* Noiseless coding Theorem or Shannon's first theorem

It states that for a code with an alphabet of r symbols, a source with an alphabet of q symbols, the average length of the code word per source symbol can be made close to the lower bound $\frac{H(S)}{\log_2 r}$ as desired by encoding each source symbol individually.

Proof:

The information of k^{th} message

$$I_k = \log(1/P_k)$$

The information carried by a message is equal to the no. of elements in that message.

$$\text{i.e., } l_k = \log(1/P_k)$$

$$\log_r(1/P_k) \leq l_k \leq \log_r((1/P_k)+1)$$

multiplying throughout by P_k and summing for all values of k, we get

$\left\{ \log_r a = \frac{\log a}{\log r} \right\}$

$$P_k \times \frac{\log(1/P_k)}{\log r} \quad \Sigma P_k \times l_k \leq \frac{\log((1/P_k)+1)}{\log r} \times P_k$$

$\log\sigma \qquad k=1 \qquad k=1 \qquad \frac{\quad}{\log\sigma}+1$

$$\frac{H(S)}{\log\sigma} \le L \le \frac{H(S)}{\log\sigma}+1$$

Taking n^{th} order extension of the source

$$\frac{H(S^n)}{\log\sigma} \le L_n \le \frac{H(S^n)}{\log\sigma}+1$$

from, $\boxed{H(S^n) = nH(S)}$

$$\therefore \frac{n\,H(S)}{\log\sigma} \le L_n < \frac{n\,H(S)}{\log\sigma}+1$$

Dividing throughout by n

$$\frac{\cancel{n}\times H(S)}{\cancel{n}\log\sigma} \le \frac{L_n}{n} < \frac{H(S)\times\cancel{n}}{\cancel{n}\log\sigma}+\frac{1}{n}$$

ie, $$\frac{H(S)}{\log\sigma} \le \frac{L_n}{n} \le \frac{H(S)}{\log\sigma}+\frac{1}{n}$$

we have $\lim_{n\to\infty} \frac{L_n}{n} = L$

$$\therefore \frac{H(S)}{\log\sigma} \le L$$

Hence proved

1. Shannon- Fano Algorithm

* Procedures

(1) → List the messages in the order of decreasing probabilities

(2) → Partition this ensemble into almost two equiprobable subgroups

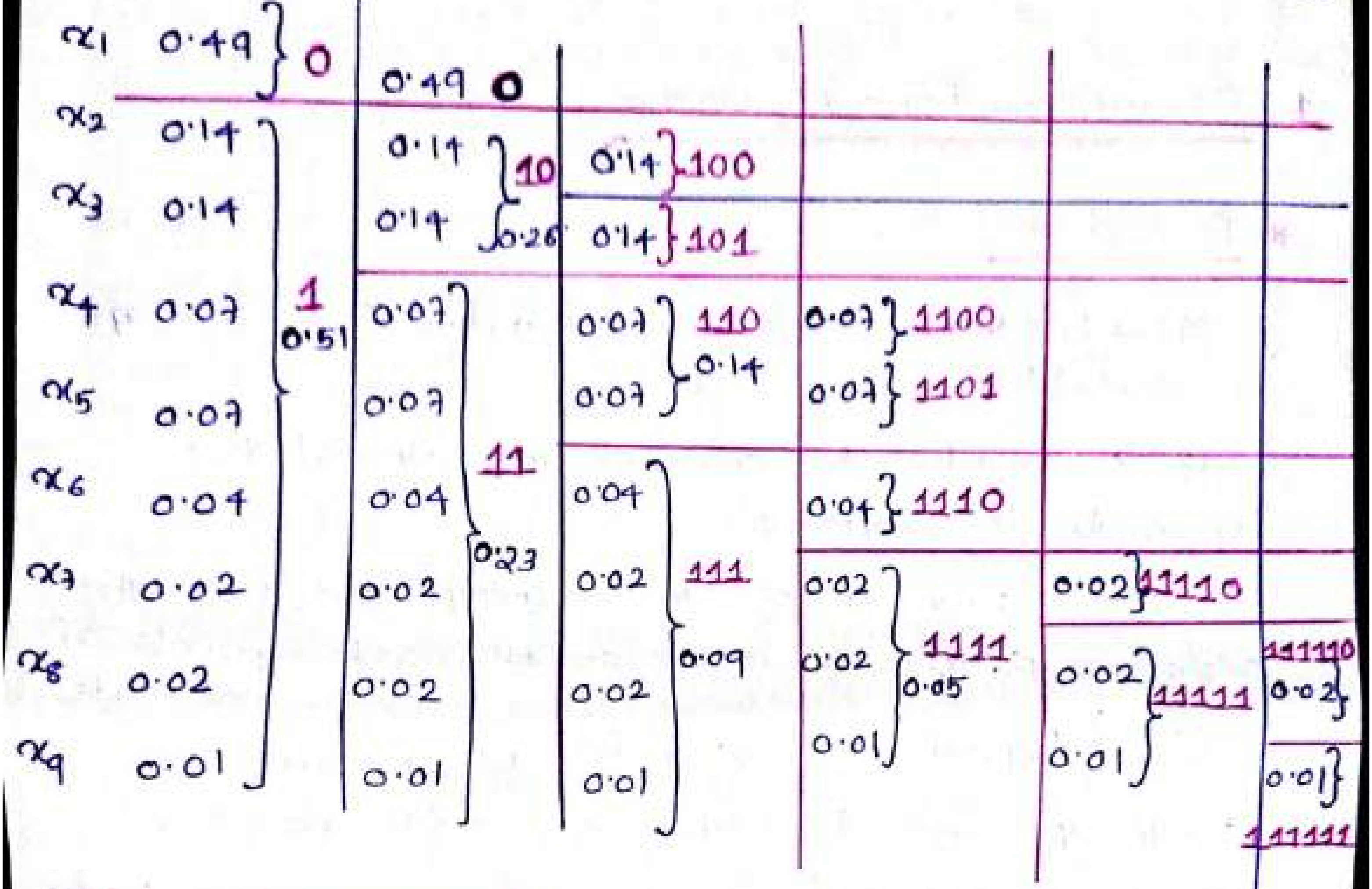
(3) → Assign a '0' to one group and '1' to the other group. These form the starting code symbols of these codes

(4) → Repeat step (2) and (3) on each of the subgroups until the subgroups contain only one source symbol, to determine the succeeding code symbols of the code words.

(5) → For convenience, a code tree may be constructed and codes read off directly.

Qn Apply shannon - fano coding procedure for the following messages.

$$P = \{0.49, 0.14, 0.14, 0.07, 0.07, 0.04, 0.02, 0.02, 0.01\}$$



Symbol	code	length.
x_1	0	1
x_2	100	3
x_3	101	3
x_4	1110	4
x_5	1101	4
x_6	1110	4
x_7	11110	5
x_8	111110	6
x_9	111111	6

$$\eta = \frac{H(X)}{\log_2 r}$$

$$L = \sum_{k=1}^{q} P_k l_k$$

$$\therefore L = 0.49\times1 + 0.14\times3 + 0.14\times3 + 0.07\times4 + 0.07\times4 + 0.04\times4 + 0.02\times5 + 0.02\times6 + 0.01\times6$$

$$= \underline{\underline{2.33}}$$

$$H(X) = \sum_{k=1}^{q} P_k \log_2 (1/P_k)$$

$$= 0.49 \log_2(1/0.49) + 0.14\times\log_2(1/0.14) + 0.14\times\log_2(1/0.14) + 0.07\times\log_2(1/0.07) + 0.04\times\log_2(1/0.04) + 0.02\times\log_2(1/0.02) + 0.02\times\log_2(1/0.02) + 0.01\times\log_2(1/0.01) + 0.07\times\log_2(1/0.07)$$

$= 2.313$ bits / symbol //

$$\eta = \frac{H(x)}{L \log_2(r)}$$

$$= \frac{2.313}{2.33 \log_2(2)}$$

$$= \frac{2.313}{2.33} = 0.9927 = 99.27\%$$

$$\therefore \epsilon = 1 - \eta = 100 - 99.27 = 0.73\%$$

Practice Problem (Assignment - 1, Qn no. 1)

Consider a message ensemble $s_1, s_2, s_3, \ldots s_8$, having probabilities $1/4, 1/4, 1/8, 1/8, 1/16, 1/16, 1/16, 1/16$. Consider code alphabet $X = \{0, 1\}$. Apply shannon - fano coding procedure and find out the efficiency of coding.

-> Procedure applicable for both binary and non-binary encoding

-> code with minimum average length, L

* Procedure:

1. List the source symbols in the decreasing order of probabilities

2. check if $q = r + \alpha(r-1)$ is satisfied and find the integer, α

3. club the last 'r' symbols into a single composite symbol whose probability of occurrence is equal to the sum of probabilities of occurence of the last r symbols involved in the step.

