RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL

New Scheme Based On AICTE Flexible Curricula

Electronics & Communication Engineering V-Semester

EC502 DIGITAL COMMUNICATION

Unit I

Sampling theorem for low pass and band pass signals, Ideal sampling, Natural sampling, Flat top sampling, crosstalk, aliasing, time division multiplexing, PAM, PWM and PPM their generation and detection.

Unit II

Pulse code modulation, Quantization, quantization noise, companding, Inter symbol interference, Eye pattern, Delta and adaptive modulation, Encoding techniques: On-Off signaling, Polar signaling, RZ signaling, Bipolar signaling, AMI, Manchester code, Differential encoding their advantage and disadvantages.

Unit III

Band pass data transmission: ASK, Binary phase shift keying (BPSK), QPSK, DPSK, coherent and non coherent BFSK, minimum shift keying, QAM, Concept of M-ary PSK and M-ary FSK. Spectral properties of QPSK and MSK.

UNIT IV

Matched filter and correlator detector. Gram Schmidt orthogonalization procedure and concept of signal space for the computation of probability of error, calculation of error probability for BPSK, QPSK, QAM and coherent BFSK, comparison of different modulation techniques.

Unit V

Concept of information theory, entropy, information rate, channel capacity, Shannon's theorem, Shannon Hartley theorem , BW and signal to noise ratio trade off, sources encoding, extension of zero memory source, Error correcting codes**:** linear block codes and cyclic codes: encoder and decoder circuits, burst error correcting codes, concept of convolution codes.

Reference Books:

1. Communication Systems –Simon Haykins, Wiley
2. Principle of Communication Systems-Taub and Schilling, Tata McGraw-Hill
3. Communication Systems-Singh and Sapre, Tata McGraw-Hill

Unit 1

Syllabus: Random variables Cumulative distribution function, Probability density function, Mean, Variance and standard deviations of random variable, Gaussian distribution, Error function, Correlation and autocorrelation, Central-limit theorem, Error probability, Power Spectral density of digital data.

1.1 Random Variable:

The outcome of an event may always need not be numbers. It may be the name of the horse in an event etc. An experiment whose outcome cannot be predicted exactly, and hence is random, is called a random experiment.

The collective outcomes of a random experiment form a sample space. A particular outcome is called a sample point or sample. Collection of outcomes is called an event. Thus an event is a subset of sample space.

A Random Variable is a real valued function defined over the sample space of random experiments. The random variables are denoted by uppercase letters such as X, Y etc and the values assumed by them are denoted by lower case letters with subscript such as x_1, x_2, y_1, y_2 etc.

Random variables are classified as discrete or continuous.

If in any finite interval *X(λ)* assumes only a finite number of distinct values then the random variable is discrete. Ex. Tossing a die.

If in any finite interval *X(λ)* assumes continuous values then the random variable is continuous. For Example, shift in the magnitude of miss of a bullet due to wind.

A random variable is a mapping from sample space Ω to a set of real numbers. What does this mean? Let's take the usual evergreen example of "flipping a coin".

In a "coin-flipping" experiment, the outcome is not known prior to the experiment, that is we cannot predict it with certainty. But we know the all possible outcomes – Head or Tail. Assign real numbers to the all possible events, say "0" to "Head" and "1" to "Tail", and associate a variable "could take these two values. This variable "X" is called a random variable, since it can randomly take any value '0' or '1' before performing the actual experiment.

Obviously, we do not want to wait till the coin-flipping experiment is done. Because the outcome will lose its significance, we want to associate some probability to each of the possible event. In the coin-flipping experiment, all outcomes are equally probable. This means that we can say that the probability of getting Head as well that of getting Tail is 0.5.

This can be written as,

$$P(X = 0) = 0.5 \; and \; P(X = 1) = 0.5$$

1.2 Cumulative Distribution Function:

The cumulative distributive function or distribution function for a discrete random variable is defined as

$$F(X) = P(X \leq x) = \sum_{u \leq x} f(u) \quad -\infty < x < \infty$$

If X can take on the values x_1, x_2, x_3, x_4 … x_n, then the distributive function is given by

$$F(x) = \begin{cases} 0 & -\infty \leq x < x_1 \\ f(x_1) & x_1 \leq x < x_2 \\ f(x_1) + f(x_2) & x_2 \leq x < x_3 \\ \vdots & \\ f(x_1) + f(x_2) + \cdots + f(x_n) & x_n \leq x < \infty \end{cases}$$

Mathematically, a complete description of a random variable is given be "Cumulative Distribution Function"- $F_{\mathbf{X}}(x)$. Here the bold faced **"X"** is a random variable and "x" is a dummy variable which is a place holder for all possible outcomes. The Cumulative Distribution Function is defined as,

$$f_X(x) = P(X \leq x)$$

If we plot the CDF for our coin-flipping experiment, it would look like the one shown in the figure

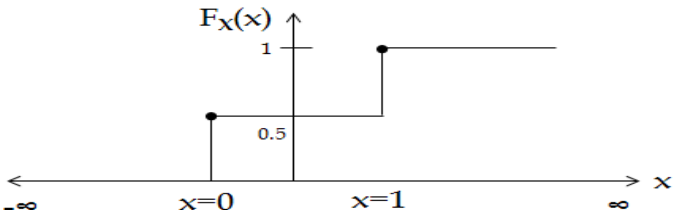


Figure 1.01 Cumulative Distribution Function

The example provided above is of discrete nature, as the values taken by the random variable are discrete and therefore the random variable is called Discrete Random Variable.
If the values taken by the random variables are of continuous nature, then the random variable is called Continuous Random Variable and the corresponding cumulative distribution function will be smoother without discontinuities.

1.3 Probability Distribution function:

Consider an experiment in which the probability of events is as follows. The probabilities of getting the numbers 1,2,3,4 individually are 1/10, 2/10,3/10,4/10 respectively. It will be more convenient for us if we have an equation for this experiment which will give these values based on the events. For example, the equation for this experiment can be given by f(x)=x/10 where x=1,2,3,4. This equation is called probability distribution function.

1.4 Probability Density function (PDF) and Probability Mass Function (PMF):

It's more common deal with Probability Density Function (PDF)/Probability Mass Function (PMF) than CDF. The PDF is given by taking the first derivate of CDF.

$$f_X(x) = \frac{d\mathrm{F}_X(x)}{dx}$$

For discrete random variable that takes on discrete values, is it common to defined Probability Mass Function.

$$f_X(x) = P(X = x)$$

The previous example was simple. The problem becomes slightly complex if we are asked to find the probability of getting a value less than or equal to 3. Now the straight forward approach will be to add the probabilities of getting the values x=1,2,3 which comes out to be 1/10+2/10+3/10 =6/10. This can be easily modeled as a probability density function which will be the integral of probability distribution function with limits 1 to 3.
Based on the probability density function or how the PDF graph looks, PDF fall into different categories like binomial distribution, Uniform distribution, Gaussian distribution, Chi-square distribution, Rayleigh distribution, Rician distribution etc. Out of these distributions, you will encounter Gaussian distribution or Gaussian Random variable in digital communication very often.

The PDF has the following properties:

a) $f_x(x) \geq 0 \quad for\ all\ x$

This results from the fact that F(x) increases monotonically as x increases, more outcomes are included in the probability of occurrence represented by F(x).

b) $\int_{-\infty}^{\infty} f_x(x)dx = 1$

This result can be seen from the fact that

$$\int_{-\infty}^{\infty} f_x(x)dx = 1 \quad F(\infty) - F(-\infty) = 1 - 0 = 1$$

c) $F(x) = \int_{-\infty}^{x} f_x(x)dx$

This result from the integration of the definition of PDF.

1.5 Mean:
The mean of a random variable is defined as the weighted average of all possible values the random variable can take. Probability of each outcome is used to weight each value when calculating the mean. Mean is also called expectation (E[X])
For continues random variable X and probability density function $f_X(x)$

$$E[X] = \int_{-\infty}^{\infty} x f_X(x)dx$$

For discrete random variable X, the mean is calculated as weighted average of all possible values (x_i) weighted with individual probability (p_i)

1.6 Variance:
Variance measures the spread of a distribution. For a continuous random variable X, the variance is defined as

$$var[X] = \int_{-\infty}^{\infty} (x - E[X])^2 f_X(x)dx$$

For discrete case, the variance is defined as

$$var[X] = \sigma^2 X = \sum_{-\infty}^{\infty} (x - \mu X)^2 pi$$

Standard Deviation (σ) is defined as the square root of variance $\sigma^2 X$

Properties of Mean and Variance:
For a constant – "c" following properties will hold true for mean

$$E[cX] = cE[X]$$
$$E[X + c] = E[X] + c$$
$$E[c] = c$$

For a constant – "c" following properties will hold true for variance

$$var[cX] = c^2 \mathrm{var}\,[X]$$
$$var[X + c] = \mathrm{var}\,[X]$$
$$var[c] = 0$$

PDF and CDF define a random variable completely. For example: If two random variables X and Y have the same PDF, then they will have the same CDF and therefore their mean and variance will be same. On the other hand, mean and variance describes a random variable only partially. If two random variables X and Y have the same mean and variance, they may or may not have the same PDF or CDF.

1.7 Binomial, Poisson and Normal (Gaussian) Distributions

The most important probability distributions are Binomial, Poisson and Normal (Gaussian). Binomial and Poisson distributions are for discrete random variables, whereas the Normal distribution is for continuous random variables.

1.7.1 Binomial Distribution:

Let us consider an event with only two possible outcomes. Such an experiment is known as Bernoulli trial.) One outcome is called success and other is called failure. Let the experiment is repeated a number of times. Let us consider that the probability of success $\boldsymbol{p}$ in each trial is same, i.e. the trials are independent. Then the probability of failure in each trial is $\boldsymbol{q = (1 - p)}$. The probability of x successes in n trials is given by a probability function known as Binomial distribution:

$$\boldsymbol{f(x) = P(X = x) = \binom{n}{x} p^x q^{n-x}} \quad \text{...1.7.1.1}$$

The properties of Binomial distribution are

$$\boldsymbol{Mean = np, variance = npq}$$

Example 1.7.1.1

A fair die is tossed 5 times. A toss is called a success if face 1 or 6 appears. Find (a) the probability of two successes, (b) the mean and the standard deviation for the number of successes.

(a)

$$\boldsymbol{N = 5, p = \frac{2}{6} = \frac{1}{3}, q = 1 - p = 1 - \frac{1}{3} = \frac{2}{3}}$$

$$\boldsymbol{P(X = 2) = \binom{5}{2}\left[\frac{1}{3}\right]^2 \left[\frac{2}{3}\right]^{5-2} = \frac{80}{243}} \qquad \boldsymbol{\binom{n}{x} = \frac{n!}{x!\,(n-x)!}}$$

(b)

$$\boldsymbol{Mean = np = 5 \times \frac{1}{3} = 1.667}$$

$$\boldsymbol{Standard\ deviation = \sqrt{npq} = \sqrt{5 \times \frac{1}{3} \times \frac{2}{3}} = 1.054}$$

1.7.2 Poisson Distribution:

Let X be a discrete random variable that can assume values 0, 1, 2… Then the probability function of X is given by Poisson distribution:

$$\boldsymbol{f(x) = P(X = x) = \frac{\lambda^x e^{-\lambda}}{x!}} \quad \text{...1.7.2.1}$$

$$\boldsymbol{x = 0, 1, 2, \ldots}$$

Where, $\boldsymbol{\lambda}$ is a positive constant. The properties of Poisson distribution are

$$\boldsymbol{Mean = \lambda, variance = \lambda}$$

1.7.3 Poisson Approximation to Binomial Distribution:

In Binomial distribution if n is large and p is close to zero, then it can be approximated by Poisson distribution with $\boldsymbol{\mu = np}$. In practice $\boldsymbol{n \geq 50}$ and $\boldsymbol{np \leq 5}$ give satisfactory approximation.

It can be seen from the equation of binomial distribution that if n is large, then calculation of a desired probability considering Binomial distribution is tedious. On the other hand, calculation of a desired probability considering Poisson distribution is fairly simple as seen from equation 1.7.2.1.

1.7.4 Normal (or Gaussian) distribution

Gaussian PDF looks like a bell. It is used most widely in communication engineering. This is the most important continuous probability distribution as most of the natural phenomenon are characterized by random variables with normal distribution. The importance of normal distribution is further enhanced because of the central limit theorem. The density function (PDF) for normal (Gaussian) distribution is given by

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-(x-\mu)^2}{2\sigma^2}} \quad -\infty < x < \infty \quad \text{...1.7.4.1}$$

Where μ and σ are mean and standard deviation, respectively.

The properties of the normal distribution are

$$\boldsymbol{Mean = \mu, variance = \sigma^2}$$

The corresponding distribution function is

$$F(x) = P(X \leq x) = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{x} e^{-(v-u)^2/2\sigma^2}\, dv$$

$$= \frac{1}{2} + \frac{1}{\sigma\sqrt{2\pi}} \int_{0}^{x} e^{-(v-u)^2/2\sigma^2}\, dv \quad \text{...1.7.4.2}$$

Let Z be the standardized random variable corresponding to X. Thus if $Z = \frac{x-\mu}{\sigma}$, then the mean of Z is zero and its variance is 1. Hence

$$F(z) = \frac{1}{\sqrt{2\pi}} e^{-z^2/2} \quad \text{...1.7.4.3}$$

$F(z)$ is known as standard normal density function. The corresponding distribution function is

$$F(z) = P(Z \leq z) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{z} e^{-u^2/2}\, du = \frac{1}{2} + \frac{1}{\sqrt{2\pi}} \int_{0}^{z} e^{-u^2/2}\, du \quad \text{...1.7.4.4}$$

The integral of equation 1.7.4.4 is not easily evaluated. However, it is related to the error function, whose tabulated values are available in mathematical tables.

Error Function

The **error function** of z is defined as

$$erf\ z = \frac{2}{\sqrt{\pi}} \int_{0}^{z} e^{-u^2}\, du \quad \text{...1.7.3.5}$$

The error function has the values between 0 and 1.

$$erf\ (0) = 0\ and\ erf\ (\infty) = 1$$

The Complementary **error function** of z is defined as

$$erfc\ z = 1 - erf\ z = \frac{2}{\sqrt{\pi}} \int_{z}^{\infty} e^{-u^2}\, du \quad \text{...1.7.3.6}$$

The relationship between $f(x)$, $erf\ z$ and $erfc\ z$ is as follows

$$f(x) = \frac{1}{2}\left[1 + erf\left(\frac{z}{\sqrt{2}}\right)\right] = 1 - \frac{1}{2}\, erfc\left(\frac{z}{\sqrt{2}}\right) \qquad \text{...1.7.3.7}$$

1.7.5 Normal approximation to Binomial Distribution

If n is large and if neither *p* nor *q* is close to zero, then the binomial distribution can be closely approximated by a normal distribution with standardized random variable given by

$$Z = \frac{X - np}{\sqrt{npq}}$$

In practice $\boldsymbol{np \geq 5}$ and $\boldsymbol{nq \geq 5}$ give the satisfactory approximation.

1.7.6 Normal approximation to Poisson Distribution

As Binomial distribution has relationship with both Poisson and Normal distributions, one would expect that there should be some relationship between Poisson and Normal distributions. In fact it is found to be so. It has been seen that the Poisson distribution approaches Normal distribution as $\boldsymbol{\lambda \to \infty}$.

1.8 Error Function:

In mathematics, the error function is a special function of sigmoid shape that occurs in probability, statistics, and partial differential equations describing diffusion. It is defined as:

$$\mathrm{erf}(x) = \frac{1}{\sqrt{\pi}}\int_{-x}^{x} e^{-t^2}\,dt$$

$$= \frac{2}{\sqrt{\pi}}\int_{0}^{x} e^{-t^2}\,dt$$

In statistics, for nonnegative values of *x*, the error function has the following interpretation: for a random variable *X* that is normally distributed with mean 0 and variance , erf(*x*) describes the probability of *X* falling in the range [−*x*, *x*].

1.9 Central Limit Theorem

Central Limit Theorem (CLT) states that irrespective of the underlying distribution of a, if you take a number of samples of size N from the population, then the “sample mean” follow a normal distribution with a mean of μ and a standard deviation of σ/ sqrt(N).The normality gets better as your sample size **N** increases. Here the underlined word signifies that irrespective of the base distribution the probability distribution curve will approach Gaussian or Normal distribution as the number of sample increases. In other words, CLT states that the sum of independent and identically distributed random variables approaches Normal distribution as $N \geq \infty$.

Applications of Central Limit Theorem:

CLT is being applied in vast range of applications including Signal processing, channel modeling, random process, population statistics, engineering research, predicting the confidence intervals, hypothesis testing, even in Casino and gambling , etc…,

One such application is deriving the response of a cascaded series of Low pass filters by applying Central limit theorem **.**The author has illustrated how the response of a cascaded series of low pass filters approaches Gaussian shape as the number of filters in the series increases.

In digital communication, channel noise is often modeled as normally distributed. Modeling a channel as normally distributed when the noise components in that channel are sufficiently large is justified by Central limit theorem.

1.10 Correlation

The Correlation (more precisely Cross Correlation) between two waveforms is the measure of similarity between one waveform and time delayed version of the other waveform. This express how much one waveform is related to the time delayed version of the other waveform.

The expression for correlation is very close to convolution. Consider two general complex functions $f_1(t)$ and $f_2(t)$, which may or may not be periodic, and not restricted, to finite interval. Then cross correlation or simply correlation $R_{1,2}(r)$ between two function is defined as follows:

$$R_{1,2}(r) = \lim_{c\to\infty} \int_{-\frac{T}{2}}^{\frac{T}{2}} f_1(t) f_2^*(t+r)dt \quad \text{...1.10.1a}$$

(The *conjugate symbol**, is removed if the function is real.)

This represents the shift of function $f_2(t)$ by an amount $-r$(i.e. towards left). A similar effect can be obtained by shifting $f_1(t)$ by an amount $+r$ (i.e. towards right). Therefore correlation may also be defined as

$$R_{1,2}(r) = \lim_{c\to\infty} \int_{-\frac{T}{2}}^{\frac{T}{2}} f_1(t-r) f_2^*(t)dt \quad \text{...1.10.1b}$$

Let us define the correlation for two cases, (i) Energy (non periodic) signal and (ii) Power (Periodic) Signals.
In the definition of correlation limit of integration may be taken as infinite for energy signals,

$$R_{1,2}(r) = \int_{-\infty}^{\infty} f_1(t) f_2^*(t+r)dt = \int_{-\infty}^{\infty} f_1(t-r) f_2^*(t)dt \quad \text{...1.10.2a}$$

For power signals of period T_0, the definition in above equation may not converge. Therefore the average correlation over a period T_0 is defined as

$$R_{1,2}(r) = \frac{1}{T_0}\int_{-\frac{T_0}{2}}^{\frac{T_0}{2}} f_1(t) f_2^*(t+r)dt = \frac{1}{T_0}\int_{-\frac{T_0}{2}}^{\frac{T_0}{2}} f_1(t-r) f_2^*(t)dt \quad \text{...1.10.2b}$$

The correlation definition represents the overlapping area between the two functions.

1.11 Auto Correlation Function

Auto correlation is a special form of cross correlation. It is defined as correlation of a function with itself. Auto correlation function is a measure of similarity between a signal & its time delayed version. It is represented with $R(r)$.

Consider a signal $f(t)$. The auto correlation function of $f(t)$ with its time delayed version is given by

$$R_{11}(r) = R(r) = \lim_{T\to\infty} \frac{1}{T}\int_{-\frac{T}{2}}^{\frac{T}{2}} f(t) f^*(t+r)dt \quad \text{(+ve Shift)}$$

$$= \lim_{T\to\infty} \frac{1}{T}\int_{-\frac{T}{2}}^{\frac{T}{2}} f(t-r) f^*(t)dt \quad \text{(-ve Shift)}$$

Properties of auto correlation

Auto correlation exhibits conjugate symmetry i.e. ($R(r) = R*(-r)$

Auto correlation function of energy signal at origin i.e. at $r = 0$ is equal to total energy of that signal, which is given as:

$$R(0) = \int_{-\infty}^{\infty} |x(t)|^2 dt$$

Auto correlation function $\infty \frac{1}{T}$,

Auto correlation function is maximum at $r = 0$ $i.e. |R(r)| \leq R(0) \forall r$

Auto correlation function and energy spectral densities are Fourier transform pairs. i.e.

$$F.T. |R(r)| = \Psi(\omega)$$

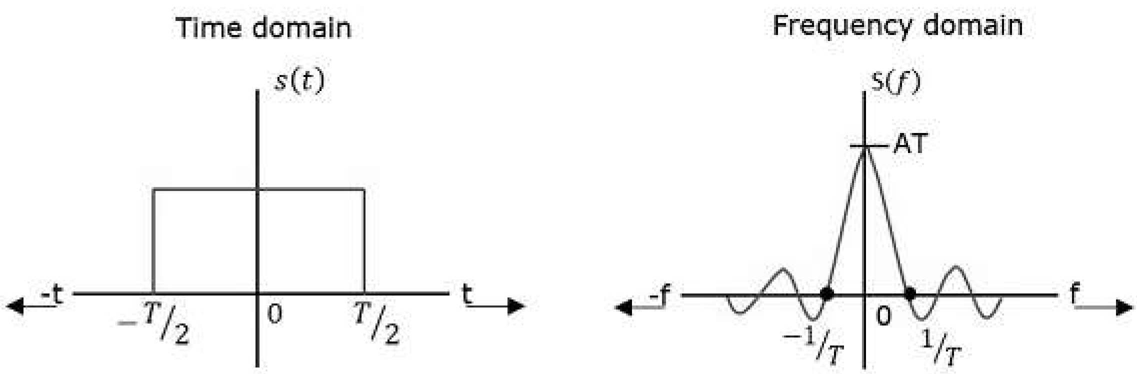
$$\Psi(\omega) = \int_{-\infty}^{\infty} R(r) e^{-j\omega t} dr$$

$$R(r) = x(t) * x(-t)$$

1.12 Power Spectral Density

The function which describes how the power of a signal got distributed at various frequencies, in the frequency domain is called as Power Spectral Density (PSD).

PSD is the Fourier Transform of Auto-Correlation. It is in the form of a rectangular pulse.



$$s(t) = A \; |t| < \frac{T}{2}$$

$$s(t) = 0 \; |t| > \frac{T}{2}$$

$$s(f) = AT \frac{\sin(\pi T f)}{\pi T f}$$

Figure 1.02 Power Spectral Density

1.13 Power Spectral Density Derivation

According to the Einstein-Wiener-Khintchine theorem, if the auto correlation function or power spectral density of a random process is known, the other can be found exactly.

