CHAPTER 6

Digital Transmission of Analog Signals: PCM, DPCM and DM

6.1 Introduction

Quite a few of the information bearing signals, such as speech, music, video, etc., are *analog* in nature; that is, they are functions of the continuous variable t and for any $t = t_1$, their value can lie anywhere in the interval, say $-A$ to A. Also, these signals are of the baseband variety. If there is a channel that can support baseband transmission, we can easily set up a baseband communication system. In such a system, the transmitter could be as simple as just a power amplifier so that the signal that is transmitted could be received at the destination with some minimum power level, even after being subject to attenuation during propagation on the channel. In such a situation, even the receiver could have a very simple structure; an appropriate filter (to eliminate the out of band spectral components) followed by an amplifier.

If a baseband channel is not available but have access to a passband channel, (such as ionospheric channel, satellite channel etc.) an appropriate CW modulation scheme discussed earlier could be used to shift the baseband spectrum to the passband of the given channel.

Interesting enough, it is possible to transmit the analog information in a digital format. Though there are many ways of doing it, in this chapter, we shall explore three such techniques, which have found widespread acceptance. These are: Pulse Code Modulation (PCM), Differential Pulse Code Modulation (DPCM) and Delta Modulation (DM). Before we get into the details of these techniques, let us summarize the benefits of digital transmission. For simplicity, we shall assume that information is being transmitted by a sequence of binary pulses.

i) During the course of propagation on the channel, a transmitted pulse becomes gradually distorted due to the non-ideal transmission characteristic of the channel. Also, various unwanted signals (usually termed interference and noise) will cause further deterioration of the information bearing pulse. However, as there are only two types of signals that are being transmitted, it is possible for us to identify (with a very high probability) a given transmitted pulse at some appropriate intermediate point on the channel and *regenerate* a clean pulse. In this way, we will be completely eliminating the effect of distortion and noise till the point of regeneration. (In long-haul PCM telephony, regeneration is done every few kilometers, with the help of *regenerative repeaters.*) Clearly, such an operation is not possible if the transmitted signal was analog because there is nothing like a reference waveform that can be regenerated.

ii) *Storing* the messages in digital form and forwarding or redirecting them at a later point in time is quite simple.

iii) *Coding* the message sequence to take care of the channel noise, encrypting for secure communication can easily be accomplished in the digital domain.

iv) *Mixing* the signals is easy. All signals look alike after conversion to digital form independent of the source (or language!). Hence they can easily be multiplexed (and demultiplexed)

6.2 The PCM system

Two basic operations in the conversion of analog signal into the digital is *time discretization* and *amplitude discretization*. In the context of PCM, the former is accomplished with the *sampling operation* and the latter by means of *quantization.* In addition, PCM involves another step, namely, conversion of quantized amplitudes into a sequence of simpler pulse patterns (usually binary), generally called as *code words.* (The word *code* in pulse code modulation refers to the fact that every quantized sample is converted to an R-bit code word.)

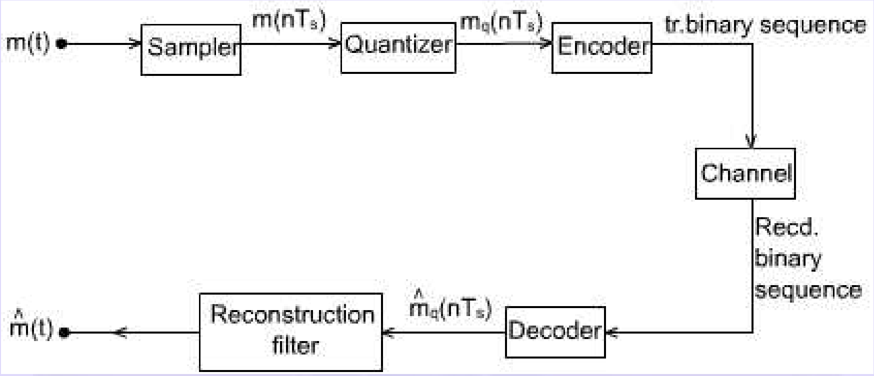


Fig. 6.1: A PCM system

Fig. 6.1 illustrates a PCM system. Here, $m(t)$ is the information bearing message signal that is to be transmitted digitally. $m(t)$ is first sampled and then quantized. The output of the sampler is $m(nT_s) = m(t)\big|_{t = nT_s}$. T_s is the sampling period and n is the appropriate integer. $f_s = \frac{1}{T_s}$ is called the sampling rate or sampling frequency. The quantizer converts each sample to one of the values that is closest to it from among a pre-selected set of discrete amplitudes. The encoder represents each one of these quantized samples by an R-bit code word. This bit stream travels on the channel and reaches the receiving end. With f_s as the sampling rate and R-bits per code word, the bit rate of the PCM system is $R f_s = \frac{R}{T_s}$ bits/sec. The decoder converts the R-bit code words into the corresponding (discrete) amplitudes. Finally, the reconstruction filter, acting on these discrete amplitudes, produces the analog signal, denoted by $\widehat{m}(t)$. If there are no channel errors, then $\widehat{m}(t) \simeq m(t)$.

6.3 Sampling

We shall now develop the sampling theorem for lowpass signals. Theoretical basis of sampling is the Nyquist *sampling theorem* which is stated below.

Let a signal $x(t)$ be band limited to W Hz; that is, $X(f) = 0$ for $|f| > W$. Let $x(nT_s) = x(t)\big|_{t = nT_s}$, $-\infty < n < \infty$ represent the samples of $x(t)$ at uniform intervals of T_s seconds. If $T_s \leq \dfrac{1}{2W}$, then it is possible to reconstruct $x(t)$ exactly from the set of samples, $\{x(nT_s)\}$.

In other words, the sequence of samples $\{x(nT_s)\}$ can provide the complete time behavior of $x(t)$. Let $f_s = \dfrac{1}{T_s}$. Then $f_s = 2W$ is the minimum sampling rate for $x(t)$. This minimum sampling rate is called the *Nyquist rate*.

Note: If $x(t)$ is a sinusoidal signal with frequency f_0, then $f_s > 2f_0$. $f_s = 2f_0$ is not adequate because if the two samples per cycle are at the zero crossings of the tone, then all the samples will be zero!

We shall consider three cases of sampling, namely, i) ideal impulse sampling, ii) sampling with rectangular pulses and iii) flat-topped sampling.

6.3.1 Ideal impulse sampling

Consider an arbitrary lowpass signal $x(t)$ shown in Fig. 6.2(a). Let

$$x_s(t) = x(t)\left[\sum_{n = -\infty}^{\infty} \delta(t - nT_s)\right] \tag{6.1a}$$

$$= \sum_{n = -\infty}^{\infty} x(t)\, \delta(t - nT_s) \tag{6.1b}$$

$$= \sum_{n = -\infty}^{\infty} x(nT_s)\, \delta(t - nT_s) \tag{6.1c}$$

where $\delta(t)$ is the unit impulse function of section 1.5.1. $x_s(t)$**,** shown in red in Fig. 6.2(b) consists of a sequence of impulses; the weight of the impulse at $t = nT_s$ is equal to $x(nT_s)$. $x_s(t)$ is zero between two adjacent impulses.

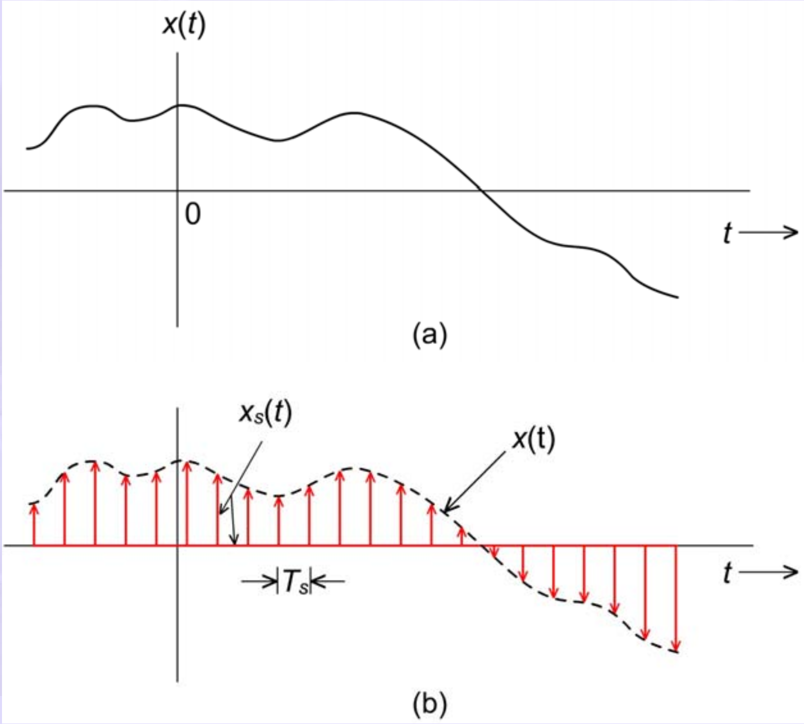


Fig. 6.2: (a) A lowpass signal $x(t)$

(b) $x_s(t)$, sampled version of $x(t)$

It is very easy to show in the frequency domain that $x_s(t)$ preserves the complete information of $x(t)$. As defined in Eq. 6.1(a), $x_s(t)$ is the product of $x(t)$ and $\sum_n \delta(t - nT_s)$. Hence, the corresponding Fourier relation is convolution. That is,

$$X_s(f) = X(f) * \frac{1}{T_s}\left[\sum_{n=-\infty}^{\infty} \delta\left(f - \frac{n}{T_s}\right)\right] \tag{6.2a}$$

$$= \frac{1}{T_s} \sum_{n=-\infty}^{\infty} X\left(f - \frac{n}{T_s}\right) \tag{6.2b}$$

$$= \frac{1}{T_s} \sum_{n=-\infty}^{\infty} X(f - n f_s) \tag{6.2c}$$

From Eq. 6.2(c), we see that $X_s(f)$ is a superposition of $X(f)$ and its shifted versions (shifted by multiples of f_s, the sampling frequency) scaled by $\frac{1}{T_s}$. This is shown in Fig. 6.3. Let $X(f)$ be a triangular spectrum as shown in Fig. 6.3(a).

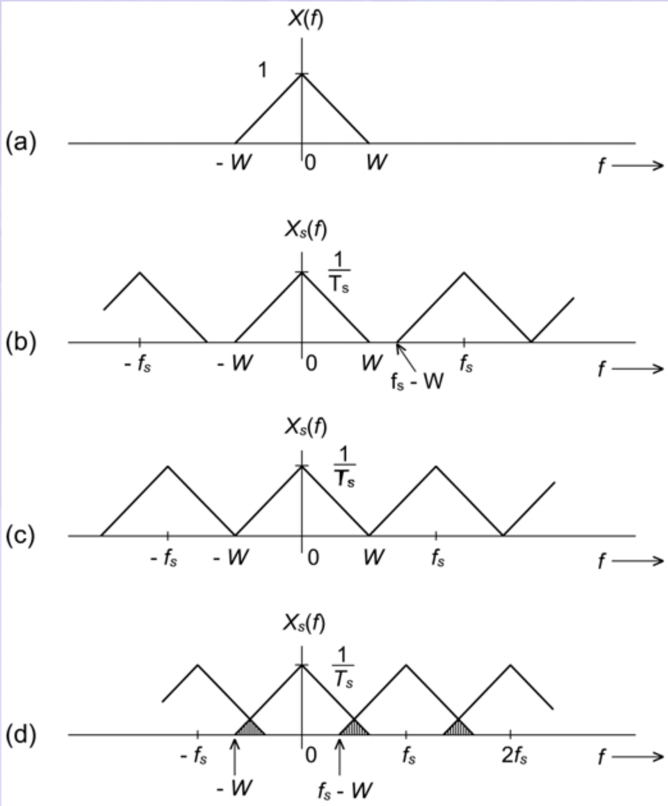


Fig. 6.3: Spectra of $x(t)$ and $x_s(t)$

(a) $X(f)$ (b) $X_s(f)$, $f_s > 2W$

(c) $X_s(f)$, $f_s = 2W$ (d) $X_s(f)$, $f_s < 2W$

From Fig. 6.3(b) and 6.3(c), it is obvious that we can recover $x(t)$ from $x_s(t)$ by passing $x_s(t)$ through an ideal lowpass filter with gain T_s and bandwidth W, as shown in Fig. 6.4.

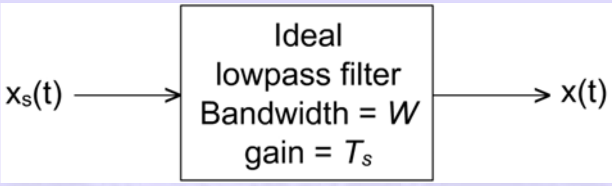


Fig. 6.4: Reconstruction of $x(t)$ from $x_s(t)$

Of course, with respect to Fig. 6.3(b), which represents the over-sampled case, reconstruction filter can have some transition band which can fit into the gap between $f = W$ and $f = (f_s - W)$. However, when $f_s < 2W$, (under-sampled case) we see that spectral lobes overlap resulting in signal distortion, called *aliasing distortion*. In this case, exact signal recovery is not possible and one must be willing to tolerate the distortion in the reconstructed signal. (To avoid aliasing, the signal $x(t)$ is first filtered by an anti-aliasing filter band-limited to $|W| \leq \frac{f_s}{2}$ and the filtered signal is sampled at the rate of f_s samples per second. In this way, even if a part of the signal spectrum is lost, the remaining spectral components can be recovered without error. This would be a better option than permitting aliasing. See Example 6.1.)

It is easy to derive an interpolation formula for $x(t)$ in terms of its samples $x(nT_s)$ when the reconstruction filter is an ideal filter and $f_s \geq 2W$. Let $H(f)$ represent an ideal lowpass filter with gain T_s and bandwidth $W' = \frac{f_s}{2}$ where

$W \leq \frac{f_s}{2} \leq f_s - W$. Then, $h(t)$ the impulse response of the ideal lowpass filter is, $h(t) = 2T_s W^{'} \sin c\left(2W^{'} t\right)$. As $x(t) = x_s(t) * h(t)$ and $2W^{'} T_s = 1$, we have

$$x(t) = \sum_{n = -\infty}^{\infty} x(nT_s) \sin c\left[2W^{'}(t - nT_s)\right] \qquad (6.3a)$$

If the sampling is done at the Nyquist rate, then $W^{'} = W$ and Eq. 6.3(a) reduces to

$$x(t) = \sum_{n = -\infty}^{\infty} x\left(\frac{n}{2W}\right) \sin c(2Wt - n) \qquad (6.3b)$$

That is, the impulse response of the ideal lowpass filter, which is a $\sin c(\ )$ function, acts as the interpolating function and given the input, $\left\{x(nT_s)\,\delta(t - nT_s)\right\}$, it interpolates the samples and produces $x(t)$ for all t.

Note that $x_s(t)$ represents a sequence of impulses. The weight of the impulse at $t = nT_s$ is equal to $x(nT_s)$. In order that the sampler output be equal to $x(nT_s)$, we require conceptually, the impulse modulator to be followed by a unit that converts impulses into a sequence of sample values which are basically a sequence of numbers. In [1], such a scheme has been termed as an “ideal C-to-D converter”. For simplicity, we assume that the output of the sampler represents the sample sequence $\left\{x(nT_s)\right\}$.

To reconstruct $x(t)$ from $\left\{x(nT_s)\right\}$, we have to perform the inverse operation, namely, convert the sample sequence to an impulse train. This has been termed as an “ideal D-to-C converter in [1]. We will assume that the reconstruction filter in Fig. 6.1 will take care of this aspect, if necessary.

6.3.2 Sampling with a rectangular pulse train

As it is not possible in practice to generate impulses, let look at a more practical method of sampling, namely, sampling with a rectangular pulse train. (Note that an impulse is a limiting case of a rectangle pulse as explained in Sec 1.5.1.)

Let $y_p(t)$ represent the periodic rectangular pulse train as shown in Fig. 6.5(b). Let

$$x_s(t) = \left[x(t)\right] y_p(t) \tag{6.4a}$$

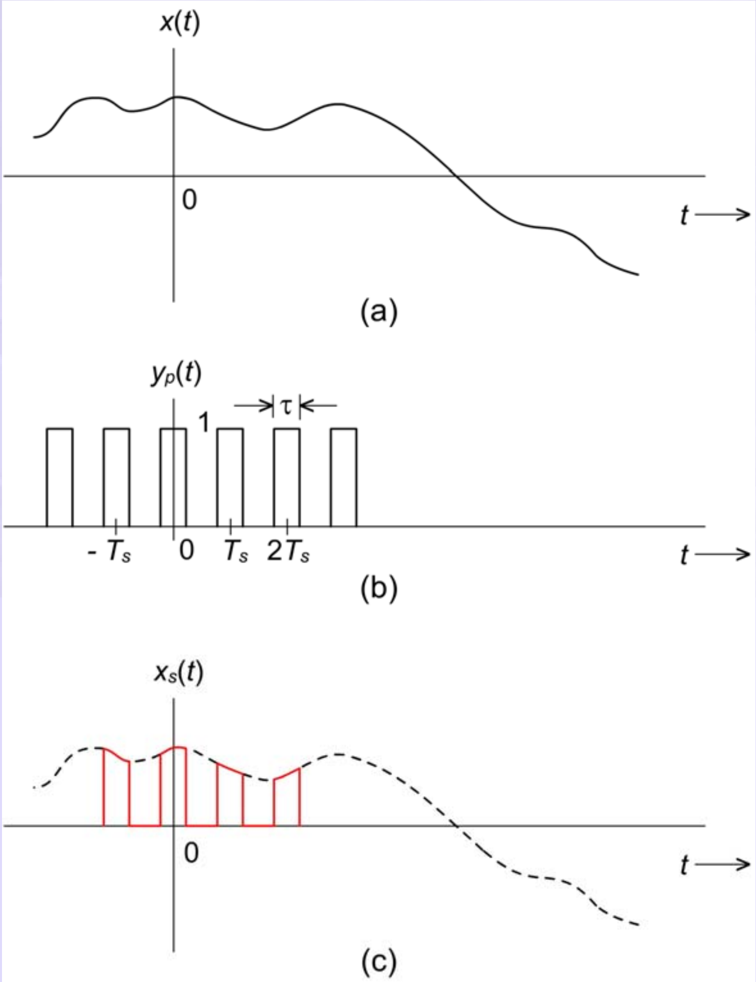


Fig. 6.5: Sampling with a rectangular pulse train

(a) $x(t)$, (b) the pulse train, (c) the sampled signal

Then, $X_s(f) = X(f) * Y_p(f)$ (6.4b)

But, from exercise 1.1, we have

$$Y_p(f) = \sum_{n = -\infty}^{\infty} \left(\frac{\tau}{T_s}\right) \text{sinc}(n f_s \tau)\, \delta(f - n f_s)$$

Hence,

$$X_s(f) = \frac{\tau}{T_s} \sum_{n = -\infty}^{\infty} \text{sinc}(n f_s \tau)\, X(f - n f_s) \tag{6.5}$$

$$= \frac{\tau}{T_s}\left[\cdots + \text{sinc}\left(\frac{\tau}{T_s}\right) X(f + f_s) + X(f) + \text{sinc}\left(\frac{\tau}{T_s}\right) X(f - f_s) + \cdots\right]$$

As $\text{sinc}\left(\frac{n\tau}{T_s}\right)$ is only a scale factor that depends on n, we find that $X(f)$ and its shifted replicas are weighted by different factors, unlike the previous case where all the weights were equal. A sketch of $X_s(f)$ is shown in Fig. 6.6 for $\frac{\tau}{T_s} = \frac{1}{10}$ and, $f_s > 2W$ and $X(f)$ of Fig. 6.3(a).

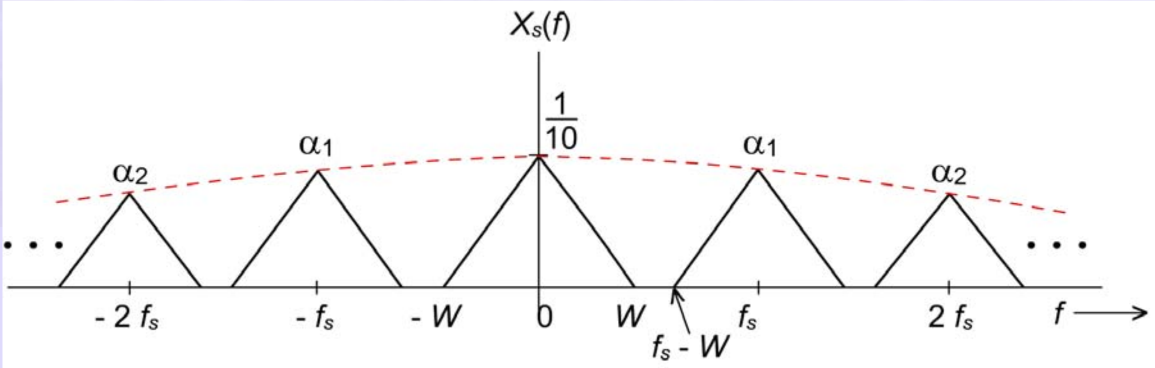


Fig. 6.6: Plot of Eq. 6.5

In the Fig. 6.6, $\alpha_1 = \frac{\text{sinc}(0.1)}{10} = 0.0983$, $\alpha_2 = \frac{\text{sinc}(0.2)}{10} = 0.0935$ etc. From Eq. 6.5 and Fig. 6.6, it is obvious that each lobe of $X_s(f)$ is multiplied by a different number. However as the scale factor is the same for a given spectral lobe of $X_s(f)$, there is no distortion. Hence $x(t)$ can be recovered from $x_s(t)$ of Eq. 6.4(a).

From Fig. 6.5(c), we see that during the time interval τ, when $y_p(t) = 1$, $x_s(t)$ follows the shape of $x(t)$. Hence, this method of sampling is also called *exact scanning.*

6.3.3 Flat topped sampling

Consider the scheme shown in Fig. 6.7.

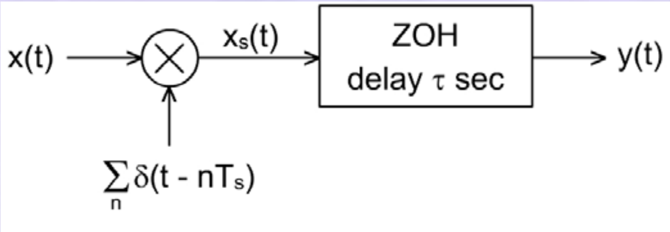


Fig. 6.7: Flat topped sampling

ZOH is the zero order hold (Example 1.15) with the impulse response

$$h(t) = ga\left(\frac{t - \tau/2}{\tau}\right)$$

$x_s(t)$ is the same as the one given by Eq. 6.1.

As $y(t) = x_s(t) * h(t)$, (6.6a)

$y(t)$ will be as shown (in red color) in Fig. 6.8.

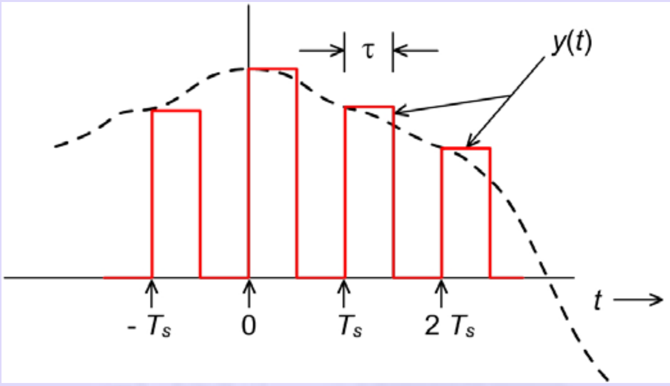


Fig. 6.8: Sampled waveform with flat top pulses

Holding the sample value constant provides time for subsequent coding operations. However, a large value of τ would introduce noticeable distortion and exact recovery of $x(t)$ is not possible unless $y(t)$ is passed through an **equalizer**, as will be seen shortly.

Taking the Fourier transform of Eq. 6.6(a), we have

$$Y(f) = X_s(f)\, H(f) \tag{6.6b}$$

where $H(f) = \tau\, e^{-j\pi f \tau} \operatorname{sinc}(f\tau)$

As $\quad X_s(f) = \dfrac{1}{T_s}\sum_n X(f - n f_s)$, we have

$$Y(f) = \frac{\tau}{T_s}\left[\sum_n X(f - n f_s)\right] e^{-j\pi f \tau} \operatorname{sinc}(f\tau) \tag{6.7}$$

Let $Y_0(f)$ denote the term for $n = 0$ in Eq. 6.7. That is,

$$Y_0(f) = \frac{\tau}{T_s} e^{-j\pi f \tau} X(f) \operatorname{sinc}(f\tau) \tag{6.8}$$

By holding the samples constant for a fixed time interval, we have introduced a delay of $\dfrac{\tau}{2}$ and more importantly $X(f)$ is multiplied with $\operatorname{sinc}(f\tau)$, which is a function of the variable f; that each spectral component of $X(f)$ is multiplied by a different number, which implies spectral distortion. Assuming that the delay of

$\frac{\tau}{2}$ is not serious, we have to *equalize* the amplitude distortion caused and this can be achieved by multiplying $Y_0(f)$ by $\frac{1}{|H(f)|}$. Of course, if $\frac{\tau}{T_s} < 0.1$, then the amplitude distortion may not be significant. The distortion caused by holding the pulses constant is referred to as the *aperture effect.*

6.3.4 Undersampling and the problem of aliasing

When $f_s < 2W$, we have seen that there is spectral overlap (Fig. 6.3d) and the resulting distortion is called the aliasing distortion. Let us try to understand this in some detail.