4. Repeat steps 1 and 3 on the resulting set of symbols until in the final step exactly r-symbols are left.

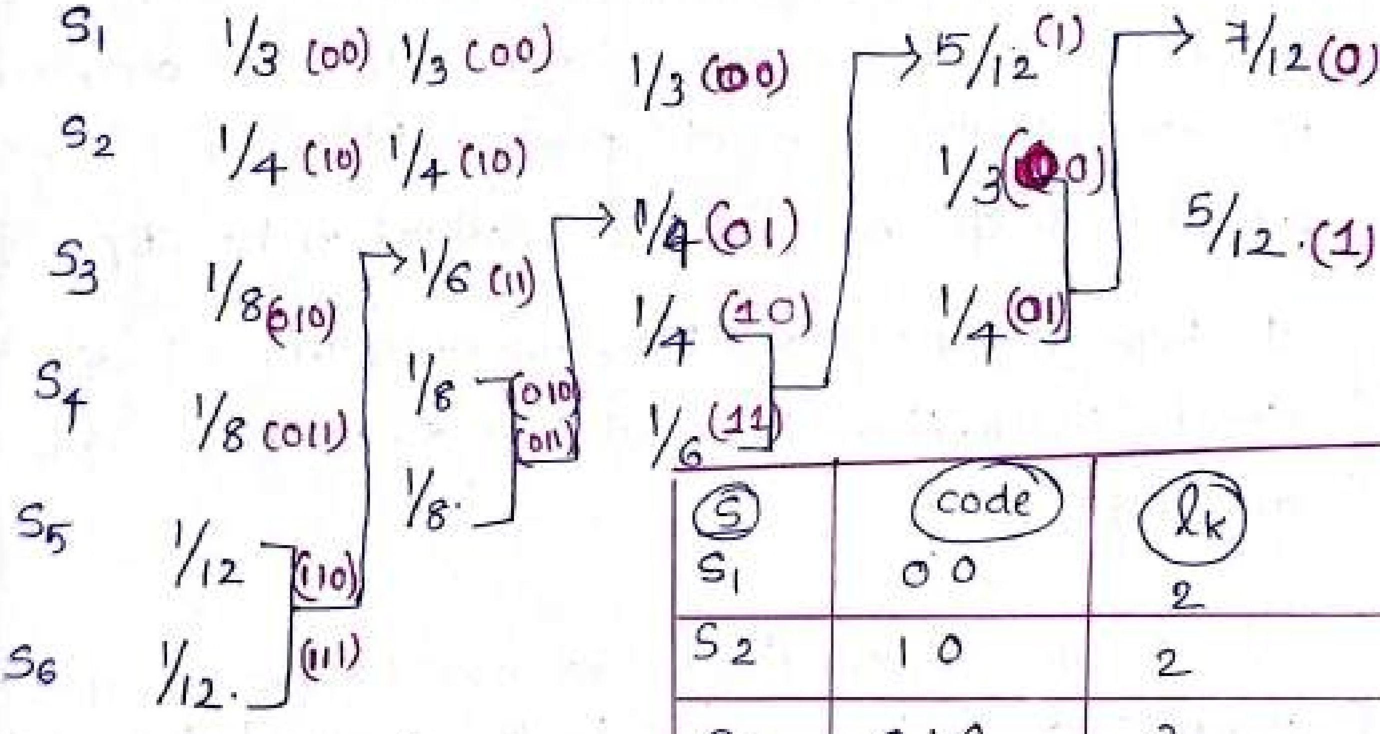
5. Assign codes freely to the last r-composite symbols & work backwards to the original source to arrive at the optimum codes.

6. Alternatively, follow the steps carefully a tree diagram can be constructed starting from

7. Discard the codes of the dummy symbols

Qn Apply Huffman's coding to the messages $S = \{s_1, s_2, s_3, s_4, s_5, s_6\}$ with probabilities $P = \{1/3, 1/4, 1/8, 1/8, 1/12, 1/12\}$ and calculate the efficiency of coding, $X = \{0, 1\}$

Ans:



S	code	l_k
S_1	00	2
S_2	10	2
S_3	010	3
S_4	011	3
S_5	110	3
S_6	111	3

$$L = \sum_{k=1}^{6} P_k l_k$$

$$= \frac{2}{3}\times 1 + 2\times\frac{1}{4} + 3\times 1/8 + 3\times 1/8 + 3\times 1/12 + 3\times 1/12$$

$$= 2.4167 \text{ bits/symbol}$$

$$H(S) = \sum_{k=1}^{6} P_k \log(1/P_k) = 2.376 \text{ bits/symbol}$$

$$\eta = \frac{H(S)}{L \log_2(2)} = 98.31\%$$

$$\epsilon = 1 - \eta = 1.69\%$$

[Pulse Code Modulation]:

It is digital pulse modulation technique. PCM is complex compared to analog pulse modulation techniques (ie PAM, PWM & PPM) in the sense that the message s/l is subjected to large no: of operations.

→ Consists of 3 parts transmitter, transmission path & receiver

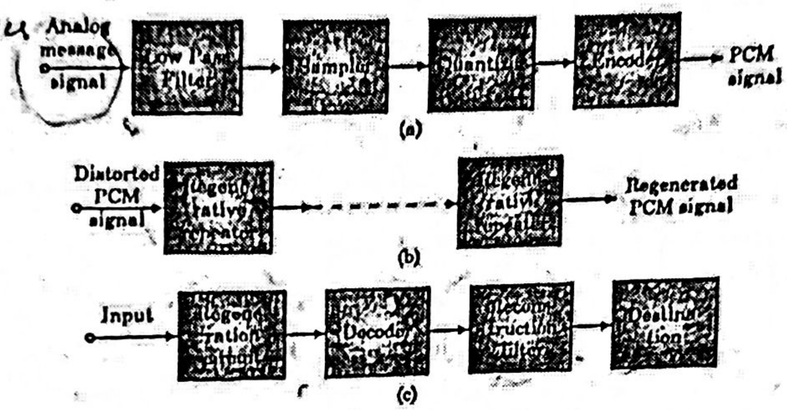


FIGURE 4.2 *The basic elements of a PCM system (a) Transmitter (b) Transmission path (c) Receiver.*

→ PCM information is txtd in form of codewords. A PCM s/m consists of PCM encoder (txtr) & a PCM decoder (rxr) i.e.

→ Essential opns are sampling, quantising & encoding in a PCM transmitter.

→ All opns are performed in same ckt called analog to digital converter.

→ Regeneration of impaired signals occurs at intermediate points along the transmission path as indicated in the middle part.

→ At the rxr, essential opns consist of one last stage of regeneration followed by decoding, then demodu of the train of quantized samples as in the bottom part.

→ The opns of decoding & reconstruction are usually performed in the same ckt called digital to analog converter.

→ After sampling & quantizing the encoding process is used to translate the discrete set of sample values to a more appropriate form of signal.

→ Representing each member of this discrete set of values as a particular arrangement of discrete events is called a code.

→ One of the discrete events in a code is called a code element or symbol.

→ A particular arrangement of symbols used in a code to represent a single value of the discrete set is called a code-word or character.

→ There are binary as well as ternary code. The max. advantage over the effects of noise in a transmission medium is obtained by using a binary code as it withstands a relatively high level of noise & is easy to regenerate.

→ There are several ways to represent binary sequences produced by ADC.

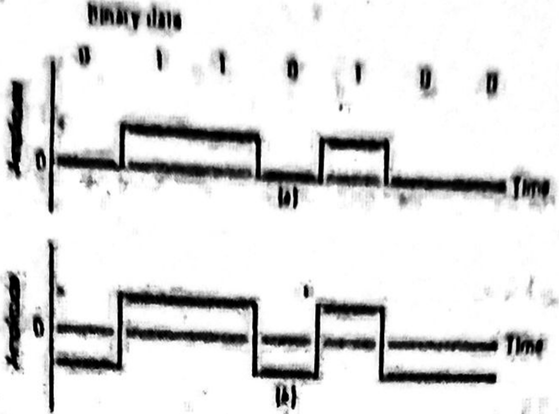


Fig: Two binary waveforms. (a) Nonreturn-to-zero unipolar. (b) Non return to zero polar

some are shown above, which are unipolar & bipolar waves.

NOISE IN PCM SYSTEM:

→ Noise is produced in txr by quantisation at the end of a PCM system by rounding off sample values of an analog baseband s/l to the nearest permissible representation levels of the quantizer.

→ Quantisation noise is signal dependent.

→ Consider a memoryless quantizer both uniform & symmetric, with L representation levels.

→ Let x denote quantizer i/p & y denote quantizer o/p. They are related by transfer characteristic as shown by,

$$y = Q(x) \quad \rightarrow ①$$

which is a staircase ~~function~~ function that befits the type of midtread or midriser quantizer.

→ Suppose that the i/p x lies inside the interval

$$\mathcal{I}_k = \{x_k < x \le x_{k+1}\} \; ; \; k = 1, 2, \ldots L$$

where x_k & x_{k+1} are decision thresholds of the interval as in fig below.

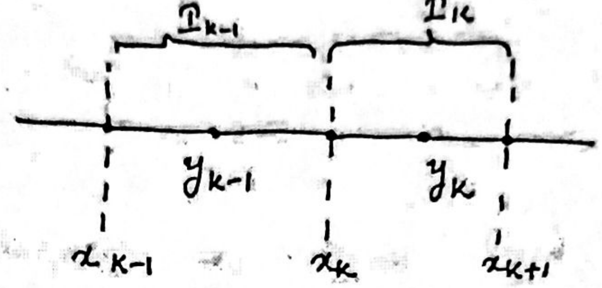


fig: Decision thresholds of the quantizer.

→ Corresponding the quantizer o/p y takes on a discrete value y_k, $k = 1, 2, \ldots L$.

ie, $y = y_k$; if x lies in interval $\mathcal{I}_k$ → ②

→ Let q denote the quantization error with values in range $-\Delta/2 \le q \le \Delta/2$. Then, $y_k = x + q$; if x lies in the interval $\mathcal{I}_k$

→ 'x' is the sample in X (random variable) with zero mean & σ_X^2 variance.

→ When fine quantization occurs (ie $L > 64$), distortion produced by noise affects PCM s/m performance as though it is determined by stepsize Δ.

→ Let Q denote quantisation error & let q denote its sample value.

the amount random variable Q is uniformly distributed over $-\Delta/2$ to $\Delta/2$ as, (29)

$$f_Q(q) = \begin{cases} 1/\Delta & , -\Delta/2 \le q \le \Delta/2 \\ 0 & ; \text{otherwise.} \end{cases} \rightarrow (3)$$

where $f_Q(q)$ is the probability density function of Q. & we must ensure the incoming s/l does not overload the quantizer.

→ Then the mean of quantization error is zero its variance σ^2_q is same as mean-square value,

$$\sigma^2_q = E[Q^2]$$

$$= \int_{-\infty}^{\infty} q^2 f_Q(q)\, dq \rightarrow (4)$$

Subst (3) in (4),

$$\sigma^2_q = \frac{1}{\Delta}\int_{-\Delta/2}^{\Delta/2} q^2\, dq = \frac{\Delta^2}{12}$$

→ Let variance be denoted by σ^2_x.