Hence, to derive the power spectral density, we shall use the time auto-correlation ($Rx(\tau)$) of a power signal $x(t)$ as shown below.

$$R_x(r) = \lim_{T_P \to \infty} \frac{1}{T_P} \int_{\frac{-T_P}{2}}^{\frac{T_P}{2}} x(t) x(t + r) \, dt$$

Since $x(t)$ consists of impulses, $Rx(\tau)$ can be written as

$$R_x[r] = \frac{1}{T} \sum_{n=-\infty}^{\infty} R_n \delta(r - nT)$$

Where $R_n = \lim_{N \to \infty} \frac{1}{N} \sum_k a_k a_{k+n}$

Getting to know that $Rn=R-n$ for real signals, we have

$$S_x(\omega) = \frac{1}{T}\left(R_0 + 2\sum_{n=1}^{\infty} R_n cos n\omega T\right)$$

Since the pulse filter has the spectrum of $(w) \leftrightarrow f(t)$, we have

$$S_y(\omega) = |F(\omega)|^2 S_x(\omega)$$

$$= \frac{|F(\omega)|^2}{T}\left(\sum_{n=\infty}^{\infty} R_n e^{-jnuT_b}\right)$$

$$= \frac{|F(\omega)|^2}{T}\left(R_0 + 2\sum_{n=1}^{\infty} R_n cos n\omega T\right)$$

Hence, we get the equation for Power Spectral Density. Using this, we can find the PSD of various line codes.

Unit 2

Digital conversion of Analog Signals: Sampling theorem, sampling of band pass signals, Pulse Amplitude Modulation (PAM), types of sampling (natural, flat-top), equalization, signal reconstruction and reconstruction filters, aliasing and anti-aliasing filter, Pulse Width Modulation (PWM), Pulse Position Modulation (PPM)
Digital transmission of Analog Signals: Quantization, quantization error, Pulse Code Modulation (PCM), companding, scrambling, TDM-PCM, Differential PCM, Delta modulation, Adaptive Delta modulation, vocoder.

PART I DIGITAL CONVERSION OF ANALOG SIGNALS

2.1 Sampling of Analog Signals

Sampling is defined as, "The process of measuring the instantaneous values of continuous-time signal in a discrete form." In the process of sampling an analog signal is converted into a corresponding sequences of samples, that are uniformly spaced in time.
This discretization of analog signal is called as Sampling. The following figure indicates a continuous-time signal x (t) and a sampled signal x_s (t). When x (t) is multiplied by a periodic impulse train, the sampled signal x_s (t) is obtained.

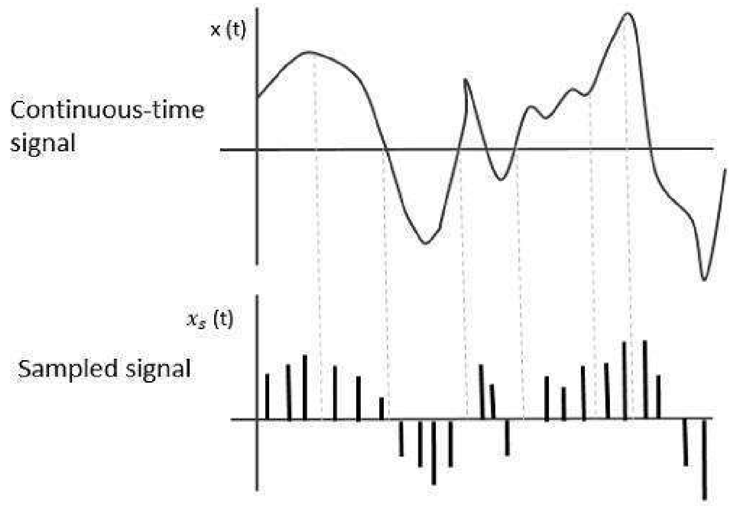


Figure 2.01 Sampling

2.1.1 Sampling Rate

To discretize the signals, the gap between the samples should be fixed. That gap can be termed as a sampling period Ts.

$$Sampling\ Frequency = 1/Ts = fs$$

Where,

$$Ts = sampling\ time$$
$$fs = sampling\ frequency\ or\ the\ sampling\ rate$$

Sampling frequency is the reciprocal of the sampling period. This sampling frequency can be simply called as Sampling rate. The sampling rate denotes the number of samples taken per second, or for a finite set of values.

For an analog signal to be reconstructed from the digitized signal, the sampling rate should be highly considered. The rate of sampling should be such that the data in the message signal should neither be lost nor it should get over-lapped. Hence, a rate was fixed for this, called as Nyquist rate.

2.1.2 Signals Sampling Techniques

There are three types of sampling techniques:

a. Impulse sampling.
b. Natural sampling.
c. Flat Top sampling.

(a) Impulse Sampling

Impulse sampling can be performed by multiplying input signal x(t) with impulse train of period 'T'. Here, the amplitude of impulse changes with respect to amplitude of input signal x(t). The output of sampler is given by

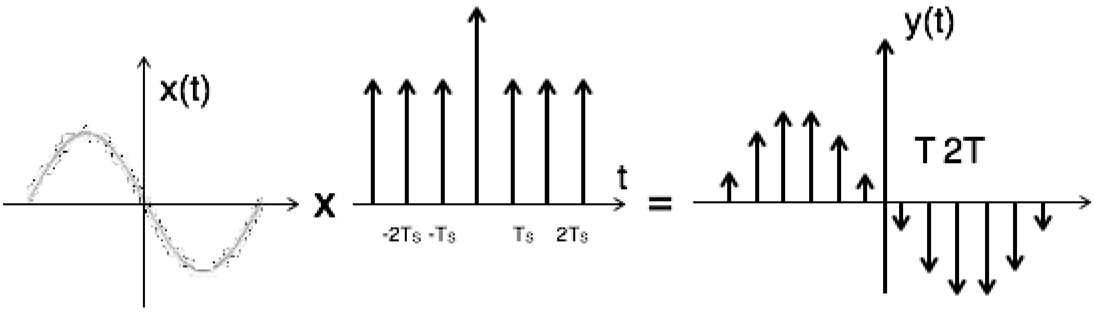


Figure 2.1.2.1 Impulse Sampling

$$y(t) = x(t) \times impulse\ train$$

$$= x(t) \times \sum_{n=-\infty}^{\infty} x(nt)\delta(t-nt)$$

$$y(t) = y_n(t) = \sum_{n=-\infty}^{\infty} x(nt)\delta(t-nt) \dots \qquad \dots(2.1.2.1)$$

To get the spectrum of sampled signal, consider Fourier transform of equation 1 on both sides

$$Y(\omega) = \frac{1}{T_0} \sum_{n=-\infty}^{\infty} X(\omega - n\omega_s)$$

This is called ideal sampling or impulse sampling. You cannot use this practically because pulse width cannot be zero and the generation of impulse train is not possible practically.

(b) Natural Sampling

Natural sampling is similar to impulse sampling, except the impulse train is replaced by pulse train of period T. i.e. you multiply input signal x(t) to pulse train $\sum_{n=-\infty}^{\infty} P(t-nT)$ as shown below

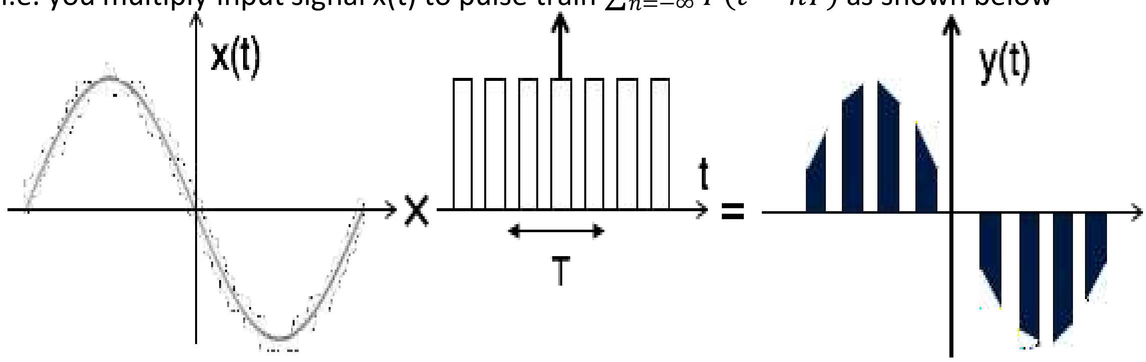


Figure 2.1.2.2 Natural Sampling

$$y(t) = x(t) \times pulse\ train$$
$$= x(t) \times p(t)$$
$$= x(t) \times \sum_{n=-\infty}^{\infty} P(t - nT) \qquad \text{...(2.1.2.2)}$$

The exponential Fourier series representation of p(t) can be given as

$$p(t) = \sum_{n=-\infty}^{\infty} F_n e^{jn\omega_0 t} \qquad \text{...(2.1.2.3)}$$

$$= \sum_{n=-\infty}^{\infty} F_n e^{j2\pi n f_0 t}$$

$$\text{Where } F_n = \frac{1}{T}\int_{-\frac{T}{2}}^{\frac{T}{2}} p(t) e^{jn\omega_0 t}\, dt$$

$$= \frac{1}{TP}(n\omega_s)$$

Substitute F_n value in equation 2.1.2.2

$$\therefore\ p(t) = \sum_{n=-\infty}^{\infty} \frac{1}{T} P(n\omega_s) e^{jn\omega_0 t}$$
$$= \frac{1}{T} \sum_{n=-\infty}^{\infty} P(n\omega_s) e^{jn\omega_0 t}$$

Substitute p(t) in equation 2.1.2.1

$$y(t) = x(t) \times p(t)$$

$$= x(t) \times \frac{1}{T} \sum_{n=-\infty}^{\infty} P(n\omega_s) e^{jn\omega_0 t}$$

$$= \frac{1}{T} \sum_{n=-\infty}^{\infty} P(n\omega_s) x(t) e^{jn\omega_0 t}$$

To get the spectrum of sampled signal, consider the Fourier transform on both sides.

$$F.T.[y(t)] = F.T. \frac{1}{T} \sum_{n=-\infty}^{\infty} [P(n\omega_s) x(t) e^{jn\omega_0 t}]$$
$$= \frac{1}{T} \sum_{n=-\infty}^{\infty} P(n\omega_s) F.T.[x(t) e^{jn\omega_0 t}]$$

According to frequency shifting property

$$F.T.[x(t) e^{jn\omega_0 t}] = X[\omega - n\omega_s]$$
$$\therefore\ Y[\omega] = \frac{1}{T} \sum_{n=-\infty}^{\infty} P(n\omega_s)\, X[\omega - n\omega_s]$$

(c) Flat Top Sampling

During transmission, noise is introduced at top of the transmission pulse which can be easily removed if the pulse is in the form of flat top. Here, the top of the samples are flat i.e. they have constant amplitude. Hence, it is called as flat top sampling or practical sampling. Flat top sampling makes use of sample and hold circuit.

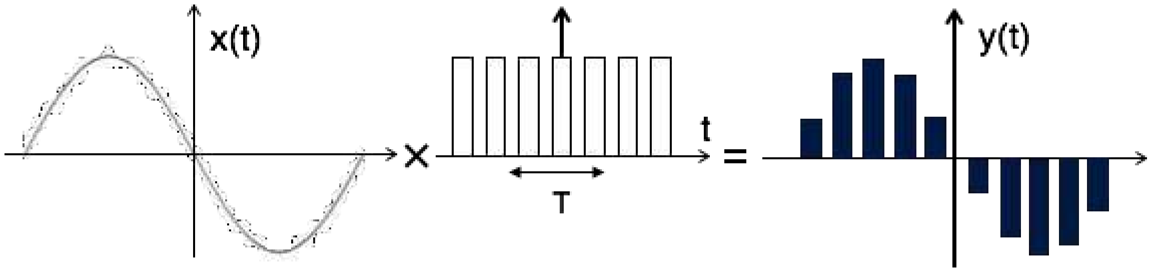


Figure 2.1.2.3 Flat Top Sampling

Theoretically, the sampled signal can be obtained by convolution of rectangular pulse p(t) with ideally sampled signal say $y_\delta(t)$ as shown in the diagram:

i.e. $y(t) = p(t) \times y_\delta(t)$ ...(2.1.2.4)

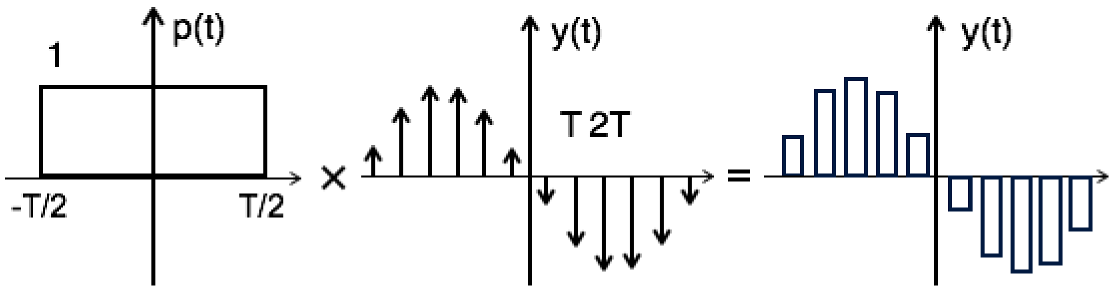


Figure 2.11 Sampling

To get the sampled spectrum, consider Fourier transform on both sides for equation 1

$$Y[\omega] = F.T.[P(t) \times y_\delta(t)]$$

By the knowledge of convolution property,

$$Y[\omega] = P(\omega)Y_\delta(\omega)$$

Here $P(\omega) = TS_a\left(\frac{\omega T}{2}\right) = 2sin\omega T/\omega$

2.1.3 Sampling Theorem

The sampling theorem states that, "a signal whose spectrum is band limited to B Hz, [G(ω) = 0 for $|\omega| > 2\pi B$] can be reproduced exactly from its samples if it is sampled at the rate fs which is greater than twice the maximum frequency ω of the signal to be sampled." Therefore minimum sampling frequency is

$$f_s = 2B\ Hz$$

Proof: Consider a continuous time signal x(t). The spectrum of x(t) is a band limited to f_m Hz i.e. the spectrum of x(t) is zero for $|\omega|>\omega_m$.

Sampling of input signal x(t) can be obtained by multiplying x(t) with an impulse train δ(t) of period T_s. The output of multiplier is a discrete signal called sampled signal which is represented with y(t) in the following diagrams:

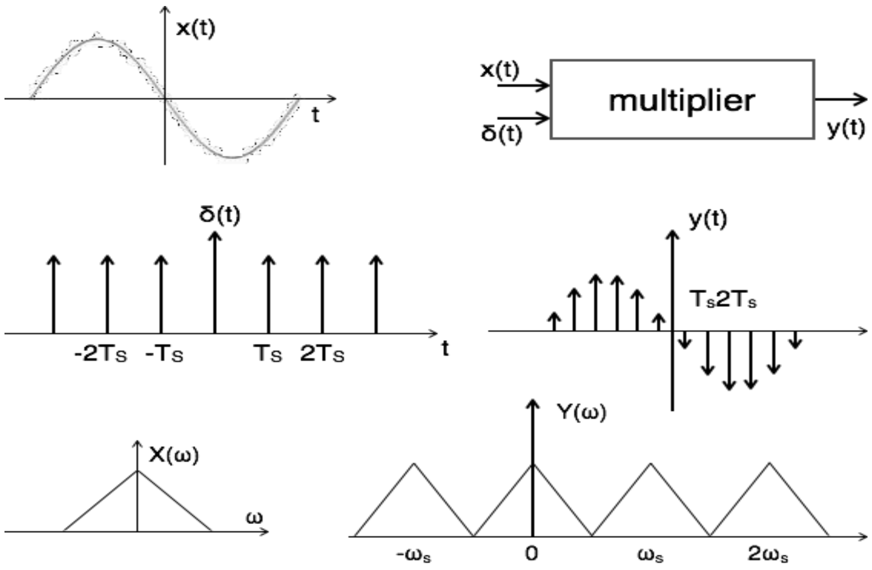


Figure 2.1.3.1 Sampling of signal x(t)

Here, you can observe that the sampled signal takes the period of impulse. The process of sampling can be understood as under.

The sampled signal is given by

$$y(t) = x(t).\,\delta(t) = x(t).\,\delta T_s(t)$$

$$= \sum_n x(nT_s).\,\delta(t - nT_s) \qquad \ldots.(2.1.3.1)$$

The impulse train $\delta T_s(t)$ is a periodic signal of period T_s, hence it can be expressed as a Fourier series

$$\delta T_s(t) = \frac{1}{T_s}\,[1 + 2\,cos\omega_s t + 2\,cos2\omega_s t + \ 2\,cos3\omega_s t + \cdots]$$

Where $\omega_s = \frac{2\pi}{T_s} = 2\pi f_s$ $\qquad \ldots.(2.1.3.2)$

Substitute $\delta(t)$in equation 1.

$$\rightarrow y(t) = x(t).\,\delta(t)$$

$$y(t) = \frac{1}{T_s}[x(t) + 2x(t)\,cos\omega_s t + \ 2x(t)\,cos2\omega_s t + \ 2\,x(t)cos3\omega_s t + \cdots]$$

Now to find the $Y(\omega)$, we have to take the fourier transform of both the sides,

$$Y(\omega) = \frac{1}{T_s}[X(\omega) + X(\omega - \omega_s) + X(\omega + \omega_s) + X(\omega - 2\omega_s) + X(\omega + 2\omega_s) + \cdots]$$

$$Y(\omega) = \frac{1}{T_s}\sum_{n=-\infty}^{\infty} X(\omega - n\omega_s) \text{ Where } n = 0, \pm 1, \pm 2, \ldots$$

The Fourier spectrum $Y(\omega)$ is shown in figure 2.1.3.1. Now If we have to recover $x(t)$ from $y(t)$, we should be able to recover $X(\omega)$ from $Y(\omega)$, and it is possible only if there is no overlapping between the successive cycles of $Y(\omega)$, and for this condition

$$f_s > 2B\ Hz$$

Therefore the sampling interval

$$T_s < \frac{1}{2B}$$

Therefore as long as the sampling frequency f_s is greater than $2B$, $Y(\omega)$, will consist of non overlapping repetitions of $X(\omega)$, and $x(t)$ can be recovered from $y(t)$ by passing $y(t)$ by an ideal low pass filter with cut off frequency B Hz.

Nyquist Rate

The minimum sampling rate $f_s = 2B$ required to recover $x(t)$ from its samples $y(t)$ is called the Nyquist rate and the corresponding sampling interval is called Nyquist Interval for $y(t)$.

Effect of Sampling Rate on $Y(m)$

Possibility of sampled frequency spectrum with different conditions is given by the following diagrams:

(a) Over Sampling

(b) Under Sampling

(c) Perfect sampling

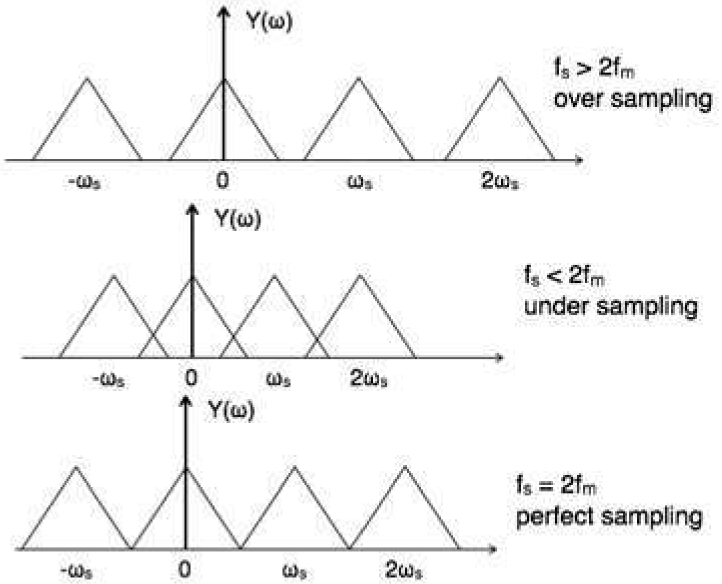


Figure 2.1.3.2 Sampling Conditions

Aliasing Effect

The overlapped region in case of under sampling represents aliasing effect, which can be removed by

- considering $fs > 2fm$
- By using anti-aliasing filters.

2.2 Sampling of Band Pass Signals

In case of band pass signals, the spectrum of band pass signal X[ω] = 0 for the frequencies outside the range $f_1 \leq f \leq f_2$. The frequency f_1 is always greater than zero. Plus, there is no aliasing effect when $f_s > 2f_2$. But it has two disadvantages:

- The sampling rate is large in proportion with f_2. This has practical limitations.
- The sampled signal spectrum has spectral gaps.

To overcome this, the band pass theorem states that the input signal x(t) can be converted into its samples and can be recovered back without distortion when sampling frequency $fs < 2f2$.

Also

$$f_s = \frac{1}{T} = \frac{2f_2}{m}$$

Where m is the largest integer < $f2\,/B$ and B is the bandwidth of the signal. If f_2=KB, then for band pass signals of bandwidth $2f_m$ and the minimum sampling rate f_s=2B=$4f_m$, the spectrum of the sampled signal is given by

$$Y[\omega] = \frac{1}{T}\sum_{n=-\infty}^{\infty} X[\omega - 2nB]$$

Aliasing

Aliasing can be referred to as "the phenomenon of a high-frequency component in the spectrum of a signal, taking on the identity of a low-frequency component in the spectrum of its sampled version."

The corrective measures taken to reduce the effect of Aliasing are −

- In the transmitter section of PCM, a low pass anti-aliasing filter is employed, before the sampler, to eliminate the high frequency components, which are unwanted.

- The signal which is sampled after filtering is sampled at a rate slightly higher than the Nyquist rate. This choice of having the sampling rate higher than Nyquist rate, also helps in the easier design of the reconstruction filter at the receiver.

2.3 Types of Modulation:

In pulse width modulation, there are different types of modulation for analog and digital as shown below:

- **PCM:** Pulse Code Modulation for Analog Modulation.
- **PPM:** Pulse Position Modulation for Digital Modulation
- **PDM:** Pulse Duration Modulation for Digital Modulation.
- **PAM:** Pulse Amplitude Modulation for Digital Modulation.