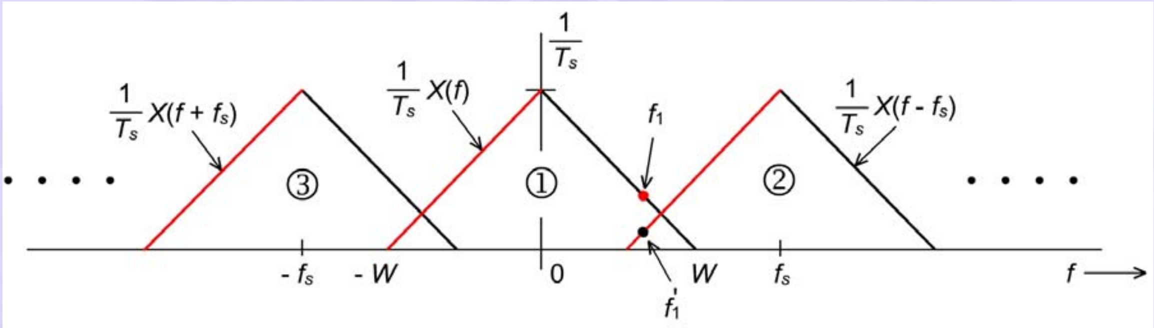


Fig. 6.9: A part of the aliased spectrum

Let $X_s(f)$ represent the aliased spectrum and let us consider the three lobes from it, namely, $\frac{1}{T_s} X(f)$, $\frac{1}{T_s} X(f - f_s)$ and $\frac{1}{T_s} X(f + f_s)$, indicated by ①, ②, and ③ respectively in Fig. 6.9. The left sides of these triangular spectra are shown in red. As can be seen from the figure, the complex exponential with the (negative) frequency f_1' is interfering with the frequency f_1. That is, $e^{-j2\pi f_1' t}$ is giving rise to the same set of samples as that $e^{j2\pi f_1 t}$, for the T_s chosen. We shall make this clear by considering the sampling of a cosine signal. Let the signal to be sampled be the cosine signal,

$$x(t) = 2\cos\left[(2\pi \times 4)t\right]$$

Its spectrum consists of two impulses, one at 4 Hz and the other at - 4 Hz as shown in Fig. 6.10(a), in which the negative frequency component has been shown with a broken line. Let $x(t)$ be sampled at the rate of 6 samples/sec.

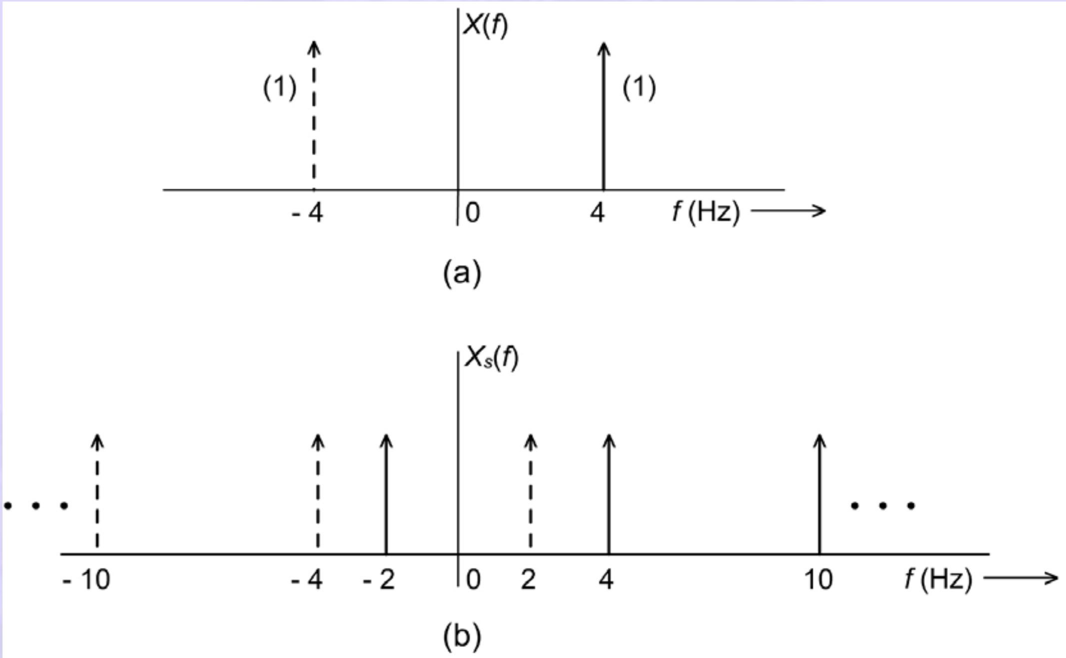


Fig. 6.10: Sampling of a cosine signal with aliasing

(Note that the Nyquist rate is greater than 8 samples per second.) The resulting spectrum is shown in Fig. 6.10(b). Notice that in $X_s(f)$, there are spectral components at ± 2 Hz. The impulse with a broken line at 2 Hz is originally - 4 Hz component. In other words, $e^{-j2\pi(4)t}$ is giving rise to same set of samples as $e^{j2\pi(2)t}$ when sampled with $T_s = \frac{1}{6}$ sec. This is easy to verify.

$$e^{-j2\pi(4)t}\Big|_{t = \frac{n}{6}} = e^{-j\frac{4\pi}{3}n} = e^{-j\frac{4\pi}{3}n} \cdot e^{j2\pi n}$$

$$= e^{j\frac{2\pi}{3}n} = e^{j2\pi(2)t}\Big|_{t = \frac{n}{6}}$$

Similarly, the set of values obtained from $e^{j8\pi t}$ for $t = nT_s$ is the same as those from $e^{-j4\pi t}$ (Notice that in Fig. 6.10(b), there is a solid line at $f = -2$). In other words, a 4 Hz cosine signal becomes an alias of 2 Hz cosine signal. If there were to be a 2 Hz component in $X(f)$, the 4 Hz will interfere with the 2 Hz component and thereby leading to distortion due to under sampling. In fact, a frequency component at $f = f_2$ is the alias of another lower frequency component at $f = f_1$ if $f_2 = k f_s \pm f_1$, where k is an integer. Hence, with $f_s = 6$, we find that the 4 Hz component is an alias the of 2 Hz component, because 4 = 6 - 2. Similarly, $\cos[2\pi(8)t]$ as 8 = 6 + 2. A few of the aliases of the 2 Hz component with $f_s = 6$ have been shown on the aliasing diagram of Fig. 6.11.

Fig. 6.11: Aliasing diagram with $f_s = 6$

As seen from the figure, upto 3 Hz there is no aliasing problem. Beyond that, the spectrum folds back, making a 4 Hz component align with a 2 Hz component. With repeated back and forth folding, we find that 8 Hz, 10 Hz, 14 Hz, etc. are aliases of 2 Hz and all of them will contribute to the output at 2 Hz, when $X_s(f)$ is filtered by an ILPF with cutoff at $\frac{f_s}{2} = 3$ Hz.

Note: Reconstruction of the analog signal is done by placing an ideal LPF with a cutoff at $\frac{f_s}{2}$. If $\cos(8\pi t)$ were to be sampled at 6 samples/sec, the output of the lowpass filter (with a cutoff at 3 Hz) is cosine signal with the frequency of 2 Hz!

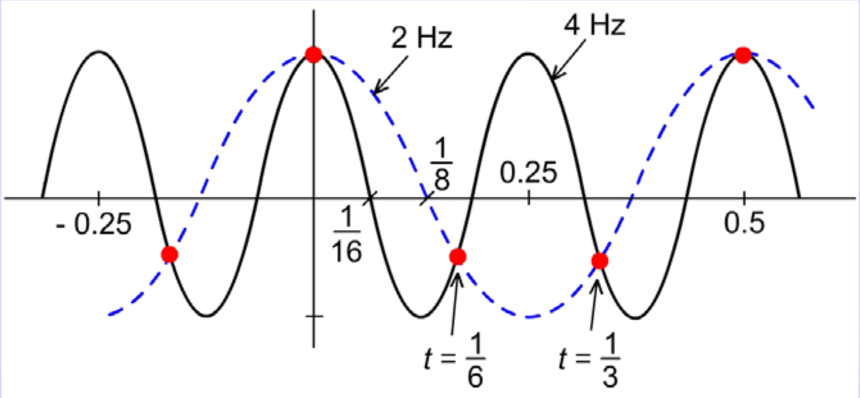


Fig. 6.12: Aliasing picture in the time domain

The concept of aliasing can also be illustrated in the time domain. This is shown in Fig. 6.12. The red dots represent the samples from the two cosine signals (one at 4 Hz and the other at 2 Hz) at $t = -\frac{1}{6}, 0, \frac{1}{6}, \frac{2}{6}, \frac{3}{6}$. By extrapolation, we find that $\cos(8\pi t)$ will give rise to same set of samples as $\cos(4\pi t)$ at $t = \frac{n}{6}$, $n = 0, \pm 1, \pm 2$ etc.

We will now demonstrate the effect of aliasing on voice and music signals. There are two voice recordings: one is a spoken sentence and the other is part of a Sanskrit sloka rendered by a person with a very rich voice. The music signal pertains to “sa re ga ma pa da ne ...” sounds generated by playing a flute. (The recording of the flute signal was done on a cell phone.)

1) a. The signal pertaining to the spoken sentence *God is subtle but not malicious*, is sampled at 44.1 ksps and reconstructed from these

samples.

b. The reconstructed signal after down sampling by a factor of 8.

2) a. Last line of a Sanskrit sloka. Sampling rate 44.1 ksps.

b. Reconstructed signal after down sampling by a factor of 8.

c. Reconstructed signal after down sampling by a factor of 12.

3) a. Flute music sampled at 8 ksps.

b. Reconstructed signal after down sampling by a factor of 2.

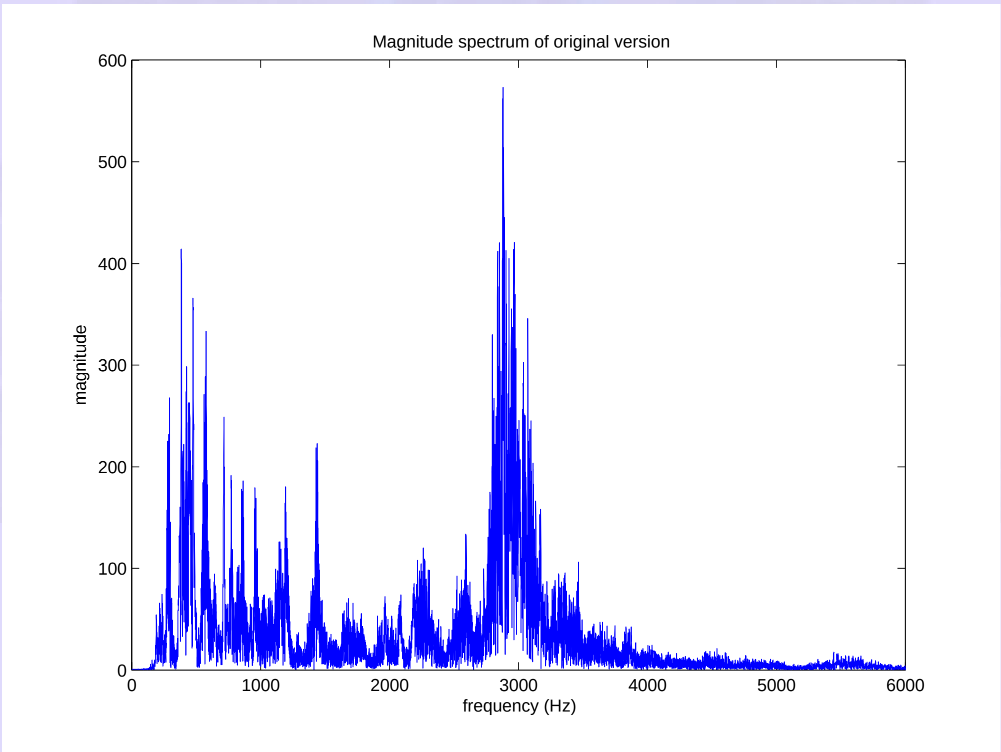


Fig. 6.13(a): Spectrum of the signal at (2a)

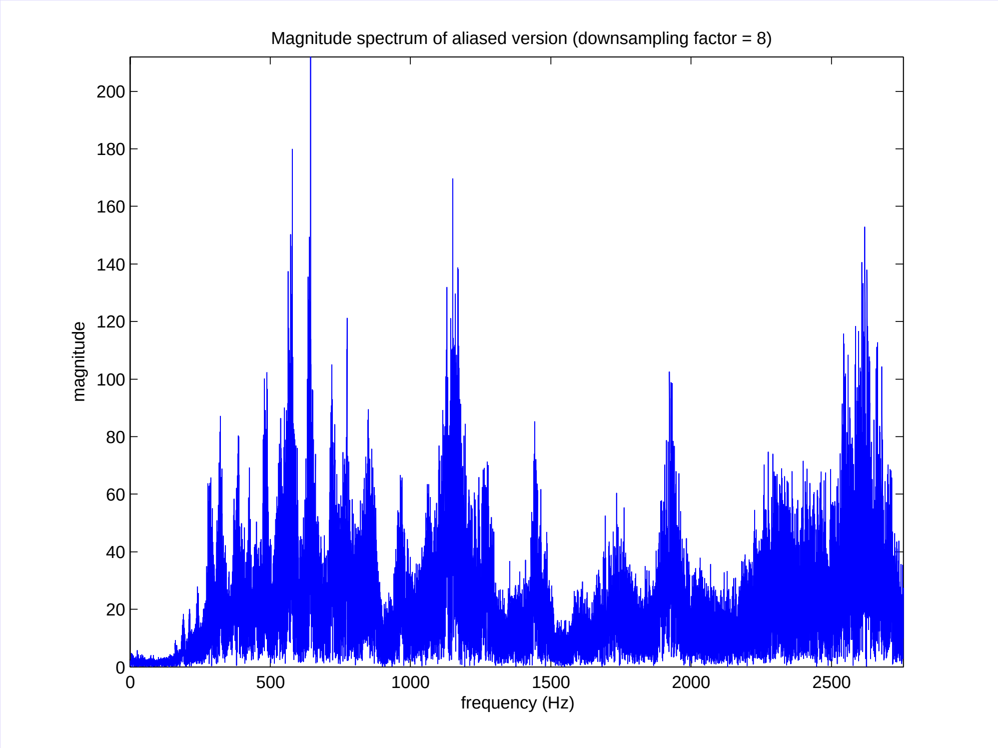


Fig. 6.13(b): Spectrum of the signal at (2b)

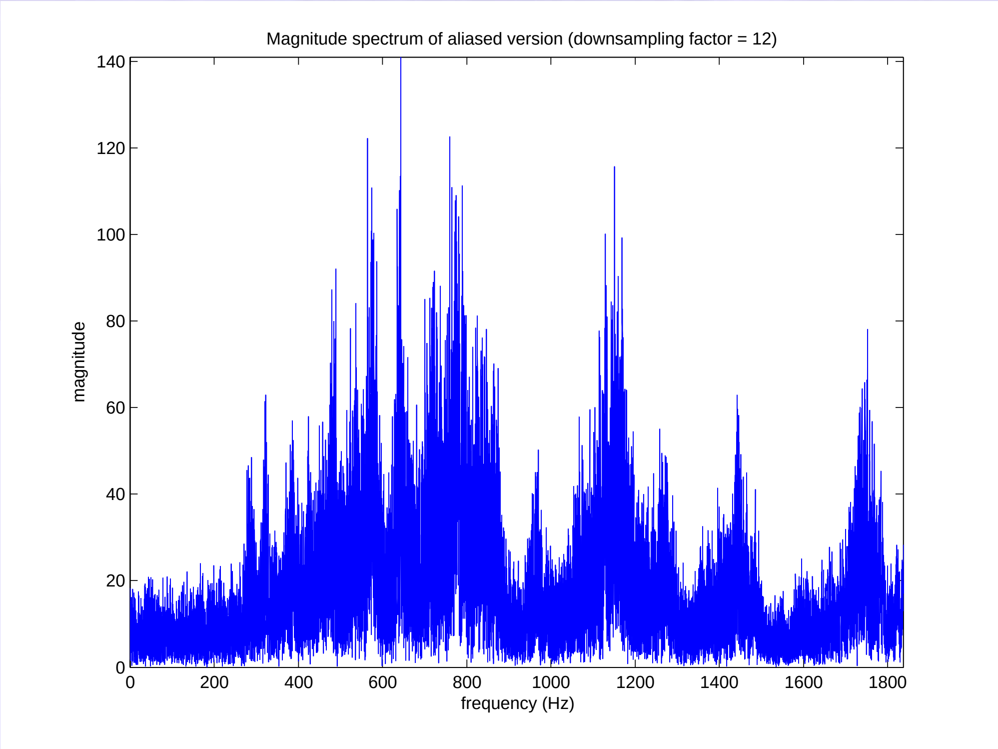


Fig. 6.13(c): Spectrum of the signal at (2c)

With respect to 1b, we find that there is noticeable distortion when the word *malicious* is being uttered. As this word gives rise to spectral components at the higher end of the voice spectrum, fold back of the spectrum results in severe aliasing. Some distortion is to be found elsewhere in the sentence.

Aliasing effect is quite evident in the voice outputs 2(b) and (c). (It is much more visible in 2(c).) We have also generated the spectral plots corresponding to the signals at 2(a), (b) and (c). These are shown in Fig. 6.13(a), (b) and (c). From Fig. 6.13(a), we see that the voice spectrum extends all the way upto 6 kHz with strong spectral components in the ranges 200 Hz to 1500 Hz and 2700 Hz to 3300 Hz. (Sampling frequency used is 44.1 kHz.)

Fig. 6.13(b) corresponds to the down sampled (by a factor of 8) version of (a); that is, sampling frequency is about 5500 samples/sec. Spectral components above 2750 Hz will get folded over. Because of this, we find strong spectral components in the range 2500-2750 Hz.

Fig. 6.13(c) corresponds to a sampling rate of 3600 samples/sec. Now spectrum fold over takes place with respect to 1800 Hz and this can be easily seen in Fig. 6.13(c).

As it is difficult to get rid of aliasing once it sets in, it is better to pass the signal through an anti-aliasing filter with cutoff at $\frac{f_s}{2}$ and then sample the output of the filter at the rate of f_s samples per second. This way, we are sure that all the spectral components upto $|f| \leq \frac{f_s}{2}$ will be preserved in the sampling process. Example 6.1 illustrates that the reconstruction mean square error is less than or equal to the error resulting from sampling without the anti-aliasing filter.

Example 6.1

An analog signal $x(t)$ is passed through an anti-aliasing filter with cutoff at f_c and then sampled with $f_s = 2 f_c$. (Note that $X(f)$ is not band limited to f_c.) Let $y_1(t)$ be the signal reconstructed from these samples and

$$e_1 = \int_{-\infty}^{\infty} \left(x(t) - y_1(t)\right)^2 dt$$

Now the anti-aliasing filter is withdrawn and $x(t)$ is sampled directly with $f_s = 2 f_c$ as before. Let $y_2(t)$ be the signal reconstructed from these samples and let $e_2 = \int_{-\infty}^{\infty} \left(x(t) - y_2(t)\right)^2 dt$. Show that $e_2 \geq e_1$.

Let $X(f)$ be as shown in the figure 6.14(a), with f_N being the highest frequency in it. $Y_1(f)$ is as shown in Fig. 6.14(b).

As can be shown from the figure, $Y_1(f)$ does not have any aliasing but that part of $X(f)$ for $f_c \leq |f| \leq f_N$ is missing. This introduces some distortion and the energy of the error signal,

$$e_1 = \int_{-\infty}^{\infty} \left(y_1(t) - x(t)\right)^2 dt$$

$$= \int_{-\infty}^{\infty} \left|Y_1(f) - X(f)\right|^2 df$$

But $\quad Y_1(f) = \begin{cases} X(f), & |f| \leq f_c \\ 0 \quad , & |f| > f_c \end{cases}.$

Hence, $e_1 = \int_{-\infty}^{-f_c} \left|X(f)\right|^2 df + \int_{f_c}^{\infty} \left|X(f)\right|^2 df \qquad (6.9a)$

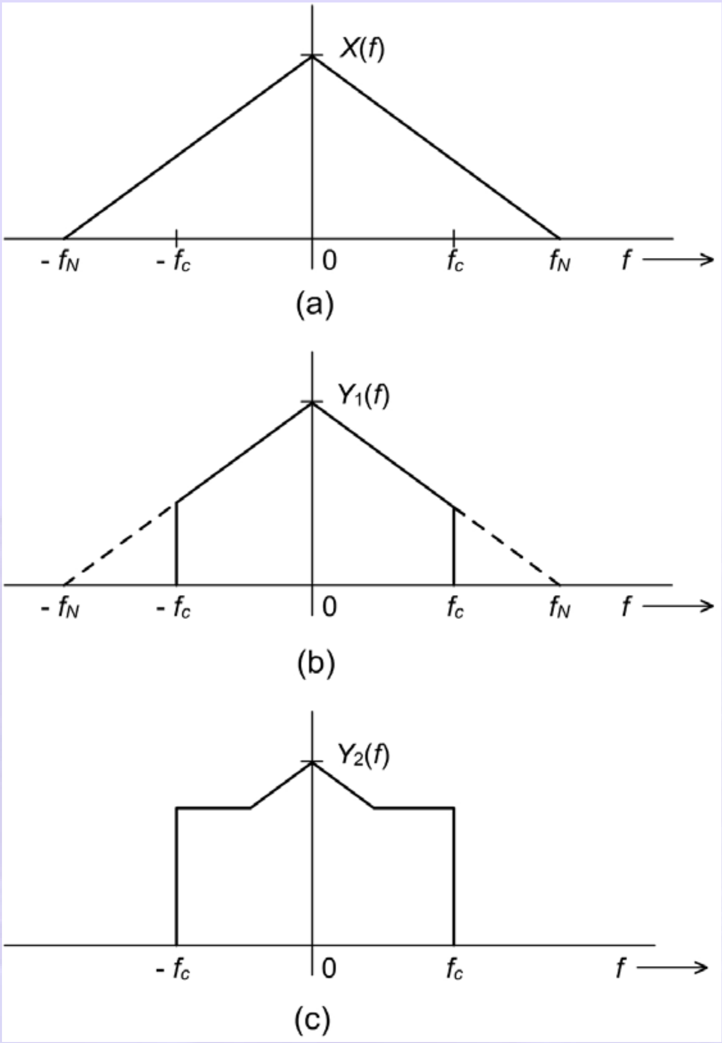


Fig. 6.14: (a) Original spectrum, $X(f)$

(b) Spectrum of the reconstructed signal after $x(t)$ is passed through an anti-aliasing filter before sampling

(c) Spectrum of the reconstructed signal with aliasing

It is evident from Fig. 6.14(c), that $y_2(t)$ suffers from aliasing and

$$e_2 = \int_{-\infty}^{-f_c} |X(f)|^2 \, df + \int_{f_c}^{\infty} |X(f)|^2 \, df + \int_{-f_c}^{f_c} |Y_2(f) - X(f)|^2 \, df \qquad (6.9b)$$

But $Y_2(f) \neq X(f)$ for $|f| \leq f_c$. Comparing Eqs. 6.9(a) and (b), we find that Eq. 6.9(b) has an extra term which is greater or equal to zero. Hence $e_2 \geq e_1$. ♦

Example 6.2

Find the Nyquist sampling rate for the signal $x(t) = \text{sinc}(200t)\,\text{sinc}^2(1000t)$.

$\text{sinc}(200t)$ has a rectangular spectrum in the interval $|f| \leq 100$ Hz and $\text{sinc}^2(1000t) = [\text{sinc}(1000t)][\text{sinc}(1000t)]$ has a triangular spectrum in the the frequency range $|f| \leq 1000$ Hz. Hence $X(f)$ has spectrum confined to the range $|f| \leq 1100$ Hz. This implies the Nyquist rate is 2200 samples/sec. ♦

Example 6.3

a) Consider the bandpass signal $x(t)$ with $X(f)$ as shown in Fig. 6.15. Let $x(t)$ be sampled at the rate of 20 samples per second. Sketch $X_s(f)$ (to within a scale factor), for the frequency range $20 \leq |f| \leq 30$ Hz.

b) Let $x(t)$ be now sampled as the rate of 24 samples per second. Sketch $X_s(f)$ (to within a scale factor) for $20 \leq |f| \leq 30$ Hz.

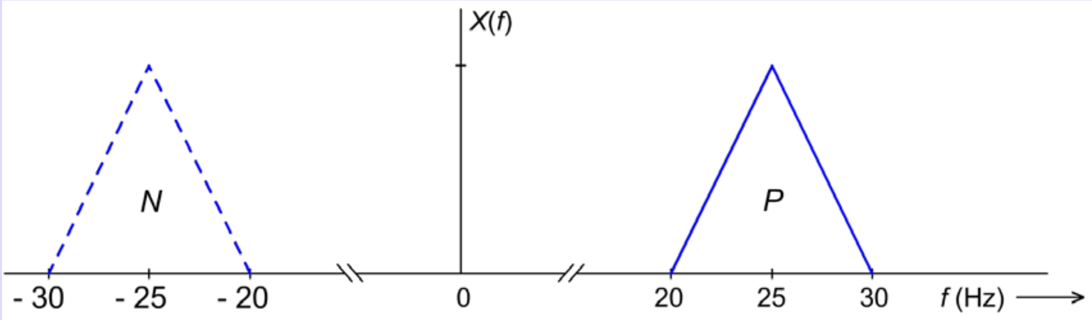


Fig. 6.15: $X(f)$ for example 6.3

a) Let P denote $X(f)$ for $f \geq 0$ and let N denote $X(f)$ for $f < 0$. (N is shown with a broken line.) Consider the right shifts of $X(f)$ in multiples of 20 Hz, which is the sampling frequency. Then P will be shifted away from the frequency interval of interest. We have to only check whether shifted N can contribute to spectrum in the interval $20 \leq f \leq 30$. It is easy to see that this will not happen. This implies, in $X_s(f)$, spectrum has the same shape as $X(f)$ for $20 \leq f \leq 30$. This is also true for the left shifts of $X(f)$. That is, for $20 \leq |f| \leq 30$, $X_s(f)$, to within a scale factor, is the same as $X(f)$.

b) Let $f_s = 24$. Consider again right shifts. When N is shifted to the right by $2f_s = 48$, it will occupy the interval $(-30 + 48) = 18$ to $(-20 + 48) = 28$. That is, we have the situation shown in Fig. 6.16.