→ SNR of o/p s/l is, $(SNR)_o = \frac{\sigma^2_x}{\sigma^2_q} = \frac{\sigma^2_x}{\Delta^2/12}$

→ Also 'x' has amplitude $-x_{max}$ to $+x_{max}$

$$\Delta = \frac{x_{max} - (-x_{max})}{L} = \frac{2x_{max}}{L}$$

→ If max & min values are normalized to 1, then, $\Delta = \frac{2}{L}$, then, $L = \frac{2}{\Delta}$

if Δ is sufficiently small then $Q_e = \frac{\Delta}{[illegible]}$

$$Q = \frac{\Delta}{2}$$

10.22 COMPANDER CHARACTERISTIC

Figure 10.20 shows the compander characteristics which is the combination of the compressor and expander characteristics. Due to the inverse nature of compressor and expander, the overall characteristics of the compander is a straight line (dotted line in figure 10.20. This indicates that all the boosted signals are brought back to their original amplitudes.*

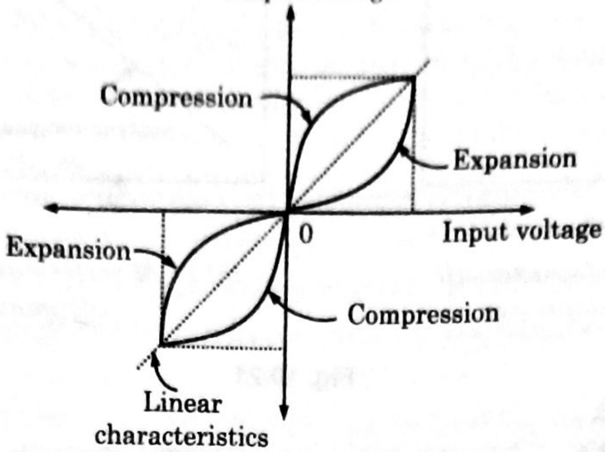


Fig. 10.20 *Companding curves for PCM system*

10.23 DIFFERENT TYPES OF COMPRESSOR CHARACTERISTICS

(GGSIPU, Delhi, Sem. Exam., 2006-07) (10 marks)

Ideally, we need a linear compressor characteristics for small amplitudes of the input signal and a logarithmic characteristic elsewhere. In pratice, this is achieved by using following two methods:

(*i*) μ-law companding

(*ii*) A-law comapding

10.23.1. μ-law Companding

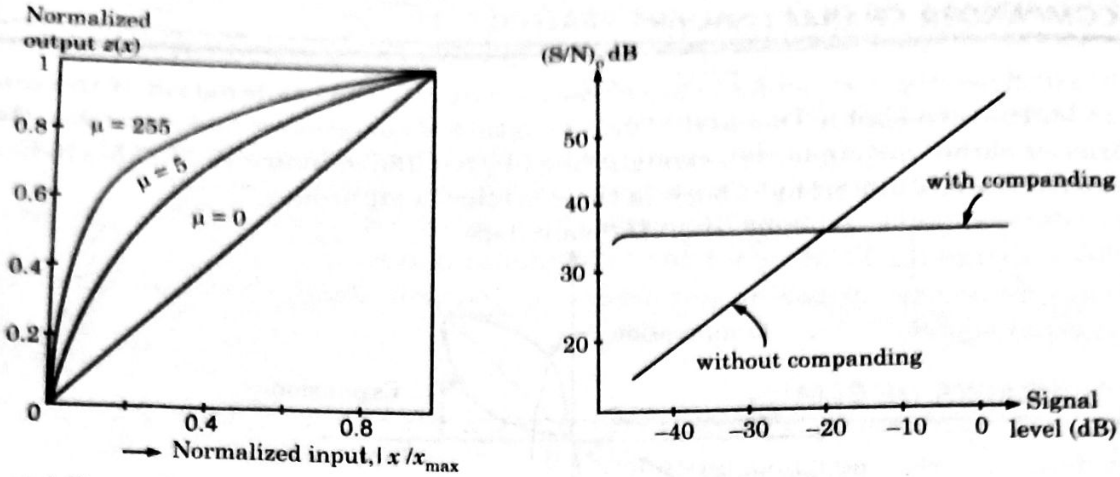
In the μ-law companding, the compressor characteristic is continuous. It is approximately linear for smaller values of input levels and logarithmic for high input levels. The μ-law compressor characteristic is mathematically expressed as under:

$$z(x) = (\text{sgn}\, x)\frac{\ln(1+\mu|x|/x_{max})}{\ln(1+\mu)} \qquad ...(10.44)$$

where $0 \le |x|/x_{max} \le 1.$

Here, $z(x)$ represents the output and x is the input to the compressor. Also, $|x|/x_{max}$ represents the normalized value of input with respect to the maximum value x_{max} . Further, (sgn x) term represents ± 1 *i.e.*, positive and negative values of input and output. The μ-law compressor characteristics for different values of μ have been shown in figure 10.21(*a*). The practically used value of μ is 255.It may be noted that the characteristic corresponding to μ = 0 corresponds to the uniform quantization. The μ-law comanding is used for speech and music signals. It is used for PCM telephone systems in United States, Canada and Japan. Figure 10.21(*b*) shows the variation of signal to quantization noise ratio with respect to signal level, with and without companding. It is obvious that SNR is almost constant at all the signal levels when companding is used.

* Non uniform quantization can be used to make the SNR a constant for all signals within the input range.



(a) Compressor characteristic of a μ-law compressor

(b) PCM performance with μ-law companding

Fig. 10.21

10.23.2. A-law Comanding

(VTU, Bangalore, Sem. Exam., 2004-2005)

In the A-law companding, the compressor characteristic is piecewise, made up of a linear segment for low level inputs and a logarithmic segment for high level inputs. Figure 10.22 shows the A-law compressor characteristics for different values of A. Corresponding to $A = 1$, we observe that the characteristic is linear which corresponds to a uniform quantization. The practically used value of A is 87.56. The A-law companding is used for PCM telephone systems in Europe. The linear segment of the characteristics is for low level inputs whereas the logarithmic segments is for high level input. *It is mathematically expressed as under:

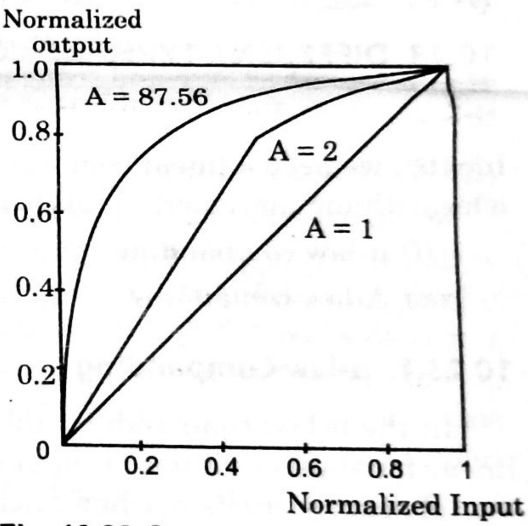


Fig. 10.22 *Compresssor charqcteristic of A-law compressor*

$$\frac{z(x)}{x_{max}} = \begin{cases} \dfrac{A\,|x|/x_{max}}{1 + \log_e A} & \text{for } 0 \le \dfrac{|x|}{x_{max}} \le 1 \\[2ex] \dfrac{1 + \log_e [A\,|x|/x_{max}]}{1 + \log_e A} & \text{for } \dfrac{1}{A} \le \dfrac{|x|}{x_{max}} \le 1 \end{cases} \quad \text{...(10.44}a\text{)}$$

10.24 APPLICATIONS OF PCM

Some of the applications of PCM may be listed as under:

(i) With the advent of fibre optic cables, PCM is used in telephony.

(ii) In space communication, space craft transmits signals to earth. Here, the transmitted power is quite small (*i.e.*, 10 or 15 W) and the distances are very large (*i.e.*, a few million km). However, due to the high noise immunity, only PCM systems can be used in such applications.

4.2. LINEAR PREDICTION

Linear prediction is a technique of time series analysis that emerges from the examination of linear systems. Using linear prediction, the parameters of such a system can be determined by Analyzing the systems inputs and outputs.

Prediction constitutes a special form of estimation specifically, the requirement is to use a finite set of present and past samples of a stationary process to predict a sample of the process in the future. The prediction is linear if it is a linear combination of the given samples of the process.

Predictor

Predictor is a kind of filter designed to perform the prediction operation.

Prediction Error

The difference between the actual sample of the process at the future time of interest and the predictor output is called the prediction error.

According to the Wiener filter theory, a predictor is designed to minimize the mean square value of the prediction error.

Consider the random samples $X_{n-1}, X_{n-2} \dots X_{n-M}$ drawn from a stationary process $X(t)$. Suppose the requirement is to make a prediction of the sample X_n.

Let $\hat{X}_n$ denote the random variable resulting from this prediction. We can write

$$\hat{X}_n = \sum_{k=1}^{M} h_{ok} X_{n-k} \quad \dots (4.1)$$

where $h_{o1}, h_{o2}, \dots h_{oM}$ are the optimum predictor coefficients. M is the number of delay elements employed in the predictor, as its order.

The predictor as a special case of the Wiener filter and it proceed as follows.

1. The variance of the sample X_n, viewed as the desired response, equals

$$\sigma_X^2 = E[X_n^2]$$

$$= R_X(0) \quad \dots (4.2)$$

Where it is assumed that X_n has zero mean.