Types of Modulation – Tree Diagram:

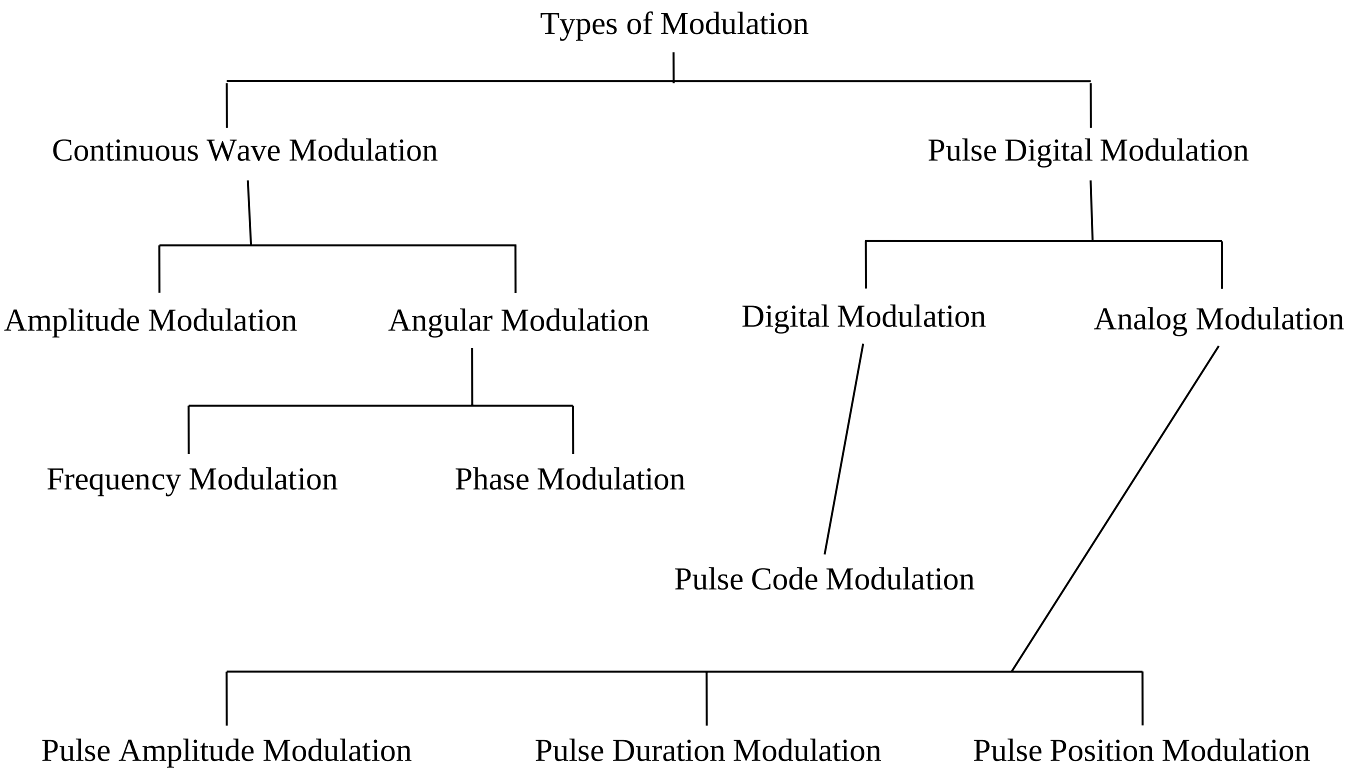


Figure 2.3.1 Types of Modulation

PCM is an important method of analog –to-digital conversion. In this modulation the analog signal is converted into an electrical waveform of two or more levels. A simple two level waveform is shown in fig 2.3.2.

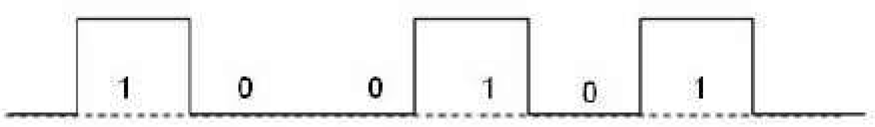


Figure 2.3.2 A Simple Binary PCM Waveform

The PCM system block diagram is shown in fig 3.2. The essential operations in the transmitter of a PCM system are Sampling, Quantizing and Coding. The Quantizing and encoding operations are usually performed by the same circuit, normally referred to as analog to digital converter. The essential operations in the receiver are regeneration, decoding and demodulation of the quantized samples. Regenerative repeaters are used to reconstruct the transmitted sequence of coded pulses in order to combat the accumulated effects of signal distortion and noise.

2.4 Pulse Amplitude Modulation (PAM):

In pulse amplitude modulation, the amplitude of regular interval of periodic pulses or electromagnetic pulses is varied in proposition to the sample of modulating signal or message signal. This is an analog type of modulation. In the pulse amplitude modulation, the message signal is sampled at regular periodic or time intervals and this each sample is made proportional to the magnitude of the message signal. These sample pulses can be transmitted directly using wired media or we can use a carrier signal for transmitting through wireless. There are two types of sampling techniques for transmitting messages using pulse amplitude modulation, they are

- **FLAT TOP PAM:** The amplitude of each pulse is directly proportional to instantaneous modulating signal amplitude at the time of pulse occurrence and then keeps the amplitude of the pulse for the rest of the half cycle.
- **Natural PAM:** The amplitude of each pulse is directly proportional to the instantaneous modulating signal amplitude at the time of pulse occurrence and then follows the amplitude of the modulating signal for the rest of the half cycle.

Flat top PAM is the best for transmission because we can easily remove the noise and we can also easily recognize the noise. When we compare the difference between the flat top PAM and natural PAM, flat top PAM principle of sampling uses sample and hold circuit. In natural principle of sampling, noise interference is minimum. But in flat top PAM noise interference maximum. Flat top PAM and natural PAM are practical and sampling rate satisfies the sampling criteria.

There are two types of pulse amplitude modulation based on signal polarity

1. Single polarity pulse amplitude modulation, 2. Double polarity pulse amplitude modulation

In single polarity pulse amplitude modulation, there is fixed level of DC bias added to the message signal or modulating signal, so the output of modulating signal is always positive. In the double polarity pulse amplitude modulation, the output of modulating signal will have both positive and negative ends.

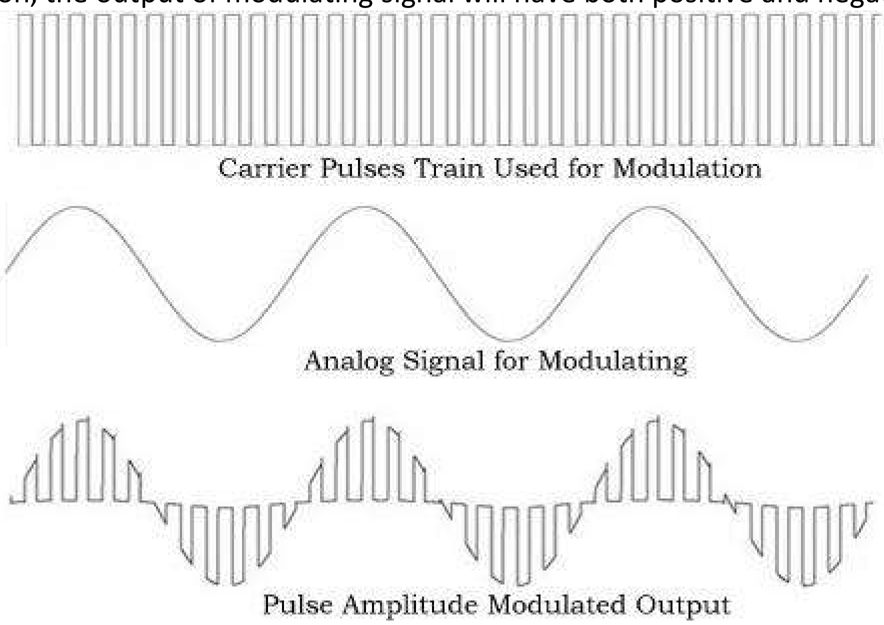


Figure 2.4.1 Pulse Amplitude Modulation

Advantages of Pulse Amplitude Modulation (PAM):

- It is the base for all digital modulation techniques and it is simple process for both modulation and demodulation technique.
- No complex circuitry is required for both transmission and reception. Transmitter and receiver circuitry is simple and easy to construct.

- PAM can generate other pulse modulation signals and can carry the message or information at same time.

Disadvantages of Pulse Amplitude Modulation (PAM):

- Bandwidth should be large for transmitting the pulse amplitude modulation signal. Due to Nyquist criteria also high bandwidth is required.
- The frequency varies according to the modulating signal or message signal. Due to these variations in the signal frequency, interferences will be there. So noise will be great. For PAM, noise immunity is less when compared to other modulation techniques. It is almost equal to amplitude modulation.
- Pulse amplitude signal varies, so power required for transmission will be more, peak power is also, even at receiving more power is required to receive the pulse amplitude signal.

Applications of Pulse Amplitude Modulation (PAM):

- It is mainly used in Ethernet which is type of computer network communication, we know that we can use Ethernet for connecting two systems and transfer data between the systems. Pulse amplitude modulation is used for Ethernet communications.
- It is also used for photo biology which is a study of photosynthesis.
- Used as electronic driver for LED lighting.
- Used in many micro controllers for generating the control signals etc.

2.5 Pulse Position Modulation (PPM):

In the pulse position modulation, the position of each pulse in a signal by taking the reference signal is varied according to the sample value of message or modulating signal instantaneously. In the pulse position modulation, width and amplitude is kept constant. It is a technique that uses pulses of the same breath and height but is displaced in time from some base position according to the amplitude of the signal at the time of sampling. The position of the pulse is 1:1 which is propositional to the width of the pulse and also propositional to the instantaneous amplitude of sampled modulating signal. The position of pulse position modulation is easy when compared to other modulation. It requires pulse width generator and mono stable multi vibrator.

Pulse width generator is used for generating pulse width modulation signal which will help to trigger the mono stable multi vibrator; here trial edge of the PWM signal is used for triggering the mono stable multi vibrator. After triggering the mono stable multi vibrator, PWM signal is converted into pulse position modulation signal. For demodulation, it requires reference pulse generator, flip-flop and pulse width modulation demodulator.

Advantages of Pulse Position Modulation (PPM):

- Pulse position modulation has low noise interference when compared to PAM because amplitude and width of the pulses are made constant during modulation.
- Noise removal and separation is very easy in pulse position modulation.
- Power usage is also very low when compared to other modulations due to constant pulse amplitude and width.

Disadvantages of Pulse Position Modulation (PPM):

- The synchronization between transmitter and receiver is required, which is not possible for every time and we need dedicated channel for it.
- Large bandwidth is required for transmission same as pulse amplitude modulation.
- Special equipments are required in this type of modulations.

Applications of Pulse Position Modulation (PPM):

- Used in non-coherent detection where a receiver does not need any Phase lock loop for tracking the phase of the carrier.
- Used in radio frequency (RF) communication.
- Also used in contactless smart card, high frequency, RFID (radio frequency ID) tags and etc.

2.6 Pulse Duration Modulation (PDM) or Pulse Width Modulation (PWM):

It is a type of analog modulation. In pulse width modulation or pulse duration modulation, the width of the pulse carrier is varied in accordance with the sample values of message signal or modulating signal or modulating voltage. In pulse width modulation, the amplitude is made constant and width of pulse and position of pulse is made proportional to the amplitude of the signal. We can vary the pulse width in three ways

1. By keeping the leading edge constant and vary the pulse width with respect to leading edge
2. By keeping the tailing constant.
3. By keeping the center of the pulse constant.

We can generate pulse width using different circuitry. In practical, we use 555 Timer which is the best way for generating the pulse width modulation signals. By configuring the 555 timer as mono stable or a stable multi vibrator, we can generate the PWM signals. We can use PIC, 8051, AVR, ARM, etc. microcontrollers to generate the PWM signals. PWM signal generation has n number of ways. In demodulation, we need PWM detector and its related circuitry for demodulating the PWM signal.

Advantages of Pulse Width Modulation (PWM):

- As like pulse position modulation, noise interference is less due to amplitude has been made constant.
- Signal can be separated very easily at demodulation and noise can also be separated easily.
- Synchronization between transmitter and receiver is not required unlike pulse position modulation.

Disadvantages of Pulse Width Modulation (PWM):

- Power will be variable because of varying in width of pulse. Transmitter can handle the power even for maximum width of the pulse.
- Bandwidth should be large to use in communication, should be huge even when compared to the pulse amplitude modulation.

Applications of Pulse Width Modulation (PWM):

- PWM is used in telecommunication systems.
- PWM can be used to control the amount of power delivered to a load without incurring the losses. So, this can be used in power delivering systems.
- Audio effects and amplifications purposes also used.
- PWM signals are used to control the speed of the robot by controlling the motors.
- PWM is also used in robotics.
- Embedded applications.
- Analog and digital applications etc.

PART II DIGITAL TRANSMISSION OF ANALOG SIGNALS

2.7 Quantization

In the process of quantization we create a new signal $m_q(t)$, which is an approximation to $m(t)$. The quantized signal $m_q(t)$, has the great merit that it is separable from the additive noise.

The operation of quantization is represented in figure 2.7.1. Here we have a signal m(t), whose amplitude varies in the range from V_H to V_L as shown in the figure.

We have divided the total range in to M equal intervals each of size S, called the step size and given by

$$S = \Delta = \frac{(V_H - V_L)}{M}$$

In our example M=8. In the centre of each of this step we located quantization levels $m0, m1, m2, \ldots m7$. The $m_q(t)$ is generated in the following manner-

Whenever the signal $m(t)$ is in the range Δ_0, the signal $m_q(t)$ maintains a constant level m_0, whenever the signal $m(t)$ is in the range Δ_1, the signal $m_q(t)$ maintains a constant level m_1 and so on. Hence the signal $m_q(t)$ will found all times to one of the levels $m0, m1, m2, \ldots m7$. The transition in $m_q(t)$ from $m0$ to $m1$ is made abruptly when $m(t)$ passes the transition level L_{01}, which is mid way between m_0 and m_1 and so on.

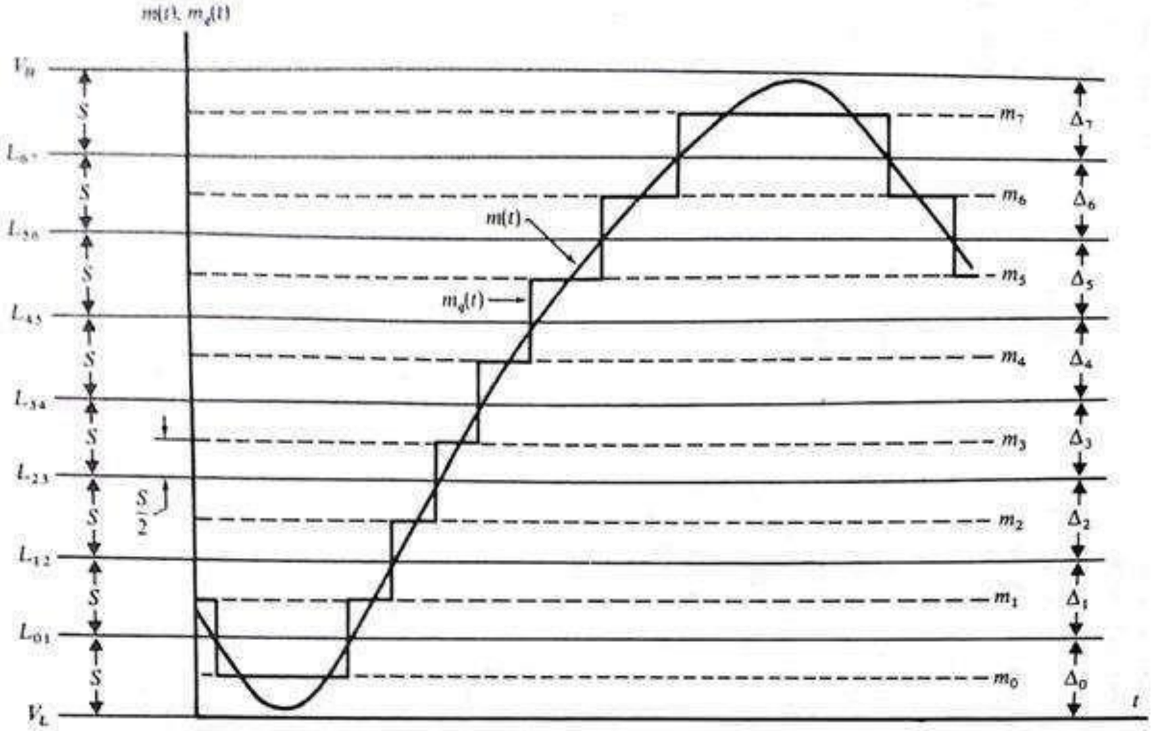


Figure 2.7.2 Quantization Process

Using quantization of signals, the effect of noise can be reduced significantly. The difference between $m(t)$ and $m_q(t)$ can be regarded as noise and is called quantization noise.

$$quantization\ noise = m(t) - m_q(t)$$

Also the quantized signal and original signal differs from one another in a ransom manner. This difference or error due to quantization process is called quantization error and is given by

$$e = m(t) - m_k$$

when $m(t)$ happens to be close to quantization level m_k, quantizer output will be m_k.

The process of transforming sampled amplitude values of a message signal into a discrete amplitude value is referred to as Quantization. The quantization Process has a two-fold effect:
1. the peak-to-peak range of the input sample values is subdivided into a finite set of decision levels or decision thresholds that are aligned with the risers of the staircase, and
2. The output is assigned a discrete value selected from a finite set of representation levels that are aligned with the treads of the staircase.

A quantizer is memory less in that the quantizer output is determined only by the value of a corresponding input sample, independently of earlier analog samples applied to the input.

Types of Quantizers:
1. Uniform Quantizer
2. Non- Uniform Quantizer

0 Ts 2Ts 3Ts Time Analog Signal Discrete Samples (Quantized)

In Uniform type, the quantization levels are uniformly spaced, whereas in non-uniform type the spacing between the levels will be unequal and mostly the relation is logarithmic. Types of Uniform Quantizers: (based on I/P - O/P Characteristics)
1. Mid-Rise type Quantizer

2. Mid-Tread type Quantizer

In the stair case like graph, the origin lies the middle of the tread portion in Mid –Tread type whereas the origin lies in the middle of the rise portion in the Mid-Rise type. Mid – tread type: Quantization levels – odd number. Mid – Rise type: Quantization levels – even number.

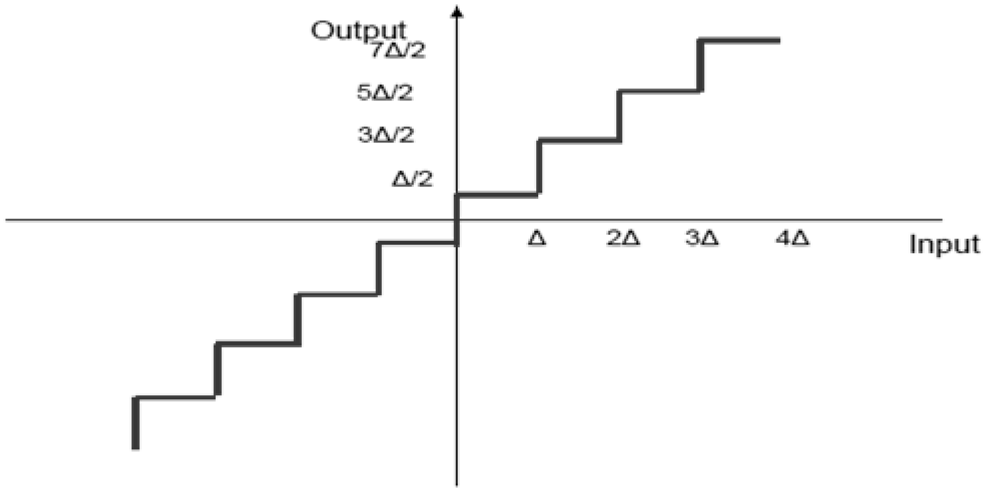


Figure 2.7.2 IO Characteristics of Mid-Rise type Quantizer

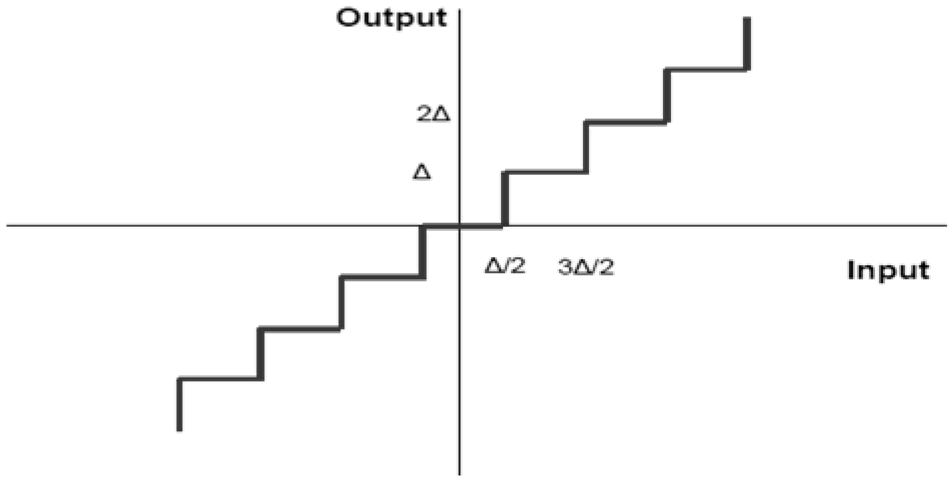


Figure 2.7.3 IO Characteristics of Mid-Tread type Quantizer

2.7.1 Quantization Noise and Signal-to-Noise:

"The Quantization process introduces an error defined as the difference between the input signal, x(t) and the output signal, y(t). This error is called the Quantization Noise."

$$q(t) = x(t) - y(t)$$

Quantization noise is produced in the transmitter end of a PCM system by rounding off sample values of an analog base-band signal to the nearest permissible representation levels of the quantizer. As such quantization noise differs from channel noise in that it is signal dependent. Let 'Δ' be the step size of a quantizer and L be the total number of quantization levels. Quantization levels are 0, ± Δ., ± 2 Δ., ±3 Δ . . . .

The Quantization error, Q is a random variable and will have its sample values bounded by [-(Δ/2) < q < (Δ/2)]. If Δ is small, the quantization error can be assumed to a uniformly distributed random variable.

Consider a memory less quantizer that is both uniform and symmetric.
L = Number of quantization levels
X = Quantizer input
Y = Quantizer output
The output y is given by

$$Y = Q(x) \quad \text{...(2.7.1.1)}$$

it is a staircase function that befits the type of mid tread or mid riser quantizer of interest. Suppose that the input 'X' lies inside the interval

$$I_k = \{X_k < X \leq X_{k-1}\} \; k=1,2,....L \quad \text{...(2.7.1.2)}$$

Where X_k and X_{k-1} are the decision thresholds of the intervals I_k as shown in figure 2.7.1.1.
Correspondingly, the quantizer output y takes on a discrete value

$Y = y_k$ if x lies in the interval I_k.

Let q = quantization error with values in the range $-\frac{\Delta}{2} \leq q \leq \frac{\Delta}{2}$ then

$Y_k = x + q$ if 'n' lies in the interval I_k

Assuming that the quantizer input 'n' is the sample value of a random variable 'X' of zero mean with variance σx^2.

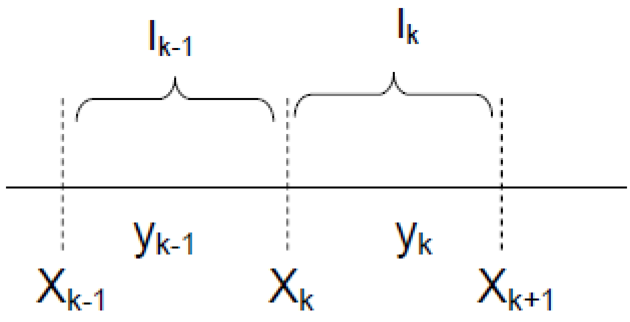


Figure 2.7.1.1 Decision Thresholds

The quantization noise uniformly distributed out the signal band, its interfering effect on a signal is similar to that of thermal noise.