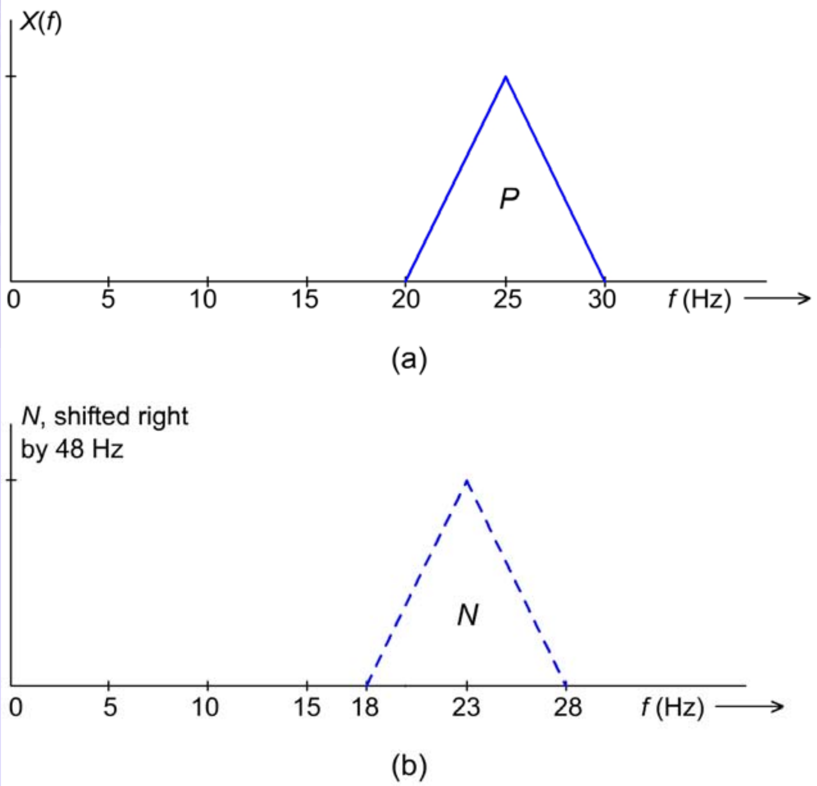


Fig. 6.16: The two lobes contributing to the spectrum for $20 \leq f \leq 30$

Let the sum of (a) and (b) be denoted by $Y(f)$. Then, $Y(f)$ is as shown in Fig. 6.17.

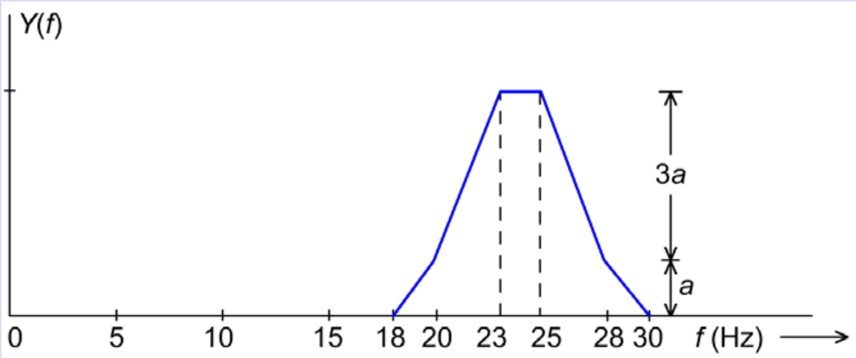


Fig. 6.17: Spectrum for $20 \leq f \leq 30$ when $f_s = 24$

For $-30 \leq f \leq -20$, $Y(f)$ is the mirror image of the spectrum shown in Fig. 6.17. Notice that for the bandpass signal, increasing f_s has resulted in the distortion of the original spectrum. ♦

Example 6.4

Let $x(t) = 2\cos(800\pi t) + \cos(1400\pi t)$. $x(t)$ is sampled with the rectangular pulse train $x_p(t)$ shown in Fig. 6.18. Find the spectral components in the sampled signal in the range 2.5 kHz to 3.5 kHz.

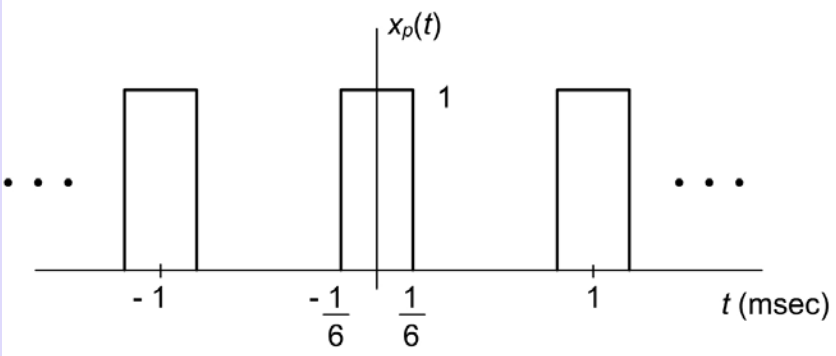


Fig. 6.18: $x_p(t)$ of the Example 6.4

$X(f)$ has spectral components at ± 400 Hz and ± 700 Hz. The impulses in $X_p(f)$ occur at ± 1 kHz, ± 2 kHz, ± 4 kHz, ± 5 kHz, etc. (Note the absence of spectral lines in $X_p(f)$ at $f = \pm 3$ kHz, ± 6 kHz, etc.) Convolution of $X(f)$ with the impulses in $X_p(f)$ at 2 kHz will give rise to spectral components at 2.4 kHz and 2.7 kHz. Similarly, we will have spectral components at $4 - (0.4) = 3.6$ kHz and $4 - (0.7) = 3.3$ kHz. Hence, in the range 2.5 kHz to 3.5 kHz, sampled spectrum will have two components, namely, at 2.7 kHz and 3.3 kHz. ♦

Exercise 6.1

A sinusoidal signal $x(t) = A\cos\left[\left(2\pi \times 10^3\right) t\right]$ is sampled with a uniform impulse train at the rate of 1500 samples per second. The sampled signal is passed through an ideal lowpass filter with a cutoff at 750 Hz. By sketching the appropriate spectra, show that the output is a sinusoid at 500 Hz. What would be the output if the cutoff frequency of the LPF is 950 Hz?

Exercise 6.2

Let $x(t)$ be the sum of two cosine signals. $X(f)$ is shown in Fig. 6.19. Let $x(t)$ be sampled at the rate of 300 samples per second. Let $x_s(t)$ be passed through an ideal LPF with a cutoff at 150 Hz.

a) Sketch the spectrum at the output of the LPF.

b) Write the expression for the output of the LPF.

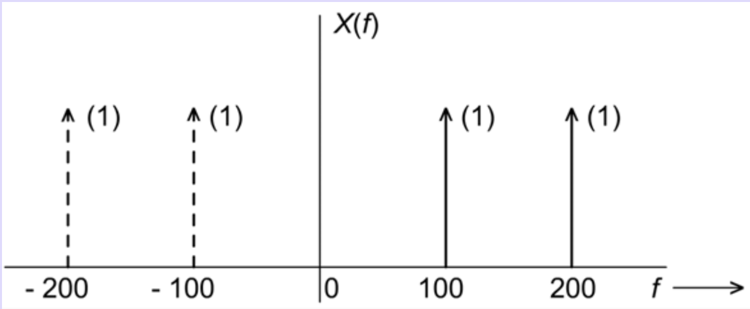


Fig. 6.19: $X(f)$ for the Exercise 6.2

6.4 Quantization

6.4.1 Uniform quantization

An analog signal, even it is limited in its peak-to-peak excursion, can in general assume any value within this permitted range. If such a signal is sampled, say at uniform time intervals, the number of different values the samples can assume is unlimited. Any human sensor (such as ear or eye) as the ultimate receiver can only detect finite intensity differences. If the receiver is not able to distinguish between two sample amplitudes, say v_1 and v_2 such that $|v_1 - v_2| < \frac{\Delta}{2}$, then we can have a set of discrete amplitude levels separated by Δ and the original signal with continuous amplitudes, may be approximated by a signal constructed of discrete amplitudes selected on a minimum error basis from an available set. This ensures that the magnitude of the error between the actual sample and its approximation is within $\frac{\Delta}{2}$ and this difference is *irrelevant* from the receiver point of view. The realistic assumption that a signal $m(t)$ is (essentially) limited in its peak-to-peak variations and any two adjacent (discrete) amplitude levels are separated by Δ will result in a **finite number** (say L) **of discrete amplitudes** for the signal.

The process of conversion of analog samples of a signal into a set of discrete (digital) values is called **quantization**. Note however, that quantization is inherently information lossy. For a given peak-to-peak range of the analog signal, smaller the value of Δ, larger is the number of discrete amplitudes and hence, finer is the quantization. Sometimes one may resort to somewhat coarse quantization, which can result in some noticeable distortion; this may, however be acceptable to the end receiver.

The quantization process can be illustrated graphically. This is shown in Fig. 6.20(a). The variable x denotes the input of the quantizer, and the variable

y represents the output. As can be seen from the figure, the quantization process implies that a straight line relation between the input and output (broken line through the origin) of a linear continuous system is replaced by a staircase characteristic.

The difference between two adjacent discrete values, Δ, is called the *step size* of the quantizer. The error signal, that is, difference between the input and the quantizer output has been shown in Fig. 6.20(b). We see from the figure that the magnitude of the error is always less than or equal to $\frac{\Delta}{2}$. (We are assuming that the input to the quantizer is confined to the range $\left(-\frac{7\Delta}{2} \text{ to } \frac{7\Delta}{2}\right)$.

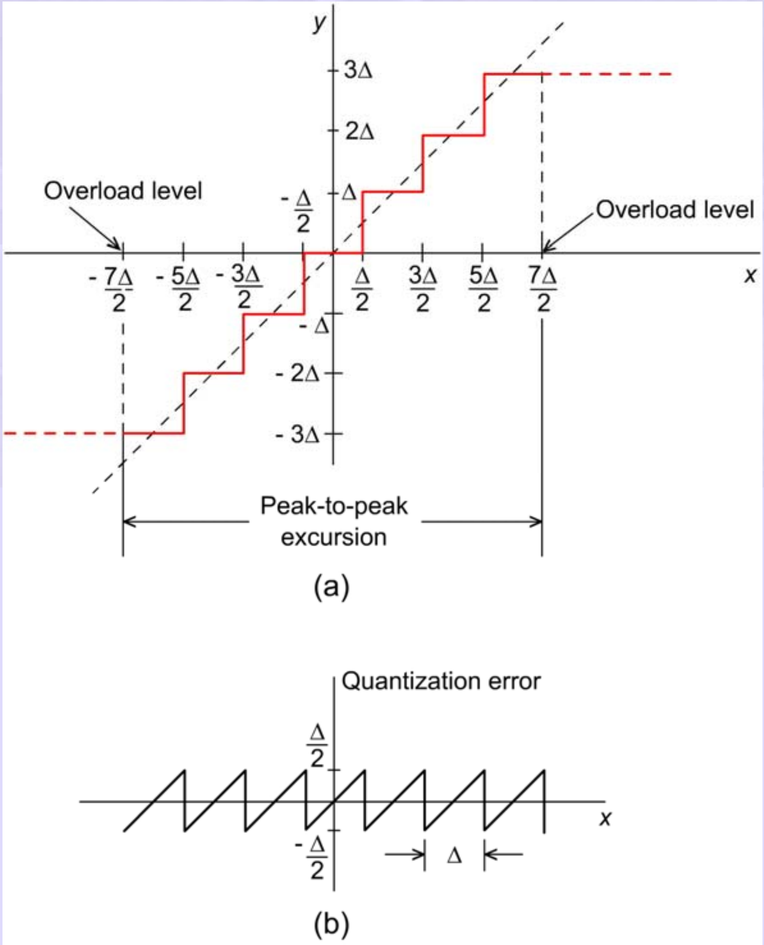


Fig. 6.20: (a) Quantizer characteristic of a uniform quantizer (b) Error signal

Let the signal $m(t)$, shown in Fig. 6.21(a), be the input to a quantizer with the quantization levels at $0, \pm 2$ and ± 4. Then, $m_q(t)$, the quantizer output is the waveform shown in red. Fig. 6.21(b) shows the error signal $e(t) = m(t) - m_q(t)$ as a function of time. Because of the noise like appearance of $e(t)$, it is a common practice to refer to it as the *quantization noise*. If the input to the quantizer is the set of (equispaced) samples of $m(t)$, shown in Fig. 6.21(c), then the quantizer output is the sample sequence shown at (d).

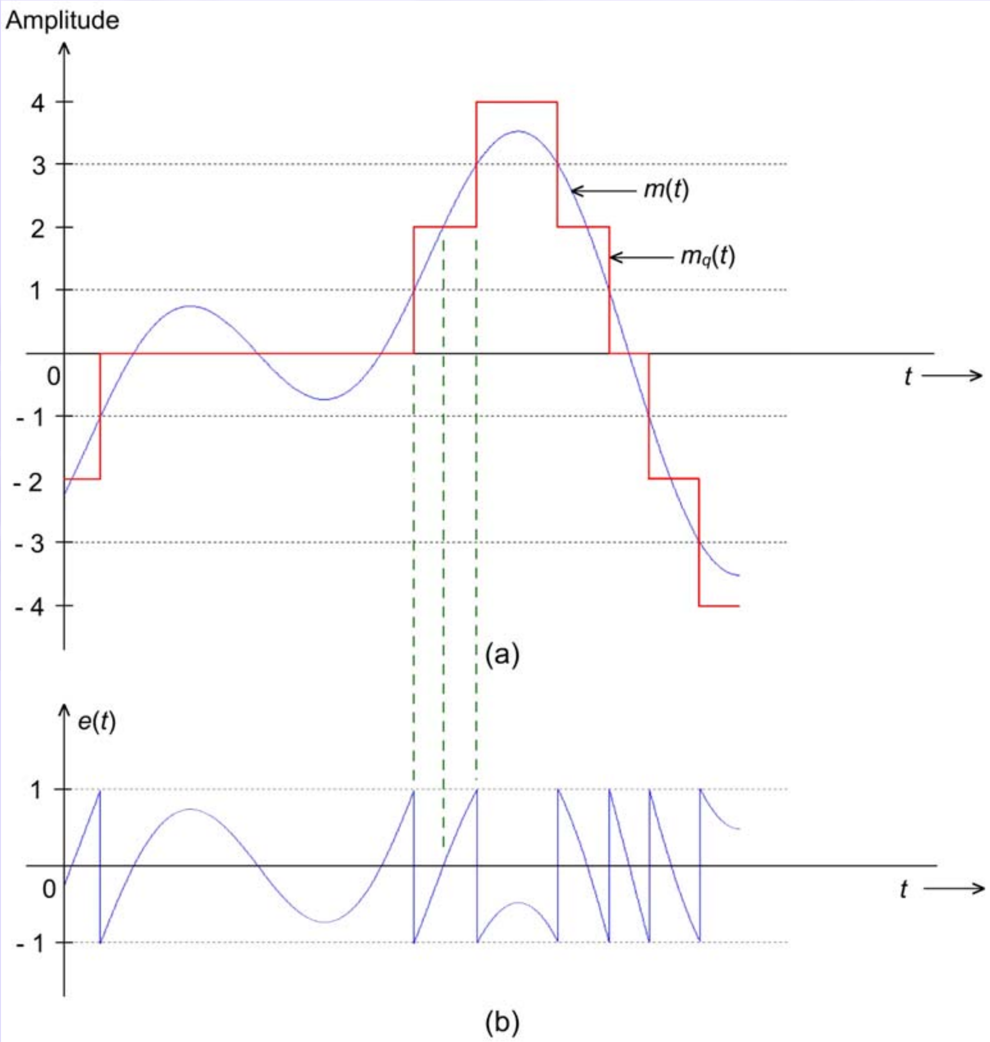


Fig. 6.21: (a) An analog signal and its quantized version (b) The error signal

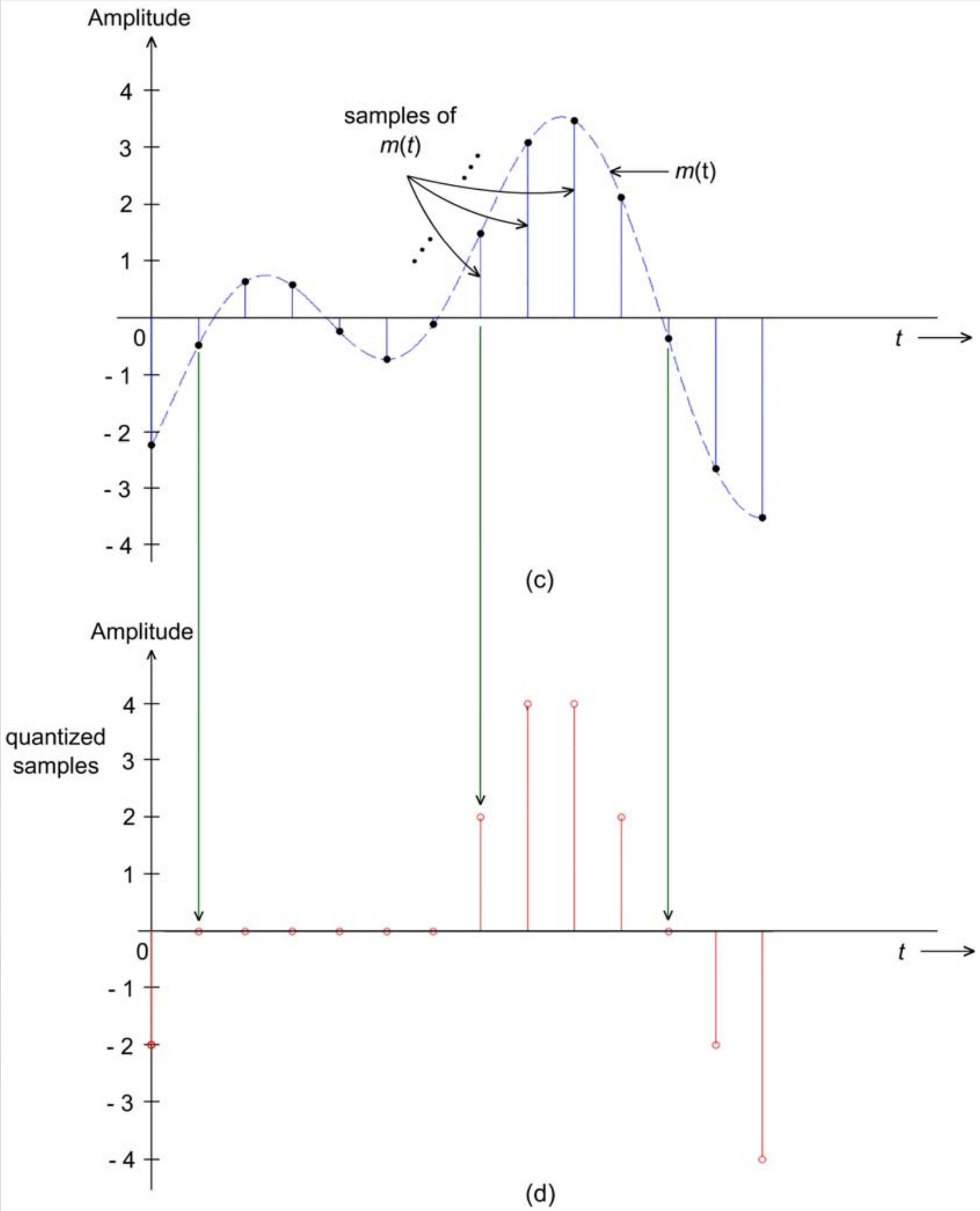


Fig. 6.21: (c) Equispaced samples of $m(t)$ (d) Quantized sample sequence

For the quantizer characteristic shown in Fig. 6.20(a), the maximum value of the quantizer output has been shown as 3Δ and the minimum as (-3Δ). Hence, if the input x is restricted to the range $\pm 3.5\Delta$ (that is,

$\left.x_{\max} = -x_{\min} \le \frac{7\Delta}{2}\right)$, the quantization error magnitude is less than or equal to $\frac{\Delta}{2}$. Hence, (for the quantizer shown), $\pm 3.5\Delta$ is treated as the *overload* level.

The quantizer output of Fig. 6.20(a) can assume values of the form $\Delta \cdot H_i$ where $\pm H_i = 0, 1, 2, \cdots$. A quantizer having this input-output relation is said to be of the *mid-tread* type, because the origin lies in the middle of a tread of a staircase like graph.

A quantizer whose output levels are given by $H_i \frac{\Delta}{2}$, where $\pm H_i = 1, 3, 5, \cdots$ is referred to as the *mid-riser* type. In this case, the origin lies in the middle of the rising part of the staircase characteristic as shown in Fig. 6.22. Note that overload level has not been indicated in the figure. The difference in performance of these two quantizer types will be brought out in the next sub-section in the context of idle-channel noise. The two quantizer types described above, fall under the category of *uniform quantizers* because the step size Δ of the quantizer is constant. A *non-uniform quantizer* is characterized by a variable step size. This topic is taken up in the sub-section 6.4.3.

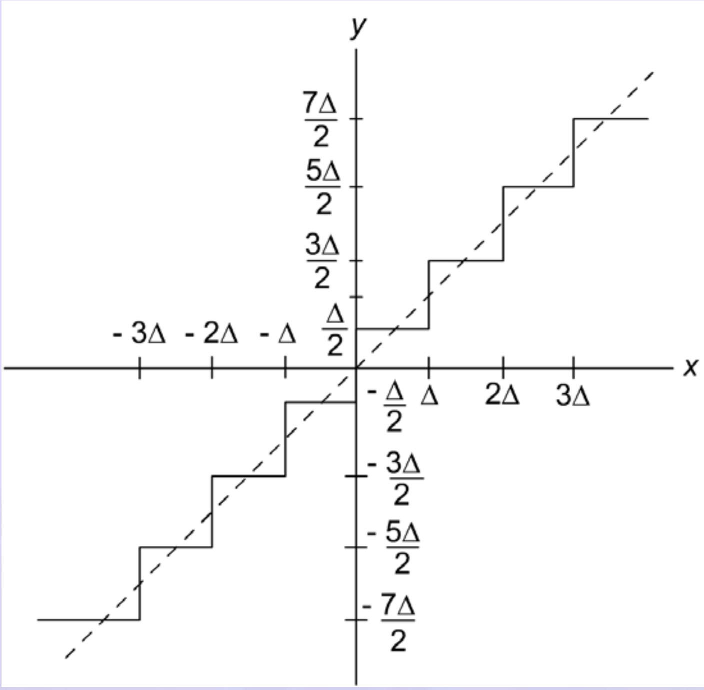


Fig. 6.22: Mid-riser type quantizer

6.4.2 Quantization Noise

Consider a uniform quantizer with a total of L quantization levels, symmetrically located with respect to zero. Let x represent the quantizer input, and $y = QZ(x)$ denote its output. Consider a part of the quantizer operating range shown in Fig. 6.23.

Let I_k be the interval defined by

$$I_k = \{x_k < x \leq x_{k+1}\}, \; k = 1, 2, \cdots, L.$$

Then $y = y_k$, if $x \in I_k$; we can also express y_k as

$$y_k = x + q, \text{ if } x \in I_k$$

where q denotes the quantization error, with $|q| \leq \frac{\Delta}{2}$.

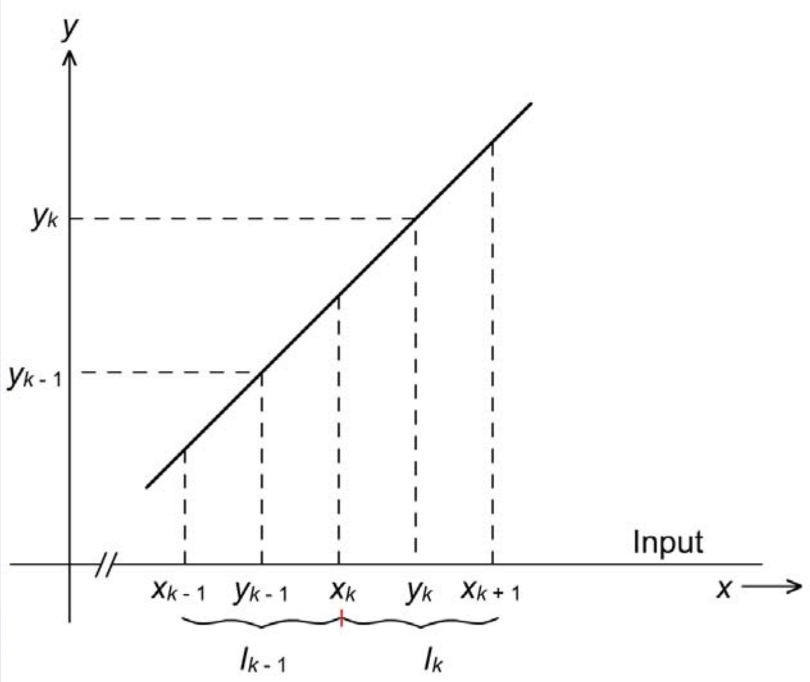


Fig. 6.23: Input intervals and the corresponding output levels of a quantizer

Let us assume that the input x is the sample value of a random variable X with zero mean and variance σ_X^2. When the quantization is fine enough, the distortion produced by the quantization affects the performance of a PCM system as though it were an additive, independent source of noise with zero mean and mean square value determined by the quantizer step size Δ.