2. The cross-correlation function of X_n, acting as the desired response, and X_{n-k}, acting as the k^{th} tap input of the predictor, is given by

$$E[X_n X_{n-k}] = R_X(k) \qquad k = 1, 2, \dots M \quad \dots (4.3)$$

3. The auto correlation function of the predictor's tap input X_{n-k} with another tap input X_{n-m} is given by

$$E[X_{n-k} X_{n-m}] = R_X(m-k) \qquad k, m = 1, 2, \dots M \quad \dots (4.4)$$

4. The normal equation to fit the linear prediction problems as follows.

$$\sum_{m=1}^{M} h_{om} R_X(m-k) = R_X(k) \qquad k = 1, 2, \dots M \quad \dots (4.5)$$

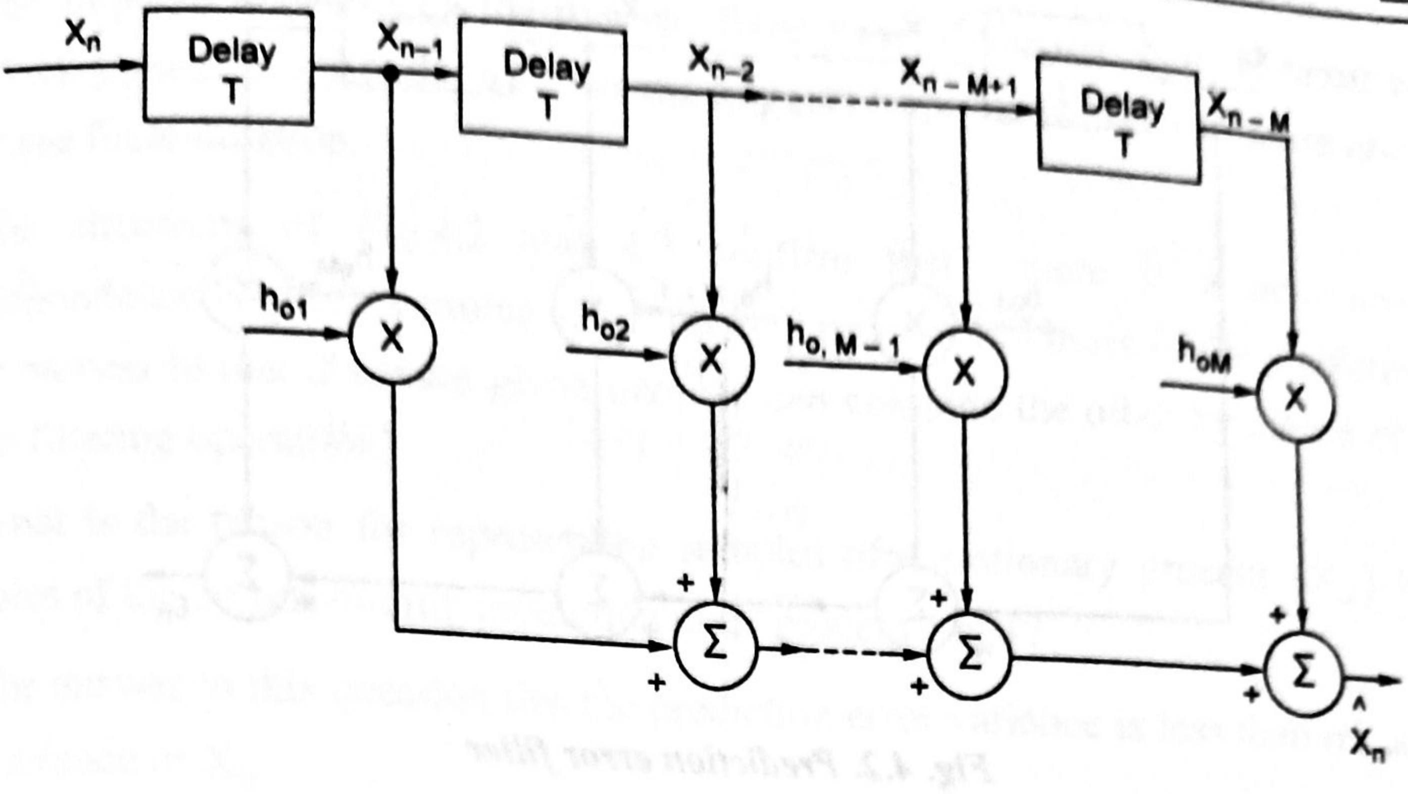


Fig. 4.1. Linear Predictor

4.2.1. PREDICTION ERROR PROCESS

The prediction error can be denoted by ε_n and is defined by

$$\Sigma_n = X_n - \hat{X}_n$$

$$= X_n - \sum_{k=1}^{M} h_{ok} X_{n-k} \qquad \text{... (4.6)}$$

The prediction error ε_n can be computed by giving the present and past samples of a stationary process, namely X_n, X_{n-1} ... X_{n-M} and given the predictor co efficients $h_{o1}, h_{o2}, \ldots h_{oM}$ by using a structure called a prediction-error filter.

The operation of prediction-error filtering is invertible. Specifically we may rearrange equation (4.6) as

$$X_n = \sum_{k=1}^{M} h_{ok} X_{n-k} + \varepsilon_n \qquad \text{... (4.7)}$$

The present sample of the original process X_n may be computed as a linear combination of past samples of the process X_{n-1}, ... X_{n-M} plus the present prediction error ε_n, where subscript n refers to the present.

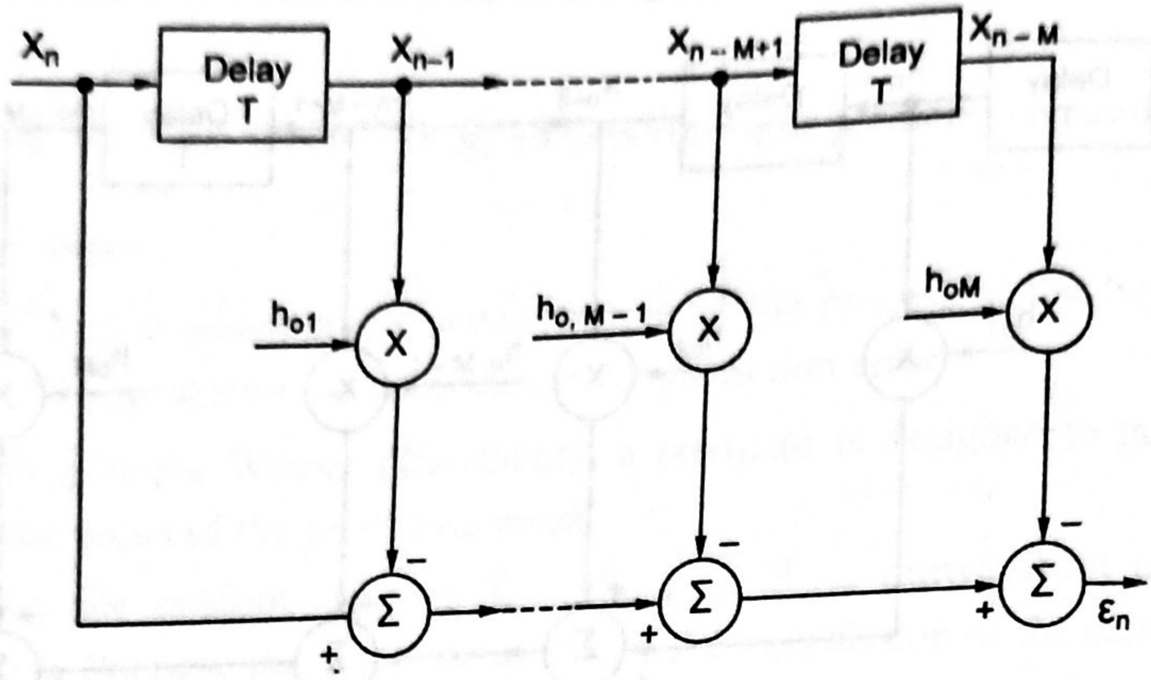


Fig. 4.2. Prediction error filter

Below figure depicts the structure for performing the inverse operation. So this structure can be called as inverse filter.