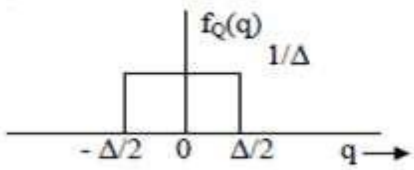
2.7.2 Expression for Quantization Noise and SNR in PCM:-

Let Q = Random Variable denotes the Quantization error
q = Sampled value of Q
Assuming that the random variable Q is uniformly distributed over the possible range

$$(-\Delta/2 \leq q \leq \Delta/2), \; as$$

$$f_Q(q) = \begin{cases} \frac{1}{\Delta} & -\Delta/2 \leq q \leq \Delta/2 \\ 0 & otherwise \end{cases} \quad \text{...(2.7.1.3)}$$

Where $f_Q(q)$ = probability density function of the Quantization error. If the signal does not overload the Quantizer, then the mean of Quantization error is zero and the variance σx^2.



Therefore

$$\sigma_Q^2 = E\{Q^2\}$$

$$\sigma_Q^2 = \int_{-\infty}^{\infty} q^2 f_q(q) dq \quad \text{...(2.7.1.4)}$$

$$\sigma_Q^2 = \frac{1}{\Delta}\int_{-\Delta/2}^{\Delta/2} q^2 dq = \frac{\Delta^2}{12} \quad \text{...(2.7.1.5)}$$

Thus the varience of the Quantization noise produced by a Uniform Quantizer, grows as the square of the step size. Equation (2.7.1.5) gives an expression for Quantization noise in PCM system.
Let σx^2= Variance of the base band signal x(t) at the input of the quantizer.
When the base band signal is reconstructed at the receiver output, we obtain original signal plus Quantization noise. Therefore output signal to Quantization noise ration (SNR) is given by

$$(SNR)_Q = \frac{Signal\ Power}{Noise\ Power} = \frac{\sigma x^2}{\sigma_Q{}^2} = \frac{\sigma x^2}{\Delta^2/12} \quad ...(2.7.1.6)$$

Smaller the step size Δ, larger will be the SNR.

2.7.3 Signal to Quantization Noise Ration:- [Mid Tread Type]

Let x = Quantizer input, sampled value of random variable X with mean X, variance σx^2. The Quantizer is assumed to be uniform, symmetric and mid trade type.
X_{max}=absolute value of the overload level of the Quantizer.
Δ= Step size,
Then L= No. of Quantization level given by

$$L = \frac{2X_{max}}{\Delta} + 1 \quad ...(2.7.1.7)$$

Let n = no. of bits used to represent each level.
In general 2^n = L, but in the mid trade quantizer, since the number of representation levels is odd,

$$L = 2^n - 1 \quad ...(2.7.1.8)$$

From equations (2.7.1.7) and (2.7.1.8),

$$2^n - 1 = \frac{2X_{max}}{\Delta} + 1$$

Or

$$\Delta = \frac{X_{max}}{2^{n-1} - 1} \quad ...(2.7.1.9)$$

The ration $\frac{X_{max}}{\sigma_x}$ is called the loading factor. To avoid significant overload distortion, the amplitude of the Quantizer input x extend from $-4\sigma_x$ to $4\sigma_x$, which correspond to loading factor of 4. Thus with $X_{max} = 4\sigma_x$, we can write equation (2.7.1.9) as,

$$\Delta = \frac{4\sigma_x}{2^{n-1} - 1} \quad ...(2.7.1.10)$$

$$(SNR)_Q = \frac{\sigma_x{}^2}{\Delta^2/12} = \frac{3}{4}[2^{n-1} - 1]^2 \quad ...(2.7.1.11)$$

For larger value of n(typically n>6), we may approximate the result as

$$(SNR)_Q = \frac{3}{4}[2^{n-1} - 1]^2 \approx \frac{3}{16}(2^{2n}) \quad ...(2.7.1.12)$$

Hence expressing the SNR in db

$$10log_{10}(SNR)_Q = 6n - 7.2 \quad ...(2.7.1.13)$$

This formula states that each bit in code word of a PCM system contributes 6db to the signal to noise ratio. For loading factor of 4, the problem of overload i.e. the problem that the sampled value of signal falls outside the total amplitude range of Quantizer, 8σx is less than 10-4. The equation 2.7.1.11 gives a good description of the noise performance of a PCM system provided that the following conditions are satisfied.
1. The Quantization error is uniformly distributed

2. The system operates with an average signal power above the error threshold so that the effect of channel noise is made negligible and performance is there by limited essentially by Quantization noise alone.
3. The Quantization is fine enough (say n>6) to prevent signal correlated patterns in the Quantization error waveform
4. The Quantizer is aligned with input for a loading factor of 4
Note: 1. Error uniformly distributed
2. Average signal power
3. n > 6
4. Loading factor = 4
From (2.7.1.13): 10 log10 (SNR)O = 6n – 7.2

In a PCM system, Bandwidth B = nW or [n=B/W], substituting the value of 'n' we get

$$10log_{10}(SNR)_Q = 6(\frac{B}{W}) - 7.2 \quad ...(2.7.1.14)$$

2.7.4 Signal to Quantization Noise Ratio:- [Mid Rise Type]
Let x = Quantizer input, sampled value of random variable X with mean X, variance σx^2. The Quantizer is assumed to be uniform, symmetric and mid rise type.
Let X_{max}=absolute value of the overload level of the Quantizer.

$$L = \frac{2X_{max}}{\Delta} \quad ...(2.7.1.15)$$

Since the number of representation levels is odd,

$$L = 2^n \text{ (Mid rise only)} \quad ...(2.7.1.16)$$

From equations (2.7.1.15) and (2.7.1.16),

$$\Delta = \frac{X_{max}}{2^n} \quad ...(2.7.1.17)$$

$$(SNR)_Q = \frac{\sigma_x{}^2}{\Delta^2/12} \quad ...(2.7.1.18)$$

Where $\sigma_x{}^2$ represent the variance or the signal power.

Consider a special case of sinusoidal signals:
Let the signal power be P_s, then $P_s = 0.5X_{max}^2$

$$(SNR)_Q = \frac{P_s}{\Delta^2/12} = \frac{12P_s}{\Delta^2} = 1.5L^2 = 1.5L^{2n} \quad ...(2.7.1.19)$$

$$\text{In decibels } (SNR)_Q = 1.76 + 6.02n \quad ...(3.20)$$

Improvement of SNR can be achieved by increasing the number of bits, n. Thus for 'n' number of bits/ sample the SNR is given by the above equation 2.7.1.19. For every increase of one bit / sample the step size reduces by half. Thus for (n+1) bits the SNR is given by

$$(SNR)_{(n+1)bit} = (SNR)_{(n)bit} + 6\ dB$$

Therefore addition of each bit increases the SNR by 6dB.

2.7.5 Classification of Quantization Noise:
The Quantizing noise at the output of the PCM decoder can be categorized into four types depending on the operating conditions, Overload noise, Random noise, Granular Noise and Hunting noise
Over Load Noise:- The level of the analog waveform at the input of the PCM encoder needs to be set so that its peak value does not exceed the design peak of V_{max} volts. If the peak input does exceed V_{max}, then

the recovered analog waveform at the output of the PCM system will have flat – top near the peak values. This produces overload noise.

Granular Noise:- If the input level is reduced to a relatively small value w.r.to to the design level (quantization level), the error values are not same from sample to sample and the noise has a harsh sound resembling gravel being poured into a barrel. This is granular noise. This noise can be randomized (noise power decreased) by increasing the number of quantization levels i.e. Increasing the PCM bit rate.

Hunting Noise:- This occurs when the input analog waveform is nearly constant. For these conditions, the sample values at the Quantizer output can oscillate between two adjacent quantization levels, causing an undesired sinusoidal type tone of frequency (0.5fs) at the output of the PCM system. This noise can be reduced by designing the quantizer so that there is no vertical step at constant value of the inputs.

2.7.6 Quantization Error

For any system, during its functioning, there is always a difference in the values of its input and output. The processing of the system results in an error, which is the difference of those values.

The difference between an input value and its quantized value is called a Quantization Error. A Quantizer is a logarithmic function that performs Quantization (rounding off the value). An analog-to-digital converter (ADC) works as a quantizer.

The following figure illustrates an example for a quantization error, indicating the difference between the original signal and the quantized signal.

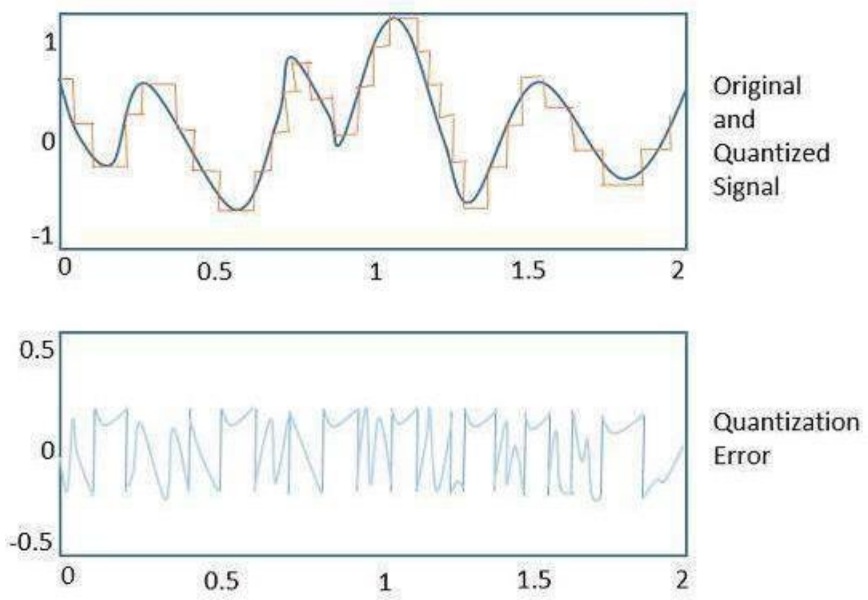


Figure 2.7.6.1 Quantization Error

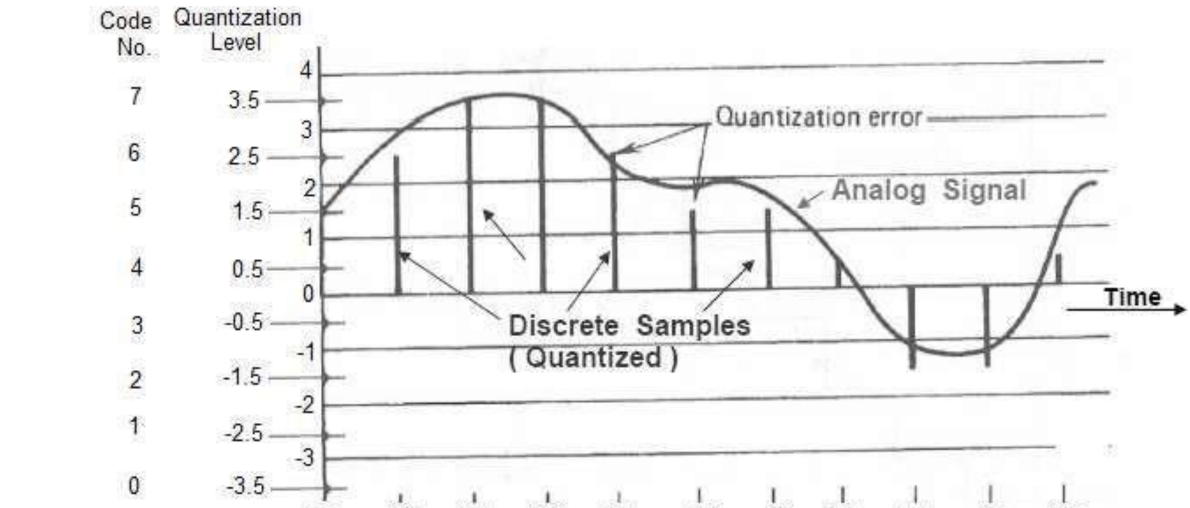
Quantization Noise

It is a type of quantization error, which usually occurs in analog audio signal, while quantizing it to digital. For example, in music, the signals keep changing continuously, where regularity is not found in errors. Such errors create a wideband noise called as Quantization Noise.

2.8 Pulse Code Modulation:

A signal which is to be quantized before transmission is sampled as well. The quantization is used to reduce the effect of noise and the sampling allows us to do the time division multiplexing. The combined operation of sampling and quantization generate a quantized PAM waveform i.e. a train of pulses whose amplitude is restricted to a number of discrete levels.

Rather than transmitting the sampled values itself, we may represent each quantization level by a code number and transmit the code number. Most frequently the code number is converted in to binary equivalent before transmission. Then the digits of the binary representation of the code are transmitted as pulses. This system of transmission is called binary Pulse Code Modulation. The whole process can be understood by the following diagram.



Sample Value	1.5	2.8	3.4	3.3	2.4	1.8	1.6	0.4	-1.4	-1.4	0.5
Nearest Quantization Level	1.5	2.5	3.5	3.5	2.5	1.5	1.5	0.5	-1.5	-1.5	0.5
Code No.	5	6	7	7	6	5	5	4	2	2	4
Binary Representation	101	110	111	111	110	101	101	100	010	010	100

Figure 2.8.1 Pulse Code Modulation

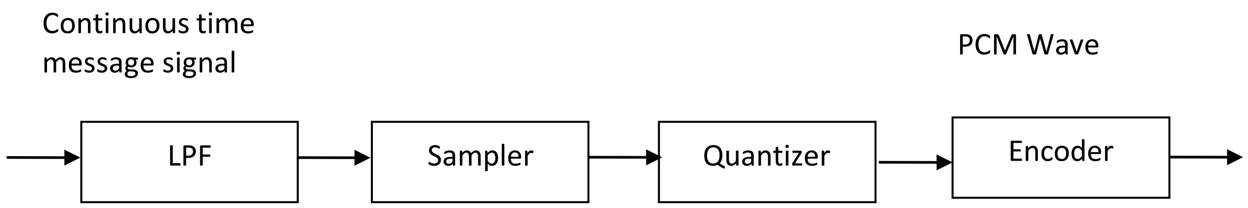
PCM Transmitter:

Basic Blocks:

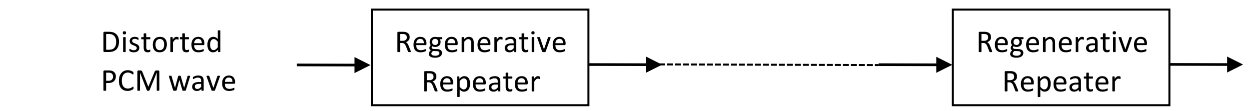
1. Anti aliasing Filter, 2. Sampler, 3. Quantizer, 4. Encoder

The block diagram of a PCM transmitter is shown in figure (a). An anti-aliasing filter is basically a filter used to ensure that the input signal to sampler is free from the unwanted frequency components. For most of the applications these are low-pass filters. It removes the frequency components of the signal which are above the cutoff frequency of the filter. The cutoff frequency of the filter is chosen such it is very close to the highest frequency component of the signal.

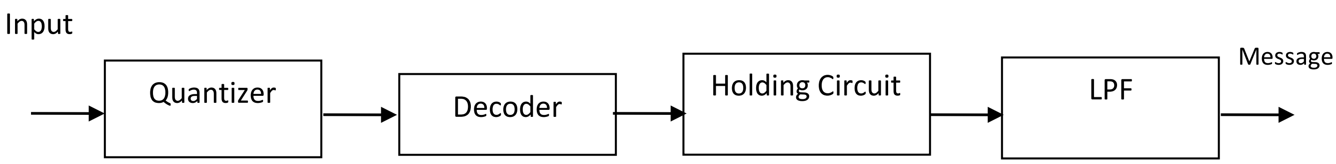
The message signal is sampled at the Nyquist rate by the sampler. The sampled pulses are then quantized by the quantizer. The encoder encodes these quantized pulses in to binary equivalent, which are then transmitted over the channel. During the channel the regenerative repeaters are used to maintain the signal to noise ratio.



(A) Transmitter



(b) Transmission Path



(c) Receiver

Figure 2.8.2 PCM System Basic Block Diagram

Figure (c) shows the receiver. The first block is again the quantizer, but this quantizer is different from the transmitter quantizer sa it has to take the decision regarding the presence or absence of the pulse only. Thus there are only two quantization levels. The output of the quantizer goes to the decoder which is an D/A converter that performs the inverse operation of the encoder. The decoder output is a sequence of quantized pulses. The original signal is reconstructed in the holding circuit and the LPF.

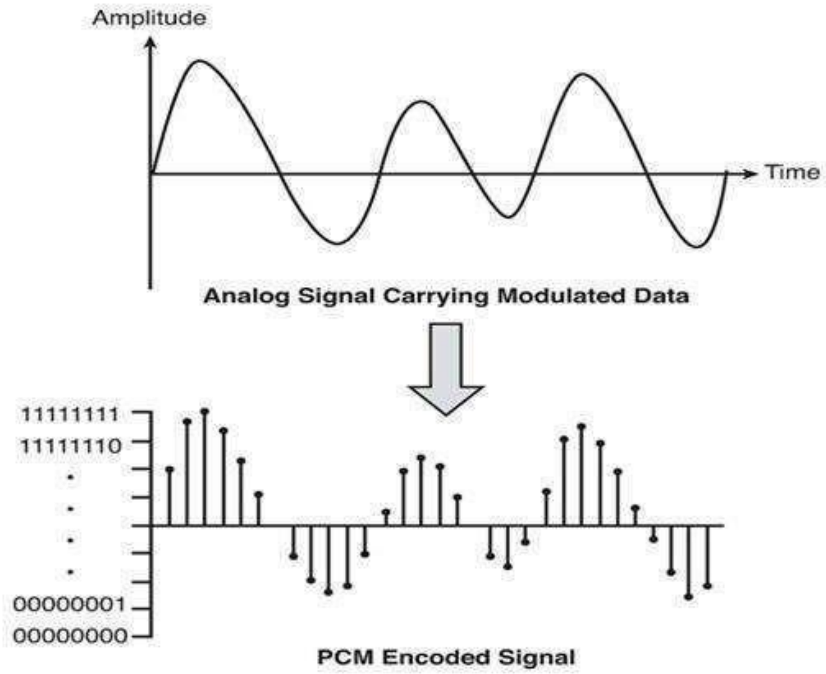


Figure 2.8.3 PCM Encoding

2.8.1 Advantages of Pulse Code Modulation:

- Pulse code modulation will have low noise addition and data loss is also very low.
- We can repeat the exact transmitted signal at the receiver. This is called repeatability. And we can retransmit the signal with any distortion loss also.
- Pulse code modulation is used in music play back CD's and also used in DVD for data storing whose sampling rate is bit higher.
- Pulse code modulation can be used in storing the data.
- PCM can encode the data also.
- Multiplexing of signals can also be done using pulse code modulation. Multiplexing is nothing for adding the different signals and transmitting the signal at same time.
- Pulse code modulation requires large bandwidth
- Pulse code modulation permits the use of pulse regeneration.

2.8.2 Disadvantages of Pulse Code Modulation:

- Specialized circuitry is required for transmitting and also for quantizing the samples at same quantized levels. We can do encoding using pulse code modulation but we need to have complex and special circuitry.

- Pulse code modulation receivers are cost effective when we compared to other modulation receivers.
- Developing pulse code modulation is bit complicated and checking the transmission quality is also difficult and takes more time.
- Large bandwidth is required for pulse code modulation when compared to bandwidth used by the normal analog signals to transmit message.
- Channel bandwidth should be more for digital encoding.
- PCM systems are complicated when compared to analog modulation methods and other systems.
- Decoding also needs special equipment's and they are also too complex.

2.8.3 Applications of Pulse Code Modulation (PCM):

- Pulse code modulation is used in telecommunication systems, air traffic control systems etc.
- Pulse code modulation is used in compressing the data that is why it is used in storing data in optical disks like DVD, CDs etc. PCM is even used in the database management systems.
- Pulse code modulation is used in mobile phones, normal telephones etc.

Remote controlled cars, planes, trains use pulse code modulations.

2.9 Companding in PCM

The word Companding is a combination of Compressing and Expanding, which means that it does both. This is a non-linear technique used in PCM which compresses the data at the transmitter and expands the same data at the receiver. The effects of noise and crosstalk are reduced by using this technique.
There are two types of Companding techniques. They are −
A-law Companding Technique

- Uniform quantization is achieved at A = 1, where the characteristic curve is linear and no compression is done.
- A-law has mid-rise at the origin. Hence, it contains a non-zero value.
- A-law companding is used for PCM telephone systems.

2.9.1 µ-law Companding Technique

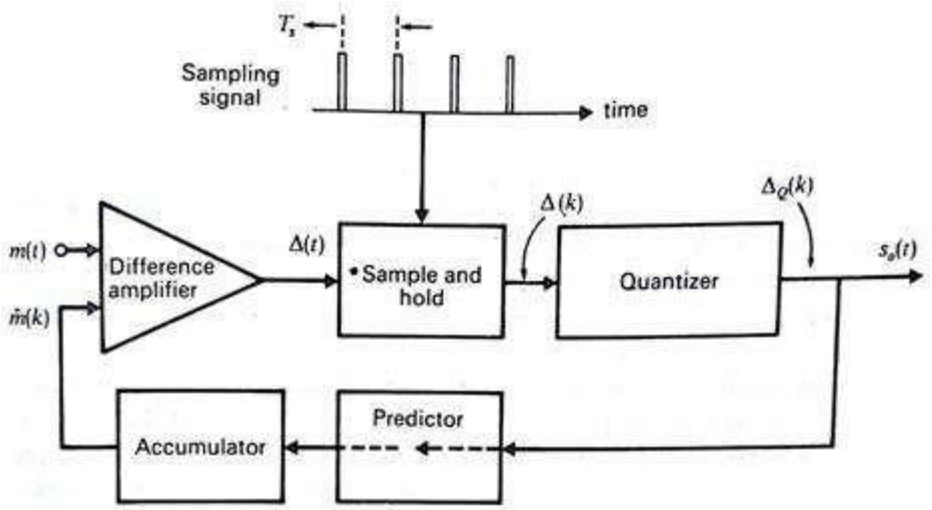
- Uniform quantization is achieved at µ = 0, where the characteristic curve is linear and no compression is done.
- µ-law has mid-tread at the origin. Hence, it contains a zero value.
- µ-law companding is used for speech and music signals.

µ-law is used in North America and Japan.
For the samples that are highly correlated, when encoded by PCM technique, leave redundant information behind. To process this redundant information and to have a better output, it is a wise decision to take a predicted sampled value, assumed from its previous output and summarize them with the quantized values. Such a process is called as Differential PCM (DPCM) technique.