Let the random variable Q denote the quantization error, and let q denote its sample value. Assuming Q is uniformly distributed in the range $\left(-\frac{\Delta}{2}, \frac{\Delta}{2}\right)$, we have

$$f_Q(q) = \begin{cases} \frac{1}{\Delta}, & -\frac{\Delta}{2} \le q \le \frac{\Delta}{2} \\ 0 \ , & otherwise \end{cases}$$

where $f_Q(q)$ is the PDF of the quantization error. Hence, the variance of the quantization error, σ_Q^2 is

$$\sigma_Q^2 = \frac{\Delta^2}{12} \tag{6.10}$$

The reconstructed signal at the receiver output, can be treated as the sum of the original signal and quantization noise. We may therefore define an *output signal-to-quantization noise ratio* $(SNR)_{0,\,q}$ as

$$(SNR)_{0,\,q} = \frac{\sigma_X^2}{\sigma_Q^2} = \frac{12\,\sigma_X^2}{\Delta^2} \tag{6.11a}$$

As can be expected, Eq. 6.11(a) states the result "for a given σ_X, smaller the value of the step size Δ, larger is the $(SNR)_{0,\,q}$". Note that σ_Q^2 is independent of the input PDF provided overload does not occur. $(SNR)_{0,\,q}$ is usually specified in dB.

$$(SNR)_{0,q}\,(\text{in dB}) = 10\,\log_{10}\left[(SNR)_{0,q}\right] \tag{6.11b}$$

Idle Channel Noise

Idle channel noise is the coding noise measured at the receiver output with zero input to the transmitter. (In telephony, zero input condition arises, for example, during a silence in speech or when the microphone of the handset is covered either for some consultation or deliberation). The average power of this form of noise depends on the type of quantizer used. In a quantizer of the mid-riser type, zero input amplitude is coded into one of two innermost representation levels, $\pm\frac{\Delta}{2}$. Assuming that these two levels are equiprobable, the idle channel noise for mid-riser quantizer has a zero mean with the mean squared value

$$\frac{1}{2}\left(\frac{\Delta}{2}\right)^2 + \frac{1}{2}\left(-\frac{\Delta}{2}\right)^2 = \frac{\Delta^2}{4}$$

On the other hand, in a quantizer of the mid-tread type, the output is zero for zero input and the ideal channel noise is correspondingly zero. Another difference between the mid-tread and mid-riser quantizer is that, the number of quantization levels is odd in the case of the former where as it is even for the latter. But for these minor differences, the performance of the two types of quantizers is essentially the same.

Example 6.5

Let a baseband signal $x(t)$ be modeled as the sample function of a zero mean Gaussian random process $X(t)$. $x(t)$ is the input to a quantizer with the overload level set at $\pm 4\sigma_X$, where σ_X is the variance of the input process (In other words, we assume that the $P\left[|x(t)| \geq 4\sigma_X\right] \simeq 0$). If the quantizer output is coded with R-bit sequence, find an expression for the $(SNR)_{0,\,q}$.

With R-bits per code word, the number of quantization levels, $L = 2^R$. For calculating the step size Δ, we shall take $x_{max} = |x(t)|_{max}$ as $4\,\sigma_X$.

Step size $\Delta = \dfrac{2\,x_{max}}{L} = \dfrac{8\,\sigma_X}{L} = \dfrac{8\,\sigma_X}{2^R}$

$$(SNR)_{0,\,q} = \frac{\sigma_X^2}{\sigma_Q^2} = \frac{\sigma_X^2}{\dfrac{\Delta^2}{12}}$$

$$= \frac{\left(12\,\sigma_X^2\right)2^{2R}}{64\,\sigma_X^2}$$

$$= \frac{3}{16} \cdot 2^{2R}$$

$$(SNR)_{0,\,q}\,(\text{in dB}) = 6.02R - 7.2 \qquad \blacklozenge \qquad (6.12)$$

This formula states that each bit in the code word of a PCM system contributes 6 dB to the signal-to-noise ratio. Remember that this formula is valid, provided,

i) The quantization error is uniformly distributed.

ii) The quantization is fine enough (say $n \geq 6$) to prevent signal correlated patterns in the error waveform.

iii) The quantizer is aligned with the input amplitude range (from $-4\sigma_X$ to $4\sigma_X$) and the probability of overload is negligible.

Example 6.6

Let $x(t)$ be modeled as the sample function of a zero mean stationary process $X(t)$ with a uniform PDF, in the range $(-a, a)$. Let us find the $(SNR)_{0,q}$ assuming an R-bit code word per sample.

Now the signal $x(t)$ has only a finite support; that is, $x_{max} = a$. Its variance, $\sigma_X^2 = \dfrac{a^2}{3}$.

Step size $\Delta = \dfrac{2a}{2^R}$

$$= a\,2^{-(R-1)}$$

Hence, $(SNR)_{0,q} = \dfrac{\sigma_M^2}{\left(\dfrac{\Delta^2}{12}\right)} = \dfrac{\dfrac{a^2}{3}}{\dfrac{a^2\,2^{-2(R-1)}}{12}}$

$$= 2^{2R}$$

$$(SNR)_{0,q}\,(\text{in dB}) = 6.02R \qquad (6.13a)$$

Even in this case, we find the 6 dB per bit behavior of the $(SNR)_{0,q}$. ♦

Example 6.7

Let $m(t)$, a sinusoidal signal with the peak value of A be the input to a uniform quantizer. Let us calculate the $(SNR)_{0,q}$ assuming R-bit code word per sample.

Step size $\Delta = \frac{2A}{2^R}$

Signal power $= \frac{A^2}{2}$

$$(SNR)_{0,\,q} = \frac{A^2}{2} \frac{2^{2R}\, 12}{4A^2}$$

$$= \frac{3}{2}\left(2^{2R}\right)$$

$$(SNR)_{0,\,q}\,(\text{in dB}) = 6.02R + 1.8 \qquad (6.13b)$$

♦

We now make a general statement that $(SNR)_{0,\,q}$ of a PCM system using a uniform quantizer can be taken as $6R + \alpha$, where R is the number of bits/code word and α is a constant that depends on the ratio $\left(\frac{x_{rms}}{x_{\max}}\right) = \left(\frac{\sigma_X}{x_{\max}}\right)$, as shown below.

$$\Delta = \frac{2x_{\max}}{2^R}$$

$$\sigma_Q^2 = \frac{\Delta^2}{12} = \frac{x_{\max}^2}{2^{2R} \times 3}$$

$$(SNR)_{0,q} = \frac{\sigma_X^2}{\sigma_Q^2} = 3\left(\frac{\sigma_X}{x_{\max}}\right)^2 \cdot 2^{2R}$$

$$10 \log_{10}(SNR)_{0,q} = 4.77 + 6.02R + 20 \log_{10}\left(\frac{\sigma_X}{x_{\max}}\right)$$

$$= 6.02\,R + \alpha \qquad (6.14a)$$

where $\alpha = 4.77 + 20 \log_{10}\left(\frac{\sigma_X}{x_{\max}}\right)$ (6.14b)

Example 6.8

Let $m(t)$ be a real bandpass signal with a non-zero spectrum (for $f > 0$) only in the range f_L to f_H, with $f_H \geq f_L$. Then, it can be shown that, the minimum sampling frequency[1] to avoid aliasing is,

$$(f_s)_{min} = 2B \frac{\left(1 + \frac{f_L}{B}\right)}{1 + \left\lfloor \frac{f_L}{B} \right\rfloor} = \frac{2 f_H}{\lfloor k \rfloor} \tag{6.15}$$

Where $B = f_H - f_L$, $k = \frac{f_H}{B}$ and $\lfloor x \rfloor$ denotes the integer part of x. Let $m(t)$ be a bandpass signal with $f_L = 3$ MHz and $f_H = 5$ MHz. This signal is to be sent using PCM on a channel whose capacity is limited to 7 Mbps. Assume that the samples of $m(t)$ are uniformly distributed in the range - 2 V to 2 V.

a) Show that the minimum sampling rate as given by Eq. 6.15 is adequate to avoid aliasing.

b) How many (uniform) quantization levels are possible and what are these levels so that the error is uniform with in $\pm \frac{\Delta}{2}$.

c) Determine the $(SNR)_{0,\, q}$ (in dB) that can be achieved.

a) Eq. 6.15 gives $(f_s)_{min}$ as

$$(f_s)_{min} = 2 \times 2 \frac{1 + \frac{3}{2}}{1 + \left\lfloor \frac{3}{2} \right\rfloor} \times 10^6$$

$$= 5 \times 10^6 \text{ samples/sec}$$

Let $M(f)$, the spectrum of $m(t)$ be as shown in Fig. 6.24.

---

[1] Note that a sampling frequency $f_s > (f_s)_{min}$ could result in aliasing (see Example 6.3) unless $f_s \geq 2 f_H$, in which case the sampling of a bandpass signal reduces to that of sampling a lowpass signal.

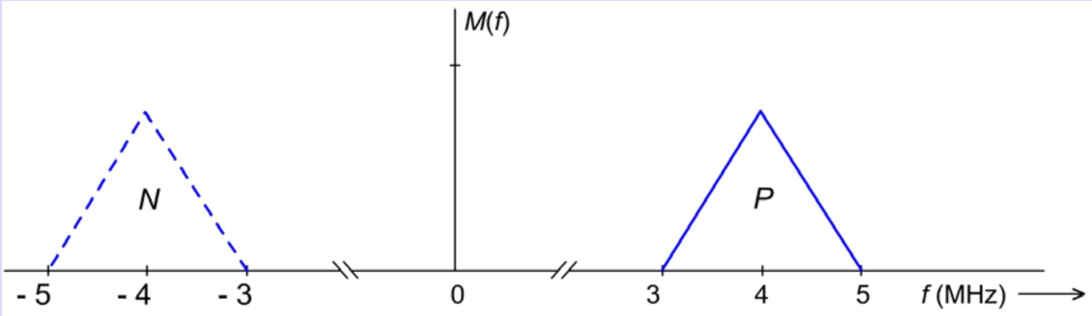


Fig. 6.24: Bandpass spectrum of Example 6.8

As $f_s = 5 \times 10^6$ samples/sec, $M(f)$ is to be repeated at intervals of 5×10^6 Hz. Shifting N to the right by 5 MHz will create a spectral lobe between 0 to 2 MHz whereas shifting by 10 MHz will result in a spectral lobe occupying the range 5 to 7 MHz. That is, right shifts of N will not interfere with P. Similarly, left shifts of P (in multiples of 5 MHz) will not fall in the range - 5 MHz to - 3 MHz. Hence, there is no aliasing in the sampling process.

b) As the sampling rate is 5×10^6 samples/sec and the capacity of the channel is limited to 7Mbps; it is possible to send only one bit per sample; that is, we have only a two-level quantization. The quantizer levels are at ± 1 so that step size Δ is 2 V and the error is uniform in the range ± 1.

c) As the signal is uniformly distributed in the range $(-2, 2)$, we have

$$\sigma_M^2 = \frac{(4)^2}{12} = \frac{4}{3}.$$

Variance of the quantization noise, $\sigma_Q^2 = \frac{(2)^2}{12} = \frac{1}{3}$

Hence $(SNR)_{0,q} = \frac{4/3}{1/3} = 4$

$(SNR)_{0,q}\,(\text{dB}) = 10 \log_{10} 4 = 6.02$ dB ♦

When the assumption that quantization noise is uniformly distributed between $\pm \frac{\Delta}{2}$, is **not** valid, then we have to take a more basic approach in calculating the noise variance. This is illustrated with the help of Example 6.9.

Example 6.9

Consider the quantizer characteristic shown in Fig. 6.25(a). Let X be the input to the quantizer with $f_X(x)$ as shown at Fig. 6.25(b). Find

a) the value of A

b) the total quantization noise variance, σ_Q^2

c) Is it is the same as $\frac{\Delta^2}{12}$?

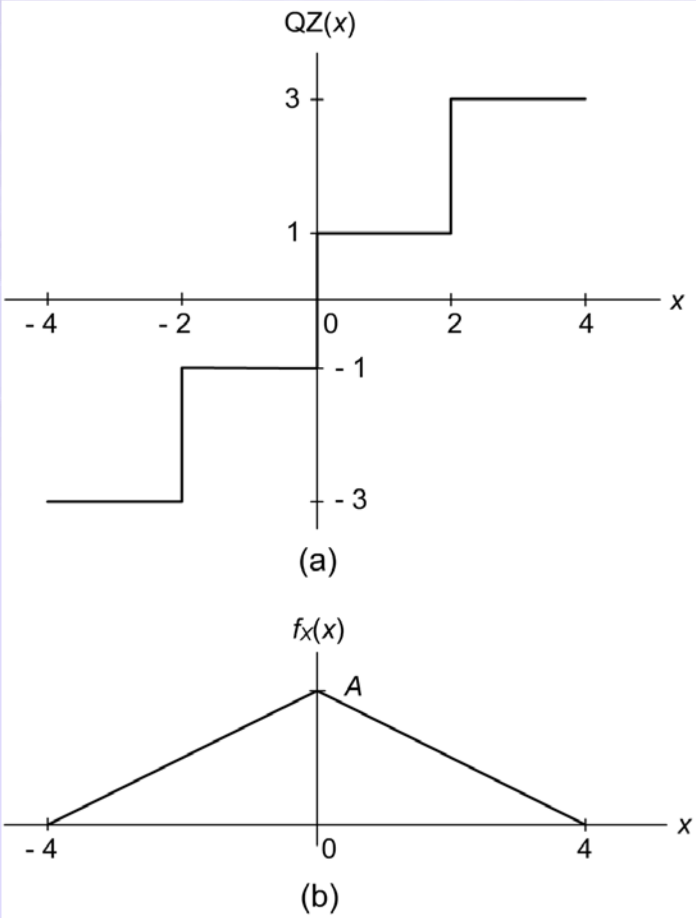


Fig. 6.25(a): Quantizer characteristic (b) the input PDF (Example 6.9)

a) As $\int_{-4}^{4} f_X(x)\, dx = 1$, $A = \frac{1}{4}$

b) $f_X(x) = \begin{cases} \frac{1}{4} - \frac{1}{16}|x|, & |x| \le 4 \\ 0 & , \ otherwise \end{cases}$

Let us calculate the variance of the quantization noise for $x \ge 0$. Total variance is twice this value. For $x > 0$, let

$$\sigma_Q'^2 = \int_0^2 (x-1)^2 f_X(x)\, dx + \int_2^4 (x-3)^2 f_X(x)\, dx$$

Carrying out the calculations, we have

$\sigma_Q'^2 = \frac{1}{6}$ and hence, $\sigma_Q^2 = \frac{1}{3}$

c) As $\frac{\Delta^2}{12} = \frac{4}{12} = \frac{1}{3}$, in this case we have σ_Q^2 the same as $\frac{\Delta^2}{12}$. ♦

Exercise 6.3

The input to the quantizer of Example 6.9 (Fig. 6.25(a)) is a random variable with the PDF,

$$f_X(x) = \begin{cases} Ae^{-|x|}, & |x| \le 4 \\ 0 & , \ otherwise \end{cases}$$

Find the answers to (a), (b) and (c) of Example 6.9.

Answers: $\sigma_Q^2 = 0.037 \ne \frac{\Delta^2}{12}$.

Exercise 6.4

A random variable X, which is uniformly distributed in the range 0 to 1 is quantized as follows:

$QZ(x) = 0, \ 0 \le x \le 0.3$

$QZ(x) = 0.7, \ 0.3 < x \le 1$

Show that the root-mean square value of the quantization is 0.198.

Exercise 6.5

A signal $m(t)$ with amplitude variations in the range $-m_p$ to m_p is to be sent using PCM with a uniform quantizer. The quantization error should be less than or equal to 0.1 percent of the peak value m_p. If the signal is band limited to 5 kHz, find the minimum bit rate required by this scheme.

Ans: 10^5 bps

6.4.3 Non-uniform quantization and companding

In certain applications, such as PCM telephony, it is preferable to use a non-uniform quantizer, where in the step size Δ is not a constant. The reason is as follows. The range of voltages covered by voice signals (say between the peaks of a loud talk and between the peaks of a fairly weak talk), is of the order of 1000 to 1. For a uniform quantizer we have seen that $(SNR)_{0,\,q} = \dfrac{12\,\sigma_X^2}{\Delta^2}$, where Δ is a constant. Hence $(SNR)_{0,\,q}$ is decided by signal power. For a person with a loud tone, if the system can provide an $(SNR)_{0,\,q}$ of the order of 40 dB, it may not be able to provide even 10 dB $(SNR)_{0,\,q}$ for a person with a soft voice. As a given quantizer would have to cater to various speakers, (in a commercial setup, there is no dedicated CODEC[1] for a given speaker), uniform quantization is not the proper method. A non-uniform quantizer with the feature that step size increases as the separation from the origin of the input -output characteristic is increased, would provide a much better alternative. This is because larger signal amplitudes are subjected to coarser quantization and the weaker passages are subjected to fine quantization. By choosing the quantization characteristic properly, it would be possible to obtain an acceptable $(SNR)_{0,\,q}$ over a wide range of input signal variation. In other words, such a

[1] CODEC stands for COder-DECoder combination

quantizer favors weak passages (which need more protection) at the expense of loud ones.

The use of a non-uniform quantizer is equivalent to passing the baseband signal through a compressor and then applying the compressed signal to a uniform quantizer. In order to restore the signal samples to their correct relative level, we must of course use a device in the receiver with a characteristic complimentary to the compressor. Such a device is called an expander. A non-uniform quantization scheme based on this principle is shown in Fig. 6.26. In the scheme shown, $C(x)$ denotes the output of the compressor for the input x. The characteristic $C(x)$ is a monotonically increasing function that has odd symmetry, $C(-x) = -C(x)$. Ideally, the compression and expansion laws are exactly inverses so that except for the effect of quantization, the expander output is equal to the compressor input. In the scheme of Fig. 6.26, $C^{-1}(.)$ denotes the expander characteristic. The combination of a COMpressor and an exPANDER is called a *compander*. We shall now derive the compressor characteristic $C(.)$, which is capable of giving rise to a fairly constant $(SNR)_{0,\,q}$ over a wide range of the input signal variation.

$$x \longrightarrow \boxed{C(.)} \xrightarrow{C(x)} \boxed{\text{U.Q}} \xrightarrow{QZ[C(x)]} \boxed{C^{-1}(.)} \longrightarrow y$$

Fig. 6.26: Non-uniform quantization through companding

Let the input signal x be bounded in the range $(-x_{\max}$ to $x_{\max})$. The characteristic $C(x)$ analytically defines non-uniform intervals Δ_k via uniform intervals of $\frac{2\,x_{\max}}{L}$ (Fig. 6.27). L is the number of quantization levels and is assumed to be large. Note that the uniform quantizer in Fig. 6.27 is the mid-riser variety. When $x \in I_6$, $C(x) \in I_6^{'}$ and the input to the UQ will be in the range $(\Delta, 2\Delta)$. Then the quantizer output is $\frac{3\Delta}{2}$. When $x \in I_k$, the compressor characteristic $C(x)$, may be approximated by a straight line segment with a slope equal to $\frac{2 x_{max}}{L \Delta_k}$, where Δ_k is the width of the interval I_k. That is,

$$C^{'}(x) = \frac{d C(x)}{d x} \simeq \frac{2 x_{max}}{L \Delta_k}, \; x \in I_k$$

As $C^{'}(x)$ is maximum at the origin, the equivalent step size is smallest at $x = 0$ and Δ_k is the largest at $x = x_{max}$. If L is large, the input PDF $f_X(x)$ can be treated to be approximately constant in any interval I_k, $k = 1, \cdots, L$. Note that we are assuming that the input is bounded in practice to a value x_{max}, even if the PDF should have a long tail. We also assume that $f_X(x)$ is symmetric; that is $f_X(x) = f_X(-x)$. Let $f_X(x) \simeq$ constant $= f_X(y_k)$, $x \in I_k$, where

$$y_k = \frac{x_{k+1} + x_k}{2}$$

and $\quad \Delta_k = length \; of \; I_k = x_{k+1} - x_k$

Then, $P_k = P[x \in I_k] = \int_{x_k}^{x_{k+1}} f_X(x) \, dx$, and, $\sum_{k=1}^{L} P_k = 1$

Let q_k denote the quantization error when $x \in I_k$.

That is, $q_k = QZ(x) - x, x \in I_k$

$$= y_k - x, x \in I_k$$

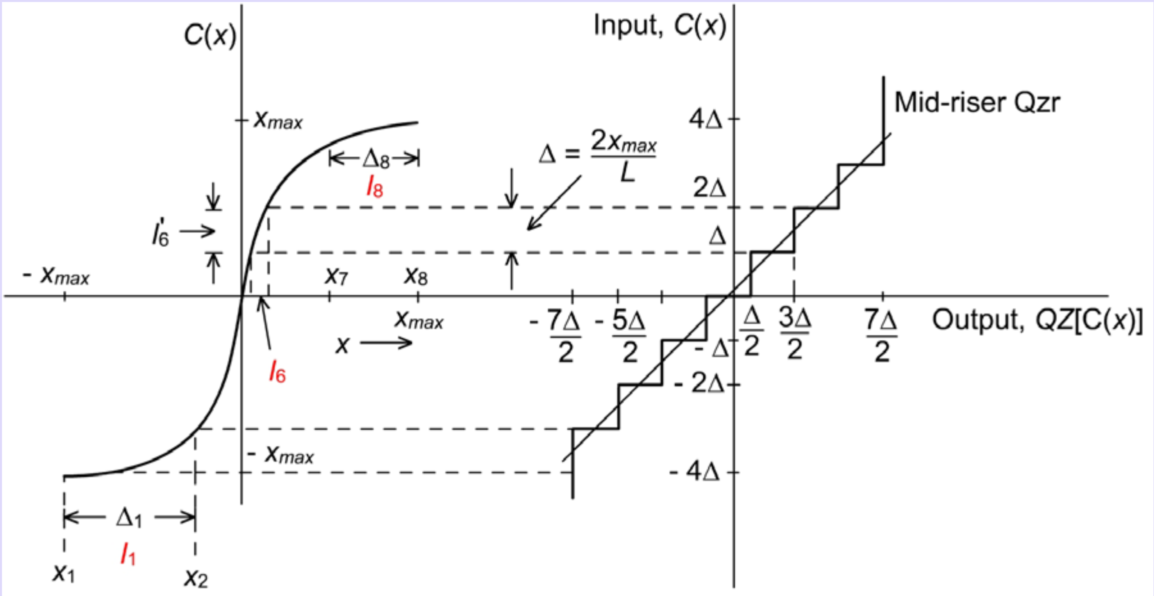


Fig. 6.27: A typical compression characteristic and non-uniform quantization levels

Let the corresponding random variable be Q_k. We can assume Q_k to be uniformly distributed in the range $\left(-\frac{\Delta_k}{2}, \frac{\Delta_k}{2}\right)$.

$$\overline{Q_k^2} = \frac{\Delta_k^2}{12}$$

σ_Q^2 = variance of the total quantization noise

$$= \sum_{k=1}^{L} P_k \overline{Q_k^2} = \sum_{k=1}^{L} P_k \frac{\Delta_k^2}{12}$$

Note that if $\Delta_k = \Delta$ for all k, then $\sigma_Q^2 = \frac{\Delta^2}{12}$, as expected.

But $\Delta_k \simeq \frac{2x_{max}}{L C'(x)}, \; x \in I_k$

Hence $\sigma_Q^2 \simeq \sum_{k=1}^{L} P_k \frac{4x_{max}^2}{L^2 \left[C'(x)\right]^2} \frac{1}{12}$

$$\simeq \frac{x_{max}^2}{3L^2} \sum_{k=1}^{L} P_k \left[C^{'}(x)\right]^{-2}$$

$$\simeq \frac{x_{max}^2}{3L^2} \sum_{k=1}^{L} \int_{x_k}^{x_{k+1}} f_X(x) \left[C^{'}(x)\right]^{-2} dx$$

$$\simeq \frac{x_{max}^2}{3L^2} \int_{-x_{max}}^{x_{max}} f_X(x) \left[C^{'}(x)\right]^{-2} dx \qquad (6.16a)$$

This approximate formula for σ_Q^2 is referred to as **Bennet's formula**.

$$(SNR)_{0,q} = \frac{\sigma_X^2}{\sigma_Q^2} \simeq \frac{3L^2}{x_{max}^2} \frac{\int_{-\infty}^{\infty} x^2 f_X(x)\, dx}{\int_{-x_{max}}^{x_{max}} f_X(x) \left[C^{'}(x)\right]^{-2} dx}$$

$$\simeq \frac{3L^2}{x_{max}^2} \frac{\int_{-x_{max}}^{x_{max}} x^2 f_X(x)\, dx}{\int_{-x_{max}}^{x_{max}} f_X(x) \left[C^{'}(x)\right]^{-2} dx} \qquad (6.16b)$$

From the above approximation, we see that it is possible for us to have a constant $(SNR)_{0,q}$ i.e. independent of σ_X^2, provided

$C^{'}(x) = \frac{1}{Kx}$ where K is a constant, or $\left(\left[C^{'}(x)\right]^{-2} = K^2 x^2\right)$

Then, $(SNR)_{0,q} = \frac{3L^2}{K^2 x_{max}^2}$

As $C^{'}(x) = \frac{1}{Kx}$, we have

$C(x) = \frac{\ln x}{K} + \alpha$, with $x > 0$ and α being a constant. (Note that $\ln x = \log_e x$.)