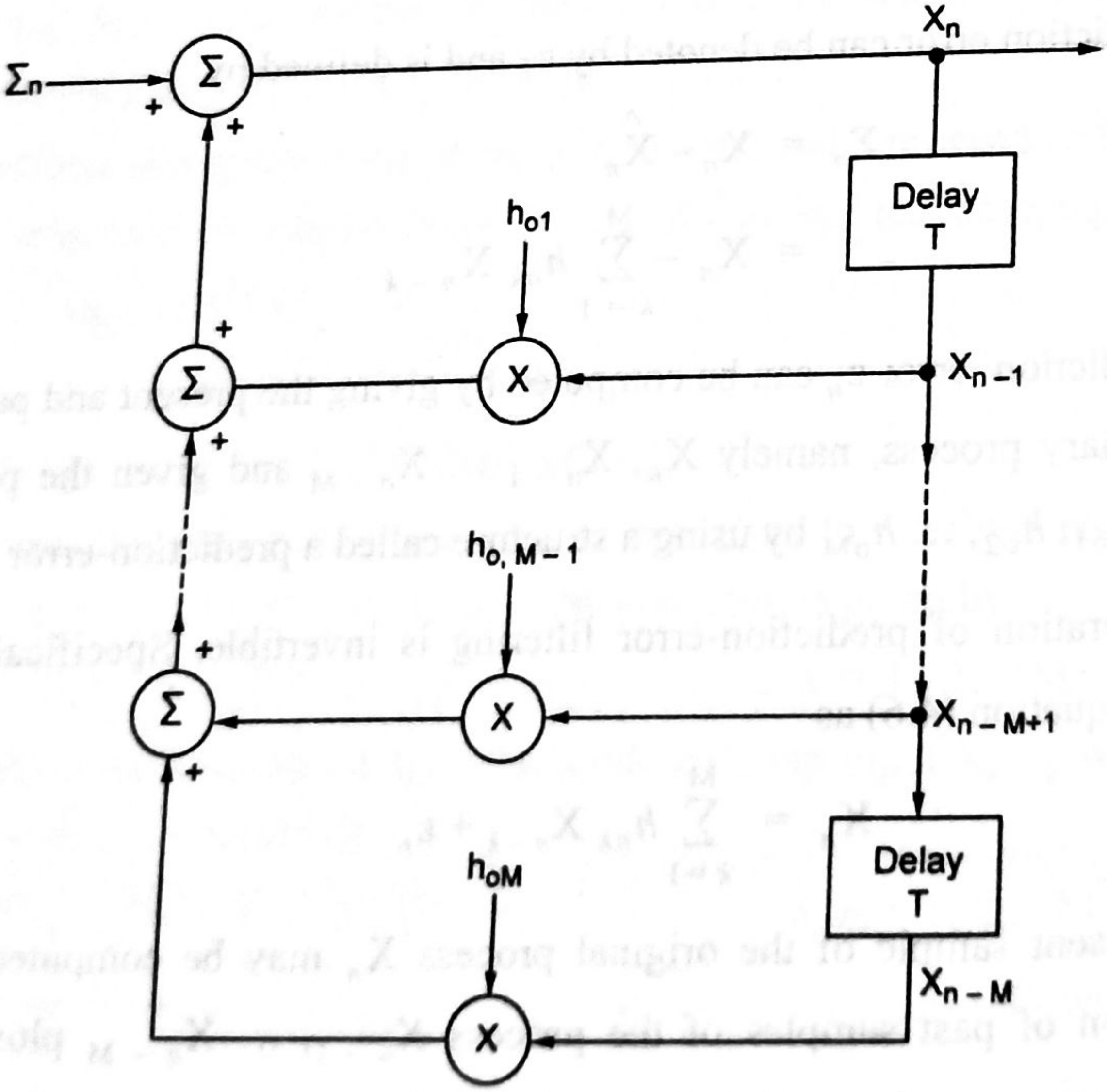


Fig. 4.3. Inverse filter

The impulse response of the inverse filter has infinite duration because of feedback present in the filter, whereas the impulse response of the prediction error filter has finite duration.

The structures of Fig.4.2 and 4.3 confirm that there is a one-to-one correspondence between samples of a stationary process and those of the prediction error process in that if we are given one, we can compute the other by means of a linear filtering operation.

What is the reason for representing samples of a stationary process $\{X_n\}$ by samples of the corresponding prediction error process $\{ \varepsilon_n \}$?

The answer to this question lies the prediction error variance is less than σ_X^2 and the variance of X_n.

If X_n has zero mean then we have ε_n. The prediction error variance equals.

$$\sigma_E^2 = E[\varepsilon_n^2]$$

$$= R_X(0) - \sum_{k=1}^{M} h_{ok} R_X(k) \qquad \ldots (4.8)$$

4.2.2. PROPERTIES OF LINEAR PREDICTION

Libnear prediction has following two important properties.

Property 1

The prediction error variance decreases with increasing predictor order.

In theory, this trend may go on indefinitely or until a critical predictor order is reached, where after there is no further reduction.

Property 2

When the prediction error variance reaches its minimum possible value, the prediction error process a assumes the form of white noise.

A prediction error filter designed to whiten a stationary process is called a **Whitening filter**. The resultant white noise sequence is known as the **innovations** process associated with the predictor input and the term " innovation" refers to "newness". Hence only new information is retained in the innovations process.

Prediction relies on the presence of correlation between adjacent samples of a stationary process.

If we increase the predictor order then we successively reduce the correlation between adjacent samples of the process, until the prediction error process consists of a sequence of uncorrelated samples.

When this condition is reached, the prediction error variance attains its minimum possible value and the whitening of the original process is accomplished.

Property 1 is exploited in the design of waveform coders.

Property 2 is exploited in the design of source coders.

4.3. LINEAR PREDICTOR VOCODERS

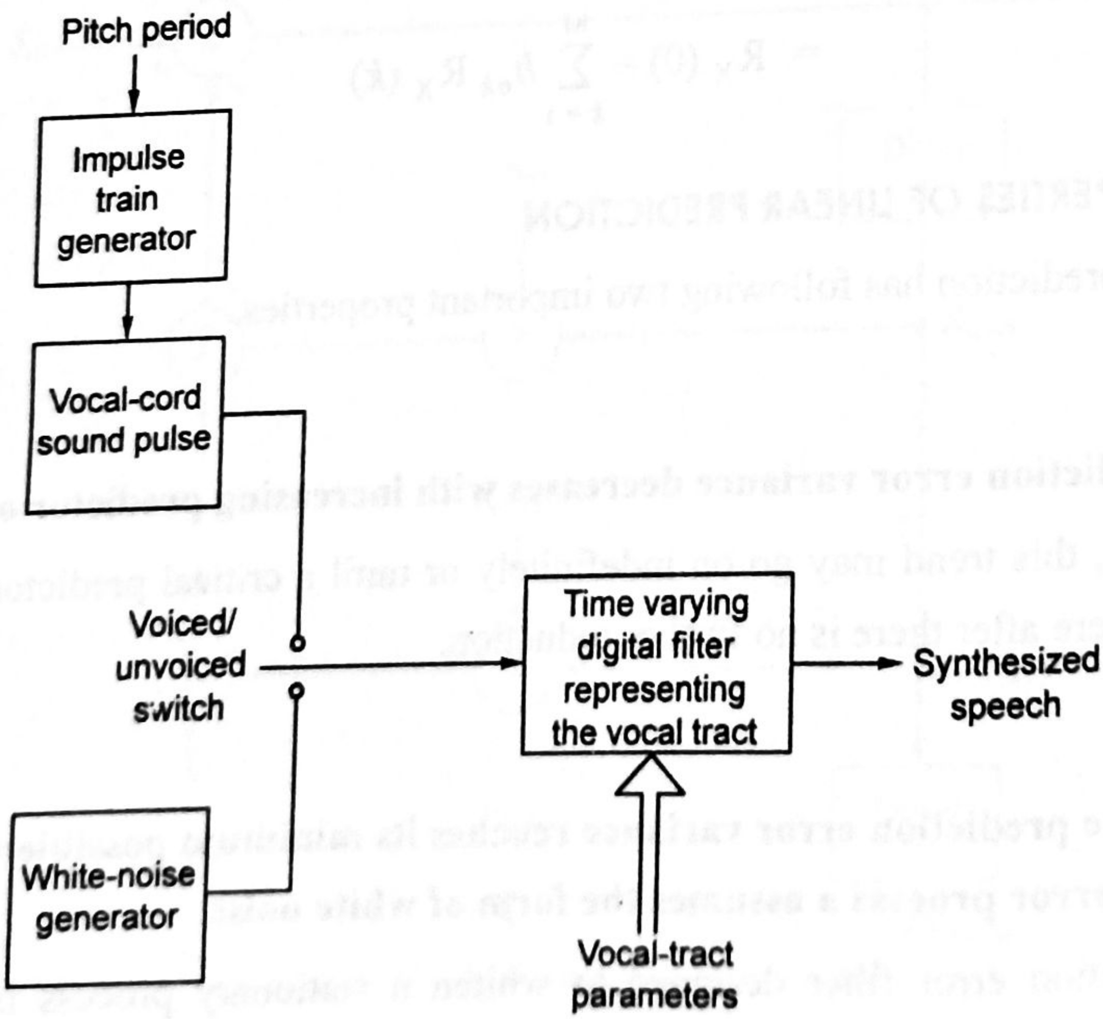


Fig. 4.4. Model of speech production process

Linear prediction provides the basis of an important source coding technique for the digitization of speech signals. The technique known as Linear predictive vocoding, relies on parameterization of speech signals according to a physical model for the speech production process.

Voiced Sounds

Voiced sounds are produced by forcing air through the glottis with the tension of the vocal cords adjusted. So that they vibrate in a realization oscillation. Thereby producing quasi-periodic pulses of air that excite the vocal tract.

Unvoiced Sounds

It is also called as Ericative sounds. These sounds are generated by forming a constriction at some point in the vocal tract and forcing air through the construction at a high enough velocity to produce turbulence.

Examples of voiced and unvoiced sounds are represented by utterances for the "A" and "S" segment in the word "Salt".

The speech waveform shown in Fig.4.5 (*a*) is the result and the utterance "every salt breeze comes from the sea" by a male subject.

The waveform shown in Fig.4.5 (*b*) corresponds to the "A" segment in the word "Salt" and magnified waveform shown in Fig.4.5 (*c*) corresponds to the "S" segment.

The generation of a voiced sound is modeled as the response of the vocal tract filter excited with a periodic equal to the pitch period.

An unvoiced sound is modeled as the response of the vocal filter excited with a white noise sequence.

The vocal tract filter is time varying, so that its coefficients can provide an adequate representation for the input segment of voiced or unvoiced sound.