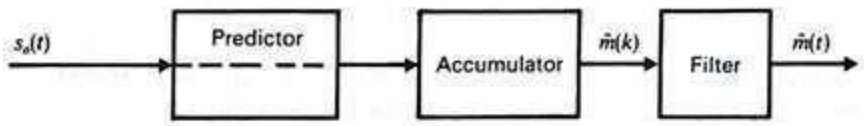
2.10 Differential Pulse Code Modulation:

Differential Pulse Code Modulation is an alternative to PCM. Instead of transmitting the sampled values itself at each sampling time; we can transmit the difference between the two successive samples. For example, we can transmit the difference between the sample value $m(k)$ at sampling time K and sample value $m(k - 1)$ at sampling time k-1. If such changes are transmitted then at the receiving end we can generate a waveform identical to the m(t) by simply adding up these changes.
The DPCM has the special merit that when these differences are transmitted by PCM. The differences $m(k) - m(k-1)$ will be smaller than the sample values themselves and fewer levels will be required to

quantize $m(k)$, and corresponding fewer bits will be needed to encode the signal. The basic principle of DPCM is shown in figure 2.10.1.



(a) Transmitter



(b) Receiver

Figure 2.10.1 Differential PCM

The receiver consists of an accumulator which adds-up the receiver quantized differences $\Delta_Q(k)$ and a filter which smoothes out the quantization noise. The output of accumulator is the signal approximation $\hat{m}(k)$ which becomes $\hat{m}(t)$ at the filter output.
At the transmitter we need to know whether the $\hat{m}(t)$ is larger or smaller than $m(t)$ and by how much amount. We may than determine whether the next difference $\Delta_Q(k)$ needs to be positive or negative and of what amplitude in order to bring $\hat{m}(t)$ as close as possible to $m(t)$. For this reason we have a duplicate accumulator at transmitter.
At each sampling time the transmitter difference amplifier compares $m(t)$ and $\hat{m}(t)$, and the sample and hold circuit holds the result of that comparison $\Delta(t)$, for the duration of interval between sampling times. The quantizer generates the signal $S_0(t) = \Delta_Q(k)$ both for the transmission to the receiver and to provide the input to the receiver accumulator in the transmitter.
The basic limitation of the DPCM scheme is that the transmitted differences are quantized and are of limited values.
Need for a predictor:
There is a correlation between the successive samples of the signal $m(t)$. To take the advantage of this correlation a predictor is included. It needs to incorporate the facility for storing past differences and carrying out some algorithm to predict then next required increment.

2.11 Delta Modulation:

Delta Modulation is a DPCM scheme in which the difference signal $\Delta(t)$ is encoded into just a single bit. The single bit providing just for two possibilities is used to increase or decrease the estimate $\hat{m}(t)[m_q(t)]$.
The Linear Delta Modulator is shown in figure 2.11.1.
The baseband signal $m(t)$ and its quantized approximation $\hat{m}(t)$ are applied as input to a comparator. The comparator has one fixed output V(H) when $m(t) > m_q(t)$ and a difference output V(L) when $m(t) < m_q(t)$. Ideally the transition between V(H) and V(L) is arbitrarily abrupt as $m(t) - m_q(t)$ passes through

zero. The up-down counter increments or decrements its count by 1 at each active edge of the clock waveform. The count direction i.e. incrementing or decrementing is determined by the voltage levels at the "Count direction command" input to the counter. When this binary input is at level V(H), the counter counts up and when this binary input is at level V(L), the counter counts down.

The digital output of the counter is converted into analog quantized approximation $m_q(t)$ by a D/A converter. The waveforms for the delta modulator of figure 2.11.1 is shown in figure 2.11.2, assuming that the active clock edge is falling edge.

It may be noted that at startup there is a brief interval when $m_q(t)$ may be a poor approximation to $m(t)$, as shown in figure 2.11.3. The initial large discrepancy between $m(t)$ and $m_q(t)$ and stepwise approach of $m_q(t)$ to $m(t)$ is shown in figure 2.11.3.

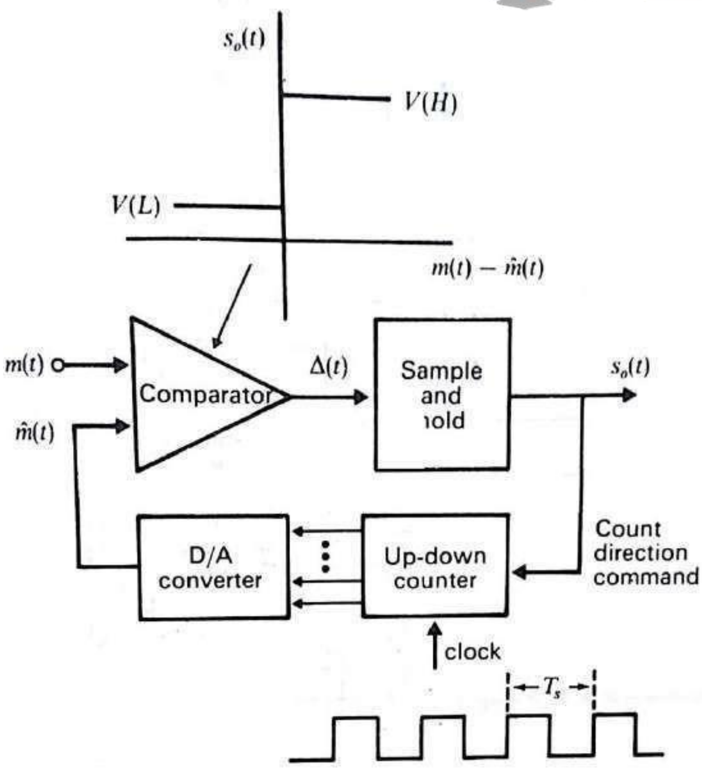


Figure 2.11.1 Delta Modulator

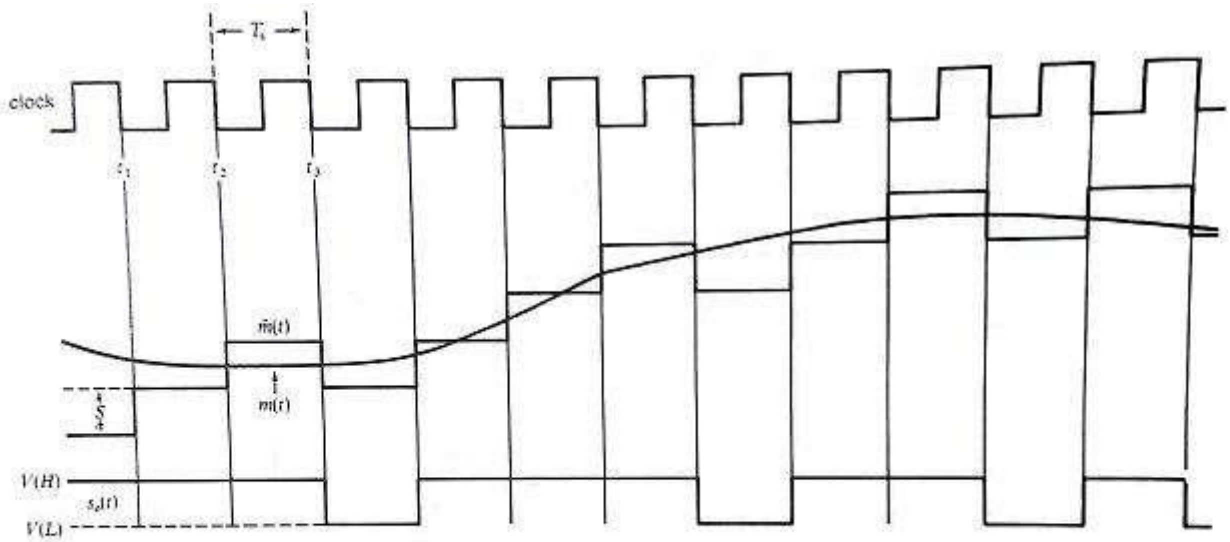


Figure 2.11.2 The response of the delta modulator to a baseband signal m(t)

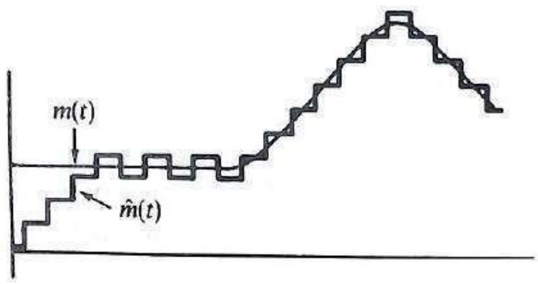


Figure 2.11.3 Startup response of DM

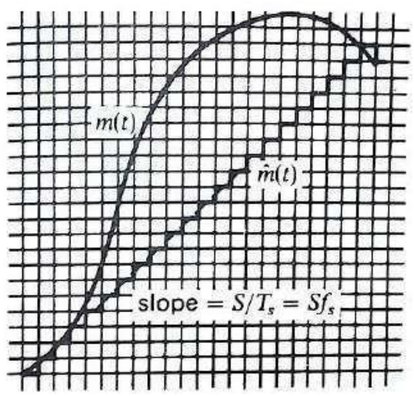


Figure 2.11.4 Slope Overload in a linear DM

It should be noted that when $m_q(t)$ has caught up $m(t)$ and even though $m(t)$ remains constant, $m_q(t)$ hunts, swinging up and down to $m(t)$.

Slope Overload

The excessive disparity between $m(t)$ and $m_q(t)$ is described as a slope overload error and occurs whenever $m(t)$ has a slope larger than the slope S/T_s which can be sustained by the waveform $m_q(t)$. The slope overload as shown in figure 3.11.4 is developed due to the small size of S. To overcome the overload we have to increase the sampling rate above the rate initially selected to satisfy the Nyquist criterion. The sampling rate f_s must satisfy the following condition

$$sf_s = 2\pi f A$$

Features of DM

Following are some of the features of delta modulation.

- An over-sampled input is taken to make full use of the signal correlation.
- The quantization design is simple.
- The input sequence is much higher than the Nyquist rate.
- The quality is moderate.
- The design of the modulator and the demodulator is simple.
- The stair-case approximation of output waveform.
- The step-size is very small, i.e., Δ (delta).
- The bit rate can be decided by the user.
- This involves simpler implementation.

2.11.3 Advantages of DM Over DPCM

- 1-bit quantizer
- Very easy design of the modulator and the demodulator
- However, there exists some noise in DM.
- Slope Over load distortion (when Δ is small)
- Granular noise (when Δ is large)

2.12 Adaptive Delta Modulation (ADM):

In digital modulation, we have come across certain problem of determining the step-size, which influences the quality of the output wave.

A larger step-size is needed in the step slope of modulating signal and a smaller step size is needed where the message has a small slope. The minute details get missed in the process. So, it would be better if we can control the adjustment of step-size, according to our requirement in order to obtain the sampling in a desired fashion. This is the concept of Adaptive Delta Modulation.

Following is the block diagram of Adaptive delta modulator.

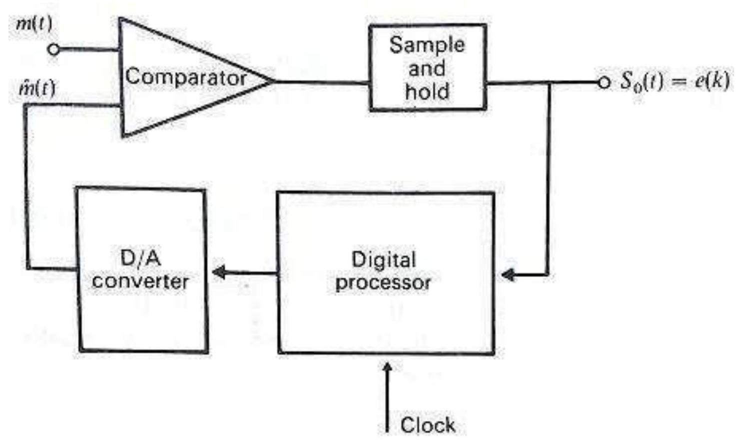


Figure 2.12.1 Adaptive Delta Modulation (ADM)

The step size S is not of fixed size but it is always a multiple of basic step size S_0. The basic step size S_0 is either added or subtracted by the accumulator as required to move $m_q(t)$ more close to $m(t)$. If the direction of the step at the clock edge K is same as at edge K-1, then the processor increases the step size by an amount S_0. If the directions are opposite then the processor decreases the magnitude of the step by S_0.

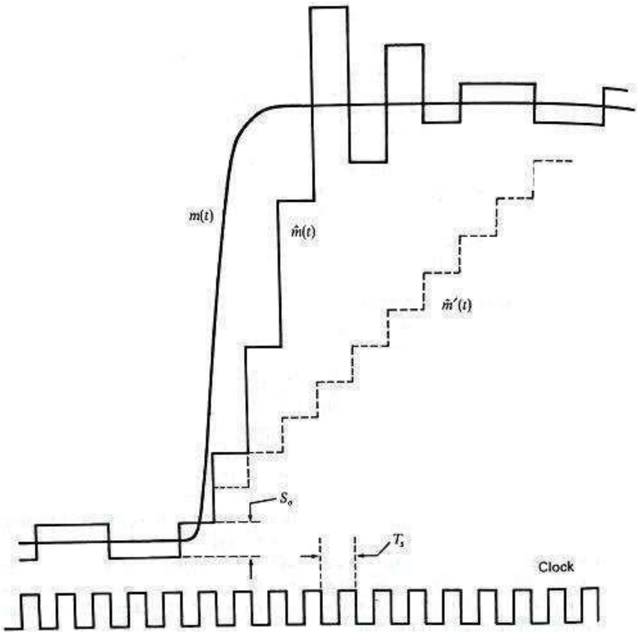


Figure 2.12.2 Waveforms comparing the response of DM and ADM

In figure 2.12.1, the output $S_0(t)$ is called $e(k)$, which represents the error i.e. the discrepancy between the $m(t)$ and $m_q(t)$, and it is either V(H) or V(L).

$e(k) = +1, if\ m(t) > m_q(t)$ immediately before Kth edge,
$e(k) = -1, if\ m(t) < m_q(t)$ immediately before Kth edge,

The features of ADM are shown in figure 2.12.2. As long as the condition $m(t) > m_q(t)$ persists the jumps in $m_q(t)$ becomes larger, that's why $m_q(t)$ catches up with $m(t)$ sooner than in the case of linear DM, as shown by $m'_q(t)$.

On the other hand, when the response to the large slope in $m(t)$, $m_q(t)$ develops large jumps and large number of clock cycles are required for these jumps to settle down. Therefore the ADM system reduces the slope overload but it increases the quantization error. Also when $m(t)$ is constant $m_q(t)$ oscillates about $m(t)$ but the oscillation frequency is half of the clock frequency.

2.13 Voice coders:

A vocoder i.e. *voice encoder* is an analysis/synthesis system, used to reproduce human speech. In the encoder, the input is passed through a multiband filter, each band is passed through an envelope follower, and the control signals from the envelope followers are communicated to the decoder. The decoder applies these control signals to corresponding filters in the (re)synthesizer.

It was originally developed as a speech coder for telecommunications applications in the 1930s, the idea being to code speech for transmission. Its primary use in this fashion is for secure radio communication, where voice has to be encrypted and then transmitted. The advantage of this method of "encryption" is that no 'signal' is sent, but rather envelopes of the band pass filters. The receiving unit needs to be set up in the same channel configuration.
Information, and recreates it, The Voder i.e. *Voice Operating Demonstrator* generates synthesized speech by means of a console with fifteen touch-sensitive keys and a pedal, basically consisting of the "second half" of the vocoder, but with manual filter controls, needing a highly trained operator.

The human voice consists of sounds generated by the opening and closing of the glottis by the vocal cords, which produces a periodic waveform with many harmonics. This basic sound is then filtered by the nose and throat (a complicated resonant piping system) to produce differences in harmonic content (formants) in a controlled way, creating the wide variety of sounds used in speech. There is another set of sounds, known as the unvoiced and plosive sounds, which are created or modified by the mouth in different fashions.
The vocoder examines speech by measuring how its spectral characteristics change over time. This results in a series of numbers representing these modified frequencies at any particular time as the user speaks. In simple terms, the signal is split into a number of frequency bands (the larger this number, the more accurate the analysis) and the level of signal present at each frequency band gives the instantaneous representation of the spectral energy content. Thus, the vocoder dramatically reduces the amount of information needed to store speech, from a complete recording to a series of numbers. To recreate speech, the vocoder simply reverses the process, processing a broadband noise source by passing it through a stage that filters the frequency content based on the originally recorded series of numbers. Information about the instantaneous frequency (as distinct from spectral characteristic) of the original voice signal is discarded; it wasn't important to preserve this for the purposes of the vocoder's original use as an encryption aid, and it is this "dehumanizing" quality of the vocoding process that has made it useful in creating special voice effects in popular music and audio entertainment.

Since the vocoder process sends only the parameters of the vocal model over the communication link, instead of a point by point recreation of the waveform, it allows a significant reduction in the bandwidth required to transmit speech.

Unit 3

Digital Transmission Techniques: Phase shift Keying (PSK)- Binary PSK, differential PSK, differentially encoded PSK, Quadrature PSK, M-ary PSK. Frequency Shift Keying (FSK)- Binary FSK (orthogonal and non-orthogonal), M-ary FSK. Comparison of BPSK and BFSK, Quadrature Amplitude Shift Keying (QASK), Minimum Shift Keying (MSK)

3.1 Digital Modulation

Digital Modulation provides more information capacity, high data security, quicker system availability with great quality communication. Hence, digital modulation techniques have a greater demand, for their capacity to convey larger amounts of data than analog modulation techniques.
There are many types of digital modulation techniques and also their combinations, as listed below.

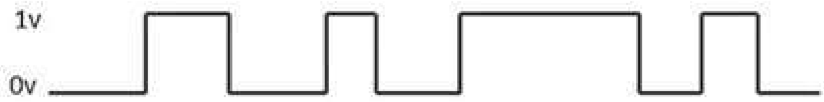
ASK – Amplitude Shift Keying
The amplitude of the resultant output depends upon the input data whether it should be a zero level or a variation of positive and negative, depending upon the carrier frequency.

FSK – Frequency Shift Keying
The frequency of the output signal will be either high or low, depending upon the input data applied.

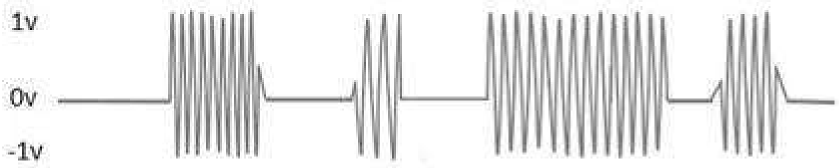
PSK – Phase Shift Keying
The phase of the output signal gets shifted depending upon the input. These are mainly of two types, namely Binary Phase Shift Keying (BPSK) and Quadrature Phase Shift Keying (QPSK), according to the number of phase shifts. The other one is Differential Phase Shift Keying (DPSK) which changes the phase according to the previous value.

M-ary Encoding
M-ary Encoding techniques are the methods where more than two bits are made to transmit simultaneously on a single signal. This helps in the reduction of bandwidth.
The types of M-ary techniques are M-ary ASK, M-ary FSK & M-ary PSK.

Amplitude Shift Keying (ASK) is a type of Amplitude Modulation which represents the binary data in the form of variations in the amplitude of a signal.
Any modulated signal has a high frequency carrier. The binary signal when ASK modulated, gives a zero value for Low input while it gives the carrier output for High input.
The figure 3.1.1 represents ASK modulated waveform along with its input.



(a) ASK Modulation



(b) ASK Modulated Wave

Figure 3.1.1 ASK Modulation

To find the process of obtaining this ASK modulated wave, let us learn about the working of the ASK modulator.

3.2 ASK Modulator

The ASK modulator block diagram comprises of the carrier signal generator, the binary sequence from the message signal and the band-limited filter. Following is the block diagram of the ASK Modulator.

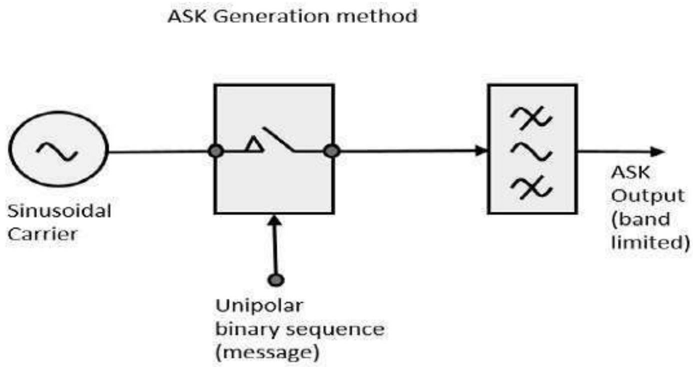


Figure 3.2.1 ASK Modulator

The carrier generator sends a continuous high-frequency carrier. The binary sequence from the message signal makes the unipolar input to be either High or Low. The high signal closes the switch, allowing a carrier wave. Hence, the output will be the carrier signal at high input. When there is low input, the switch opens, allowing no voltage to appear. Hence, the output will be low.
The band-limiting filter, shapes the pulse depending upon the amplitude and phase characteristics of the band-limiting filter or the pulse-shaping filter.

ASK Demodulator

There are two types of ASK Demodulation techniques. They are −

- Asynchronous ASK Demodulation/detection
- Synchronous ASK Demodulation/detection

The clock frequency at the transmitter when matches with the clock frequency at the receiver, it is known as a Synchronous method, as the frequency gets synchronized. Otherwise, it is known as Asynchronous.

Asynchronous ASK Demodulator

The Asynchronous ASK detector consists of a half-wave rectifier, a low pass filter, and a comparator. Following is the block diagram for the same.

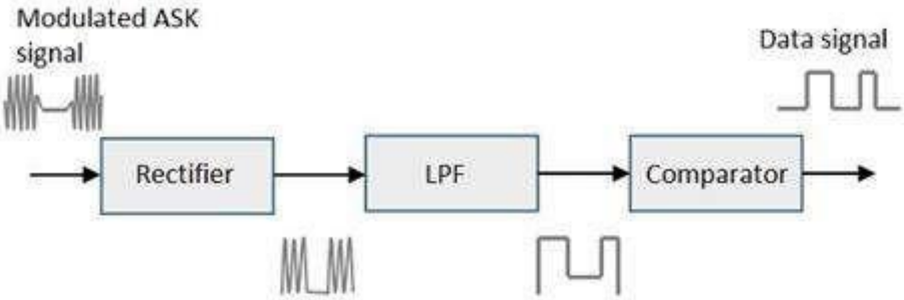


Figure 3.2.2 ASK Demodulator

The modulated ASK signal is given to the half-wave rectifier, which delivers a positive half output. The low pass filter suppresses the higher frequencies and gives an envelope detected output from which the comparator delivers a digital output.

Synchronous ASK Demodulator

Synchronous ASK detector consists of a Square law detector, low pass filter, a comparator, and a voltage limiter. Following is the block diagram for the same.