The constant α is arrived at by using the condition, when $x = x_{max}$, $C(x) = x_{max}$. Therefore, $\alpha = x_{max} - \frac{\ln x_{max}}{K}$.

Hence, $C(x) = \frac{\ln x}{K} + x_{max} - \frac{\ln x_{max}}{K}$

$$= \frac{1}{K} \ln\left(\frac{x}{x_{max}}\right) + x_{max}\,, \quad x > 0$$

As $C(-x) = -C(x)$ we have

$$C(x) = \left(\frac{1}{K} \ln\left(\frac{|x|}{x_{max}}\right) + x_{max}\right) \operatorname{sgn}(x), \tag{6.17}$$

where, $\operatorname{sgn}(x) = \begin{cases} 1, & x > 0 \\ -1, & x < 0 \end{cases}$

As the non-uniform quantizer with the above $C(x)$ results in a constant $(SNR)_{0,\,q}$, it is also called "robust quantization".

The above $C(x)$ is not realizable because $\lim_{x \to 0} C(x) \to \infty$. We want $\lim_{x \to 0} C(x) \to 0$. That is, we have to modify the theoretical companding law. We discuss two schemes, both being linear at the origin.

(i) A-law companding:

$$C(x) = \begin{cases} \dfrac{A\,|x|}{1 + \ln A} \operatorname{sgn}(x) & , \; 0 \le \dfrac{|x|}{x_{max}} \le \dfrac{1}{A} \\ x_{max} \left[\dfrac{1 + \ln\left(\dfrac{A|x|}{x_{max}}\right)}{1 + \ln A}\right] \operatorname{sgn}(x), & \dfrac{1}{A} \le \dfrac{|x|}{x_{max}} \le 1 \end{cases} \tag{6.18}$$

The constant A decides the compression characteristic, with $A = 1$ providing a linear I-O relationship. The behavior of $C(x)$ for $A = 2$ and 87.56 are shown in Fig. 6.28.

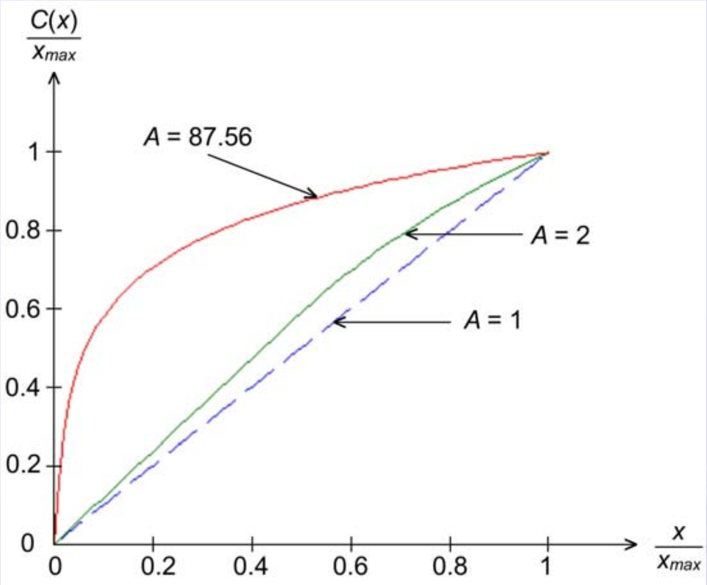


Fig. 6.28: A-law compression characteristic

$A = 87.56$ is the European commercial PCM standard which is being followed in India.

(ii) μ-law companding:

$$C(x) = x_{max}\left[\frac{\ln\left(1+\mu\frac{|x|}{x_{max}}\right)}{\ln(1+\mu)}\right]\operatorname{sgn}(x) \tag{6.19}$$

$C(x)$ is again quasi-logarithmic. For small values of x $(x > 0)$, such that $\mu x \ll x_{max}$, $C(x) \simeq \frac{\mu}{\ln(1+\mu)}x$, which indicates a linear relationship between the input x and output $C(x)$. Note that $C(-x) = -C(x)$. If x is such that $\mu|x| \gg x_{max}$, then $C(x)$ is logarithmic, given by

$$C(x) \simeq \frac{x_{max}}{\ln(1+\mu)}\ln\left(\frac{\mu|x|}{x_{max}}\right)\operatorname{sgn}(x)$$

μ is a constant, whose value decides the curvature of $C(x)$; $\mu = 0$ corresponds to no compression and as the value of μ increases, signal compression increases, as shown in Fig. 6.29.

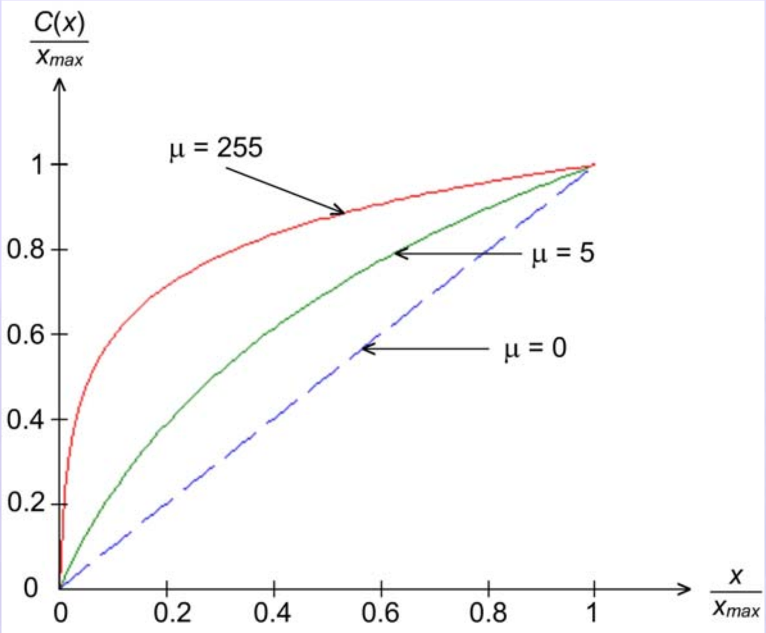


Fig. 6.29: μ-law compression characteristic

$\mu = 255$ is the north American standard for PCM voice telephony. Fig. 6.30 compares the performance of the 8-bit μ-law companded PCM with that of an 8 bit uniform quantizer; input is assumed to have the bounded Laplacian PDF which is typical of speech signals. The $(SNR)_{0,q}$ for the uniform quantizer is based on Eq. 6.14. As can be seen from this figure, companding results in a nearly constant signal-to-quantization noise ratio (within 3 dB of the maximum value of 38 dB) even when the mean square value of the input changes by about 30 dB (a factor of 10^3). G_c is the companding gain which is indicative of the improvement in SNR for small signals as compared to the uniform quantizer. In Fig. 6.30, G_c has been shown to be about 33 dB. This implies that the smallest step size, Δ_{min} is about 32 times smaller than the step size of a corresponding uniform quantizer with identical x_{max} and L, the number of quantization levels. Δ_{min} would be smaller by a factor more than 32, as a G_c of 30 dB would give rise to $\Delta_{min} = \frac{1}{32}.\ (\Delta_{uniform})$ The A - law characteristic, with $A = 87.56$ has a similar performance as that of $\mu = 255$ law, with a companding gain of about 24 dB; that is, for low amplitude signals, a uniform quantizer would require four additional bits to have the same $(SNR)_{0,q}$ performance of the $A = 87.56$ companded quantizer. (See Exercise 6.9)

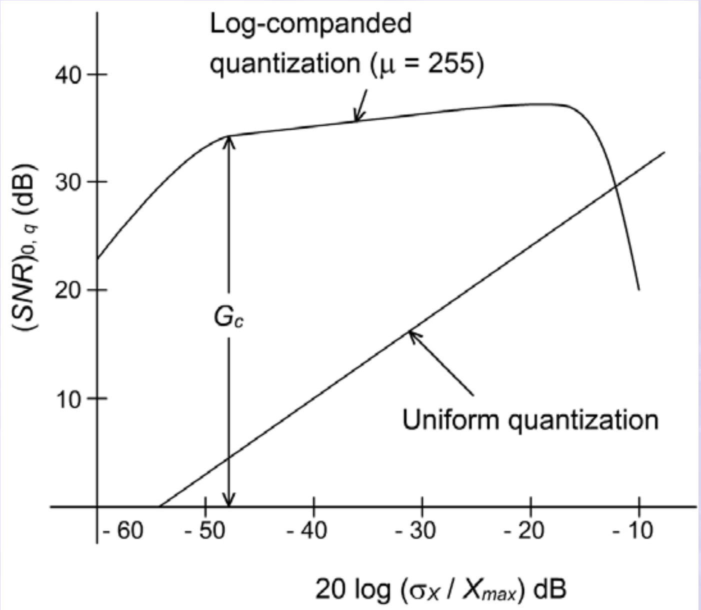


Fig. 6.30: $(SNR)_{0,q}$ performance of μ-law companded and uniform quantization

A number of companies manufacture CODEC chips with various specifications and in various configurations. Some of these companies are: Motorola (USA), National semiconductors (USA), OKI (Japan). These ICs include the anti-aliasing bandpass filter (300 Hz to 3.4 kHz), receive lowpass filter, pin selectable A-law / μ-law option. Listed below are some of the chip numbers manufactured by these companies.

Motorola: MC145500 to MC145505

National: TP3070, TP3071 and TP3070-X

OKI: MSM7578H / 7578V / 7579

Details about these ICs can be downloaded from their respective websites.

Example 6.10

A random variable X with the density function $f_X(x)$ shown in Fig. 6.31 is given as input to a non-uniform quantizer with the levels $\pm q_1$, $\pm q_2$ and $\pm q_3$.

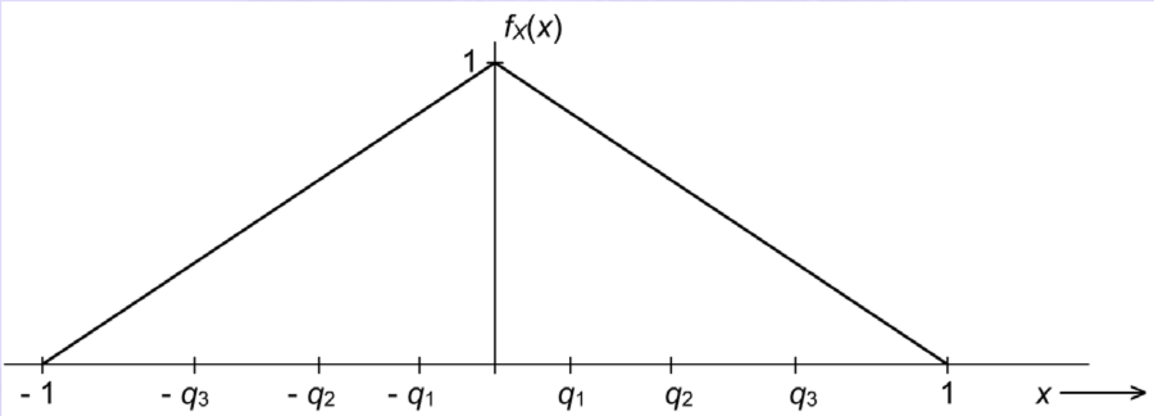


Fig. 6.31: Input PDF of Example 6.10

q_1, q_2 and q_3 are such that

$$\int_{-q_3}^{-q_2} f_X(x)\,dx = \int_{-q_2}^{-q_1} f_X(x)\,dx = \cdots = \int_{q_2}^{q_3} f_X(x)\,dx = \frac{1}{6} \tag{6.20}$$

a) Find q_1, q_2 and q_3.

b) Suggest a compressor characteristic that should precede a uniform quantizer such that Eq. 6.20 is satisfied. Is this unique?

a) $$\int_{-q_1}^{q_1} f_X(x)\,dx = 2\int_{0}^{q_1} f_X(x)\,dx = \frac{1}{6}$$

Hence, $$\int_{0}^{q_1} (1-x)\,dx = \frac{1}{12}$$

Solving, we obtain $q_1 = 0.087$

Similarly we have equations

$$\int_{0.087}^{q_2} f_X(x)\,dx = \frac{1}{6} \qquad (6.21a)$$

$$\int_{q_2}^{q_3} f_X(x)\,dx = \frac{1}{6} \qquad (6.21b)$$

From Eq. 6.21(a), we get $q_2 = 0.292$ and solving Eq. 6.21(b) for q_3 yields $q_3 = 0.591$.

b) A uniform quantizer will have levels at $\pm\frac{1}{6}$, $\pm\frac{1}{2}$ and $\pm\frac{5}{6}$.

Let $C(x)$ be the compressor characteristic which is a non-decreasing function of its argument and is anti-symmetrical about $x = 0$. Any compression characteristic that does the following mapping will take care of the requirements of the problem.

x	**$C(x)$**
0.0087	$\frac{1}{6}$
0.292	$\frac{1}{2}$
0.591	$\frac{5}{6}$
1	1

As such, $C(x)$ is not unique. ♦

Example 6.11

Consider a companded PCM scheme using the μ-law characteristic as given by

$$C(x) = \frac{\ln(1 + \mu|x|)}{\ln(1 + \mu)} \operatorname{sgn}(x), \ \mu > 0, \ |x| \le 1$$

Find the $(SNR)_{0,\,q}$ expected of this system when the input X is uniformly distributed in the range $(-1,\,1)$.

Using Bennet's formula, Eq. 6.16(a), let us compute σ_Q^2. From the $C(x)$ given, we have

$$\frac{d\,C(x)}{d\,x} = C^{'}(x) = \frac{\mu}{\ln(1+\mu)} \frac{1}{1 + \mu|x|}$$

$$\left[C^{'}(x)\right]^{-2} = \left[\frac{\ln(1+\mu)}{\mu}\right]^2 (1 + \mu|x|)^2$$

$$\sigma_Q^2 = \frac{2}{3L^2} \int_0^1 f_X(x) \left[C^{'}(x)\right]^{-2} d\,x$$

But $\quad f_X(x) = \begin{cases} \frac{1}{2}, & |x| \le 1 \\ 0, & \text{otherwise} \end{cases}$

Carrying out the integration,

$$\sigma_Q^2 = \frac{\left[\ln(1+\mu)\right]^2}{3L^2\mu^2} \left[1 + \frac{\mu^2}{3} + \mu\right]$$

$$\sigma_X^2 = \frac{1}{3}$$

$$(SNR)_{0,\,q} = \left(\frac{\mu L}{\ln(1+\mu)}\right)^2 \left(1 + \frac{\mu^2}{3} + \mu\right)^{-1}$$

If $\mu \ge 100$, then $\frac{\mu^2}{3} >> \mu$ and $\mu >> 1$, and $(SNR)_{0,\,q} \simeq \frac{3L^2}{\left[\ln(1+\mu)\right]^2}$ ♦

Example 6.12

Let $(SNR)_{0,\,q}$ of a μ-law companded PCM be approximated as

$$(SNR)_{0,\,q} \simeq \frac{3L^2}{\left[\ln(1+\mu)\right]^2}$$

We will show that $(SNR)_{0,\,q}$ (in dB) follows the $(6R + \alpha)$ rule.

Let $L = 2^R$. Then

$$(SNR)_{0,\,q} \simeq \frac{3}{\left[\ln(1+\mu)\right]^2} \cdot \left(2^R\right)^2$$

$$\simeq C\,4^R, \text{ where } C = \frac{3}{\left[\ln(1+\mu)\right]^2}$$

Then $(SNR)_{0,\,q}\,(\text{in dB}) = 10\log_{10} C + 6R$

$$= \alpha + 6R$$ ♦

Example 6.13

A music signal band-limited to 15 kHz is sampled at the rate of 45×10^3 samples/sec and is sent using 8 bit μ-law (μ = 255) companded PCM. $(SNR)_{0,\,q}$ of this system was found be inadequate by atleast 10 dB. If the sampling rate is now reduced to 35×10^3 samples/sec, let us find the expected improvement in $(SNR)_{0,\,q}$, when the bit rate is not to exceed the previous case.

With $f_s = 45 \times 10^3$ samples/sec, and assuming 8 bits/samples, we have the transmitted bit rate as $45 \times 8 \times 10^3 = 36 \times 10^4$ bits/sec.

With $(f_s) = 35 \times 10^3$, number of bits/sample that can be used

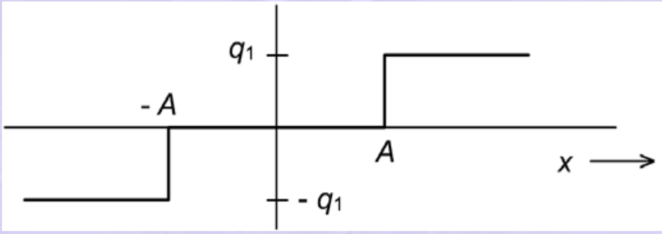
$$R = \frac{36 \times 10^4}{35 \times 10^3} = 10.28$$

As R has to be an integer, it can be taken as 10. As R has increased by two bits, we can expect 12 dB improvement in SNR. ♦

Exercise 6.6

Consider the three level quantizer (mid-tread type) shown in Fig. 6.32. The input to the quantizer is a random variable X with the PDF $f_X(x)$ give by

$$f_X(x) = \begin{cases} \frac{1}{4}, & |x| \le 1 \\ \frac{1}{8}, & 1 < |x| \le 3 \end{cases}$$



Fig, 6.32: Quantizer characteristic for the Exercise 6.5

a) Find the value of A such that all the quantizer output levels are equiprobable. (Note that A has to be less than 1, because with $A = 1$, we have $P\left[QZ(x) = 0\right] = \frac{1}{2}$)

b) Show that the variance of the quantization noise for the range $|x| \le A$, is $\frac{4}{81}$.

Exercise 6.7

Let the input to μ-law compander be the sample function $x_j(t)$ of a random process $X(t) = \cos(2\pi f_m t + \Theta)$ where Θ is uniformly distributed in the range $(0, 2\pi)$. Find the $(SNR)_{0,q}$ expected of the scheme.

Note that samples of $x_j(t)$ will have the PDF,

$$f_X(x) = \begin{cases} \dfrac{1}{\pi\sqrt{1-x^2}} , & |x| \le 1 \\ 0 , & \textit{otherwise} \end{cases}$$

Answer: $(SNR)_{0,q} = \dfrac{3}{2}\left(\dfrac{\mu L}{\ln(1+\mu)}\right)^2 \left[1 + \dfrac{\mu^2}{2} + \dfrac{4\mu}{\pi}\right]^{-1}$

Note that if $\dfrac{\mu^2}{2} >> \mu >> 1$, we have $(SNR)_{0,q} \simeq \dfrac{3L^2}{\left[\ln(1+\mu)\right]^2}$

Exercise 6.8

Show that for values of x such that $A|x| >> x_{max}$, $(SNR)_{0,q}$ of the A-law PCM is given by

$$(SNR)_{0,q} = 6R + \alpha$$

where $\alpha = 4.77 - 20\log_{10}[1 + \ln A]$

Exercise 6.9

Let $C(x)$ denote the compression characteristic. Then $\left.\dfrac{dC(x)}{dx}\right|_{x \to 0}$ is called the companding gain, G_c. Show that

a) G_c (A-law) with $A = 87.56$ is 15.71 (and hence $20\log_{10} G_c \simeq 24$ dB)

b) G_c (μ-law) with $\mu = 255$ is 46.02 (and hence $20\log_{10} G_c \simeq 33$ dB.)

Exercise 6.10

Show that for the μ-law,

$$\frac{\text{maximum step size}}{\text{minimum step size}} = \frac{\Delta_{max}}{\Delta_{min}} = \frac{\lim\limits_{x \to 0} C'(x)}{\lim\limits_{x \to x_{max}} C'(x)} = (\mu + 1).$$

6.5 Encoding

The encoding operation converts the quantized samples into a form that is more convenient for the purpose of transmission. It is a one-to-one representation of the quantized samples by using *code elements* or *symbols* of the required length per sample.

In the *binary code*, each symbol may be either of two distinct values or kinds, which are customarily denoted as 0 and 1. In a *ternary* code, each symbol may be one of three distinct values; similarly for the other codes. By far, the most popular from the point of view of implementation are the binary codes. With R-binary digits (bits) per sample, we can have 2^R distinct code words and we require $2^R \geq$ (number of quantization levels), so that it is possible for us to maintain a one-to-one relationship between the code words and the quantization levels.

Let us identify the R-bit sequence as $b_R \; b_{R-1} \cdots b_3 \; b_2 \; b_1$. In the natural binary code, this sequence represents a number (or level) N, where

$$N = b_R \left(2^{R-1}\right) + b_{R-1}\left(2^{R-2}\right) + \cdots + b_2 \, 2^1 + b_1 \, 2^0 \qquad (6.22)$$

Natural binary code results when the codeword symbols or digits are assigned to N, with N listed in an increasing or decreasing (decimal) order; that is, though the quantized samples could be either positive or negative, we simply label the quantized levels decimally without regard to the polarity of the samples. The first column in Table 6.1 illustrates a decimal assignment for a 16-level quantization in which half the levels are for positive samples (say 8 to 15) and the remaining half for negative samples. Column 2 shows the corresponding natural binary code.

Table 6.1: Various binary code words for a 4-bit Encoder

1	2				3				4				5				
Decimal Level No.	Natural Binary code				Folded Binary code				Inverted folded binary				Gray code				Comments
	b_4	b_3	b_2	b_1									g_4	g_3	g_2	g_1	
15	1	1	1	1	1	1	1	1	1	0	0	0	1	0	0	0	Levels assigned to positive message samples
14	1	1	1	0	1	1	1	0	1	0	0	1	1	0	0	1	
13	1	1	0	1	1	1	0	1	1	0	1	0	1	0	1	1	
12	1	1	0	0	1	1	0	0	1	0	1	1	1	0	1	0	
11	1	0	1	1	1	0	1	1	1	1	0	0	1	1	1	0	
10	1	0	1	0	1	0	1	0	1	1	0	1	1	1	1	1	
9	1	0	0	1	1	0	0	1	1	1	1	0	1	1	0	1	
8	1	0	0	0	1	0	0	0	1	1	1	1	1	1	0	0	
7	0	1	1	1	0	0	0	0	0	1	1	1	0	1	0	0	Levels assigned to negative message samples
6	0	1	1	0	0	0	0	1	0	1	1	0	0	1	0	1	
5	0	1	0	1	0	0	1	0	0	1	0	1	0	1	1	1	
4	0	1	0	0	0	0	1	1	0	1	0	0	0	1	1	0	
3	0	0	1	1	0	1	0	0	0	0	1	1	0	0	1	0	
2	0	0	1	0	0	1	0	1	0	0	1	0	0	0	1	1	
1	0	0	0	1	0	1	1	0	0	0	0	1	0	0	0	1	
0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	

The other codes shown in the table are derived from the natural binary code. The *folded binary code* (also called the *sign-magnitude* representation) assigns the first (left most) digit to the sign and the remaining digits are used to code the magnitude as shown in the third column of the table. This code is superior to the natural code in masking transmission errors when encoding speech. If only the amplitude digits of a folded binary code are complemented (1's changed to 0's and 0's to 1's), an i*nverted folded binary code* results; this code has the advantage of higher density of 1's for small amplitude signals, which are most probable for voice messages. (The higher density of 1's relieves some system timing errors but does lead to some increase in cross talk in multiplexed systems).

With natural binary encoding, a number of codeword digits can change even when a change of only one quantization level occurs. For example, with reference to Table 6.1, a change from level 7 to 8 entails every bit changing in the 4-bit code illustrated. In some applications, this behavior is undesirable and a code is desired for which only one digit changes when any transition occurs between adjacent levels. The Gray Code has this property, if we consider the extreme levels as adjacent. The digits of the *Gray code*, denoted by g_k, can be derived from those of the natural binary code by

$$g_k = \begin{cases} b_R & , \ k = R \\ b_{k+1} \oplus b_k , & k < R \end{cases}$$

where $\oplus$ denotes modulo-2 addition of binary digits. $(0 \oplus 0 = 0; 0 \oplus 1 = 1 \oplus 0 = 1$ and $1 \oplus 1 = 0$; note that, if we exclude the sign bit, the remaining three bits are mirror images with respect to red line in the table.)

The reverse behavior of the Gray code does not hold. That is, a change in anyone of code digits does not necessarily result in a change of only one code level. For example, a change in digit g_4 from 0 when the code is 0001 (level 1) to

1 will result in 1001 (the code word for level 14), a change spanning almost the full quantizer range.

6.6 Electrical Waveform Representation of Binary Sequences

For the purposes of transmission, the symbols 0 and 1 should be converted to electrical waveforms. A number of waveform representations have been developed and are being currently used, each representation having its own specific applications. We shall now briefly describe a few of these representations that are considered to be the most basic and are being widely used. This representation is also called as **line coding** or **transmission coding**. The resulting waveforms are called line codes or transmission codes for the reason that they are used for *transmission* on a telephone *line.*

In the **Unipolar** format (also known as On-Off signaling), symbol 1 represented by transmitting a pulse, where as symbol 0 is represented by switching off the pulse. When the pulse occupies the full duration of a symbol, the unipolar format is said to be of the *nonreturn-to-zero* (NRZ) type. When it occupies only a fraction (usually one-half) of the symbol duration, it is said to be of the *return-to-zero* (RZ) type. The unipolar format contains a DC component that is often undesirable.

In the *polar* format, a positive pulse is transmitted for symbol 1 and a negative pulse for symbol 0. It can be of the NRZ or RZ type. Unlike the unipolar waveform, a polar waveform has no dc component, provided that 0's and 1's in the input data occur in equal proportion.

