$, the receiver makes decisions based on the largest matched filter output.

An upper bound for the probability of symbol error is obtained by applying the union bound. The resulting bound is given by,

$$P_e \le \frac{1}{2}(M-1)erfc\left(\sqrt{\frac{E}{2N_0}}\right) \qquad --(2)$$

For fixed *M,* this bound becomes increasingly fixed as E/N_o is increased. It becomes a good approximation to P_e for values of $P_e \le 10^{-3}$. Moreover, for $M = 2$ (i.e., binary FSK), the bound of Eq. (2) becomes an equality.

Coherent detection of M-ary FSK requires the use of exact phase references, it is costly and difficult to maintain. This is overcome by using noncoherent detection, but the performance is slightly lesser. In a noncoherent receiver, the individual matched filters are followed by envelope detectors that destroy the phase information.

The probability of symbol error for the noncoherent detection of M-ary FSK is given by

$$P_e = \sum_{k=1}^{M-1}\frac{(-1)^{k+1}}{k+1}\binom{M-1}{k}\exp\left(-\frac{kE}{(k+1)N_0}\right) \qquad --(3)$$

where $\binom{M-1}{k}$ is a binomial coefficient, that is,

$$\binom{M-k}{k} = \frac{(M-1)!}{(M-1-k)!\,k!}$$

The first term of the series in Eq. (3) provides an upper bound on the probability of symbol error for the noncoherent detection of M-ary FSK:

$$P_e \leq \frac{M-1}{2}\exp\left(-\frac{E}{2N_0}\right) \qquad --(4)$$

For fixed *M,* this bound becomes increasingly close to the actual value of P_e as E/N_o is increased. For $M = 2$ (i.e., binary FSK), the bound of Eq. (4) becomes an equality

3.5.4 Comparison of M-ary Digital Modulation Techniques

In Table 3.3 typical values of power-bandwidth requirements for coherent binary and M-ary PSK schemes are summarized, assuming an average probability of symbol error equal to 10^{-4} and that the systems operate in identical noise environments. This table shows that, among M-ary PSK signals, QPSK (corresponding to $M = 4$) offers the best trade-off between power and bandwidth requirements. For $M > 8$, power requirements become high; accordingly, M-ary PSK schemes with $M > 8$ are not as widely used in practice. Also, coherent M-ary PSK schemes require more complex equipment than coherent binary PSK schemes for signal generation or detection, especially when $M > 8$. Basically, M-ary PSK and M-ary QAM have similar spectral and bandwidth characteristics. For $M > 4$, however, the two schemes have different signal constellations. For M-ary PSK the signal constellation is circular, whereas for M-ary QAM it is rectangular. Moreover, a comparison of these two constellations shows that the distance between the message points of M-ary PSK is smaller than the distance between the message points of M-ary QAM, for a fixed peak transmitted power. This basic difference between the two schemes is illustrated in Fig. 3.26 for $M = 16$. Accordingly, in an AWGN channel, M-ary QAM outperforms the corresponding M-ary PSK in error performance for $M > 4$. However, the superior performance of M-ary QAM can be realized only if the channel is free from nonlinearities.

Table 3.3 Comparison of Power-Bandwidth Requirements for M-ary PSK with Binary PSK. Probability of Symbol Error = 10^{-4}

Value of M	$\frac{(\text{Bandwidth})_{\text{M-ary}}}{(\text{Bandwidth})_{\text{Binary}}}$	$\frac{(\text{Average power})_{\text{M-ary}}}{(\text{Average power})_{\text{Binary}}}$
4	0.5	0.34 dB
8	0.333	3.91 dB
16	0.25	8.52 dB
32	0.2	13.52 dB

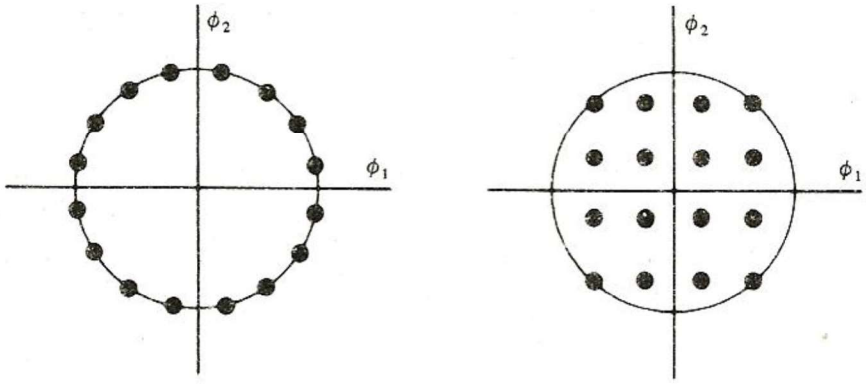


Figure 3.26 Signal constellations. (8) M-ary QPSK, and *(b)* M-ary QAM for $M = 16$