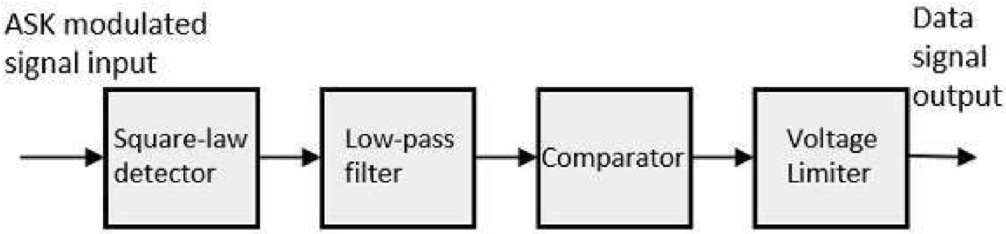


Figure 3.2.3 Synchronous ASK Demodulator

The ASK modulated input signal is given to the Square law detector. A square law detector is one whose output voltage is proportional to the square of the amplitude modulated input voltage. The low pass filter minimizes the higher frequencies. The comparator and the voltage limiter help to get a clean digital output.

3.3 Frequency Shift Keying

Frequency Shift Keying (FSK) is the digital modulation technique in which the frequency of the carrier signal varies according to the digital signal changes. FSK is a scheme of frequency modulation. The output of a FSK modulated wave is high in frequency for a binary High input and is low in frequency for a binary Low input. The binary 1s and 0s are called Mark and Space frequencies. The following image is the diagrammatic representation of FSK modulated waveform along with its input.

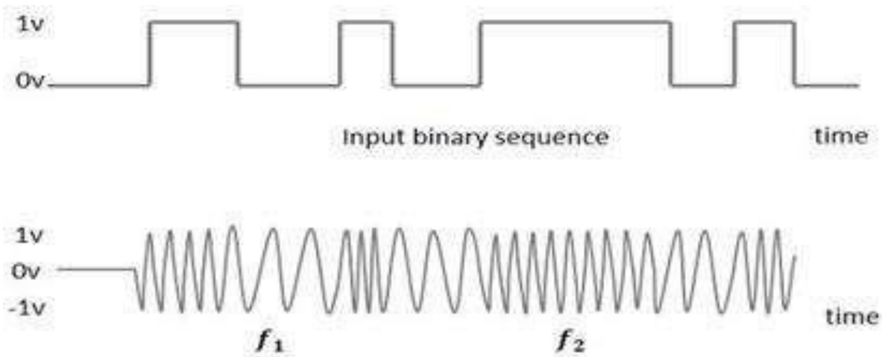


Figure 3.3.1 Frequency Shift Keying (FSK)

To find the process of obtaining this FSK modulated wave, let us know about the working of a FSK modulator.

FSK Modulator

The FSK modulator block diagram comprises of two oscillators with a clock and the input binary sequence. Following is its block diagram.

FSK Transmitter

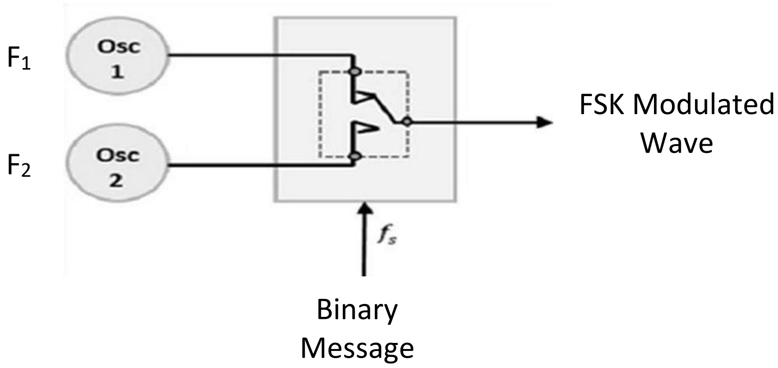


Figure 3.3.2 FSK Modulator

The two oscillators, producing a higher and a lower frequency signals, are connected to a switch along with an internal clock. To avoid the abrupt phase discontinuities of the output waveform during the transmission of the message, a clock is applied to both the oscillators, internally. The binary input sequence is applied to the transmitter so as to choose the frequencies according to the binary input.

FSK Demodulator

There are different methods for demodulating a FSK wave. The main methods of FSK detection are asynchronous detector and synchronous detector. The synchronous detector is a coherent one, while asynchronous detector is a non-coherent one.

Asynchronous FSK Detector

The block diagram of Asynchronous FSK detector consists of two band pass filters, two envelope detectors, and a decision circuit. Following is the diagrammatic representation.

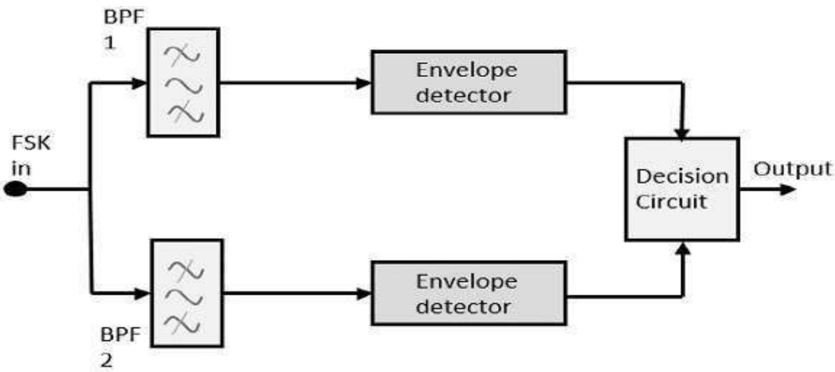


Figure 3.3.3 Asynchronous FSK Detector

The FSK signal is passed through the two Band Pass Filters (BPFs), tuned to Space and Mark frequencies. The output from these two Band Pass Filters look like ASK signal, which is then applied to the envelope detector. The signal in each envelope detector is modulated asynchronously.

The decision circuit chooses which output is more likely and selects it from any one of the envelope detectors. It also re-shapes the waveform to a rectangular one.

Synchronous FSK Detector

The block diagram of Synchronous FSK detector consists of two mixers with local oscillator circuits, two band pass filters and a decision circuit. Following is the diagrammatic representation.

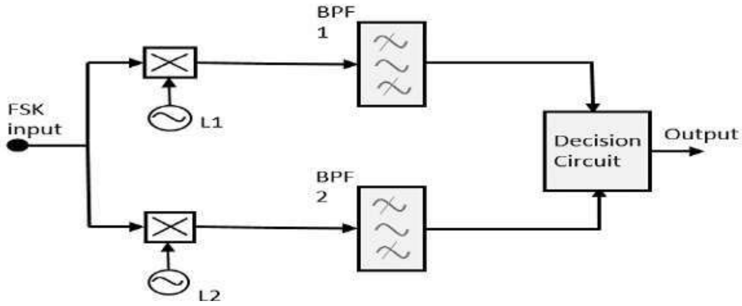


Figure 3.3.4 Synchronous FSK Detector

The FSK signal input is given to the two mixers with local oscillator circuits. These two are connected to two band pass filters. These combinations act as demodulators and the decision circuit chooses which output is more likely and selects it from any one of the detectors. The two signals have a minimum frequency separation.

For both of the demodulators, the bandwidth of each of them depends on their bit rate. This synchronous demodulator is a bit complex than asynchronous type demodulators.

3.4 Phase Shift Keying (PSK)

Phase Shift Keying (PSK) is the digital modulation technique in which the phase of the carrier signal is changed by varying the sine and cosine inputs at a particular time. PSK technique is widely used for wireless LANs, bio-metric, contactless operations, along with RFID and Bluetooth communications.

3.4.1 Binary Phase Shift Keying (BPSK)

In binary phase-shift keying (BPSK) the transmitted signal is a sinusoid of fixed amplitude. It has one fixed phase when the data is at one level and when the data is at the other level the phase is different by 180°. If the sinusoid is of amplitude A it has a power

$$P_s = \frac{1}{2} A^2 \; therefore \; A = \sqrt{2P_s}$$

Thus the transmitted signal is either

$$V_{BPSK}(t) = \sqrt{2P_s}.\cos(\omega_o t) \qquad \text{...3.4.1.1}$$

or

$$V_{BPSK}(t) = -\sqrt{2P_s}.\cos(\omega_o t + \pi) \qquad \text{...3.4.1.2(a)}$$

$$V_{BPSK}(t) = -\sqrt{2P_s}.\cos(\omega_o t) \qquad \text{...3.4.1.2(b)}$$

In BPSK the data b(t) is a stream of binary digits with voltage levels which, as a matter of convenience, we take to be at + 1V and - 1V. When b(t) = 1V we say it is at logic level 1 and when b(t) = -I V we say it is at logic level 0.

Hence $V_{BPSK}(t)$ can be written, as

$$V_{BPSK}(t) = b(t)\sqrt{2P_s}.\cos(\omega_o t) \qquad \text{...3.4.1.3}$$

In practice, a BPSK signal is generated by applying the waveform $\cos(\omega_o t)$, as a carrier, to a balanced modulator and applying the baseband signal b(t) as the modulating waveform. In this sense BPSK can be thought of as an AM signal.

BPSK Modulator:

The block diagram of Binary Phase Shift Keying consists of the balance modulator which has the carrier sine wave as one input and the binary sequence as the other input.

Following is the diagrammatic representation.

BPSK Modulator

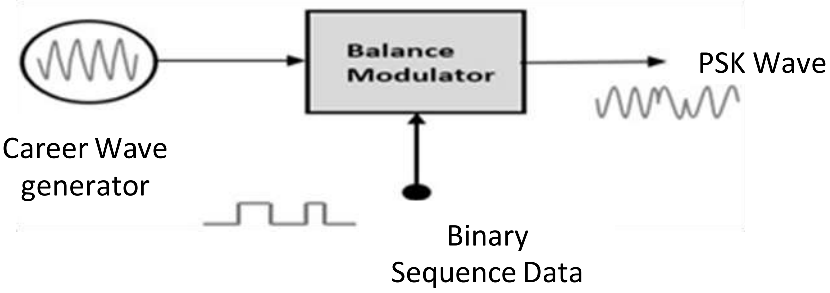


Figure 3.4.1 BPSK Modulator

The modulation of BPSK is done using a balance modulator, which multiplies the two signals applied at the input. For a zero binary input, the phase will be 0° and for a high input, the phase reversal is of 180°.

Following is the diagrammatic representation of BPSK Modulated output wave along with its given input.

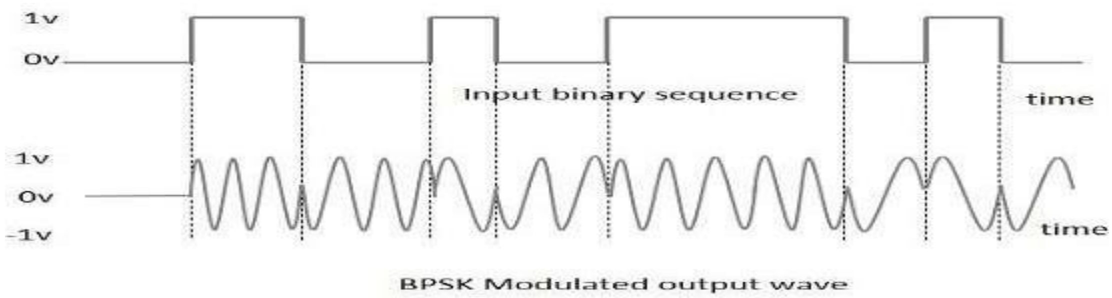


Figure 3.4.2 BPSK Modulated Waveform

The output sine wave of the modulator will be the direct input carrier or the inverted (180° phase shifted) input carrier, which is a function of the data signal.

Reception of BPSK:

The received signal has the form

$$V_{BPSK}(t) = b(t)\sqrt{2P_s}.\cos(\omega_o t + \theta) = b(t)\sqrt{2P_s}.\cos\omega_o(t + \theta/\omega_o) \quad \text{...3.4.1.4}$$

Here θ is a nominally fixed phase shift corresponding to the time delay θ/ω_o which depends on the length of the path from transmitter to receiver and the phase shift produced by the amplifiers in the" front-end" of the receiver preceding the demodulator. The original data b(t) is recovered in the demodulator. The demodulation technique usually employed is called synchronous demodulation and requires that there be available at the demodulator the waveform $\cos(\omega_o t + \theta)$. A scheme for generating the carrier at the demodulator and for recovering the baseband signal is shown in Fig. 3.4.1.1.

The received signal is squared to generate the signal

$$\cos^2(\omega_o t + \theta) = \frac{1}{2} + \frac{1}{2}\cos 2(\omega_o t + \theta) \quad \text{...3.4.1.5}$$

The DC component is removed by the band pass filter whose pass band is centered around 2fo and we then have the signal whose waveform is that of $\cos 2(\omega_o t + \theta)$. A frequency divider (composed of a flip-flop and narrow-band filter tuned to fo is used to regenerate the waveform $\cos(\omega_o t + \theta)$. Only the waveforms of the signals at the outputs of the squarer, filter and divider are relevant, not their amplitudes.

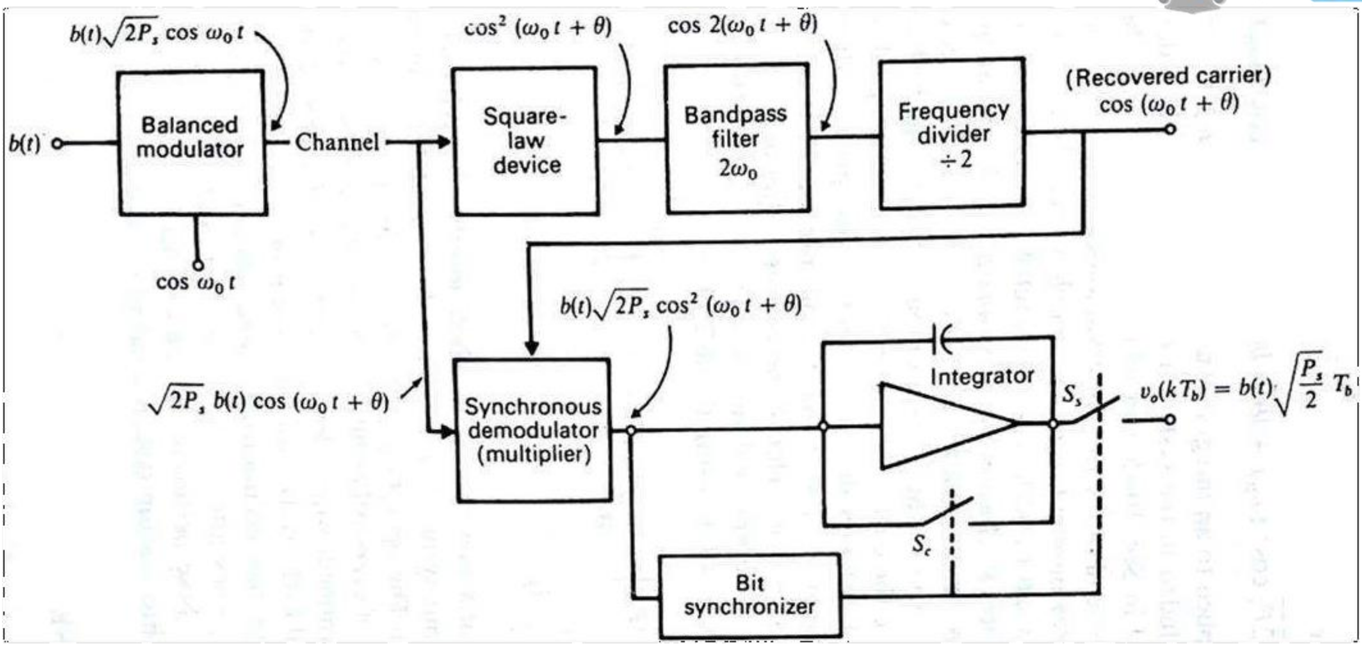


Figure 3.4.1.1 Reception of BPSK

Accordingly in Fig. 3.4.1.1 we have arbitrarily taken amplitudes to be unity. In practice, the amplitudes will be determined by features of these devices which are of no present concern. In any event, the carrier having been recovered, it is multiplied with the received signal to generate

$$b(t)\sqrt{2P_s}\cos^2(\omega_o t+\theta) = b(t)\sqrt{2P_s}\left[\frac{1}{2}+\frac{1}{2}\cos 2(\omega_o t+\theta)\right] \qquad \text{...3.4.1.6}$$

which is then applied to an integrator as shown.

We have included in the system a bit synchronizer. This device, whose operation is able to recognize precisely the moment which corresponds to the end of the time interval allocated to one bit and the beginning of the next. At that moment, it closes switch S, very briefly to discharge (dump) the integrator capacitor and leaves the switch S, open during the entire course of the ensuing bit interval, closing switch S, again very briefly at the end of the next bit time, etc. (This circuit is called an "integrate-and-dump" circuit.) The output signal of interest to us is the integrator output at the end of a bit interval but immediately before the closing of switch So. This output signal is made available by switch S; which samples the output voltage just prior to dumping the capacitor.

Let us assume for simplicity that the bit interval Tb is equal to the duration of an integral number n of cycles of the carrier of frequency f_o that is, $n\ .\ 2\pi\ =\ \omega_0 T_b$ In this case the output voltage $v_0(kT_b)$ at the end of a bit interval extending from time $(k-\ 1)T_b$ to $(k)T_b$ is, using Eq. (3.4.1.6).

$$v_0(kT_b) = b(kT_b)\sqrt{2P_s}\int_{(k-1)T_b}^{kT_b}\frac{1}{2}\,dt + b(kT_b)\sqrt{2P_s}\int_{(k-1)T_b}^{kT_b}\frac{1}{2}\cos 2(\omega_o t+\theta)\,dt \qquad \text{...3.4.1.7(a)}$$

$$v_0(kT_b) = b(kT_b)\sqrt{\frac{P_s}{2}}\,T_b \qquad \text{...3.4.1.7(b)}$$

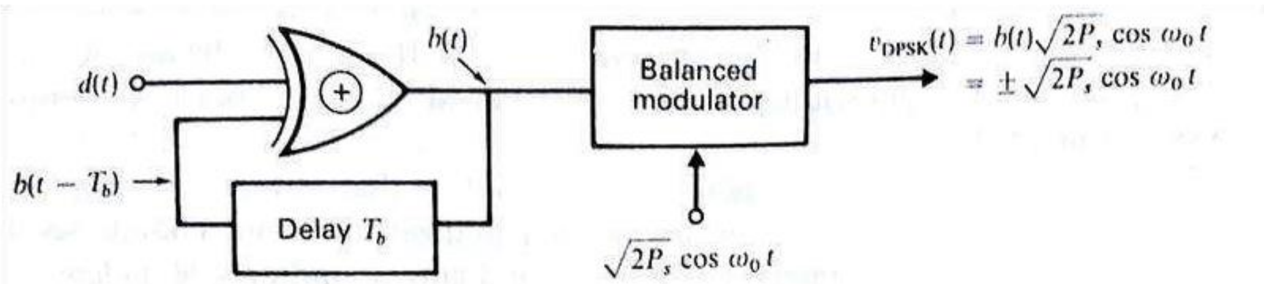
Since the integral of a sinusoid over a whole number of cycles has the value zero. Thus we see that our system reproduces at the demodulator output the transmitted bit stream b(t). The operation of the bit synchronizer allows us to sense each bit independently of every other bit. The brief closing of both switches, after each bit has been determined, wipes clean all influence of a preceding bit and allows the receiver to deal exclusively with the present bit.

3.4.2 Differential Phase Shift Keying (DPSK)

In BPSK, to regenerate the carrier we start by squaring $b(t)\sqrt{2P_s}.\ \cos(\omega_o t)$. Accordingly, if the received signal were instead $-b(t)\sqrt{2P_s}.\ \cos(\omega_o t)$ the recovered carrier would remain as before. Therefore we shall not be able to determine whether the received baseband signal is the transmitted signal $b(t)$ or it's negative $-\ b(t)$.

Differential phase-shift keying (DPSK) and differential encoded PSK (DEPSK) are modifications of BPSK which have the merit that they eliminate the ambiguity about whether the demodulated data is or is not inverted. In addition DPSK avoids the need to provide the synchronous carrier required at the demodulator for detecting a BPSK signal.

A means for generating a DPSK signal is shown in Fig. 3.4.2.1. The data stream to be transmitted, $d(t)$, is applied to one input of an exclusive-OR logic gate. To the other gate input is applied the output of the exclusive or gate $b(t)$ delayed by the time T_b, allocated to one bit. This second input is then $b(t\ -\ T_b)$. In Fig. 3.4.2.2 we have drawn logic waveforms to illustrate the response b(t) to an input d(t). The upper level of the waveforms corresponds to logic 1, the lower level to logic O. The truth table for the exclusive-OR gate is given in Fig 3.4.2.1, and with this table we can easily verify that the waveforms for (t), $b(t\ -\ T_b)$, and $b(t)$ are consistent with one another. We observe that, as required, $b(t\ -\ T_b)$ is indeed b(t) delayed by one bit time and that in any bit interval the bit b(t) is given $b(t) = d(t) \oplus b(t - T_b)$.



$d(t)$		$b(t - T_b)$		$b(t)$	
logic level	voltage	logic level	voltage	logic level	voltage
0	−1	0	−1	0	−1
0	−1	1	1	1	1
1	1	0	−1	1	1
1	1	1	1	0	−1

Figure 3.4.2.1 Means for generating DPSK

Because of the feedback involved in the system of Fig. 3.4.2.2 there is a difficulty in determining the logic levels in the interval in which we start to draw the waveforms (interval 1 in Fig. 3.4.2.2). We cannot determine b(t) in this first interval of our waveform unless we know b(k = 0). But we cannot determine b(0) unless we know both d(0) and b(-1), etc. Thus, to justify any set of logic levels in an initial bit interval we need to know the logic levels in the preceding interval. But such a determination requires information about the interval two bit times earlier and so on. In the waveforms of Fig. 3.4.2.2 we have circumvented the problem by arbitrarily assuming that in the first interval b(0) = 0. It is shown below that in the demodulator, the data will be correctly determined regardless of our assumption concerning b(0).

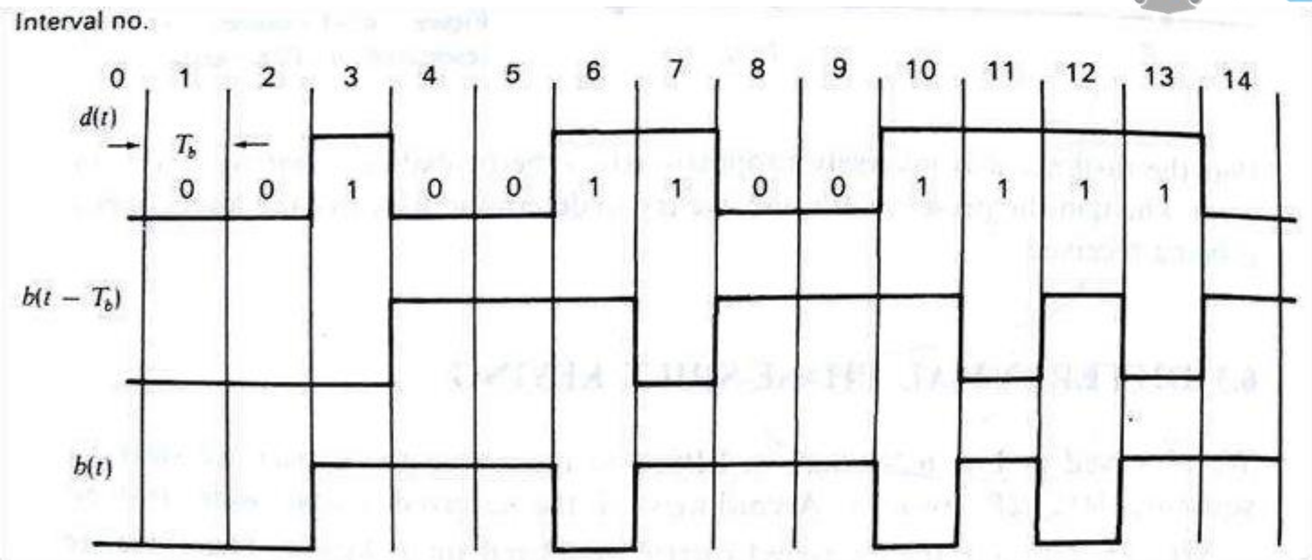


Figure 3.4.2.2 Logic waveforms to illustrate the response *b(t)* to an input *d(t)*.