In the *bipolar* format (also known as *pseudo ternary* signaling or Alternate Mark Inversion, **AMI**), positive and negative pulses are used alternatively for the transmission of 1's (with the alternation taking place at every occurrence of a 1) and no pulses for the transmission of 0's. Again it can be of the NRZ or RZ type. Note that in this representation there are *three* levels: +1, 0, -1. An attractive feature of the bipolar format is the absence of a DC component, even if the input binary data contains large strings of 1's or 0's. The absence of DC permits transformer coupling during the course of long distance transmission. Also, the bipolar format eliminates ambiguity that may arise because of *polarity inversion* during the course of transmission. Because of these features, bipolar format is used in the commercial PCM telephony. (Note that, some authors use the word bipolar to mean polar)

In the *Manchester format* (also known as *biphase* or *split phase signaling*), symbol 1 is represented by transmitting a positive pulse for one-half of the symbol duration, followed by a negative pulse for the remaining half of the symbol duration; for symbol 0, these two pulses are transmitted in the reverse order. Clearly, this format has no DC component; moreover, it has a built in synchronization capability because there is a predictable transition during each bit interval. The disadvantage of the *Manchester format* is that it requires twice the bandwidth when compared to the NRZ unipolar, polar and bipolar formats. Fig. 6.33 illustrates some of the waveform formats described above. Duration of each bit has been taken as T_b sec and the levels as 0 or $\pm a$.

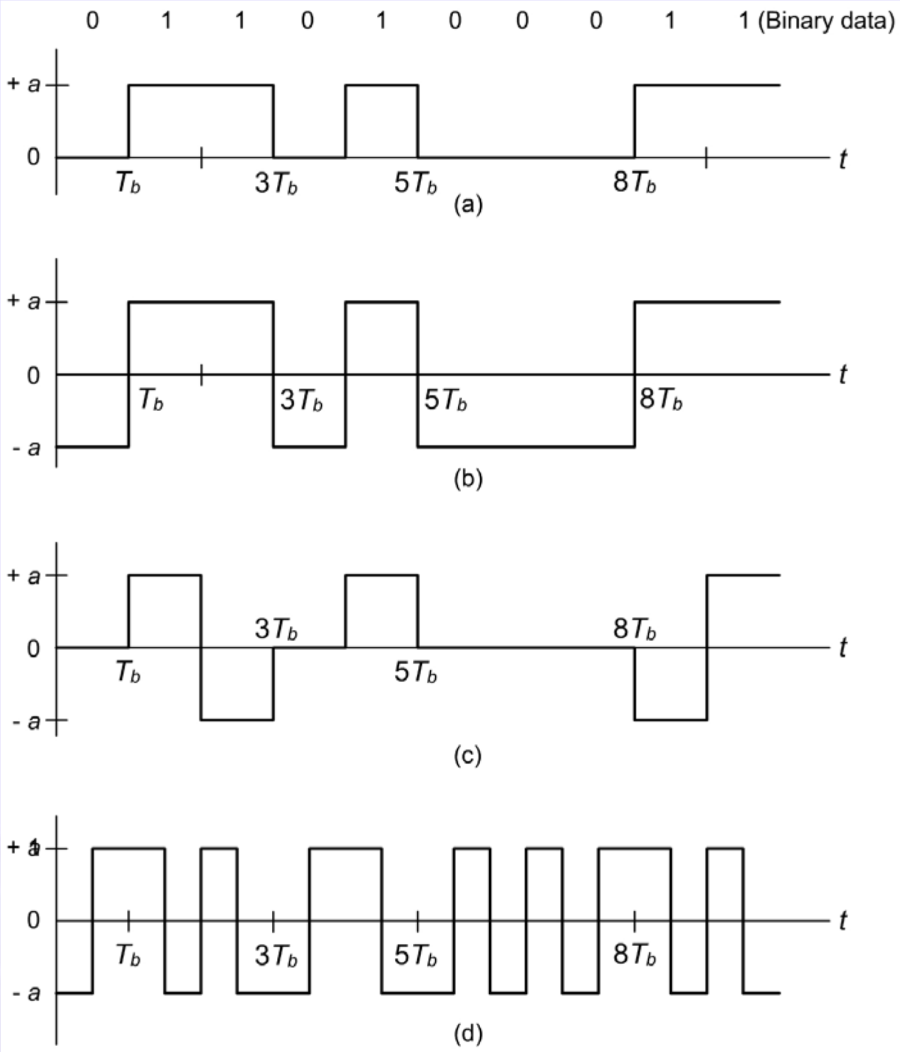


Fig. 6.33: Binary data waveform formats

NRZ: (a) on-off (b) polar (c) bipolar

(d) Manchester format

6.7 Bandwidth requirements of PCM

In Appendix A6.1, it has been shown that if

$$X(t) = \sum_{k = -\infty}^{\infty} A_k \, p(t - kT - T_d),$$

then, the power spectral density of the process is,

$$S_X(f) = \frac{1}{T} |P(f)|^2 \sum_{n = -\infty}^{\infty} R_A(n) e^{j 2 \pi n f T}$$

By using this result, let us estimate the bandwidth required to transmit the waveforms using any one of the signaling formats discussed in Sec. 6.6

6.7.1 Unipolar Format

In this case, A_k 's represent an on-off sequence. Let us assume that 0's and 1's of the binary sequence to be transmitted are equally likely and '0' is represented by level 0 and '1' is represented by level 'a'. Then,

$$P(A_k = 0) = P(A_k = a) = \frac{1}{2}$$

Let us compute $R_A(n) = E\left[A_k A_{k+n}\right]$. For $n = 0$, we have $R_A(0) = E\left[A_k^2\right]$. That is,

$$R_A(0) = (0)^2 P(A_k = 0) + (a)^2 P(A_k = a)$$

$$= \frac{a^2}{2}$$

Next consider the product $A_k A_{k+n}$, $n \neq 0$. This product has four possible values, namely, 0, 0, 0 and a^2. Assuming that successive symbols in the binary sequence are statistically independent, these four values occur with equal probability resulting in,

$$E\left[A_k A_{k+n}\right] = 0 \cdot \left(\frac{3}{4}\right) + a^2 \left(\frac{1}{4}\right)$$

$$= \frac{a^2}{4}, \; n \neq 0$$

Hence,

$$R_A(n) = \begin{cases} \dfrac{a^2}{2}, & n = 0 \\ \dfrac{a^2}{4}, & n \neq 0 \end{cases}$$

Let $p(t)$ be rectangular pulse of unit amplitude and duration T_b sec. Then, $P(f)$ is $\quad P(f) = T_b \operatorname{sinc}(f T_b)$

and $S_X(f)$ can be written as

$$S_X(f) = \left(\frac{a^2 T_b}{4}\right) \operatorname{sinc}^2(f T_b) + \frac{a^2 T_b}{4} \operatorname{sinc}^2(f T_b) \sum_{n = -\infty}^{\infty} \exp[j 2 \pi n f T_b]$$

But from Poisson's formula,

$$\sum_{n = -\infty}^{\infty} \exp[j 2 \pi n f T_b] = \frac{1}{T_b} \sum_{m = -\infty}^{\infty} \delta\left(f - \frac{m}{T_b}\right)$$

Noting that $\operatorname{sinc}(f T_b)$ has nulls at $f = \frac{n}{T_b}$, $n = \pm 1, \pm 2, \cdots$, $S_X(f)$ can be simplified to

$$S_X(f) = \frac{a^2 T_b}{4} \operatorname{sinc}^2(f T_b) + \frac{a^2}{4} \delta(f) \qquad (6.23a)$$

If the duration of $p(t)$ is less than T_b, then we have **unipolar RZ sequence**. If $p(t)$ is of duration $\frac{T_b}{2}$ seconds, then $S_X(f)$ reduces to

$$S_X(f) = \frac{a^2 T_b}{16} \operatorname{sinc}^2\left(\frac{f T_b}{2}\right)\left[1 + \frac{1}{T_b} \sum_{m = -\infty}^{\infty} \delta\left(f - \frac{m}{T_b}\right)\right] \qquad (6.23b)$$

From equation 6.23(b) it follows that unipolar RZ signaling has discrete spectral components at $f = 0, \pm \frac{1}{T_b}, \frac{3}{T_b}$ etc. A plot of Eq. 6.23(b) is shown in Fig. 6.34(a).

6.7.2 Polar Format

Assuming, $P[A_k = + a] = P[A_k = - a] = \frac{1}{2}$ and 0's and 1's of the binary data sequence are statistically independent; it is easy to show,

$$R_A(n) = \begin{cases} a^2 , & n = 0 \\ 0 , & n \neq 0 \end{cases}$$

With $p(t)$ being a rectangular pulse of duration T_b sec. (NRZ case), we have

$$S_X(f) = a^2 T_b \, \text{sinc}^2(f T_b) \tag{6.24a}$$

The PSD for the case when $p(t)$ is duration of $\frac{T_b}{2}$ (RZ case), is given by

$$S_X(f) = a^2 \frac{T_b}{4} \, \text{sinc}^2\left(\frac{f T_b}{2}\right) \tag{6.24b}$$

6.7.3 Bipolar Format

Bipolar format has three levels: $a, 0$ and $(-a)$. Assuming that 1's and 0's are equally likely, we have $P[A_k = a] = P[A_k = -a] = \frac{1}{4}$ and $P[A_k = 0] = \frac{1}{2}$. We shall now compute the autocorrelation function of a bipolar sequence.

$$R_A(0) = \frac{1}{4}(a \cdot a) + \frac{1}{4}(-a)(-a) + \frac{1}{2}(0 \cdot 0)$$

$$= \frac{a^2}{2}$$

To compute $R_A(1)$, we have to consider the four two bit sequences, namely, 00, 01, 10, 11. As the binary ‘0’ is represented by zero volts, we have only one non-zero product, corresponding to the binary sequence (11). As each one of these two bit sequences occur with a probability $\frac{1}{4}$, we have

$$R_A(1) = -\frac{a^2}{4}$$

To compute $R_A(n)$, for $n \geq 2$, again we have to take into account only those binary n-tuples which have ‘1’ in the first and last position, which will result in the product $\pm a^2$. It is not difficult to see that, in these product terms, there are as many terms with $-a^2$ as there are with a^2 which implies that the sum of the product terms would be zero. (For example, if we take the binary 3-tuples, only

101 and 111 will result in a non-zero product quantity. 101 will yield $-a^2$ whereas 111 will result in a^2). Therefore $R_A(n) = 0$ for $n \geq 2$. That is, for the bipolar format,

$$R_A(n) = \begin{cases} \dfrac{a^2}{2}, & n = 0 \\ -\dfrac{a^2}{4}, & n = \pm 1 \\ 0, & otherwise \end{cases}$$

For the NRZ case, with a rectangular $p(t)$, we obtain

$$S_X(f) = a^2 T_b \operatorname{sinc}^2(f T_b) \sin^2(\pi f T_b) \tag{6.25a}$$

Similarly for the return-to-zero case, we have

$$S_X(f) = \frac{a^2 T_b}{4} \operatorname{sinc}^2\left(\frac{f T_b}{2}\right) \sin^2(\pi f T_b) \tag{6.25b}$$

6.7.4 Manchester Format

If the input binary data consists of independent equally likely symbols, $R_A(n)$ for the Manchester format is the same as that of the polar format. The pulse $p(t)$ for the Manchester format is a doublet of unit amplitude and duration T_b. Hence,

$$P(f) = j T_b \operatorname{sinc}\left(\frac{f T_b}{2}\right) \sin\left(\frac{\pi f T_b}{2}\right)$$

and $$S_X(f) = a^2 T_b \operatorname{sinc}^2\left(\frac{f T_b}{2}\right) \sin^2\left(\frac{\pi f T_b}{2}\right) \tag{6.26}$$

Plots of Eq. 6.24(b), 6.25(b) and 6.26 are shown in Fig. 6.34(b).

From Fig. 6.34(a), we see that most of the power of the unipolar return- to-zero signal is in a bandwidth of $\frac{2}{T_b}$ (of course, for the NRZ case, it would be $\frac{1}{T_b}$).

Similarly, the bandwidth requirements of the Manchester format and return-to-zero polar format can be taken as $\frac{2}{T_b}$; the spectral width of the return-to-zero bipolar format is essentially $\frac{1}{T_b}$.

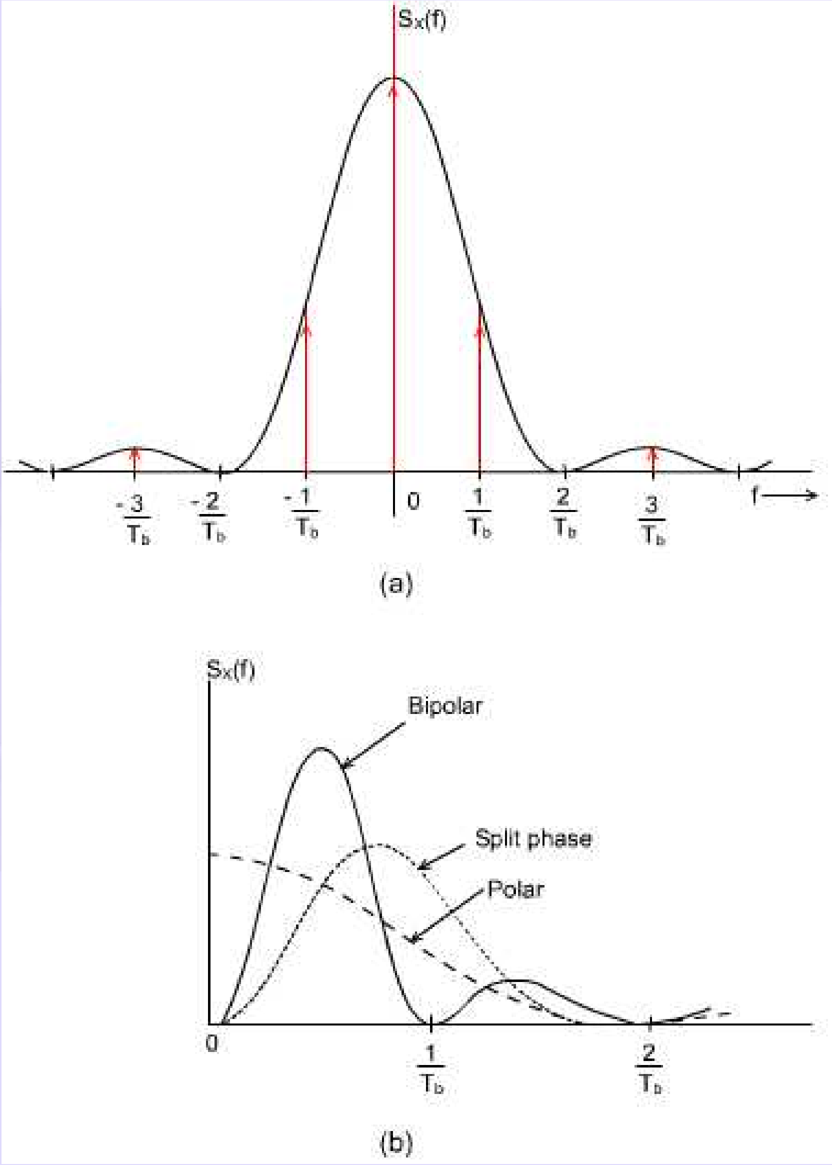


Fig. 6.34: Power spectral density of some signaling schemes
(a) RZ unipolar (b) polar (RZ), bipolar (RZ) and split phase (Manchester) formats

Bipolar signaling has several advantages:

1) The spectrum has a D.C. null; hence it is possible to transformer-couple such a signal
2) Its transmission bandwidth is not excessive
3) It has single error detection capability because an isolated error, whether it causes the deletion or erection of a pulse will violate the pulse alternation property of the bipolar format
4) It is insensitive to polarity inversion during the course of transmission
5) If a return-to-zero bipolar pulse is rectified, we get an on-off signal that has a discrete component at the clock frequency. This feature can be used to regenerate the clock at the receiving end. (See Example 6.15.)

For these reasons, bipolar signaling is used in the PCM voice telephony.

Now let us look at the bandwidth requirement of a PCM system. Consider the case of a speech signal. In the commercial PCM telephony, this signal is sampled at 8 kHz and each sample is converted to an 8 bit sequence which implies a bit rate of 64 kbps. Assuming bipolar signaling, we require a bandwidth of 64 kHz which is an order of magnitude larger than analog baseband transmission whose bandwidth requirement is just about 4 kHz!

Example 6.14

A PCM voice communication system uses bipolar return-to-zero pulses for transmission. The signal to be transmitted has a bandwidth of 3.5 kHz with the peak-to-peak and RMS values of 4 V and 0.2 V respectively. If the channel bandwidth is limited to 50 kHz, let us find the maximum $(SNR)_{0,\,q}$ (in dB) that can be expected out of this system.

Assuming that the bandwidth requirements of return-to-zero bipolar pulse as $\frac{1}{T_b}$, it would be possible for us to send upto 50,000 pulses per second on this channel. As the bit rate is the product of sampling rate and the number of bits/sample, we choose the minimum sampling rate that is permitted so that the number of bits/sample can be as high as possible. (This would result in the maximization of $(SNR)_{0,\,q}$). Minimum sampling rate = 7000 samples/sec. Hence, we could choose a 7-bit quantizer resulting in 2^7 quantization levels.)

$$\text{Step size } \Delta = \frac{4}{2^7} = 2^{-5}$$

$$\text{Noise variance, } \sigma_Q^2 = \left(\frac{\Delta^2}{12}\right) = \frac{2^{-10}}{12}$$

$$(SNR)_{0,\,q} = \frac{\sigma_x^2}{\sigma_Q^2} = \frac{48 \times 2^{10}}{100} = 48 \times 2^8$$

$$(SNR)_{0,\,q}\,(\text{in dB}) = 26.8$$ ♦

Example 6.15

In a digital communication system - such as a PCM system - bit clock recovery at the receiver is very important. Bipolar pulses enable us to recover the clock without too much difficulty.

Consider the scheme shown in Fig. 6.35.

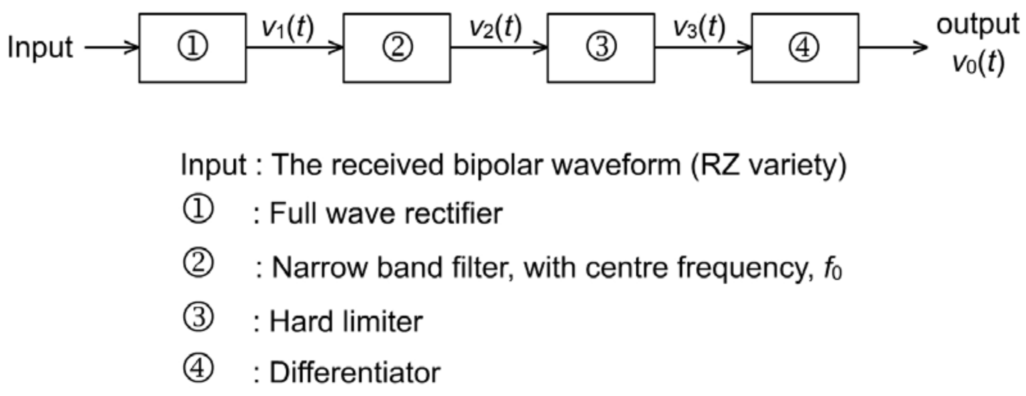


Fig. 6.35: Clock recovery scheme for bipolar signals

The bit rate of the bipolar waveform is 10^5 bps. Let us find the output of the system when

(a) $f_0 = 100$ kHz (b) $f_0 = 200$ kHz

a) $v_1(t)$, output of the full wave rectifier, is a RZ, on-off waveform. From Eq. 6.23(b), we know that this waveform process has a discrete spectral component at $f = \frac{1}{T_b}$ which is 100 kHz in this case. This component will be selected by the tuned filter; hence $v_2(t)$ is a sinusoidal signal with frequency 100 kHz. The output of the hard limiter, $v_3(t)$, is a square wave at 100 kHz. Differentiation of this result in a sequence of positive and negative going impulses with two positive (or negative) impulses being separated by 10^{-5} sec. This can be used as the clock for the receiver.

b) When the centre frequency of the tuned filter is 200 kHz, the output $v_2(t)$ is zero as there is no discrete component at 200 kHz in $v_1(t)$. ♦

Exercise 6.11

Let a random process $X(t)$ be given by

$$X(t) = \sum_{k = -\infty}^{\infty} B_k \, p(t - kT_b - T_d)$$

It is given that $B_k = A_k + A_{k-1}$ where A_k's are discrete random variables with $\overline{A_k \, A_{k \pm n}} = R_A(n)$. $A_k = a$ if the input binary symbol is '1' and $A_k = -a$ if the input symbol is '0'. 1's and 0's are equally likely and are statistically independent. T_b is the bit duration and T_d is a random delay with a uniform distribution in the range $\left(-\frac{T_b}{2}, \frac{T_b}{2}\right)$. A_k's are independent of T_d.

Show that

a) $$\overline{B_k \, B_{k \pm n}} = R_B(n) = \begin{cases} 2a^2, & n = 0 \\ a^2, & n = \pm 1 \\ 0, & otherwise \end{cases}$$

b) $S_X(f) = a^2 \, T_b \, \text{sinc}^2\left(\frac{f\,T_b}{2}\right) \cos^2(\pi f T_b)$ when $p(t)$ is a unit amplitude rectangular pulse of duration $\left(\frac{T_b}{2}\right)$ seconds.

6.8 Differential Pulse Code Modulation (DPCM)

DPCM can be treated as a variation of PCM; it also involves the three basic steps of PCM, namely, sampling, quantization and coding. But, in the case of DPCM, what is quantized is the difference between the actual sample and its *predicted* value, as explained below.

Let $x(t)$ represent the analog signal that is to be DPCM coded, and let it be sampled with a period T_s. The sampling frequency $f_s = \frac{1}{T_s}$ is such that there is no aliasing in the sampling process. Let $x(n) = x(t)\big|_{t = nT_s}$. As T_s is a fixed quantity, which is known apriori, it is the integer n that carries the time information of the signal $x(t)$. Quite a few real world signals such as speech signals, biomedical signals (ECG, EEG, etc.), telemetry signals (temperature inside a space craft, atmospheric pressure, etc.) do exhibit sample-to-sample correlation. This implies that $x(n)$ and $x(n+1)$ (or $x(n)$ and $x(n-1)$) do not differ significantly. In fact, given a set of previous M samples, say $x(n-1)$, $x(n-2), \cdots, x(n-M)$, it may be possible for us to predict (or estimate) $x(n)$ to within a small percentage error. Let $\hat{x}(n)$ denote the predicted value of $x(n)$ and let

$$e(n) = x(n) - \hat{x}(n) \tag{6.27}$$

In DPCM, it is the error sequence $e(n)$ that is quantized, coded and sent on the channel. At the receiver, samples are reconstructed using the quantized version of the error sequence. If there are no transmission errors, the reconstructed sample sequence $y(n)$ will follow the sequence $x(n)$ to within the limits of quantization error.

From the point of view of calculating the $(SNR)_{0,q}$, we can treat $e(n)$ to be a specific value of the random variable $\mathrm{E}(n)$. Similarly, we will assume that $x(n)$ and $\hat{x}(n)$ represent specific values of the random variable $X(n)$ and $\widehat{X}(n)$. Hence

$$\mathrm{E}(n) = X(n) - \widehat{X}(n) \tag{6.28}$$

The rationale behind quantizing the sequence $\mathrm{E}(n)$ is the following: if the assumption that $X(n)$ sequence is correlated is true in practice, then the variance of $\mathrm{E}(n)$ would be less that $X(n)$. As such, for a given number of quantization levels, the quantization step size Δ_ε to discretize $\mathrm{E}(n)$ would be smaller than Δ_X, the step size required to quantize $X(n)$. This implies that $\frac{\Delta_\varepsilon^2}{12} < \frac{\Delta_X^2}{12}$ and there would be net improvement in the overall signal-to-quantization noise ratio. In other words, for a given signal-to-quantization ratio, DPCM requires fewer bits/input sample than what is required by PCM.

We shall now look at a few prediction schemes and show that the variance of the error sequence, σ_ε^2, is less than σ_X^2, the variance of the sequence $X(n)$.

Case i) $\widehat{X}(n)$ is a scaled version of $X(n-1)$.

Let $\widehat{X}(n) = \alpha_1 X(n-1)$, (6.29)

where α_1 is a constant. Then,

$$\mathrm{E}(n) = X(n) - \widehat{X}(n)$$

$$= X(n) - \alpha_1 X(n-1)$$

$$E\left[\mathrm{E}^2(n)\right] = \sigma_\varepsilon^2 = E\left\{\left[X(n) - \alpha_1 X(n-1)\right]^2\right\}$$

$$\sigma_\varepsilon^2 = E\left[X^2(n) + \alpha_1^2 X^2(n-1) - 2\alpha_1 X(n)\, X(n-1)\right]$$

But, $X(n) = X(t)\big|_{t = nT}$ and if $X(t)$ is zero mean, WSS process, then

$E\left[X^2(n)\right] = E\left[X^2(n-1)\right] = \sigma_X^2$. Therefore,

$$\sigma_\varepsilon^2 = \sigma_X^2\left\{1 + \alpha_1^2 - 2\alpha_1 \frac{E\left[X(n)X(n-1)\right]}{\sigma_X^2}\right\}$$

Let ρ_1 denote the adjacent sample correlation coefficient; that is,

$$\rho_1 = \frac{E\left[X(n)\,X(n-1)\right]}{\sigma_X^2}.$$

Hence, $\sigma_\varepsilon^2 = \sigma_X^2\left[1 + \alpha_1^2 - 2\alpha_1\rho_1\right]$ (6.30)

Let us find the value of α_1 that minimizes σ_ε^2. Differentiating Eq. 6.30 with respect to α_1 and setting the result to zero, we will end up with the result,

$$\alpha_1 = \rho_1 \tag{6.31a}$$

Using this value in Eq. 6.30, gives us the result

$$\sigma_\varepsilon^2 = \sigma_X^2\left(1 - \rho_1^2\right) \tag{6.31b}$$

If ρ_1 is 0.8, then $\sigma_\varepsilon^2 = 0.36\,\sigma_X^2$, which is about a third of the input signal variance.