As for M-ary FSK, we find that for a fixed probability of error, increasing M results in a reduced power requirement. However, this reduction in transmitted power is achieved with increased channel bandwidth.

3.6 EYE PATTERN

The eye pattern is used to study the effect of ISI in baseband digital transmission.

In an operational environment, the overall system performance, particularly the effect of lSI and the effect the equalising filters, needs to be monitored regularly. A tool for this monitoring is called **eye pattern**. It is defined as the synchronised superposition of all possible combinations of signal, noise and lSI in the system, viewed between a particular symbol interval. The received signal is fed to the vertical input of a CRO and the horizontal sweep rate is set to T_s. The truly random data is sometimes positive and sometimes negative, allows us to see the pattern shape as an eye. The resulting oscilloscope display is called *eye pattern* because of its resemblance to the human eye. At any particular time, the minimum separation between the upper and the lower portion of the eye pattern is called the **eye opening**.

Eye pattern is very important in delivering information about lSI and noise impairments of the system. Specifically,

- In a linear system with truly random data, all the eye openings would be identical. In an eye pattern, therefore, the asymmetry between patterns indicates nonlinearities in the channel
- The width of the eye opening defines the timing error allowed at the receiver that would not cause lSI The inference is that even if an error is occured in sampling time by that amount, detection is possible. The preferred time for sampling is at the point where the vertical opening of the eye is largest. This is shown in the figure as "best sampling time".

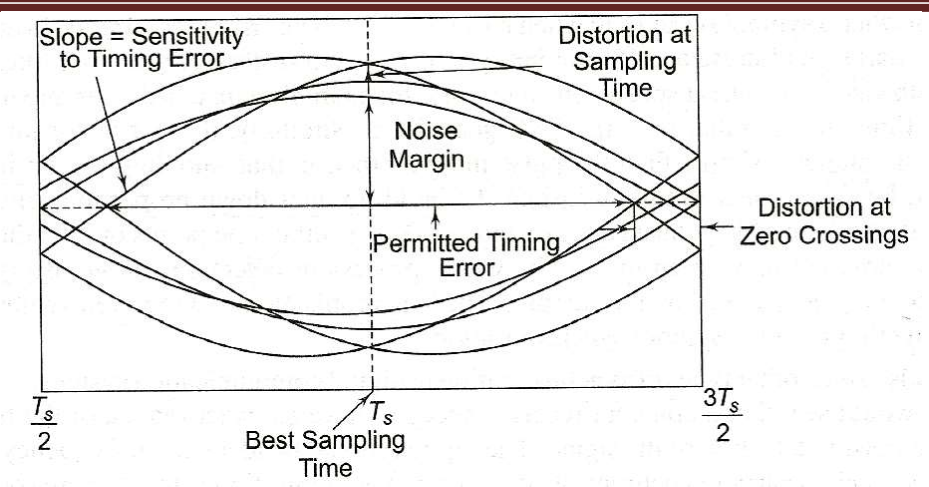


Fig.3.27 ***Eye Pattern***

- The sensitivity to the timing error is given by the slope of the open eye (evaluated at or near the zero crossing point)
- The maximum distortion is given by the vertical distance of the upper (or lower) portion of the eye from the pulse amplitude, at the sampling time.
- When the effect of ISI is severe, traces from the upper portion of the eye pattern, cross traces from the lower portion, thereby closing the eye. In case of a closed eye pattern, the detection is bound to be erroneous.

  In case of an M-ary system, the eye pattern contains *(M* - 1) eye openings stacked up vertically one on the other.

3.7 EQUALIZATION

The baseband processor block in the transmitter shapes the signal for eliminating lSI. However, due to the unpredictable nature of the channel, an equalizing filter is included at the receiver. The equalizing filter in the receiver cancels any residual lSI present in the received signal and gives a lSI free signal to the detector block.