We now observe that the response of b(t) to d(t) is that b(t) changes level at the beginning of each interval in which d(t) = 1 and b(t) does not change level when d(t) = 0. Thus during interval 3, d(3) = 1, and correspondingly b(3) changes at the beginning at that interval. During intervals 6 and 7, d(6) = d(7) = 1 and there are changes in b(t) at the beginnings of both intervals. During bits 10, 11, 12, and 13 d(t) = 1 and there are changes in b(t) at the beginnings of each of these intervals. This behavior is to be anticipated from the truth table of the exclusive-OR gate. For we note that when $d(t) = 0$, $b(t) = b(t - T_b)$ so that, whatever the initial value of $b(t - T_b)$,it reproduces itself. On the other hand when d(t) = 1, then $b(t) = b(t - T_b)$. Thus, in each successive bit interval b(t) changes from its value in the previous interval. Note that in some intervals where d(t) = 0 we have b(t) = 0 and in other intervals when d(t) = 0 we have b(t) = 1. Similarly, when d(t) = 1 sometimes b(t) = 1 and sometimes b(t) = 0. Thus there is no correspondence between the levels of d(t) and b(t), and the only invariant feature of the system is that a change (sometimes up and sometimes down) in b(t) occurs whenever d(t) = 1, and that no change in b(t) will occur whenever d(t) = 0.

Finally, we note that the waveforms of Fig. 3.4.2.2 are drawn on the assumption that, in interval 1, b(0) = 0. As is easily verified, if not intuitively apparent, if we had assumed b(0) = 1, the invariant feature by which we have characterized the system would continue to apply. Since b(0) must be either b(0) = 0 or b(0) = 1, there being no other possibilities, our result is valid quite generally. If, however, we had started with b(0) = 1 the levels b(l) and b(0) would have been inverted.

As is seen in Fig. 3.4.2.1 b(t) is applied to a balanced modulator to which is also applied the carrier $\sqrt{2P_s}.\cos(\omega_o t)$. The modulator output, which is the transmitted signal, is

$$V_{DPSK}(t) = b(t)\sqrt{2P_s}.\cos(\omega_o t)$$
$$= \pm\sqrt{2P_s}.\cos(\omega_o t) \qquad \text{...3.4.2.1}$$

Thus altogether when d(t) = 0 the phase of the carrier does not change at the beginning of the bit interval, while when d(t) = 1 there is a phase change of magnitude π.

Reception:

A method of recovering the data bit stream from the DPSK signal is shown in Fig. 3.4.2.3. Here the received signal and the received signal delayed by the bit time 1'" are applied to a multiplier. The multiplier output is

$b(t).b(t-T_b)2P_s\cos(\omega_o t+\theta).\cos[\omega_o(t-T_b)+\theta]$

$$= b(t).b(t-T_b)P_s\{\cos\omega_o T_b + \cos[2\omega_o(t-\tfrac{T_b}{2})+2\theta]\}$$

...3.4.2.2

and is applied to a bit synchronizer and integrator as shown in Fig. 6.2-1 for the BPSK demodulator. The first term on the right-hand side of Eq.(3.4.2.1) is, aside from a multiplicative constant, the waveform $b(t).b(t-T_b)$ which, as we shall see is precisely the signal we require. As noted previously in connection with BPSK, and so here, the output integrator will suppress the double frequency term. We should select $\omega_o T_b$ so that $\omega_o T_b = 2n\pi$ with n an integer. For, in this case we shall have $\cos\omega_o T_b = +1$ and the signal output will be as large as possible.

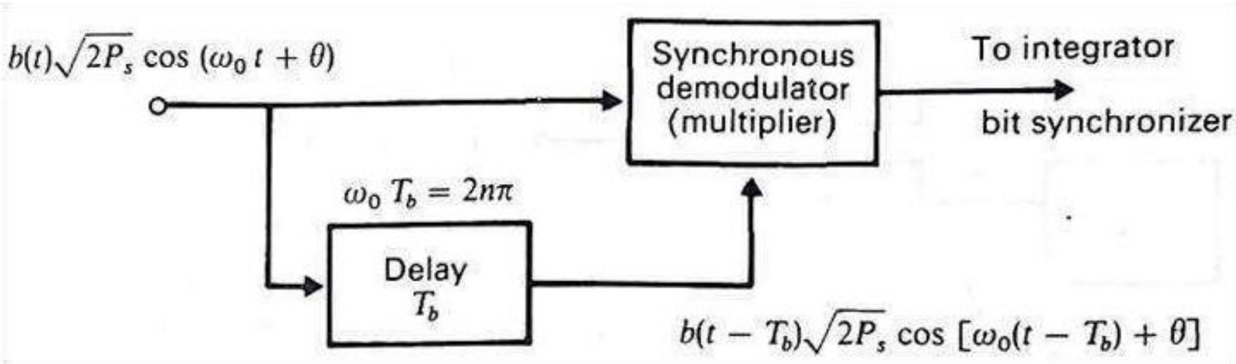


Figure 3.4.2.3 Methods of recovering data from DPSK

Further, with this selection, the bit duration encompasses an integral number of clock cycles and the integral of the double-frequency term is exactly zero.

The transmitted data bit d(t) can readily be determined from the product $b(t).b(t-T_b)$. If d(t) = 0 then there was no phase change and $b(t) = b(t-T_b)$ both being + 1V or both being - 1V. In this case $b(t).b(t-T_b)$= 1. If however, d(t) = 1 then there was a phase change and either b(t) = 1V with $b(t-T_b)$ = - I V or vice versa. In either case $b(t).b(t-T_b) = -1$.

OR

Differential Phase Shift Keying (DPSK) the phase of the modulated signal is shifted relative to the previous signal element. No reference signal is considered here. The signal phase follows the high or low state of the previous element. This DPSK technique doesn't need a reference oscillator.

The following figure represents the model waveform of DPSK.

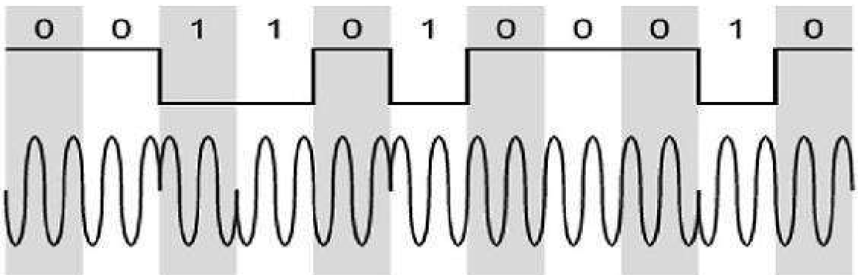


Figure 3.4.9 Differential Phase Shift Keying (DPSK)

It is seen from the above figure that, if the data bit is Low i.e., 0, then the phase of the signal is not reversed, but continued as it was. If the data is a High i.e., 1, then the phase of the signal is reversed, as with NRZI, invert on 1 (a form of differential encoding).

If we observe the above waveform, we can say that the High state represents an M in the modulating signal and the Low state represents a W in the modulating signal.

DPSK Modulator

DPSK is a technique of BPSK, in which there is no reference phase signal. Here, the transmitted signal itself can be used as a reference signal. Following is the diagram of DPSK Modulator.

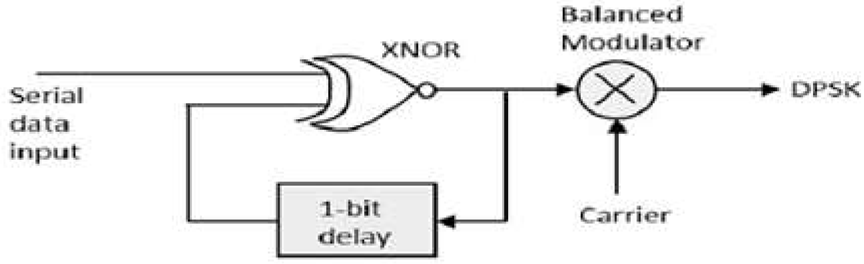


Figure 3.4.10 DPSK Modulator

DPSK encodes two distinct signals, i.e., the carrier and the modulating signal with 180° phase shift each. The serial data input is given to the XNOR gate and the output is again fed back to the other input through 1-bit delay. The output of the XNOR gate along with the carrier signal is given to the balance modulator, to produce the DPSK modulated signal.

DPSK Demodulator

In DPSK demodulator, the phase of the reversed bit is compared with the phase of the previous bit. Following is the block diagram of DPSK demodulator.

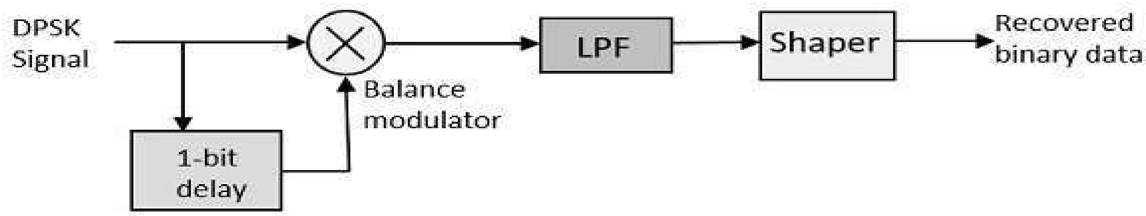


Figure 3.4.11 DPSK Demodulator

From the above figure, it is evident that the balance modulator is given the DPSK signal along with 1-bit delay input. That signal is made to confine to lower frequencies with the help of LPF. Then it is passed to a shaper circuit, which is a comparator or a Schmitt trigger circuit, to recover the original binary data as the output.

The word binary represents two bits. M represents a digit that corresponds to the number of conditions, levels, or combinations possible for a given number of binary variables.

This is the type of digital modulation technique used for data transmission in which instead of one bit, two or more bits are transmitted at a time. As a single signal is used for multiple bit transmission, the channel bandwidth is reduced.

3.4.3 Differentially Encoded Phase Shift Keying (DEPSK)

The DPSK demodulator requires a device which operates at the carrier frequency and provides a delay of T_b. Differentially-encoded PSK eliminates the need for such a piece of hardware. In this system, synchronous demodulation recovers the signal b(t), and the decoding of b(t) to generate d(t) is done at baseband.

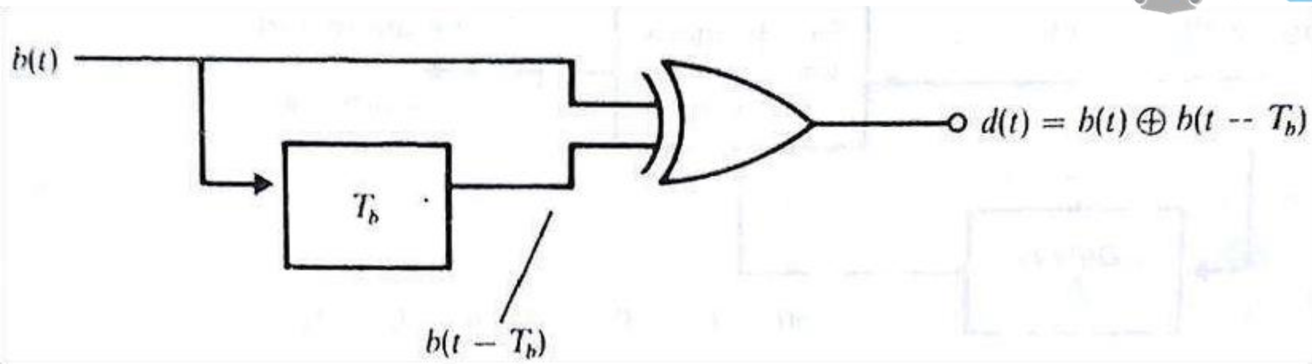


Figure 3.4.3.1 Baseband Decoder to obtain d(t) from b(t)

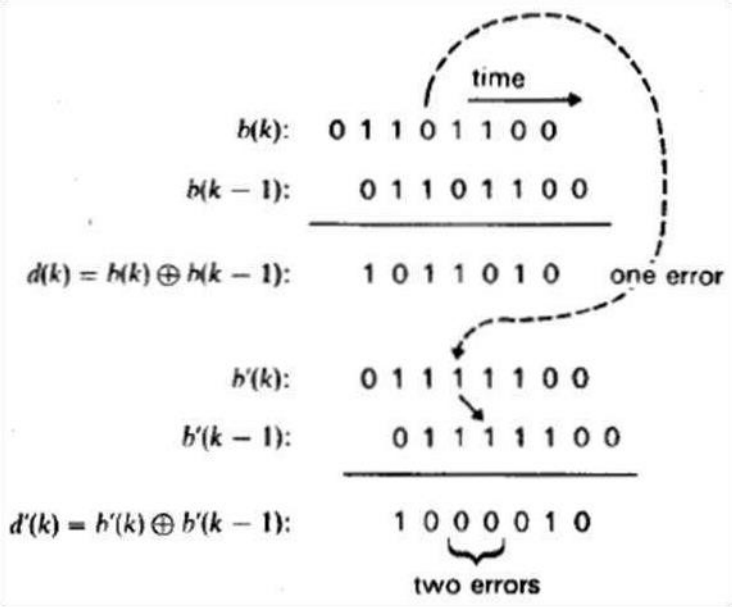


Figure 3.4.3.2 Errors in DEPSK occurs in pairs

The transmitter of the DEPSK system is identical to the transmitter of the DPSK system shown in Fig. 3.4.2.1, The signal b(t) is recovered in exactly the manner shown in Fig. 3.4.2.1 for a BPSK system. The recovered signal is then applied directly to one input of an exclusive-OR logic gate and to the other input is applied $b(t - T_b)$ (see Fig. 3.4.3.1). The gate output will be at one or the other of its levels depending on whether $b(t) = b(t - T_b)$ or $b(t) = \overline{b(t - T_b)}$. In the first case b(t) did not change level and therefore the transmitted bit is d(t) = 0. In the second case d(t) = 1.

We have seen that in DPSK there is a tendency for bit errors to occur in pairs but that single bit errors are possible. In DEPSK errors always occur in pairs. The reason for the difference is that in DPSK we do not make a hard decision, in each bit interval about the phase of the received signal. We simply allow the received signal in one interval to compare itself with the signal in an adjoining interval and, as we have seen, a single error is not precluded. In DEPSK, a firm definite hard decision is made in each interval about the value of b(t). If we make a mistake, then errors must result from a comparison with the preceding and succeeding bit. This result is illustrated in Fig.3.4.3.2. It is shown the error-free signals $b(k), b(k-1)$ and $d(k) = b(k) \oplus b(k-1)$. We have assumed that b'(k) has a single error. Then b'(k - 1) must also have a single error. We note that the reconstructed waveform d'(k) now has two errors.

3.4.4 Quadrature Phase Shift Keying (QPSK)

This is the phase shift keying technique, in which the sine wave carrier takes four phase reversals such as 0°, 90°, 180°, and 270°.

If these kinds of techniques are further extended, PSK can be done by eight or sixteen values also, depending upon the requirement.

QPSK Modulator

The mechanism by which a bit stream b(t) generates a QPSK signal for transmission is shown in Fig. 3.4.4.1 and relevant waveforms are shown in Fig. 3.4.4.2. In these waveforms we have arbitrarily assumed that in every case the active edge of the clock waveforms is the downward edge. The toggle flip-flop is driven by a clock waveform whose period is the bit time Tb. The toggle flip-flop generates an odd clock waveform and an even waveform. These clocks have periods 2Tb. The active edge of one of the clocks and the active edge of the other are separated by the bit time Tb. The bit stream $b(t)$ is applied as the data input to both type-D flip-flops, one driven by the odd and one driven by the even clock waveform. The flip-flops register alternate bits in the stream b(t) and hold each such registered bit for two bit intervals, that is for a time Tb. In Fig. 3.4.4.2 we have numbered the bits in $b(t)$. Note that the bit stream $b(t)$ (which is the output of the flip-flop driven by the odd clock) registers bit 1 and holds that bit .for time 2Tb, then registers bit 3 for time 2Tb, then bit 5 for 2Tb, etc. The even bit stream $b_e(t)$ holds, for times 2Tb each the alternate bits numbered 2.4, 6, etc.
The bit stream $b(t)$ (which. as usual, we take to be $b_e(t)$= ± 1 volt) is superimposed on a carrier $\sqrt{P_s}.\ \sin(\omega_o t)$ by the use of two multipliers (i.e., balanced modulators) as shown, to generate two signals $S_e(t)$ and $S_o(t)$. These signals are then added to generate the transmitted output signal Vm(t) which is

$$v_m(t) = \sqrt{P_s}.\, b_o(t)\sin(\omega_o t) + \sqrt{P_s}.\, b_e(t)\cos(\omega_o t)$$

As may be verified, the total normalized power of $v_m(t)$ is P_s.

The QPSK Modulator uses a bit-splitter, two multipliers with local oscillator, a 2-bit serial to parallel converter, and a summer circuit. Following is the block diagram for the same.

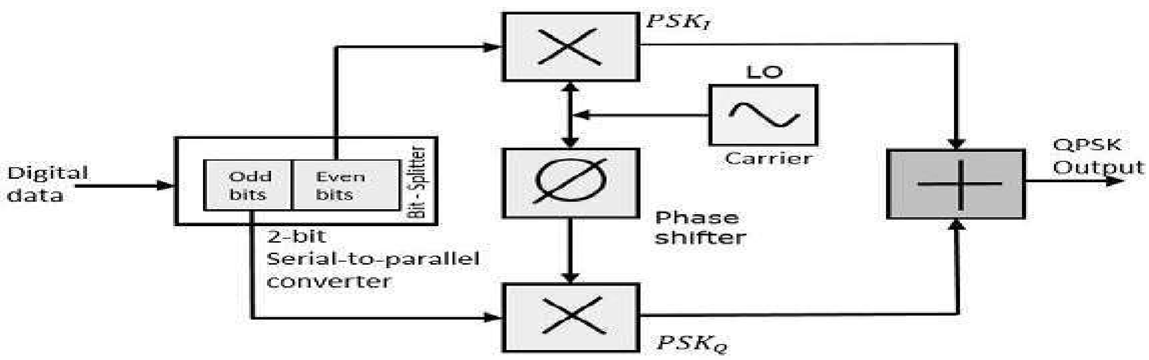


Figure 3.4.6 QPSK Modulator

At the modulator's input, the message signal's even bits (i.e., 2nd bit, 4th bit, 6th bit, etc.) and odd bits (i.e., 1st bit, 3rd bit, 5th bit, etc.) are separated by the bits splitter and are multiplied with the same carrier to generate odd BPSK (called as PSK_I) and even BPSK (called as PSK_Q). The PSK_Q signal is anyhow phase shifted by 90° before being modulated.

The QPSK waveform for two-bits input is as follows, which shows the modulated result for different instances of binary inputs.

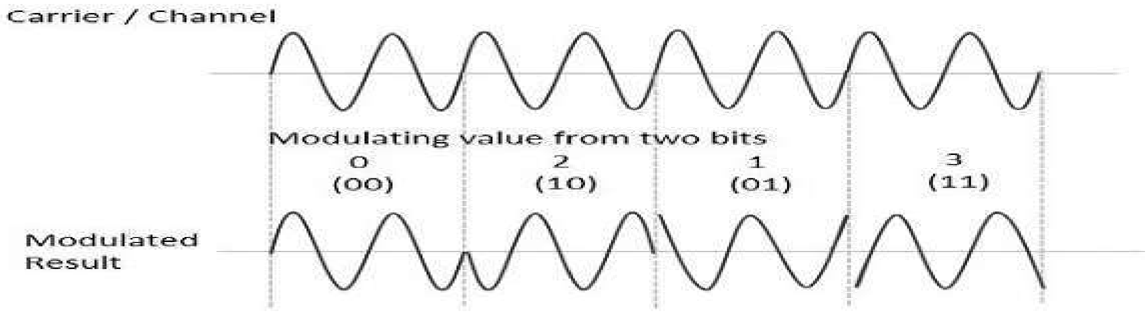


Figure 3.4.7 QPSK Waveforms

QPSK Demodulator

The QPSK Demodulator uses two product demodulator circuits with local oscillator, two band pass filters, two integrator circuits, and a 2-bit parallel to serial converter. Following is the diagram for the same.

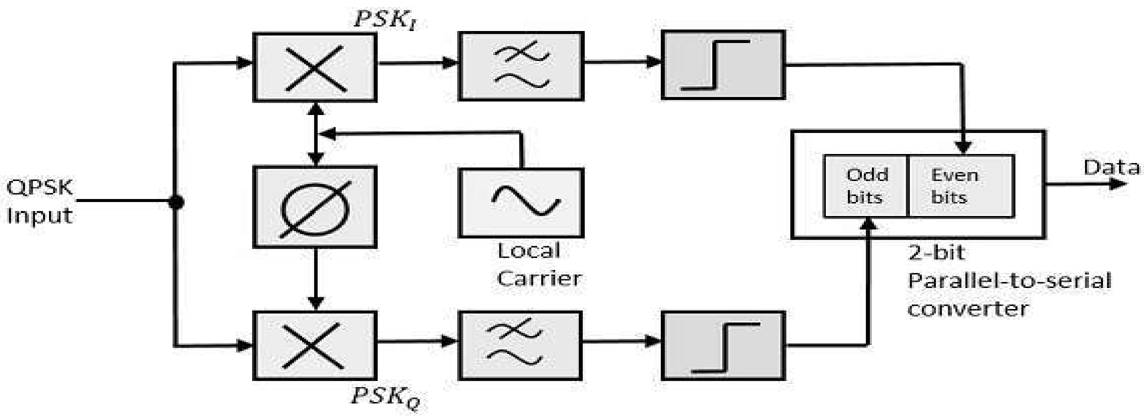


Figure 3.4.8 QPSK Demodulator

The two product detectors at the input of demodulator simultaneously demodulate the two BPSK signals. The pair of bits are recovered here from the original data. These signals after processing, are passed to the parallel to serial converter.