Case ii) Let us generalize the predictor of Eq. 6.29. Let

$$\begin{aligned}\widehat{X}(n) &= \alpha_1 X(n-1) + \alpha_2 X(n-2) + \cdots + \alpha_M X(n-M)\\ &= \sum_{i=1}^{M} \alpha_i\, X(n-i)\end{aligned} \tag{6.32}$$

where α_1 to α_M are constants.

Eq. 6.32 expresses $\widehat{X}(n)$ in terms weighted linear combination of past M input samples and represents the I-O relation of a **linear predictor of order M**. α_1 to α_M are called Linear Predictor Coefficients (LPCs).

We would like to choose $\alpha_1, \alpha_2, \cdots, \alpha_M$ such that

$E\left[\mathrm{E}^2(n)\right] = E\left\{\left[X(n) - \sum_{i=1}^{M} \alpha_i \, X(n-i)\right]^2\right\}$ is minimized. To obtain the optimum $\alpha = (\alpha_1, \alpha_2, \cdots, \alpha_M)$, let us introduce some notation. Let $E\left[X_{n-i} \, X_{n-j}\right]$ be denoted by $r_x(j-i)$ and let

$$R_X = \begin{bmatrix} r_x(0) & r_x(1) & \cdots & r_x(M-1) \\ r_x(1) & r_x(0) & \cdots & r_x(M-2) \\ . & & & \\ . & & & \\ . & & & \\ r_x(M-1) & r_x(M-2) & \cdots & r_x(0) \end{bmatrix}$$

$$\boldsymbol{t}_X = \begin{bmatrix} r_x(1) \\ r_x(2) \\ . \\ . \\ . \\ r_x(M) \end{bmatrix}$$

Then, it has been shown that (see Haykin [2], Jayant and Noll [3]), the optimum predictor vector

$(\alpha)_{opt}^T = (\alpha_1, \alpha_2, \cdots, \alpha_M)^T$, where the superscript T denotes the transpose, is given by

$$(\alpha)_{opt}^T = R_X^{-1} \boldsymbol{t}_X \tag{6.33}$$

and $$\left(\sigma_\varepsilon^2\right)_{\min} = r_X(0) - \sum_{i=1}^{M} \alpha_i \, r_X(i) \tag{6.34a}$$

$$= \sigma_X^2 \left[1 - \sum_{i=1}^{M} \alpha_i \, \rho_i\right] \tag{6.34b}$$

where $$\rho_j = \frac{r_X(j)}{r_X(0)} \tag{6.34c}$$

Let us look at some special cases of Eq. 6.33 and 6.34.

i) First order predictor $(M = 1)$

Then, α is the scalar α_1 and

$$\alpha_1 = \frac{r_x(1)}{r_x(0)} = \rho_1$$

and $$\left(\sigma_\varepsilon^2\right)_{min} = r_x(0) - \alpha_1 r_x(1)$$

$$= r_x(0)\left(1 - \rho_1 \frac{r_x(1)}{r_x(0)}\right)$$

$$= \sigma_x^2\left(1 - \rho_1^2\right)$$

which are the same as given by Eq. 6.31.

ii) Second order predictor $(M = 2)$

$$\alpha_{opt}^T = \begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix}$$ is obtained from

$$\begin{bmatrix} r_x(0) & r_x(1) \\ r_x(1) & r_x(0) \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} = \begin{bmatrix} r_x(1) \\ r_x(2) \end{bmatrix}$$

Solving for α_1 and α_2, we obtain

$$\alpha_1 = \frac{\rho_1 - \rho_1 \rho_2}{1 - \rho_1^2} \tag{6.35a}$$

$$\alpha_2 = \frac{\rho_2 - \rho_1^2}{1 - \rho_1^2} \tag{6.35b}$$

where $\rho_2 = \dfrac{r_x(2)}{r_x(0)}$

$\left(\sigma_\varepsilon^2\right)_{min}$ is obtained by using 6.35(a) and 6.35(b) in Eq. 6.34.

The block diagram of a DPCM transmitter is shown in Fig. 6.36(a) and the corresponding receiver is shown in Fig. 6.36(b).

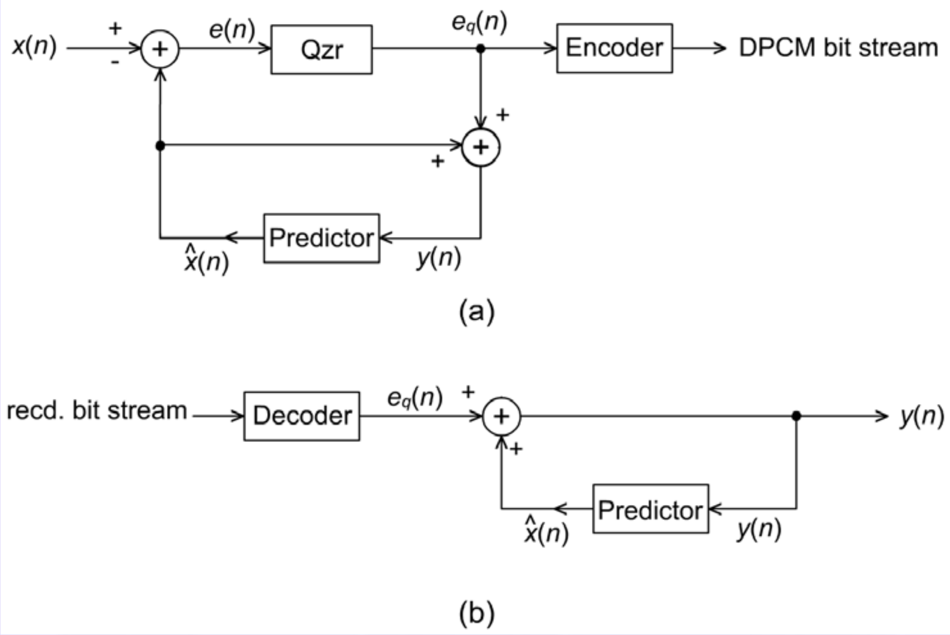


Fig. 6.36: (a) DPCM transmitter (b) DPCM receiver

In the scheme of Fig. 6.36(a), $y(n)$, the input to the predictor, is a quantized version of $x(n)$. That is,

$$\hat{x}(n) = \sum_{i=1}^{M} \alpha_i \, x_q(n-i) \tag{6.36}$$

This is essential to enable the receiver to track the predictor in the transmitter, as will be made clear later on in this section. Let us denote the quantizer operation by

$$\begin{aligned} e_q(n) &= QZ\left[e(n)\right] \\ &= e(n) + q(n) \end{aligned} \tag{6.37}$$

where $q(n)$ is the quantization error. Then,

$$\begin{aligned} y(n) &= \hat{x}(n) + e_q(n) \\ &= \hat{x}(n) + e(n) + q(n) \\ &= x(n) + q(n) = x_q(n) \end{aligned} \tag{6.38}$$

That is, the predictor input differs from the input sequence $x(n)$ by the quantization error, which can be assumed to be fairly small (much smaller than $x(n)$ most of the time). In such a situation, predictor output $\hat{x}(n)$ can be taken as being predicted from the input sequence $x(n)$ itself. If the predictor is good, then $\hat{x}(n) \simeq x(n)$. As such, σ_{ε}^2, the variance of the error quantity $E(n)$ should be much less than σ_x^2. Therefore, the quantizer with a given number of levels will produce much less quantization error, when compared to quantization of $x(n)$ directly.

Let us assume that there are no transmission errors on the channel. Then the decoder output in the receiver $e_q(n)$ is the same as one in the transmitter. Hence the sequence $y(n)$ at the receiver will follow the sequence $x(n)$ within the limits of the quantizer error.

Let σ_Q^2 denote the variance of the quantization error. Then,

$$\left[(SNR)_{0,\,q}\right]_{DPCM} = \frac{\sigma_x^2}{\sigma_Q^2}$$

$$= \frac{\sigma_x^2}{\sigma_{\varepsilon}^2} \cdot \frac{\sigma_{\varepsilon}^2}{\sigma_Q^2}, \text{ where } \sigma_{\varepsilon}^2 = \overline{E^2(n)}$$

$$= G_p\,(SNR)_p \qquad (6.39a)$$

$G_p = \dfrac{\sigma_x^2}{\sigma_{\varepsilon}^2}$, is called the *prediction gain* and $(SNR)_p = \dfrac{\sigma_{\varepsilon}^2}{\sigma_Q^2}$

From Eq. 6.39(a), we find that the signal-to-quantization noise ratio of the DPCM scheme depends on two things: the performance of the predictor (indicated by G_p) and how well the quantizer performs in discretizing the error sequence $e(n)$, indicated by $(SNR)_p$. A value of G_p greater than unity represents the gain in SNR that is due to the prediction operation of the DPCM scheme.

$$G_p = \frac{\sigma_x^2}{\sigma_\varepsilon^2} = \frac{\sigma_X^2}{\sigma_X^2 \left[1 - \sum_{i=1}^{M} \alpha_i \rho_i\right]}$$

$$= \frac{1}{1 - \sum_{i=1}^{M} \alpha_i \rho_i} \qquad (6.39b)$$

where α_i's are the optimum predictor coefficients. Jayant and Noll [3], after a great deal of empirical studies, have reported that for voice signals G_p can be of the order of 5-10 dB. In the case of TV video signals, which have much higher correlation than voice, it is possible to achieve a prediction gain of about 12 dB. With a 12 dB prediction gain, it is possible to reduce the number of bits/ sample by 2.

Example 6.16

The designer of a DPCM system finds by experimental means that a third order predictor gives the required prediction gain. Let α_{pi} denote the i^{th} coefficient of an optimum p^{th} order predictor. Given that $\alpha_{11} = -\frac{2}{3}$, $\alpha_{22} = \frac{1}{5}$ and $\alpha_{33} = \frac{1}{4}$, find the prediction gain given by the third order predictor. you can assume $r_x(0) = 1$.

We have $\alpha_{11} = \frac{r_x(1)}{r_x(0)} = \rho_1 = -\frac{2}{3}$

$\alpha_{22} = \frac{\rho_2 - \rho_1^2}{1 - \rho_1^2}$ (See Eq. 6.35(b))

$$\frac{1}{5} = \frac{\rho_2 - \left(-\frac{2}{3}\right)^2}{1 - \left(-\frac{2}{3}\right)^2}$$

Solving for ρ_2, we have $\rho_2 = \frac{5}{9} = r_X(2)$. We can obtain α_{31} and α_{32} from the following equation.

$$\begin{bmatrix} r_x(0) & r_x(1) & r_x(2) \\ r_x(1) & r_x(0) & r_x(1) \end{bmatrix} \begin{bmatrix} \alpha_{31} \\ \alpha_{32} \\ \alpha_{33} \end{bmatrix} = \begin{bmatrix} r_X(1) \\ r_X(2) \end{bmatrix}$$

That is,

$$\begin{bmatrix} 1 & -\frac{2}{3} & \frac{5}{9} \\ -\frac{2}{3} & 1 & -\frac{2}{3} \end{bmatrix} \begin{bmatrix} \alpha_{31} \\ \alpha_{32} \\ \alpha_{33} \end{bmatrix} = \begin{bmatrix} -\frac{2}{3} \\ \frac{5}{9} \end{bmatrix}$$

Solving for α_{31} and α_{32}, we obtain $\alpha_{31} = -\frac{7}{12}$ and $\alpha_{32} = \frac{1}{3}$

From Eq. 6.39(b),

$$G_p = \frac{1}{1 - \sum_{i=1}^{3} \alpha_{3i}\, \rho_i}$$

To calculate G_p, we require $\rho_3 = r_X(3)$. This can be obtained from the equation

$\rho_2 \alpha_{31} + \rho_1 \alpha_{32} + \alpha_{33} = \rho_3$. That is,

$$\rho_3 = \left(\frac{5}{9}\right)\left(-\frac{7}{12}\right) + \left(-\frac{2}{3}\right)\frac{1}{3} + \frac{1}{4}$$

$$= -\frac{8}{27}$$

Hence, $G_p = 2$. ♦

DPCM scheme can be made *adaptive* (ADPCM); that is, the quantizer and the predictor are made to adapt to the short term statistics of their respective inputs. Evidently, we can expect better performance from a ADPCM than a scheme that is fixed. 32 kbps (8 kHz sampling, 4 bits/sample coding) ADPCM coders for speech have been developed whose performance is quite comparable with that of 64 kbps companded PCM. Also 64 kbps DPCM has been developed for encoding audio signals with a bandwidth of 7 kHz. This coder uses a sampling rate of 16 kHz and a quantizer with 16 levels. Details can be found in Benevuto *et al* [4] and Decina and Modena [5].

At the start of the section, we had mentioned that DPCM is a special case of PCM. If we want, we could treat PCM to be a special case of DPCM where the predictor output is always zero!

6.9 Delta Modulation

Delta modulation, like DPCM is a predictive waveform coding technique and can be considered as a special case of DPCM. It uses the simplest possible quantizer, namely a two level (one bit) quantizer. The price paid for achieving the simplicity of the quantizer is the increased sampling rate (much higher than the Nyquist rate) and the possibility of slope-overload distortion in the waveform reconstruction, as explained in greater detail later on in this section.

In DM, the analog signal is highly over-sampled in order to increase the adjacent sample correlation. The implication of this is that there is very little change in two adjacent samples, thereby enabling us to use a simple one bit quantizer, which like in DPCM, acts on the difference (prediction error) signals.

In its original form, the DM coder approximates an input time function by a series of linear segments of constant slope. Such a coder is therefore referred to as a Linear (or non-adaptive) Delta Modulator (LDM). Subsequent developments have resulted in delta modulators where the slope of the approximating function is a variable. Such coders are generally classified under *Adaptive Delta Modulation (ADM)* schemes. We use DM to indicate either of the linear or adaptive variety.

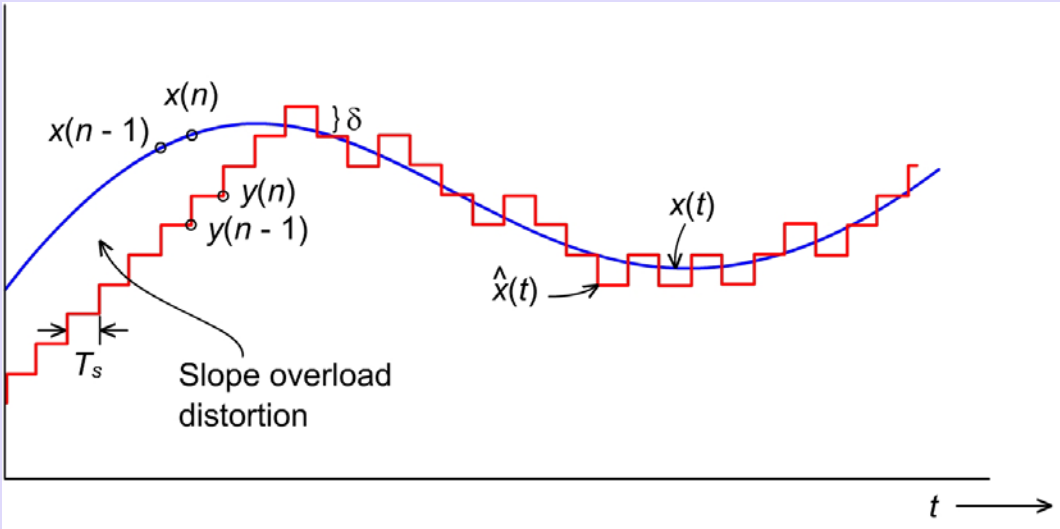


Fig. 6.37: Waveforms illustrative of LDM operation

6.9.1 Linear Delta Modulation (LDM)

The principle of operation of an LDM system can be explained with the help of Fig. 6.37. The signal $x(t)$, band-limited to W Hz is sampled at the rate $f_s >> 2W$. $x(n)$ denotes the sample of $x(t)$ at $t = nT_s$. The staircase approximation to $x(t)$, denoted by $\hat{x}(t)$, is arrived as follows. One notes, at $t = nT_s$, the polarity of the difference between $x(n)$ and the latest staircase approximation to it; that is, $\hat{x}(t)$ at $t = nT_s$ which is indicated by $y(n-1)$ in the figure. $y(n-1)$ is incremented by a step of size δ if $\left[x(n) - y(n-1)\right]$ is positive or decremented by δ, if this is of negative polarity. Mathematically, let

$$e(n) = x(n) - \hat{x}(n) \tag{6.40a}$$

$$= x(n) - y(n-1) \tag{6.40b}$$

and $\quad b(n) = \delta\, \text{sgn}\left[e(n)\right]$ (6.40c)

Then, $y(n) = y(n-1) + b(n)$ (6.41)

For each signal sample, the transmitted channel symbol is the single bit $b(n)$ and the bit rate of the coder is f_s.

Figure 6.38 shows the (discrete-time) implementation of an LDM system as implied by Eq. 6.41.

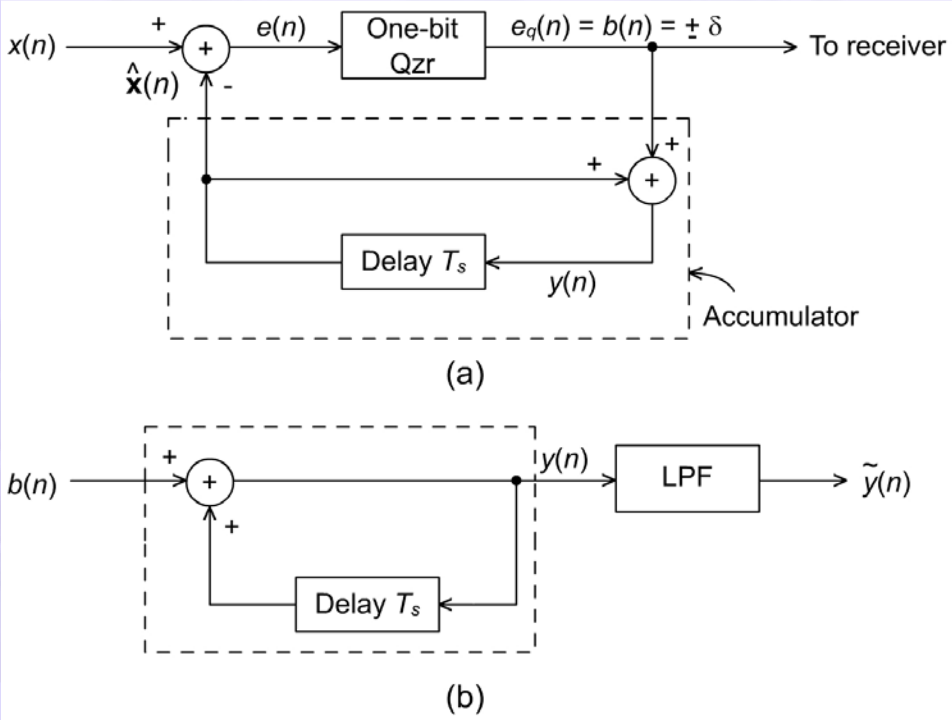


Fig. 6.38: Discrete-time LDM system (a) Transmitter (b) Receiver

In the transmitter we have the delay element acting as the first order predictor so that $\hat{x}(n) = y(n-1)$. The boxed portion of the transmitter is generally referred to as the 'accumulator'. Assuming that the initial contents of the accumulator are zero, it can easily be shown that $y(n) = \sum_{i=1}^{n} b(i)$, justifying the use of the term accumulator for the boxed portion. At each sampling instant, the accumulator increments the approximation to the input signal by $\pm\,\delta$, depending on the output of the quantizer. (Note that the quantizer step size is $\Delta = 2\,\delta$). In the receiver, the sequence $y(n)$ is reconstructed by passing the incoming sequence of positive and negative pulses through an accumulator in a manner similar to that used in the transmitter. The sequence $y(n)$ is passed through a discrete time LPF resulting in $\tilde{y}(n)$. This lowpass filtering operation is quite essential in the case of DM because it improves the $(SNR)_{0,\,q}$ of a DM scheme as explained later.

Figure 6.39 shows a single integration DM coder-decoder system in an 'analog implementation' notation that is very common in the DM literature. Note that the coder input is $x(t)$, a function of the continuous parameter t, and that the sampling clock appears after the two level quantizer. The outputs corresponding to $\hat{x}(n)$ and $y(n)$ of Fig. 6.38 have not been indicated in Fig. 6.39(a). These quantities $\hat{x}(n)$ and $y(n)$, are realized at the output of the integrator just *before* and just *after* each clock operation. The time-continuous notation in Fig. 6.39 is very relevant for implementation, though for analytical reasons, the use of discrete time notation may be preferred.

Figure 6.39 also depicts the waveforms involved at various stages of the LDM coder. These illustrations are self explanatory.

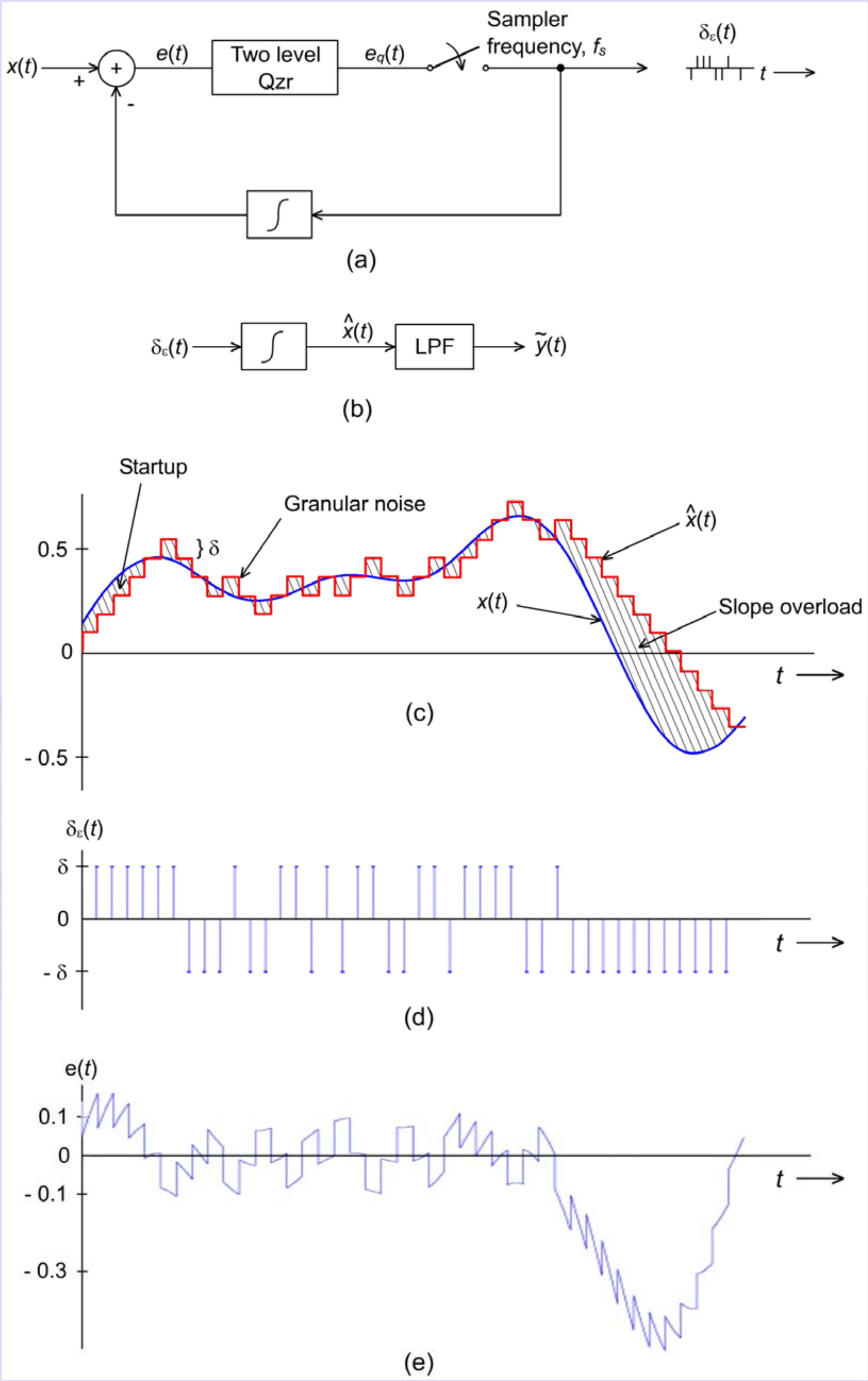


Fig. 6.39: LDM system (analog) with representative waveforms

Figures 6.37 and Fig. 6.39 indicate two types of quantization errors in DM: slope overload distortion (noise) and granular noise. Slope-overload is said to occur when the step size is too small to follow a steep segment of the input waveform $x(t)$. Granularity, on the other hand, refers to a situation where the staircase function $\hat{x}(t)$ hunts around a relatively flat segment of the input function, with a step size that is too large relative to the local slope characteristic of the input. It is therefore clear that for a given statistics of the input signal slope relatively small values of δ accentuate slope-overload while relatively large values of δ increase granularity.