There are two broad categories of equalization.

1. The first category, **Maximum-Likelihood Sequence Estimation (MLSE)**, estimates the sequence from the received distorted pulse sequence based on some predefined metric. An MLSE does not reshape or compensate the distorted samples, instead it adjusts itself in such a way that it, it can better deal with the distorted pulse sequence.
2. The second category, **Equalization with Filters**, uses filters to compensate for the distorted pulse. The detector is presented with a sequence of received samples that the equalizer has cleaned up from the effects of ISI. Such filters can be further subdivided into **Transversal Filters or Decision Feedback Equalizer** depending on whether they contain only feedforward elements or both feedforward and feedback elements

respectively. They can be also grouped according to the automatic nature of their operation, which may be either **preset or adaptive**. They can also be grouped according to the filter's resolution or update rate or whether the predetection samples are provided only on symbol boundaries, i.e., one sample per symbol (symbol spaced) or multiple samples (fractionally spaced) etc.

MLSE equalizer has superior equalization capability. But the computational complexity grows exponentially with the length of the channel dispersion. Example of a MLSE equalizer is Viterbi Equalization. If the size of the symbol is *M* and the number of interfering symbols contributing to lSI is *N,* then the Viterbi algorithm computes M^{N+1} metrics for each new received symbol. Due to its complexity, it is not widely used. However, the computational complexity of linear transversal filter is a linear function of the channel dispersion length.

3.7.1 Linear Transversal Filter

Since the transmitter sends discrete-time samples at a rate of $1/T_s$ symbols/sec and the sampler at the receiver also produces samples at the same rate but with a transmission delay, it is assumed that the signal at the transversal filter output will also have the time separation of T_s. As a result, we have a $(2N + 1)$ - tap transversal filter impulse response $h_e(t)$ that extents a time interval of $2NT_s$ seconds for a channel with a maximum *N* number of interfering symbols contributing to ISI.

The filter, shown in Fig. 3.28, consists of delay lines with a single period of delay. The current and past values of the received signal are linearly weighted with equalizer coefficients or tap weights $\{c_n\}$ and are then summed to produce the output. The principal contribution is from the central tap, with the other taps contributing echoes of the main signal at symbol intervals on either side of the main signal. If the filter is implemented with infinite number of taps, then the tap weights can be chosen to force the system impulse response to go to zero at all but one of the sampling times, thus making $H_e(f)$ correspond exactly to the inverse of the channel transfer function. Obviously an infinite length filter is not realizable, practical finite-length filters can be designed that approximate this ideal one.

In Fig. 3.28, the outputs of the weighted taps are summed and fed to a decision device. The tap weights $\{c_n\}$ is chosen to subtract the effects of interference from symbols adjacent in time to the desired symbol. Consider that there are $(2N + 1)$ taps with weights $c_{-N}, c_{-N+1}, \dots, c_0, c_1, \dots, c_N$.

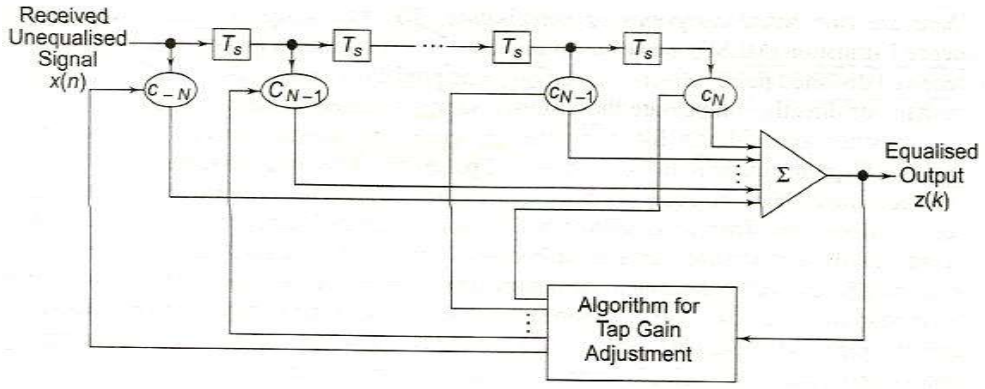


Fig. 3.28 ***Transversal Filter***

Output samples $\{z(k)\}$ of the equalizer are then found by convolving the input samples $\{x(k)\}$ and tap weights $\{cn\}$ as follows:

$$z(k) = \sum_{n=-N}^{N} x(k-n)c_n\,, \qquad k = -2N, \ldots\ldots, 2N$$

$$n = -N, \ldots\ldots, N \qquad --(1)$$

where $k = 0, \pm 1, \pm 2, \ldots$ is a time index. The index n is used in two ways - as a time offset and as a filter coefficient identifier. Now we define three matrices **z, c** and **x** as follows,

$$\mathbf{z} = \begin{bmatrix} Z(-2N) \\ \cdot \\ \cdot \\ Z(0) \\ \cdot \\ \cdot \\ Z(2N) \end{bmatrix} \quad \mathbf{c} = \begin{bmatrix} c_{-N} \\ \cdot \\ \cdot \\ c_0 \\ \cdot \\ \cdot \\ c_N \end{bmatrix}$$

$$\mathbf{x} = \begin{bmatrix} x(-N) & 0 & 0 & \ldots & 0 & 0 \\ x(-N+1) & x(-N) & 0 & \ldots & \ldots & \ldots \\ \cdot & & & \cdot & & \cdot \\ \cdot & & & \cdot & & \cdot \\ \cdot & & & \cdot & & \cdot \\ x(N) & x(N-1) & x(N-2) & \ldots & x(-N+1) & x(-N) \\ \cdot & & & \cdot & & \cdot \\ \cdot & & & \cdot & & \cdot \\ \cdot & & & \cdot & & \cdot \\ 0 & 0 & 0 & \ldots & x(N) & x(N-1) \\ 0 & 0 & 0 & \ldots & 0 & x(N) \end{bmatrix}$$

and in terms of these newly defined matrices, we can relate the output and the input of the transversal filter by,

$$[\boldsymbol{z}] = [\mathbf{x}][\boldsymbol{c}] \qquad --(2)$$

Whenever matrix [**x**] is square with its dimensions equal to the dimensions of **[c]**, we can find **[c]** by solving the equation

$$[\boldsymbol{c}] = [\mathbf{x^{-1}}][\boldsymbol{z}] \qquad --(3)$$

The size of the matrix [z] and the number of rows in the matrix [x] can be chosen to be any value. In Eq. (1) and in defining the matrix of **[x],** index '*k*' was arbitrarily chosen to allow for *4N* + 1 sample points. Vectors [z] and [c] have dimensions *4N* + 1 and *2N* + 1 respectively and matrix [xl is non-square with dimensions *4N* + 1 by *2N* + 1. Equations like Eq. (3) are referred to as overdetermined set (i.e. there are more equations than unknowns). We can solve such an equation in a deterministic way known as **zero-forcing solution**, or, in a statistical way, known as the **Minimum Mean-Square Error (MMSE) solution**.

3.7.1.1 Zero-forcing Solution

By disposing of the top *N* and bottom *N* rows of the matrix [x], we can solve for [c] from Eq. (3). [x] now becomes a square matrix with dimension *2N* + 1 and [z] a vector of dimension *2N* + 1. This particular solution minimizes the peak lSI distortion by selecting the weights $\{c_n\}$ so that the equalizer output goes to zero at *N* sample points on either side of the desired pulse. Hence, it is called a **zero-forcing equaliser**. Here the weights are chosen as,

$$z(k) = \begin{cases} 1, & for\ k = 0 \\ 0, & for\ k = \pm 1, \pm 2, \ldots, \pm N \end{cases}$$

Equation (3) is then used to solve for **[c]**. The required length of the filter (i.e. number of tap weights) is a function of how much distortion the channel may introduce to the signal. For such an equalizer with finite length, the peak distortion is minimized only if the eye pattern is initially open. For high-speed transmission and channels introducing significant lSI, the eye is often closed before equalization. For those applications, the zero-forcing equalizer cannot be used as this type of equalizer requires the main pulse to go to maximum.