Linear predictive vocoder consists of two things

1. Transmitter

2. Receiver

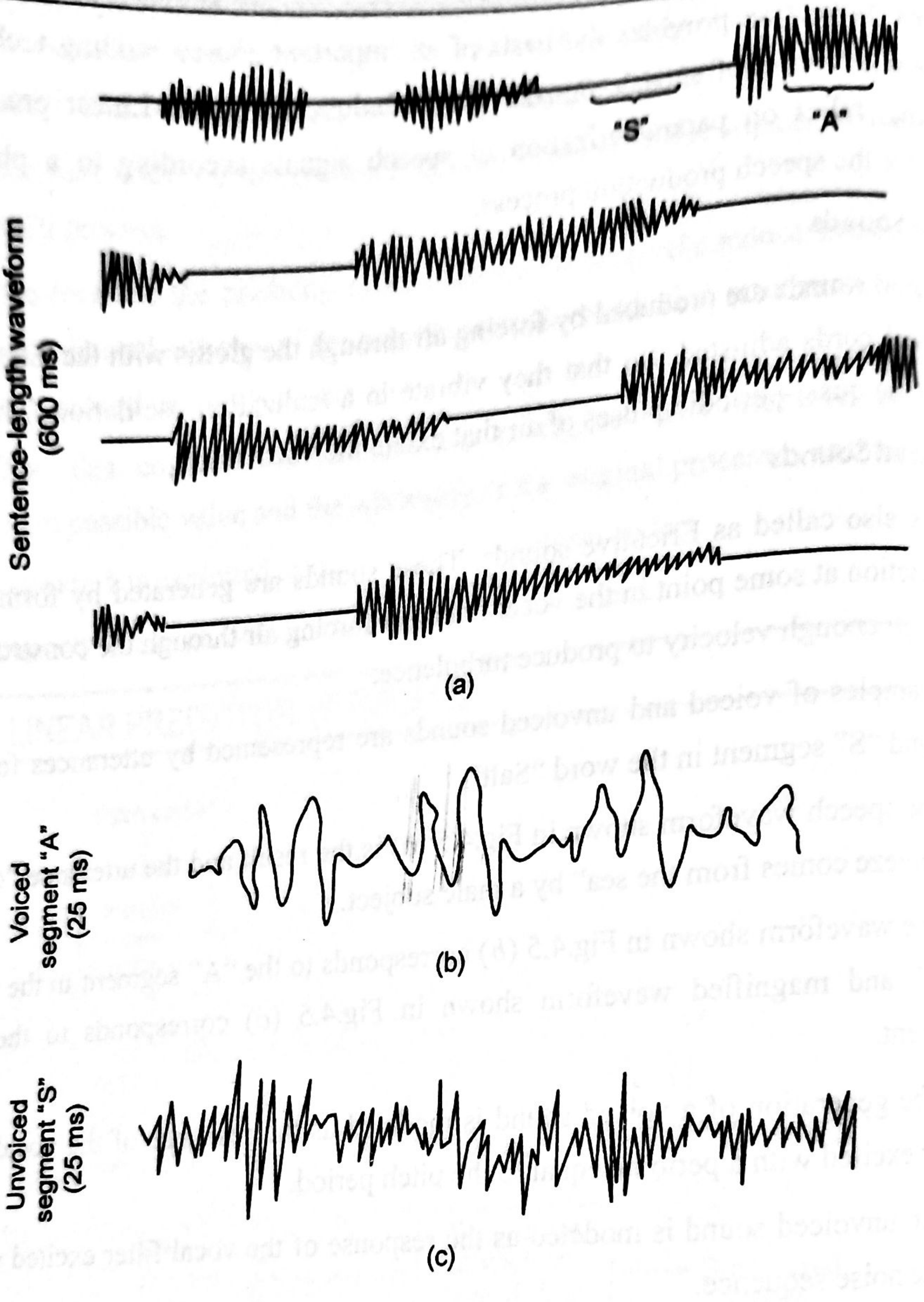


Fig. 4.5.

4.3.1. TRANSMITTER

First transmitter will perform analysis on the input speech signal, block by block. Each block is 10-30 ms long, for which the speech production process may be treated as essentially stationary.

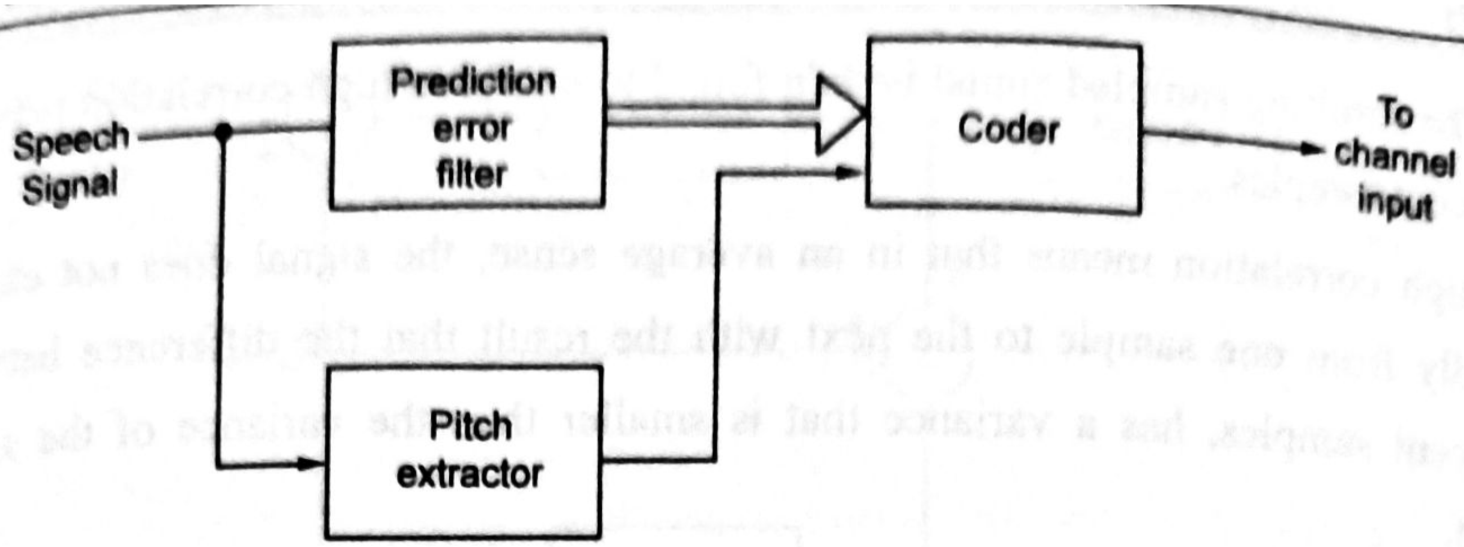


Fig. 4.6. Linear Predictive Vocoder Transmitter

The resulting parameters of analysis called as Prediction Error Filt Co-efficients and a voiced/unvoiced parameter.

The pitch period will provide a complete description for the particular segme of the input speech signal. A digital representation of the parameters constitutes th transmitted signal.

4.3.2. RECEIVER

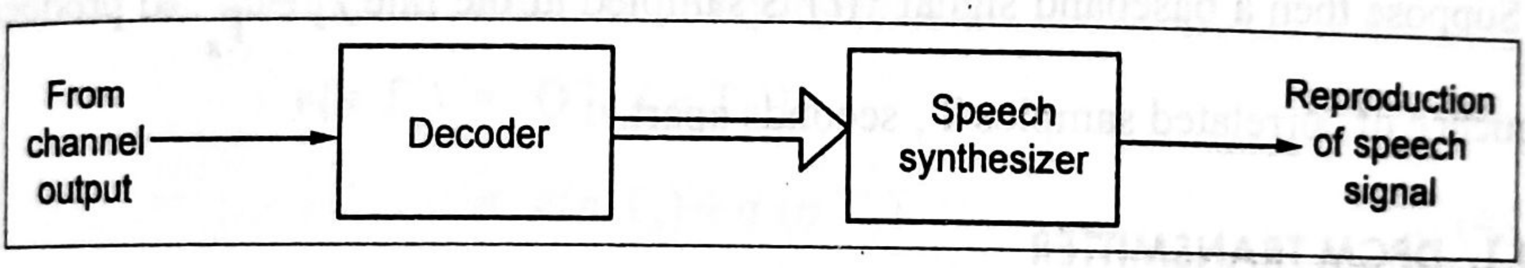


Fig. 4.7. Linear Predictive Vocoder - Receiver

First the receiver will performs decoding followed by synthesis of the speec signal and the latter operation utilizes the model of speech production process.

The artificial sounding reproduction of the original speech signal is the standar result of this analysis/synthesis.

Some poor reproduction quality of a linear predictive vocoder is tolerated fo secure military communications it requires very low bit rates. (4 kb/s or less).

3. The Wiener Filter

3.1 The Wiener-Hopf Equation

The Wiener filter theory is characterized by:

1. The assumption that both signal and noise are random processed with known spectral characteristics or, equivalently, known auto- and cross-correlation functions.
2. The criterion for best performance is minimum mean-square error. (This is partially to make the problem mathematically tractable, but it is also a good physical criterion in many applications.)
3. A solution based on scalar methods that leads to the optimal filter weighting function (or transfer function in the stationary case).

$z(n) = s(n) + v(n)$ → $h(i)$ or $H(z)$ → $\hat{s}(n)$

JWSS   LTI   WSS

R_s, R_v, R_z, R_{sz} or $S_s(z), S_v(z), S_z(z), S_{sz}(z)$ known

$$J = E\left\{\left[s(n) - \hat{s}(n)\right]^2\right\}$$

Fig. 3.1-1 Wiener Filter Problem

We now consider the filter optimization problem that Wiener first solved in the 1940s. Referring to Fig. 3.1-1, we assume the following:

1. The filter input is an additive combination of signal and noise, both of which are jointly wide-sense stationary (JWSS) with known auto- and cross-correlation functions (or corresponding spectral functions).
2. The filter is linear and time-invariant. No further assumption is made as to its form.
3. The output is also wide-sense stationary.
4. The performance criterion is minimum mean-square error.

The estimate $\hat{s}(n)$ of a signal $s(n)$ is given by the convolution representation

$$\hat{s}(n) = h(n) * z(n) = \sum_{i=-\infty}^{\infty} h(n-i)z(i) = \sum_{i=-\infty}^{\infty} h(i)z(n-i), \tag{3.1-1}$$

where $z(i)$ is the measurement and $h(n)$ is the impulse response of the estimator. Let $\mathcal{H}$ denote the region of support of $h(n)$, defined by

$$\mathcal{H} = \{n : h(n) \neq 0\}.$$

Then, Eq. (3.1-1) can be rewritten as

$$\hat{s}(n) = \sum_{i \in \mathcal{H}} h(i) z(n-i). \tag{3.1-2}$$

Let the mean-square estimation error (MSE) J be

$$\begin{aligned} J &= E\left\{[s(n) - \hat{s}(n)]^2\right\} \\ &= E\left\{\left[s(n) - \sum_{i \in \mathcal{H}} h(i) z(n-i)\right]\left[s(n) - \sum_{j \in \mathcal{H}} h(j) z(n-j)\right]\right\} \\ &= E\left\{s^2(n)\right\} - 2\sum_{i \in \mathcal{H}} h(i) E\left\{s(n) z(n-i)\right\} + \sum_{i \in \mathcal{H}} \sum_{j \in \mathcal{H}} h(i) h(j) E\left\{z(n-i) z(n-j)\right\}. \end{aligned} \tag{3.1-3}$$

To minimize the MSE, take the partial derivatives of J with respect to $h(i)$, for each $h(i) \neq 0$. Then, set the result equal to zero

$$\begin{aligned} \frac{\partial J}{\partial h(i)} &= -2E\left\{s(n) z(n-i)\right\} + 2\sum_{j \in \mathcal{H}} h(j) E\left\{z(n-i) z(n-j)\right\} \\ &= 0. \end{aligned} \tag{3.1-4}$$

Solving Eq. (3.1-4), we find

$$\sum_{j \in \mathcal{H}} h(j) E\left\{z(n-i) z(n-j)\right\} = E\left\{s(n) z(n-i)\right\}, \quad i \in \mathcal{H} \tag{3.1-5}$$

which we may express in the form of

$$\sum_{j\in\mathcal{H}} h(j)R_z(j-i) = R_{sz}(i), \quad i \in \mathcal{H}, \tag{3.1-6}$$

where $R_z(k)$ is the autocorrelation function of z(n) and $R_{sz}(k)$ is the crosscorrelation function of s(n) and z(n).