Slope overload distortion is a basic drawback of the LDM system. This distortion can be avoided provided $f_s \delta \geq \left| x'(t) \right|_{max}$ where $x'(t)$ refers to the slope of the input waveform. This is because $\hat{x}(t)$ (or $y(t)$) changes by $\pm \delta$ every $T_s = \frac{1}{f_s}$ seconds; hence the maximum LDM slope is $f_s \delta$ and if this quantity is greater than or equal to $\left| x'(t) \right|_{max}$ for all t, then LDM will be able to track the slope changes of $x(t)$. Consider the case of tone signal, $x(t) = A_m \cos(2 \pi f_m t)$. Then

$$x'(t) = -2 \pi f_m A_m \sin(2 \pi f_m t),$$

and $$\left| x'(t) \right|_{max} = 2 \pi f_m A_m = \omega_m A_m$$

Hence to avoid slope overload, we require

$$f_s \delta \geq \left| x'(t) \right|_{max} = 2 \pi f_m A_m \tag{6.42a}$$

or $$\frac{A_m}{\delta} \leq \frac{1}{2 \pi} \left(\frac{f_s}{f_m} \right) \tag{6.42b}$$

The RHS of Eq. 6.42(b) may be viewed as a kind of spectral characteristic for a basic delta modulator, since it is a function of f_m. Taken in this light, we expect that (single integration) LDM is quite suitable to waveforms that have PSDs that decrease as the reciprocal of the square of the frequency. Examples of such signals are speech and monochrome TV video.

Example 6.17

Find the minimum sampling frequency $(f_s)_{min}$, to avoid slope overload when $x(t) = \cos(2\pi\, 800 t)$ and $\delta = 0.1$.

To avoid slope-overload, we require (Eq. 10.42(a)),

$$f_s \geq 2\pi f_m \left(\frac{A_m}{\delta}\right)$$

Hence, for $f_m = 800$ Hz and $\frac{A_m}{\delta} = 10$,

$$(f_s)_{min} = 2\pi \times 800 \times 10$$
$$\simeq 50 \text{ kHz}$$

Note that this sampling rate is 10π times the Nyquist rate of 1.6 kHz. ♦

Example 6.18

Let a message signal $m(t)$ be the input to a delta modulator where $m(t) = 6 \sin\left[\left(2\pi \times 10^3\right)t\right] + 4 \sin\left[\left(4\pi \times 10^3\right)t\right]$ V, with t in seconds. Determine the minimum pulse rate that will prevent slope overload, if the step size is 0.314 V.

$$m(t) = 6 \sin\left[\left(2\pi \times 10^3\right)t\right] + 4 \sin\left[\left(4\pi \times 10^3\right)t\right]$$

$$\frac{d\,m(t)}{d\,t} = 12\pi \times 10^3 \cos\left[\left(2\pi \times 10^3\right)t\right] + 16\,\pi \times 10^3 \cos\left[\left(4\pi \times 10^3\right)t\right]$$

$\frac{d\,m(t)}{d\,t} = m^{'}(t)$ is maximum at $t = 0$.

$$\left(m^{'}(t)\right)_{\max} = 28\pi \times 10^3$$

To avoid slope overload, we require

$$f_s\,\delta \geq 28\pi \times 10^3$$

$$f_s \geq \frac{28\pi \times 10^4 \times 0.314}{0.314} = 280 \times 10^3$$

Pulse rate $= 280 \times 10^3$ /sec. ♦

Exercise 6.12

The input to a linear delta modulator is a sinusoidal signal whose frequency can vary from 200 Hz to 4000 Hz. The input is sampled at eight times the Nyquist rate. The peak amplitude of the sinusoidal signal is 1 Volt.

a) Determine the value of the step size in order to avoid slope overload when the input signal frequency is 800 Hz.

b) What is the peak amplitude of the input signal to just overload the modulator, when the input signal frequency is 200 Hz.

c) Is the modulator overloaded when the input signal frequency is 4 kHz.

Ans: (a) $\frac{\pi}{40}$ (b) 4 V (c) Yes

The performance of the DM system depends on the granular noise, slope-overload noise (together they are termed as quantization noise in the context of DM) and regeneration errors (that is, errors due to channel noise). In the analysis that follows, we shall derive an expression for the output SNR of a DM system, neglecting the slope-overload-noise and regeneration errors.

For negligible slope-overload, we require $|e(t)| = |x(t) - \hat{x}(t)| \leq \delta$. Let $\mathrm{E}(t)$ denote the random process of which $e(t)$ is a sample function. We will assume that $\mathrm{E}(t)$ is zero mean WSS process and the samples of $E(t)$ are uniformly distributed in the range $(-\delta, \delta)$. Then σ_ε^2, the variance of $\mathrm{E}(t)$ is, $\sigma_\varepsilon^2 = \dfrac{\delta^2}{3}$. Experimental results confirm that the PSD of $\mathrm{E}(t)$ is essentially flat for $|f| \leq \dfrac{1}{T_s} = f_s$. That is,

$$S_\varepsilon(f) = \frac{\sigma_\varepsilon^2}{2 f_s} = \frac{\delta^2}{6 f_s}, \text{ for } |f| \leq f_s$$

Lowpass filtering the output of the accumulator in the receiver rejects the out-of-band noise components (remember that in DM, $f_s >> 2W$), giving the output granular noise component, N_g, as

$$N_g = \int_{-W}^{W} S_\varepsilon(f)\, d f$$

$$= \frac{W}{f_s} \frac{\delta^2}{3} \tag{6.43a}$$

and $$(SNR)_{0,\, g} = \frac{3 f_s}{\delta^2 W} \sigma_X^2 \tag{6.43b}$$

where $\sigma_X^2 = \overline{X^2(t)}$

Recall that in order to avoid slope-overload, δ and f_s must satisfy Eq. 6.42.

Exercise 6.13

The input to an LDM scheme is the sinusoidal signal $A_m \cos\left[\left(2\pi f_m\right) t\right]$. The scheme uses a sampling frequency of f_s and the step size δ is such that it is the minimum value required to avoid slope overload. Show that the $(SNR)_{0,\,g}$ expected of such a system, at the output of the final lowpass filter with cutoff at f_m, is

$$(SNR)_{0,\,g} = \frac{3}{8\pi^2}\left(\frac{f_s}{f_m}\right)^3$$

Now, if the PSD of $X(t)$ is $S_X(f)$, then the PSD of $\frac{d\,X(t)}{d\,t}$ is $(2\pi f)^2 S_X(f)$.

Hence the *mean square signal slope* can be put in the *form*

$$\overline{\left[\frac{d\,X(t)}{d\,t}\right]^2} = \int_{-\infty}^{\infty} (2\pi f)^2 S_X(f)\,df$$

$$= \left(2\pi\sigma_X B_{rms}\right)^2$$

where B_{rms}, the RMS bandwidth of the lowpass process $X(t)$, is defined as

$$B_{rms} = \frac{1}{\sigma_x}\left[\int_{-\infty}^{\infty} f^2 S_X(f)\,df\right]^{\frac{1}{2}} \qquad (6.44)$$

We now introduce the so called *slope loading factor*

$$\xi = \frac{f_s\,\delta}{2\pi\sigma_X B_{rms}} \qquad (6.45)$$

which is the ratio of maximum DM slope to the RMS signal slope. For any given signal $x(t)$, the parameters σ_x and B_{rms} are fixed. Hence, given the sampling frequency, ξ is a function of only δ. A reasonably large value of ξ ensures negligible slope-overload. Expressing δ in terms of ξ and using this value in Eq. 6.43(b), we obtain

$$(SNR)_{0,\,g} = \frac{3}{4\pi^2}\,\frac{f_s^3}{\xi^2 B_{rms}^2 W}$$

$$= \frac{6}{\pi^2}\left(\frac{W}{B_{rms}}\right)^2 \frac{F^3}{\xi^2} \qquad (6.46)$$

where $F = \dfrac{f_s}{2W}$ is the *over-sampling ratio*.

Equation 6.46 indicates that for a given value of ξ, the SNR in LDM with single integration increases as the cube of the bit rate (9 dB/octave). The bit rate, in turn, decides the bandwidth requirements of the system.

Computer simulations indicate that Eq. 6.46 is also valid for $(SNR)_{0,\,q}$ provided $\ln 2F \leq \xi < 8$. If $\xi < \ln 2F$, then slope-overload noise dominates (δ too small) and SNR drops off quite rapidly. Figure 6.40 illustrates the typical variation of $(SNR)_{0,\,q}$ with ξ.

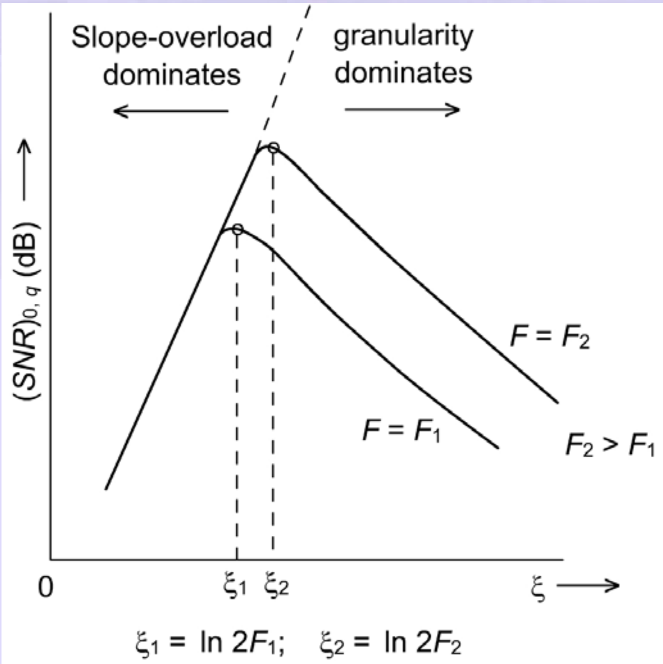


Fig. 6.40: LDM performance versus slope loading factor

For a specified value of F, DM performance is maximized by taking the empirically determined optimum slope loading factor,

$$\xi_{opt} \simeq \ln 2F$$

The maximum value of $(SNR)_{0,\,g} \simeq (SNR)_{0,\,q}$ is then given by Eq. 6.46 with $\xi = \xi_{opt}$. Note that as the over sampling ratio is fixed for each curve, ξ is dependent only on δ. Hence, when $\xi < \xi_{opt}$, δ is not large enough to prevent slope overload. Similarly, as the value of ξ increases away from ξ_{opt}, the size of δ increases which implies an increase in the granular noise level.

Example 10.20

A signal $x(t)$ with $\sigma_X^2 = 1$ has the spectral density $S_X(f)$ given by

$$S_X(f) = \begin{cases} \dfrac{1}{500}\ , & |f| \le 125 \text{ Hz} \\ \dfrac{1}{1000}\ , & 125 < |f| < 375 \text{ Hz} \end{cases}$$

$x(t)$ is over sampled by a factor of 10 and the samples are applied to DM coder. Find δ_{opt}, the optimum value of δ.

We will take $\xi_{opt} = \ln 20 = 3$. From Eq. 6.45,

$$\delta_{opt} = \frac{2\pi B_{rms}\,\xi_{opt}}{f_s}.$$

We need to compute B_{rms}.

$$B_{rms}^2 = 2\left[\int_0^{125} f^2 \cdot \frac{1}{500}\,df + \int_{125}^{375} f^2 \cdot \frac{1}{1000}\,df\right]$$

$$= \frac{1}{250}\left[\frac{f^3}{3}\right]_0^{125} + \frac{1}{500}\left[\frac{f^3}{3}\right]_{125}^{275}$$

$$= \frac{1}{750}\left[1953125\right] + \frac{1}{1500}\left[50781250\right]$$

$$= 7812.5 + 33854.16 = 41666.66$$

$$B_{rms} = 204.12$$

$$\delta_{opt} = \frac{2\pi \times 3 \times 204.12}{7500}$$

$$= 0.512$$

6.9.2 Adaptive delta modulation

Consider Fig. 6.40. We see that for a given value of F, there is an optimum value of ξ which gives rise to maximum $(SNR)_{0,\,q}$ from a DM coder. Let us now change ξ from ξ_{opt} by varying σ_X with δ fixed. We then find that the range of σ_X for which near about this maximum $(SNR)_{0,\,q}$ is possible is very limited. (This situation is analogous to the uncompanded PCM). Adaptive Delta Modulation, a modification of LDM, is a scheme that circumvents this deficiency of LDM.

The principle of ADM is illustrated in Fig. 6.41. Here the step size Δ of the quantizer is not a constant but varies with time. We shall denote by $\Delta(n) = 2\,\delta(n)$, the step size at $t = nT_s \cdot \delta(n)$ increases during a steep segment of the input and decreases when the modulator is quantizing a slowly varying segment of x(t). A specific step-size adaptation scheme is discussed in detail later on in this section.

The adaptive step size control which forms the basis of an ADM scheme can be classified in various ways such as: *discrete* or *continuous*; *instantaneous* or *syllabic* (fairly gradual change); *forward* or *backward*. Here, we shall describe an adaptation scheme that is backward, instantaneous and discrete. For extensive coverage on ADM, the reader is referred to Jayant and Noll [3].

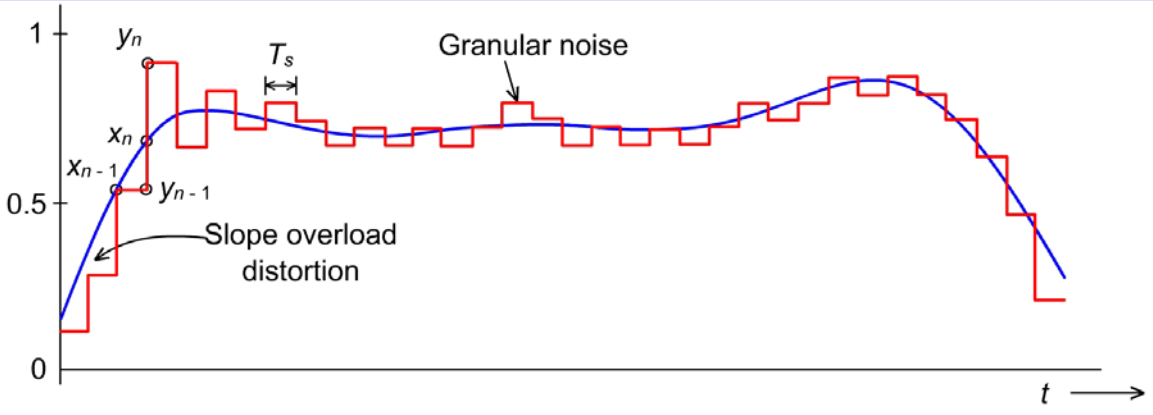


Fig. 6.41: Waveforms illustrative of ADM operation

Figure 6.42 gives the block diagram of an ADM scheme. In practical implementation, the step size $\delta(n)$ is constrained to be between some predetermined minimum and maximum values. In particular, we write

$$\delta_{min} \leq \delta(n) \leq \delta_{max} \tag{6.47}$$

The upper limit, δ_{max}, controls the amount of slopeoverload distortion. The lower limit, δ_{min}, controls the granular noise. Within these limits, the adaptive rule for $\delta(n)$ can be expressed in the general form

$$\delta(n) = g(n)\,\delta(n-1) \tag{6.48}$$

where the time varying gain $g(n)$ depends on the present binary output $b(n)$ and M previous values, $b(n-1), \cdots, b(n-M)$. The algorithm is usually initiated with δ_{min}.

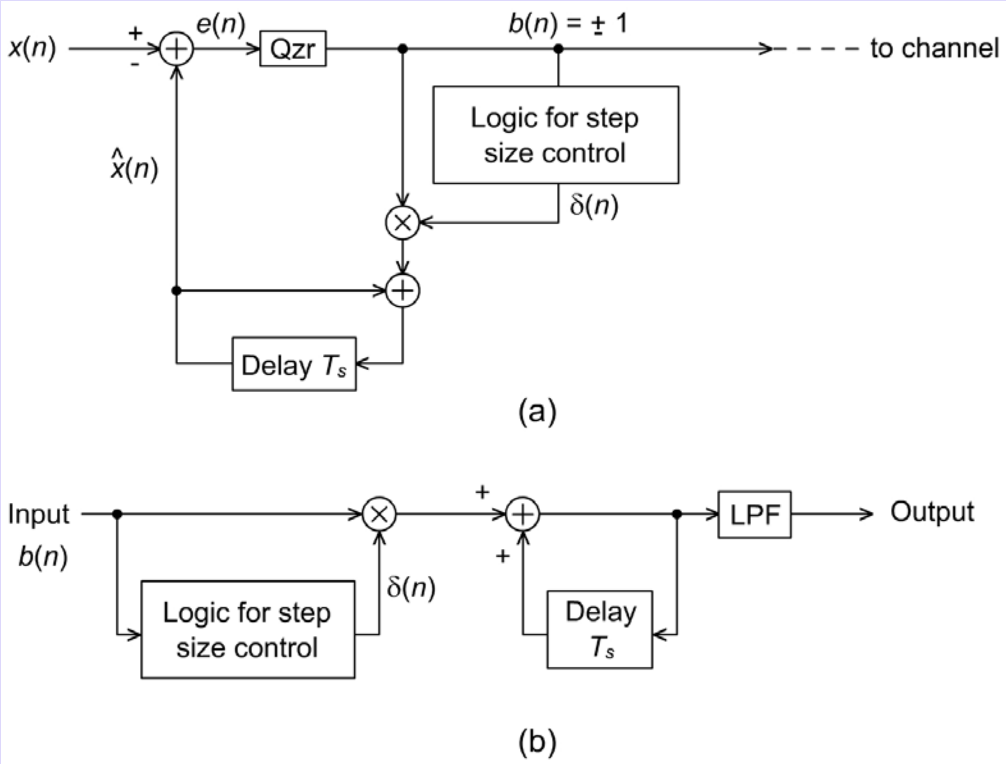


Fig. 6.42: ADM (a) Transmitter (b) Receiver

In a conceptually least complex $(M = 1)$ realization of the logic implied by Eq. 6.48, successive bits $b(n)$ and $b(n-1)$ are compared to detect probable slope-overload $[b(n) = b(n-1)]$, or probable granularity $[b(n) \neq b(n-1)]$. Then, $g(n)$ is arrived at

$$g(n) = \begin{cases} P\,, & \text{if } b(n) = b(n-1) \\ \dfrac{1}{P}\,, & \text{if } b(n) \neq b(n-1) \end{cases} \tag{6.49}$$

where $P \geq 1$. Note that $P = 1$ represents LDM. Typically, a value of $P_{opt} = 1.5$ minimizes the quantization noise for speech signals. In general, for a broad class of signals, $1 < P < 2$, does seem to yield good results. In an illustrative computer simulation with real speech, with $W = 3.3$ kHz and $f_s = 60$ kHz, ADM $(P = 1.5)$ showed an SNR advantage of more than 10 dB over LDM. The ADM scheme with $g(n)$ given by Eq. 6.49 is referred to as *constant factor ADM with one bit memory*.

Results have been reported in the literature which compare the $(SNR)_{0,\,q}$ performance of μ-law PCM and the ADM scheme discussed above. One such result is shown in Fig. 6.43 for the case of bandpass filtered (200-3200 Hz) speech. For PCM telephony, the sampling frequency used is 8 kHz. As can be seen from the figure, the SNR comparison between ADM and PCM is dependent on the bit rate. An interesting consequence of this is, below 50 kbps, ADM which was originally conceived for its simplicity, out-performs the logarithmic PCM, which is now well established commercially all over the world. A 60 channel ADM (continuous adaptation) requiring a bandwidth of 2.048 MHz (the same as used by the 30 channel PCM system) was in commercial use in France for sometime. French authorities have also used DM equipment for airborne radio communication and air traffic control over Atlantic via satellite. However, DM has not found wide-spread commercial usage simply because PCM was already there first!

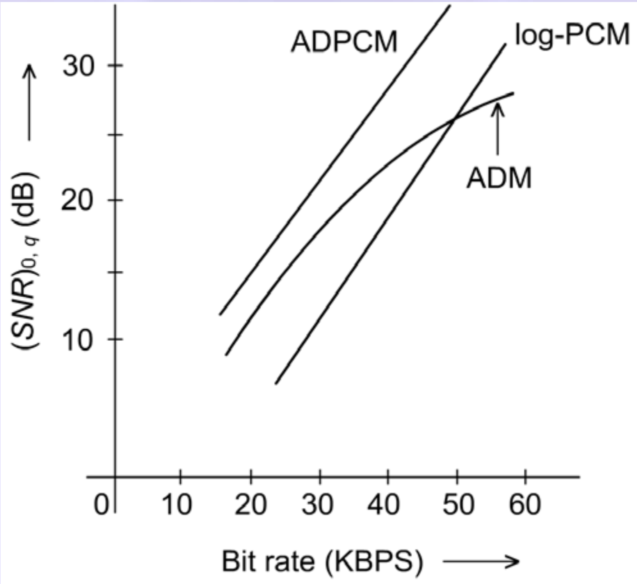


Fig. 6.43: Performance of PCM and ADM versus bit rate

Instantaneously adapting delta modulators (such as the scheme described above) are more vulnerable to channel noise than the slowly adapting or linear coders. Therefore, although instantaneously adapting delta modulators are very simple and efficient (SNR-wise) in a relatively noise-free environments (say, bit error probability $< 10^{-4}$), a number of syllabically adaptive delta modulators have been designed for use over noisy channels and these coders are particularly attractive for low-bit-rate (< 20 k bits/sec) applications. For details on syllabic adaptation, the reader is referred to [3].

Appendix A6.1

PSD of a waveform process with discrete amplitudes

A sequence of symbols are transmitted using some basic pulse $p(t)$. Let the resulting waveform process be denoted by $X(t)$, given by

$$X(t) = \sum_{k = -\infty}^{\infty} A_k \, p(t - kT - T_d) \tag{A6.1}$$

A_k's are discrete random variables, and T_d [1] is a random variable uniformly distributed in the interval $\left(-\frac{T}{2}, \frac{T}{2}\right)$. A_k's are independent of T_d. T is the symbol duration. (For the case of *binary transmission*, we use T_b instead of T; bit rate $= \frac{1}{T_b}$.) We shall assume that the autocorrelation of the discrete amplitude sequence is a function of only the time difference. That is,

$$E\left[A_k \, A_{k+n}\right] = R_A(n) = E\left[A_k \, A_{k-n}\right] \tag{A6.2}$$

With these assumptions, we shall derive an expression for the autocorrelation of the process $X(t)$

$$R_X(t + \tau, t) = E\left[X(t + \tau) \, X(t)\right]$$

$$= E\left[\sum_{k = -\infty}^{\infty} \sum_{j = -\infty}^{\infty} A_k \, A_j \, p(t + \tau - kT - T_d) \, p(t - jT - T_d)\right]$$

Let $j = k + n$ for some n. Then,

$$R_X(t + \tau, \, t) = \sum_k \sum_n \overline{A_k \, A_{k+n}} \left[\frac{1}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} p(\alpha) \, p(\beta) \, d\, t_d\right]$$

where $\alpha = (t + \tau - kT - t_d)$ and $\beta = (t - jT - t_d)$.

---

[1] For an alternative derivative for the PSD, without the random delay T_d in $X(t)$, see [6].

(Note that A_k's are independent of T_d; hence the expectation with respect to T_d can be separated from $\overline{A_k A_j}$)

$$R_X(t+\tau,\ t) = \sum_n R_A(n) \sum_k \frac{1}{T} \left[\int_{-\frac{T}{2}}^{\frac{T}{2}} p(\alpha)\, p(\beta)\, d t_d \right]$$

Let $t - kT - t_d = u$; then, $du = -\, d t_d$

$$R_X(t+\tau,\ t) = \sum_n R_A(n) \sum_k -\frac{1}{T} \int_{t-\left(k-\frac{1}{2}\right)T}^{t-\left(k+\frac{1}{2}\right)T} p(u+\tau)\, p(u-nT)\, du$$

$$= \frac{1}{T} \sum_n R_A(n) \sum_k \int_{t-\left(k+\frac{1}{2}\right)T}^{t-\left(k-\frac{1}{2}\right)T} p(u+\tau)\, p(u-nT)\, du$$

$$= \frac{1}{T} \sum_n R_A(n) \int_{-\infty}^{\infty} p(u+\tau)\, p(u-nT)\, du$$

Let $r_p(\tau)$ be the autocorrelation of the pulse $p(t)$; that is,

$$r_p(\tau) = \int_{-\infty}^{\infty} p(u+\tau)\, p(u)\, du .$$

Then, $R_X(t+\tau,\ t) = \dfrac{1}{T} \sum_n R_A(n)\, r_p(\tau + nT)$ (A6.3)

As the RHS of Eq. A6.3 is only a function of τ, we have,

$$R_X(\tau) = \frac{1}{T} \sum_{n=-\infty}^{\infty} R_A(n)\, r_p(\tau + nT) \qquad \text{(A6.4)}$$

As $r_p(\tau) \longleftrightarrow |P(f)|^2$, Fourier transform of A6.4 results in the equation

$$S_X(f) = \frac{1}{T} |P(f)|^2 \sum_{n=-\infty}^{\infty} R_A(n)\, e^{j 2\pi n f T}$$

where $S_X(f)$ is the power spectral density of the process.

References

1) A. V. Oppenheim, R. W. Schafer and J. R. Buck, Discrete-Time signal processing, (2nd ed.), Pearson Education, 2000

2) Simon Haykin, Communication Systems, (4th ed.) John Wiley, 2001

3) N. S. Jayant and P. Noll, Digital Coding of Waveforms: Principles and applications to speech and video, Prentice Hall, 1984

4) Benevuto et al, The 32 kbps coding standard, AT & T technical journal, vol. 65, no.5, pp12-22, Sept-Oct, 1986

5) M. Decina and G. Modena, CCITT standards on digital speech processing, IEEE journal on selected areas of communication, Vol. 6, Feb. 1988, pp 227-234

6) Leon W. Couch, II, Digital and analog communication systems, Pearson Education Asia, 2001