3.7.1.2 Minimum MSE Solution

A more robust equalizer is obtained if the $\{c_n\}$ tap weights are so chosen to minimize the mean-square error (MSE) of the sum of all the lSI terms and the noise terms at the output of the equalizer. MSE is defined as the expected value of the squared difference between the desired data symbol and the estimated data symbol. The set of overdetermined equations is used to obtain a minimum MSE solution by multiplying X^T to both sides of Eq. (2).

$$[\mathbf{x^T}][\mathbf{z}] = [\mathbf{x^T}][\mathbf{x}][\mathbf{c}] \quad \text{or} \quad [\mathbf{R}_{xz}] = [\mathbf{R}_{xx}][\mathbf{c}]$$

where $\boldsymbol{Rxz} = \mathbf{x^T}\,\boldsymbol{z}$ is called the **cross-correlation vector** and $\boldsymbol{Rxx} = \mathbf{x^T x}$, the **autocorrelation matrix** of the input noisy signal. In practice, $\boldsymbol{R}_{xz}$ and $\boldsymbol{R}_{xx}$ are not known, but can be estimated by sending a test signal over the channel and solving for the tap weights in a time average way.

$$\mathbf{c} = \mathbf{R}_{xx}^{-1}\mathbf{R}_{xz}$$

In case of deterministic zero-forcing solution, the **x** matrix must be square. But to achieve the minimum MSE solution in a statistical sense, a non-square x matrix gets transformed to a square autocorrelation matrix $\boldsymbol{R}_{xx}$ yielding a set of $2N + 1$ simultaneous equations, whose solution leads to tap weights that minimize the MSE. Size of the vector **c** and the number of the columns of the matrix **x** correspond to the number of taps in the equalizing filter. Most high-speed telephone line modems use an MSE weight criterion because it is superior to a zero-forcing criterion; it is more robust in the presence of noise and large lSI.

3.7.2 Decision Feedback Equalizer

The linear transversal equalizer is good for telephone channels, because telephone channels do not exhibit nulls in the frequency response. However, mobile radio channels exhibit such spectral nulls. In such cases a nonlinear equalizer is necessary. A Decision Feedback Equalizer (DFE) is a nonlinear equalizer that uses previous detector decisions to eliminate the lSI on currently being detected pulses. The lSI in the present pulse is generally caused by previous pulses. So, a DFE subtracts the distortion on a current pulse that was caused by previous pulses.

Figure 3.29 shows a DFE where the forward filter and the feedback filter can each be a linear filter, such as a transversal filter. As shown in figure, filter tap weights are updated adaptively. The nonlinearity of DFE starts form the nonlinear detection process that provides input to the feedback filter. The basic idea of a DFE is that if the values of the symbols previously detected are assumed to be correct, then the lSI contributed by them in the present symbol can be canceled out exactly at the output of the forward filter subtracting past symbol values with appropriate weightage.

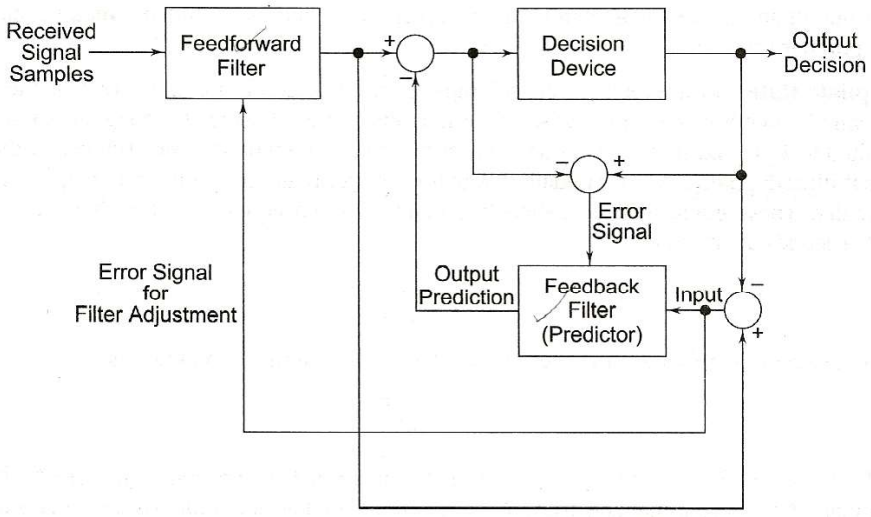


Fig. 3.29 ***Decision Feedback Equalizer***

The forward and feedback tap weights can be adjusted parallely to fulfill some error criterion like minimum MSE or peak error criterion. When only a forward filter is used, the output of the filter contains channel noise, contributed from every sample in the filter. The advantage of a DFE implementation is that the feedback filter operates on noiseless quantized levels, and thus its output is free of channel noise.