Eq. (3.1-6) is the discrete-time ***Wiener-Hopf equation***. It is the basis for the derivation of the Wiener filter.

4.6. DELTA MODULATION (DM)

Delta Modulation (DM), which is the one-bit (or two level) version of DPCM. It provides a staircase approximation to the oversampled version of an input baseband signal.

Delta Modulation transmits only one bit per sample, i.e., the present sample value is compared with the previous sample value and indication.

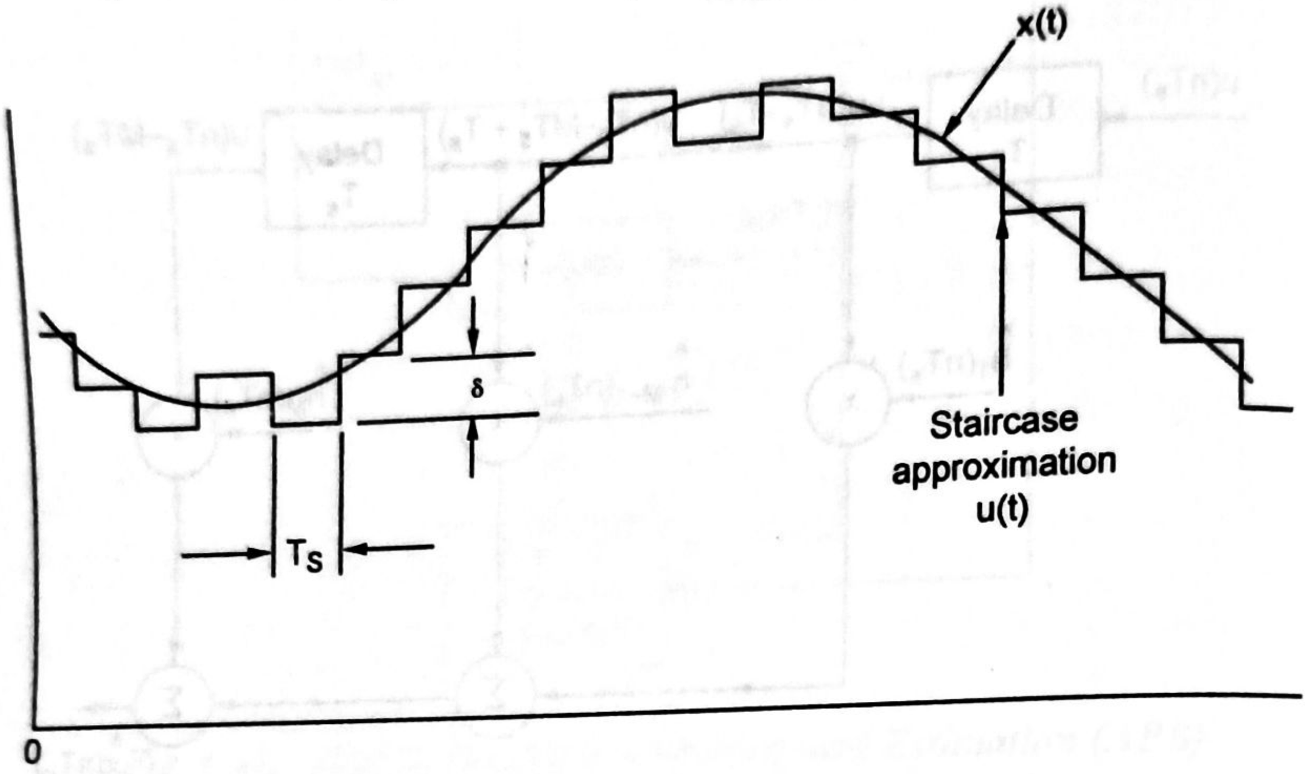


Fig. 4.15.

Input signal $x(t)$ is approximated to step signal by the delta modulator. This step size is fixed. The difference between the input and the approximation is quantized into only two levels namely, $\pm\,\delta$ corresponding to positive and negative differences respectively.

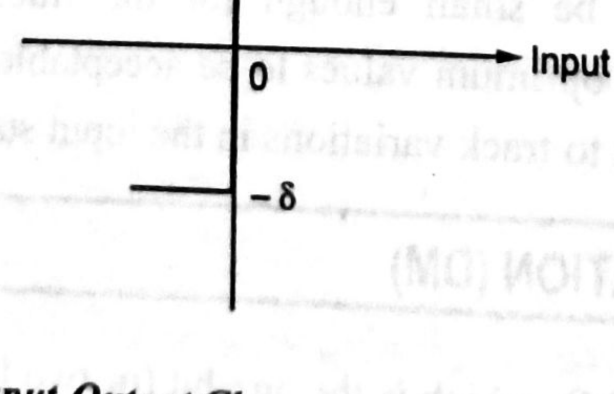


Fig. 4.16. Input-Output Characteristic of two-level quantizes

If the approximation falls below the signal at any sampling epoch, it is increased by δ.

If the approximation lies above the signal, it is diminished by δ.

Thus, the signal does not change too rapidly from sample to sample, there the staircase approximation remains within ± δ of the input signal.

The δ denotes the absolute value of the two representation levels of the one-bit quantizer used in the DM. These two levels are indicated in the transfer characteristics of above Figure.

The step size Δ of the quantizer is related to δ by

$$\Delta = 2\delta \quad \ldots (4.22)$$

It denotes the input signal as $x(t)$ and the staircase approximation to it as $u(t)$. then, the basic principle of delta modulation may be formalized in the following set of discrete time relations.

$$e(n\,T_s) = x(n\,T_s) - \hat{x}(n\,T_s)$$

$$= x(n\,T_s) - u(n\,T_s - T_s) \quad \ldots (4.23)$$

$$b(n\,T_s) = \delta \operatorname{sgn}[e(n\,T_s)] \quad \ldots (4.24)$$

$$u(n\,T_s) = u(n\,T_s - T_s) + b(n\,T_s) \quad \ldots (4.25)$$

where T_s is the sampling period.

$e(n\,T_s)$ is a prediction error representing the difference between the present sample value $x(n\,T_s)$ of the input signal and the latest approximation to it.

The binary quantity $b(n\,T_s)$ is the algebraic sign of the error $e(n\,T_s)$ except for the scaling factor δ. The $b(n\,T_s)$ is the one-bit word transmitted by the DM system.

4.6.1. DM TRANSMITTER

It consists of a summer, a two-level quantizer and an accumulator interconnected. The accumulator is initially set to zero.

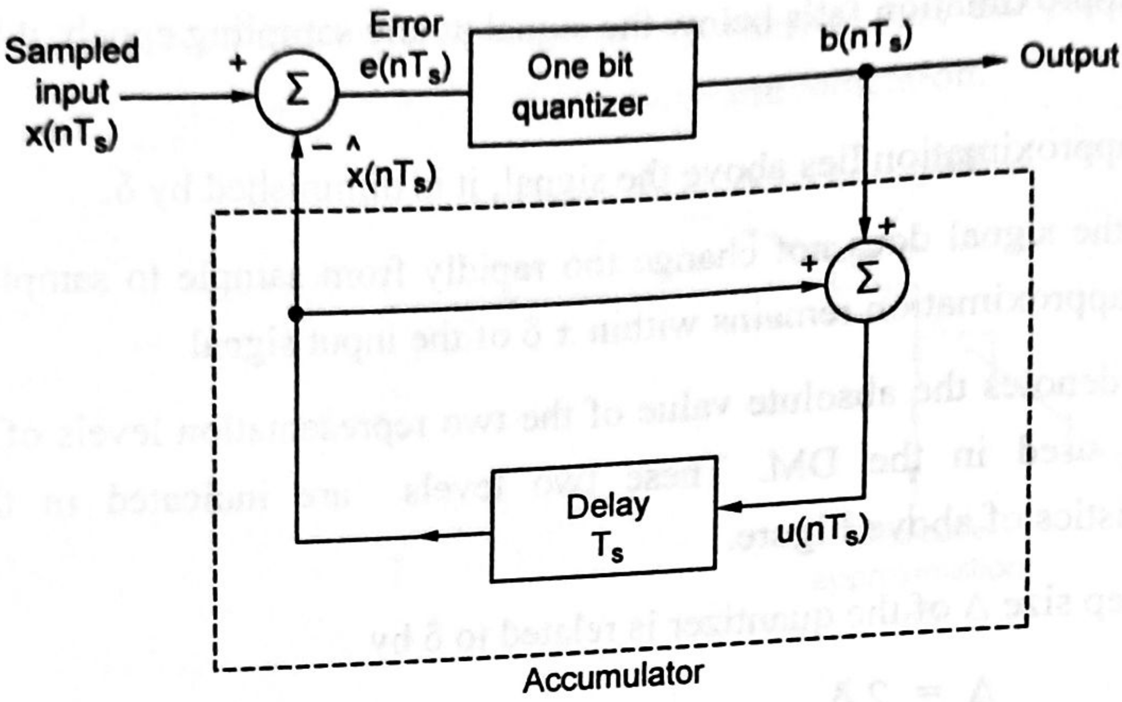


Fig. 4.17. DM transmitter

We may solve equations (4.23) – (4.25) for the accumulator output, obtaining the result.

$$u(n\,T_s) = \delta \sum_{i=1}^{n} \operatorname{sgn}\,[e\,(i\,T_s)]$$

$$= \sum_{i=1}^{n} b\,(i\,T_s) \quad \ldots (4.26)$$

Thus, at each sampling instant, the accumulator increments the approximation to the input signal by $\pm\,\delta$, depending on the binary output of the modulator. Indeed, the accumulator does the best it can to track the input by an increment $+\,\delta$ or $-\,\delta$ at a time.