3.7.3 Preset and Adaptive Equalizer

On channels whose frequency responses are known and are time invariant, the channel characteristics can be measured and the filter's tap weights can be adjusted accordingly. If the weights remain fixed during transmission of data, the equalization is called **preset**. One method of preset equalization is setting the tap weights according to some average knowledge of the channel. This is used for data transmission over voice grade telephone lines transmitting at less than 2.4 kbps. Another preset method consists of transmitting a training sequence that is compared at the receiver with locally generated sequence. The deviation of the received sequence from the known training sequence provides an estimate of the existing channel condition. This knowledge is used to set the tap weights $\{c_n\}$. In this method, it is done once either at the start of transmission or when transmission is broken and needs to be re-established. The disadvantages of this method are that (a) training is to be invoked at the start of transmission and (b) this method is not suitable for time varying channels.

Another equalization scheme capable of tracking a slowly varying channel characteristic is adaptive equalization. Periodically a preamble or short training sequence is transmitted. The receiver knows the preamble and apart from equalization, it can also be used to detect start of transmission, to set the AGC level, to align internal clocks, local oscillator and so on. Continual adjustments are accomplished by replacing the known preamble with a sequence of the symbols estimated from the equalizer output and treated as known by the receiver. This continuous and automatic equalization is a very popular scheme.

3.7.3.1 Filter Update Rate

A transversal filter with taps spaced T_s seconds apart, where T_s, is the symbol time, is called a symbol-spaced equalizer. Aliasing may occur if the signal is not strictly band limited to $1/T_s$ Hz. It gives rise to spectral null and thereby the performance of the linear transversal filter degrades. A filter update rate that is greater than the symbol rate helps to reduce this difficulty. These equalizers are called fractionally-spaced equalizer. The filter taps now are spaced T' seconds apart, where,

$$T' \leq \frac{T_s}{1+r}$$

and ***'r'*** denotes the roll-off factor. In other words, the received signal bandwidth is,

$$W' \geq \frac{1+r}{T_s}$$

The goal is to choose T' so that the equalizer transfer function $H_e(f)$ becomes broad to accommodate the whole signal spectrum. Note that the signal at the output of the equalizer is

still sampled at a rate $1/T_s$, but since the taps are spaced T' seconds apart (the equalizer input signal is sampled at a rate $1/T'$), the equalization action operates on the received signal before its frequency components are aliased. Equalizer simulations over voice-grade telephone lines with $T' = T_s/2$, confirm, that such fractionally-spaced equalizer's performance is superior to the performance of the symbol-spaced equalizer.

2 MARKS

1. What is an intersymbol interference (lSI) in baseband binary PAM systems?

In baseband binary PAM, symbols are transmitted one after another. These symbols are separated by sufficient time durations. The transmitter, channel and receiver acts as a filter to this baseband data. Because of the filtering characteristics, transmitted PAM pulses are spread in time. Let the transmitted waveform be represented as,

$$x(t) = \sum_{k=-\infty}^{\infty} A_k \, g(t - kT_b)$$

Here A_k is the amplitude of k^{th} pulse.

And $g(t)$ is shaping pulse.

The output pulse at $t = iT_b$ can be expressed as,

$$y(t_i) = \mu A_i + \mu \sum_{\substack{k=-\infty \\ k \neq i}}^{\infty} A_k \, p[(i-k)T_b]$$

Here T_b is the bit duration and $'t_i'$ indicates instant of i^{th} pulse. μA_i is the contribution of i^{th} transmitted bit. The second term in above equation occurs due to filtering nature of the transmitter receiver and channel. The second term represents the residual effect (time spread) of all other bits transmitted before and after t_i. This presence of outputs (second term) due to other bits (symbols) interfere with the output of required bit (symbol).This effect is called Intersymbol Interference (ISI).

2. What are eye patterns?

Eye pattern is used to study the effect of ISI in baseband transmission.

i) Width of eye opening defines the interval over which the received wave can be sampled without error from ISI.

ii) The sensitivity of the system to timing error is determined by the rate of closure of the eye as the sampling time is varied.