4.6.2. DM RECEIVER

The staircase approximation $u(t)$ is reconstructed by passing the incoming sequence of positive and negative pulses through an accumulator.

The out-of-band quantization noise in the high-frequency staircase waveform $u(t)$ is rejected by passing it through a low-pass filter with a bandwidth equal to the original signal bandwidth.

By comparing DM with DPCM, a delta modulation is a special case of differential pulse code modulation except for an output low-pass filter.

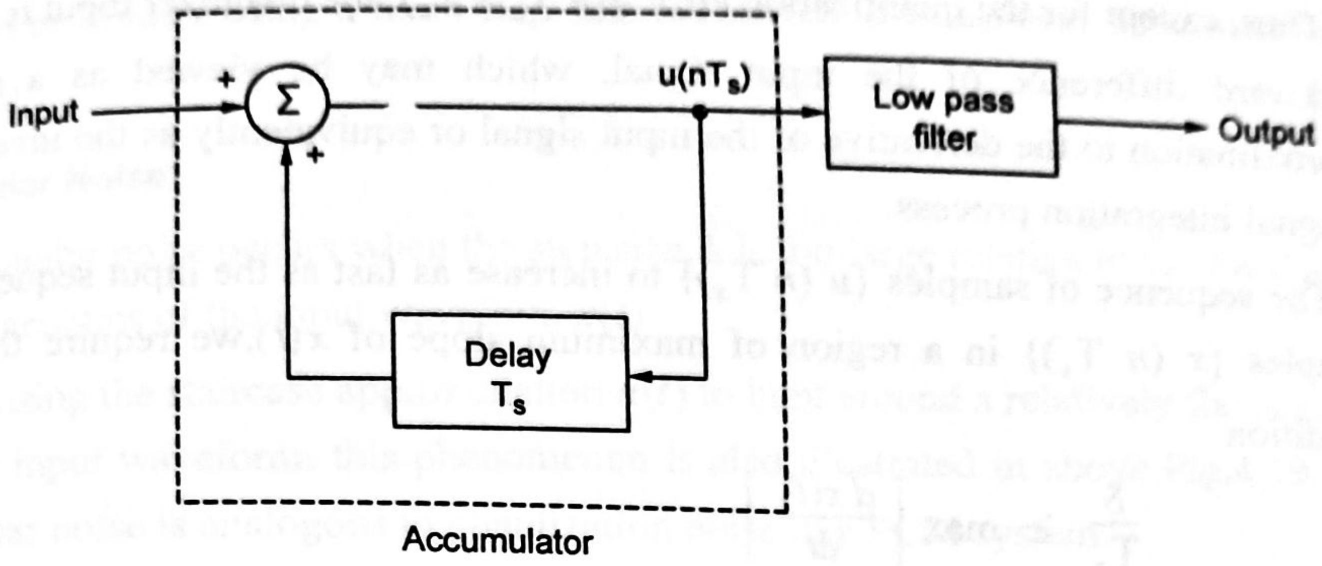


Fig. 4.18. DM Receiver

4.6.3. FEATURES OF DELTA MODULATION

Delta modulation offers two unique features

1. DM has a one-bit-code word for the output, which eliminates the need for word framing.
2. DM has the simplicity of design for both the transmitter and receiver.

These two features make the use of delta modulation attractive for some types of digital communications and for digital voice storage.

4.6.4. QUANTIZATION NOISE

Delta modulation has two types of quantization error.

1. Slope-overload distortion
2. Granular noise

Slope Overload Distortion

Let $q(n\,T_s)$ denote the quantizing error.

$$u(n\,T_s) = x(n\,T_s) + q(n\,T_s) \qquad \dots (4.27)$$

By rearranging the above equation to express the prediction error $e(n\,T_s)$ as

$$e(n\,T_s) = x(n\,T_s) - x(n\,T_s - T_s) - q(n\,T_s - T_s) \quad \ldots (4.28)$$

Thus, except for the quantization error $q(n\,T_s - T_s)$, the quantizer input is a first backward difference of the input signal, which may be viewed as a digital approximation to the derivative of the input signal or equivalently as the inverse of a digital integration process.

The sequence of samples $\{u\,(n\,T_s)\}$ to increase as fast as the input sequence of samples $\{x\,(n\,T_s)\}$ in a region of maximum slope of $x(t)$,we require that the condition

$$\frac{\delta}{T_s} \geq \max\left|\frac{d\,x(t)}{dt}\right| \quad \ldots (4.29)$$

be satisifed. Otherwise, we find that the step size $\Delta = 2\delta$ is too small for the staircase approximation $u(t)$ to follow a steep segment of the input waveform $x(t)$, with the result $u(t)$ falls behind $x(t)$, as illustrated in below Fig.4.19. This condition is called slope-overload and the resulting quantization error is called slope-overload distortion (noise).

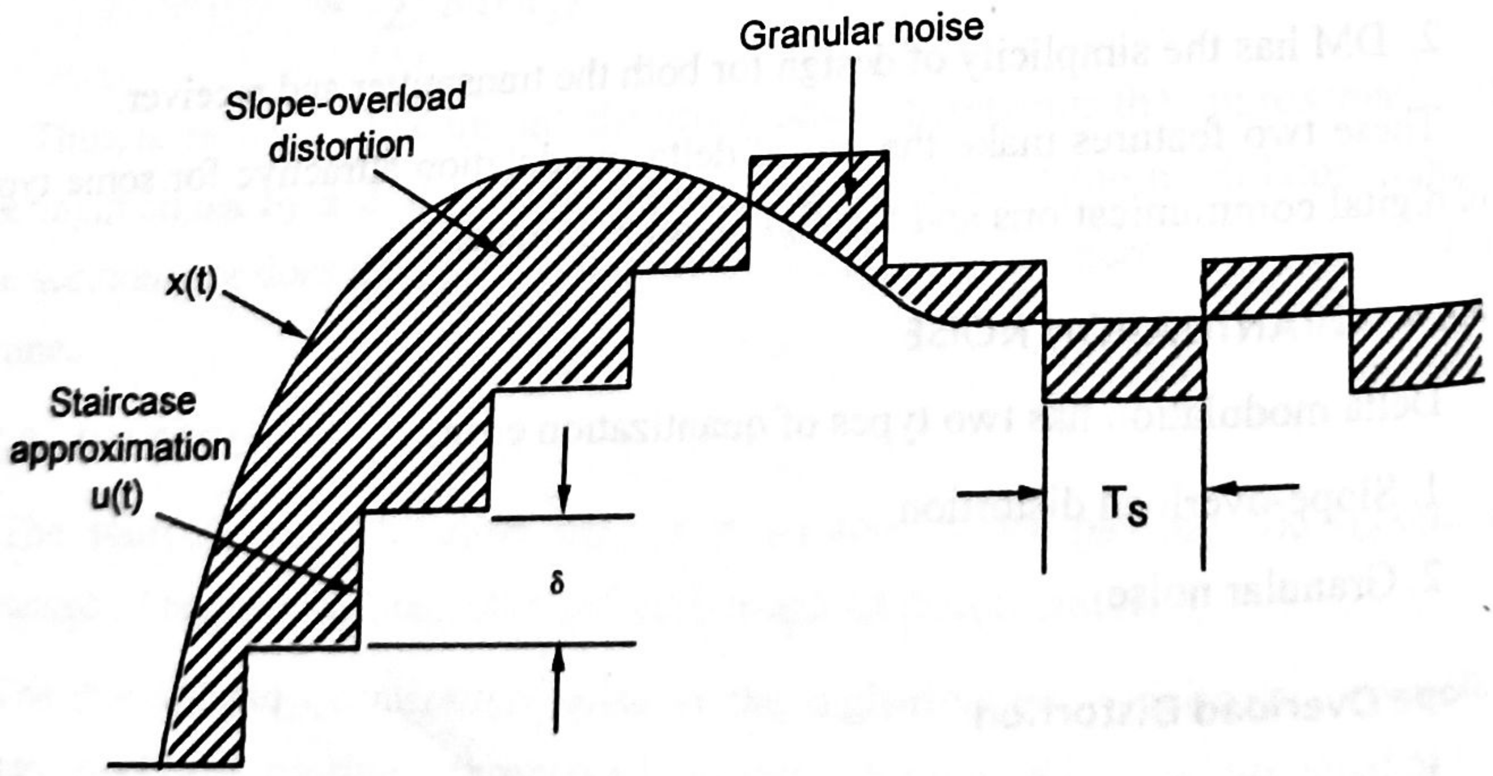


Fig. 4.19.

The maximum slope of the staircase approximation $u(t)$ is fixed by the step size Δ, increases and decreases in $u(t)$ tend to occur along straight lines. For this reason, a delta modulator using a fixed step size is referred to as a linear delta modulator (LDM).

Granular Noise

Granular noise occurs when the step size Δ is too large relative to the local slope characteristics of the input waveform $x(t)$.

Causing the staircase approximation $u(t)$ to hunt around a relatively flat segment of the input waveform; this phenomenon is also illustrated in above Fig.4.19. The granular noise is analogous to quantization noise in a PCM system.