3.5 M-ary Equation

If a digital signal is given under four conditions, such as voltage levels, frequencies, phases, and amplitude, then M = 4. The number of bits necessary to produce a given number of conditions is expressed mathematically as $N=\log 2M$. Where N is the number of bits necessary M is the number of conditions, levels, or combinations possible with N bits.

The above equation can be re-arranged as

$$2N=M$$

For example, with two bits, $2^2 = 4$ conditions are possible.

Types of M-ary Techniques

In general, Multi-level (M-ary) modulation techniques are used in digital communications as the digital inputs with more than two modulation levels are allowed on the transmitter's input. Hence, these techniques are bandwidth efficient.

There are many M-ary modulation techniques. Some of these techniques, modulate one parameter of the carrier signal, such as amplitude, phase, and frequency.

M-ary ASK

This is called M-ary Amplitude Shift Keying (M-ASK) or M-ary Pulse Amplitude Modulation (PAM).

The amplitude of the carrier signal, takes on M different levels.

Representation of M-ary ASK

$S_m(t) = A_m \cos(2\pi f_c t) \qquad A_m \in (2m - 1 - M)\Delta, m = 1,2, \ldots \ldots M$ and $0 \le t \le T_s$

Some prominent features of M-ary ASK are −

- This method is also used in PAM.
- Its implementation is simple.
- M-ary ASK is susceptible to noise and distortion.

M-ary FSK

This is called as M-ary Frequency Shift Keying (M-ary FSK).

The frequency of the carrier signal, takes on M different levels.

Representation of M-ary FSK

$$S_i(t) = \sqrt{\frac{2E_s}{T_s}} \cos\left(\frac{\pi}{T_s}(n_c + i)t\right) \qquad 0 \le t \le T_s \quad i = 1,2, \ldots \ldots M$$

Where $fc = nc/\ 2Ts$

for some fixed integer n.

Some prominent features of M-ary FSK are −

- Not susceptible to noise as much as ASK.
- The transmitted M number of signals are equal in energy and duration.
- The signals are separated by 12Ts
- Hz making the signals orthogonal to each other.
- Since M signals are orthogonal, there is no crowding in the signal space.
- The bandwidth efficiency of M-ary FSK decreases and the power efficiency increases with the increase in M.

M-ary PSK

This is called as M-ary Phase Shift Keying (M-ary PSK).

The phase of the carrier signal, takes on M different levels.

Representation of M-ary PSK

$$S_i(t) = \sqrt{\frac{2E}{T}} \cos(\omega_0 t + \phi_i t) \qquad 0 \le t \le T \ and \quad i = 1,2, \ldots \ldots M$$

$$\phi_i(t) = \frac{2\pi i}{M} \text{ where } i = 1,2, \ldots \ldots M$$

Some prominent features of M-ary PSK are −

- The envelope is constant with more phase possibilities.
- This method was used during the early days of space communication.
- Better performance than ASK and FSK.
- Minimal phase estimation error at the receiver.
- The bandwidth efficiency of M-ary PSK decreases and the power efficiency increases with the increase in M.

So far, we have discussed different modulation techniques. The output of all these techniques is a binary sequence, represented as 1s and 0s. This binary or digital information has many types and forms, which are discussed further.

3.6 Comparison of BPSK and BFSK

The two modulation techniques can be compared in the following manners:

1. The bandwidth required for transmitting the BFSK signal is 4f (f is the frequency of the data signal), whereas the bandwidth requirement for the BPSK signal is only 2f. Therefore BPSK is a better option.

2. In BPSK the information of the message is stored in the phase variations of the career wave whereas in case of the BFSK scheme, the information is available as the frequency variations of the career wave. Now we know that the noise can affect the frequency of the career wave but cannot affect the phase of the career signal. Therefore the BPSK Scheme is again a better option.

3. The noise will also occupy some frequency it may damage the signal flatly or frequency selectively. But in PSK there is very less chance in the changing of phase f the signal. Hence for noisy channel PSK is better than FSK.

3.7 Quadrature Phase Shift Keying (QPSK)

We have seen that when a data stream whose bit duration is T_b is to be transmitted by BPSK the channel bandwidth must be nominally $2f_b$ where $f_b = I/T_b$.
Quadrature phase-shift keying allows bits to be transmitted using half the bandwidth. In QPSK system we use the type-D flip-flop as a one bit storage device.

D Flip Flop

The type-D flip-flop represented in Fig3.7.4.1 has a single data input terminal (D) to which a data stream d(t) is applied. The operation of the flip-flop is such that at the" active" edge of the clock waveform the logic level at D is transferred to the output Q. Representative waveforms are shown in Fig. 3.7.4.2 We assume arbitrarily that the negative-going edge of the clock waveform is the active edge.

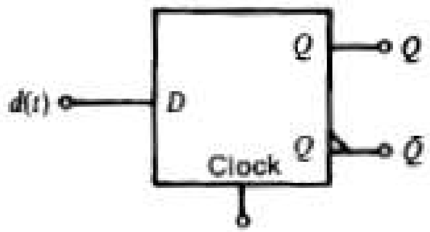


Figure 3.7.4.1 D Flip Flop

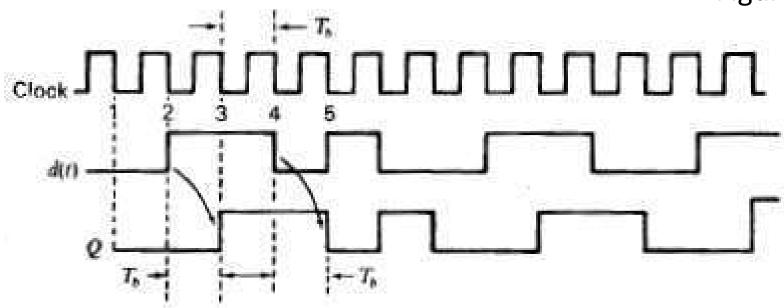


Figure 3.7.4.2 Waveforms showing D Flip Flop Characteristics

QPSK Transmitter (Modulator):
The mechanism by which a bit stream b(t) generates a QPSK signal for transmission is shown in Fig. 3.7.4.3 and relevant waveforms are shown in Fig. 3.7.4.4. In these waveforms we have arbitrarily assumed that in every case the active edge of the clock waveforms is the downward edge. The toggle flip-flop is driven by a clock waveform whose period is the bit time Tb. The toggle flip-flop generates an odd clock waveform and an even waveform. These clocks have periods 2Tb. The active edge of one of the clocks and the active edge of the other are separated by the bit time Tb. The bit stream $b(t)$ is applied as the data input to both type-D flip-flops, one driven by the odd and one driven by the even clock waveform. The flip-flops register alternate bits in the stream b(t) and hold each such registered bit for two bit intervals, that is for a time Tb. In Fig. 3.4.4.4 we have numbered the bits in $b(t)$. Note that the bit stream $b(t)$ (which is the output of the flip-flop driven by the odd clock) registers bit 1 and holds that bit .for time 2Tb, then registers bit 3 for time 2Tb, then bit 5 for 2Tb, etc. The even bit stream $b_e(t)$ holds, for times 2Tb each the alternate bits numbered 2.4, 6, etc.

The bit stream $b(t)$ (which. as usual, we take to be $b_e(t)$= ± 1 volt) is superimposed on a carrier $\sqrt{P_s}.\ \sin(\omega_o t)$ by the use of two multipliers (i.e., balanced modulators) as shown, to generate two signals $S_e(t)$ and $S_o(t)$. These signals are then added to generate the transmitted output signal Vm(t) which is

$$v_m(t) = \sqrt{P_s}.\, b_o(t)\sin(\omega_o t) + \sqrt{P_s}.\, b_e(t)\cos(\omega_o t)$$

As may be verified, the total normalized power of $v_m(t)$ is P_s.

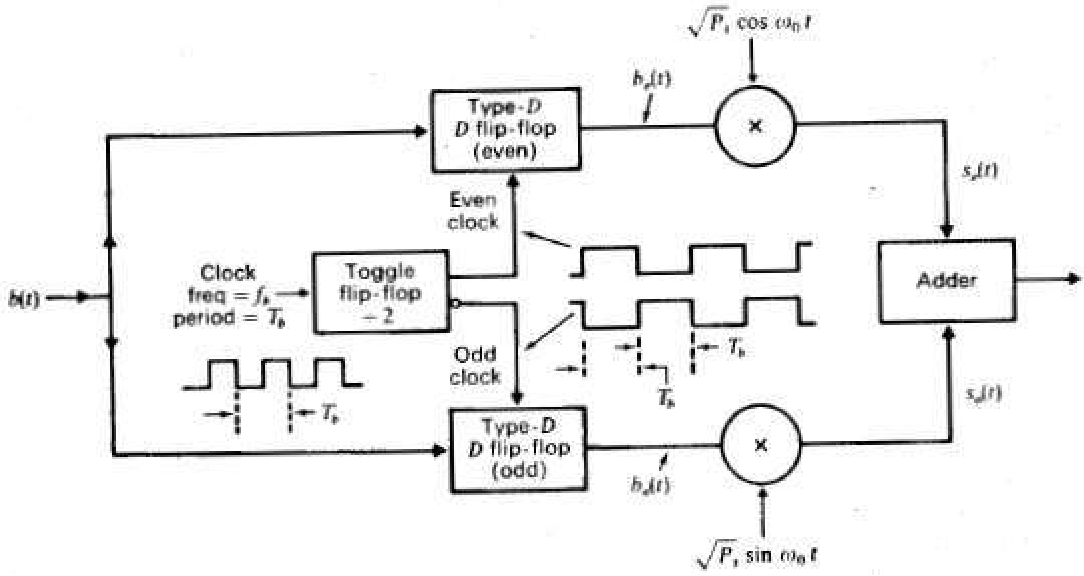


Figure 3.7.4.3 QPSK Transmitter

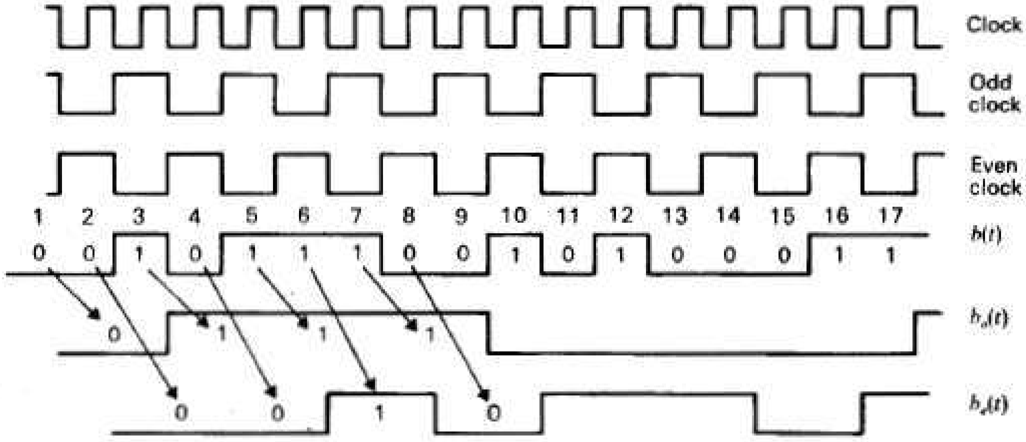


Figure 3.7.4.3 Waveforms for the QPSK Transmitter

In BPSK, the bit duration is T_b, and the generated signal has a nominal bandwidth of $2x1/T_b$. In the waveforms of $b_o(t)$ and $b_e(t)$, the bit times are each $1/2T_b$, hence both $b_o(t)$ and $b_e(t)$ have nominal bandwidth which are half of the bandwidth in BPSK.

Phasor Diagram:

When b_0 = 1 the signal $s_o(t)$ = $\sqrt{P_s}.\ \sin(\omega_o t)$, and $s_o(t)$ = $-\sqrt{P_s}.\ \sin(\omega_o t)$ when b_o = -1. Correspondingly, for $b_e(t)$ = ± 1, $s_e(t)$ = ±.$\sqrt{P_s}.\ (t)\ \cos(\omega_o t)$ These four signals have been represented as phasors in Fig. 3.7.4.4. They are in mutual phase quadrature. Also drawn are the phasors representing the four possible output signals $v_m(t)\ =\ so(t)\ +\ Se(t)$. These four possible output signals have equal amplitude $\sqrt{2P_s}$ and are in phase quadrature; they have been identified by their corresponding values of b_o and b_e. At the end of each bit interval (i.e., after each time T_b) either b_o, or b_e can change, but both cannot change at the same time. Consequently, the QPSK system shown in Fig. 3.7.4.3 is called offset or staggered QPSK and abbreviated OQPSK. After each time T_b, the transmitted signal, if it changes, changes phase by 90° rather than by 180° as in BPSK.

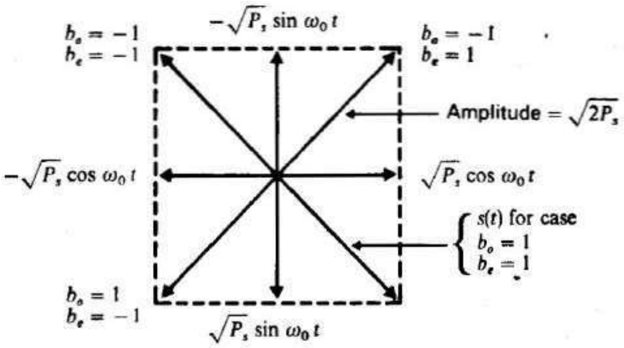


Figure 3.7.4.4 Phasor diagrams for the sinusoids of Fig 3.7.4.2

Non-offset QPSK

Suppose that in Fig. 3.7.4.3 we introduce an additional flip-flop before either the odd or even flip-flop. Let this added flip-flop be driven by the clock which runs at the rate f_b. Then one or the other bit streams, odd or even, will be delayed by one bit interval. As a result, we shall find that two bits which occur in time sequence (i.e., serially) in the input bit stream b(t) will appear at the same time (i.e., in parallel) at the outputs of the odd and even flip-flops. In this case $b_e(t)$ and $b_o(t)$ can change at the same time, after each time $2T_b$, and there can be a phase change of 180° in the output signal. There is no difference, in principle, between a staggered and non-staggered system.

In practice, there is often a significant difference between QPSK and OQPSK. At each transition time, T" for OQPSK and $2T_b$ for QPSK, one bit for OQPSK and perhaps two bits for QPSK change from 1V to -1V or -1V to 1V. Now the bits be(t) and bo(t) can, not change instantaneously and, in changing, must pass through zero and dwell in that neighborhood at least briefly. Hence there will be brief variations in the amplitude of the transmitted waveform. These variations will be more pronounced in QPSK than in OQPSK since in the first case both be(t) and bo(t) may be zero simultaneously so that the signal amplitude may actually be reduced to zero temporarily.

Symbol versus Bit Transmission

In BPSK we deal individually with each bit of duration T_b. In QPSK we lump two bits together to form what is termed a symbol. The symbol can have anyone of four possible values corresponding to the two-bit sequences 00, 01, 10, and 11. We therefore arrange to make available for transmission four distinct signals. At the receiver each signal represents one symbol and, correspondingly, two bits. When bits are transmitted, as in BPSK, the signal changes occur at the bit rate. When symbols are transmitted the changes occur at the symbol rate which is one-half the bit rate. Thus the symbol time is $T_s=2T_b$.

The QPSK Receiver

A receiver for the QPSK signal is shown in Fig. 3.7.4.5. Synchronous detection is required and hence it is necessary to locally regenerate the carriers $\cos\,\omega_o t$ and $\sin\,\omega_o t$. The scheme for carrier regeneration is similar to that employed in BPSK. In that earlier case we squared the incoming signal, extracted a waveform at twice the carrier frequency by filtering, and recovered the carrier by frequency dividing by two. In the present case, it is required that the incoming signal be raised to the fourth power after which filtering recovers a waveform at four times the carrier frequency and finally frequency division by four regenerates the carrier. In the present case, also, we require both $\cos\omega_o t$ and $\sin\omega_o t$.

The incoming signal is also applied to two synchronous demodulators consisting, as usual, of a multiplier (balanced modulator) followed by an integrator. The integrator integrates over a two-bit interval of duration $T_s=2T_b$ and then dumps its accumulation. As noted previously, ideally the interval 2Tb = T_s should encompass an integral number of carrier cycles. One demodulator uses the carrier $\cos\,\omega_o t$ and the other the carrier $\sin\,\omega_o t$. We recall that when sinusoids in phase quadrature are multiplied, and the product is integrated over an integral number of cycles, the result is zero. Hence the demodulators will selectively respond to the parts of the incoming signal involving respectively $b_e(t)$ or $b_o(t)$.

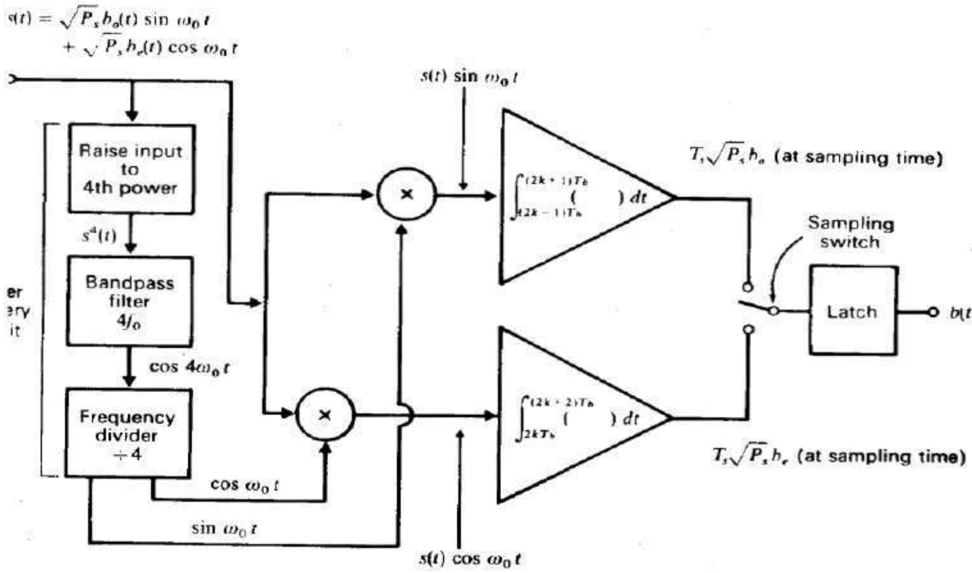


Figure 3.7.4.5 A QPSK Receiver

Of course, as usual, a bit synchronizer is required to establish the beginnings and ends of the bit intervals of each bit stream so that the times of integration can be established. The bit synchronizer is needed as well to operate the sampling switch. At the end of each integration time for each individual integrator, and just before the accumulation is dumped, the integrator output is sampled. Samples are taken alternately from one and the other integrator output at the end of each bit time T_b and these samples are held in the latch for the bit time T_b Each individual integrator output is sampled at intervals 2 T_b. The latch output is the recovered bit stream b(t).

The voltages marked on Fig. 3.7.4.5 are intended to represent the waveforms of the signals only and not their amplitudes. Thus the actual value of the sample voltages at the integrator outputs depends on the amplitude of the local carrier, the gain, if any, in the modulators and the gain in the integrators. We have however indicated that the sample values depend on the normalized power P_s of the received signal and on the duration T_s of the symbol.

3.8 M ARY PSK

In BPSK we transmit each bit individually. Depending on whether b(t) is logic 0 or logic 1, we transmit one or another of a sinusoid for the bit time T_b, the sinusoids differing in phase by 2π/2 = 180^0. In QPSK we lump together two bits. Depending on which of the four two-bit words develops, we transmit one or another of four sinusoids of duration 2T_b the sinusoids differing in phase by amount 2π/4 = 90°. The scheme can be extended. Let us lump together N bits so that in this N-bit symbol, extending over the time NT_b, there are 2N = M possible symbols. Now let us represent the symbols by sinusoids of duration NT_b= T_s which differ from one another by the phase 2 π / M. Hardware to accomplish such M-ary communication is available.

Thus in M-ary PSK the waveforms used to identify the symbols are

$$v_m(t) = \sqrt{2P_s}\cos(\omega_o t + \emptyset_m) \text{ (m=0, 1, …, M-1)} \qquad \text{…3.8.1}$$

Where phase angle is given by

$$\emptyset_m = (2m+1)\frac{\pi}{M} \qquad \text{…3.8.2}$$

The waveforms of Eq. (3.8.1) are represented by the dots in Fig. 3.8.1 in a signal space in which the coordinate axes are the orthonormal waveforms $u1(t) = \sqrt{2/T_s}\cos(\omega_o t)$ and $u2(t) = \sqrt{2/T_s}\sin(\omega_o t)$. The distance of each dot from the origin is $\sqrt{E_s} = \sqrt{P_s T_s}$

From Eq. (3.5.1) we have

$$v_m(t) = (\sqrt{2P_s}\cos\emptyset_m)\cos(\omega_o t) - (\sqrt{2P_s}\sin\emptyset_m)\sin(\omega_o t) \qquad \text{…3.8.3}$$

Defining p_e and p_o by

$$p_e = \sqrt{2P_s}\cos \emptyset_m \qquad ...3.8.4a$$
$$p_o = \sqrt{2P_s}\sin \emptyset_m \qquad ...3.8.4b$$

Equation 3.5.3 becomes

$$v_m(t) = p_e\cos(\omega_o t) - p_o\sin(\omega_o t) \qquad ...3.8.5$$

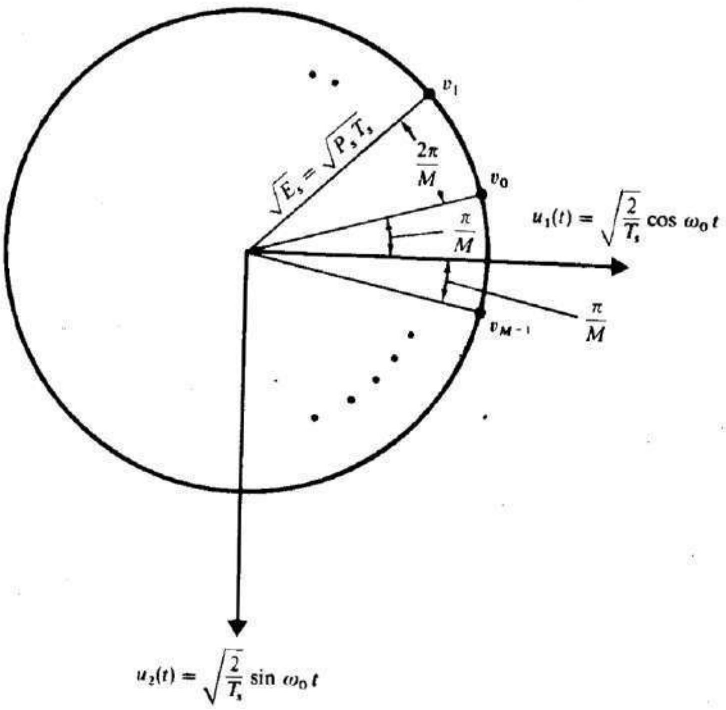


Figure 3.8.1 Graphical representation of M-ary PSK Signals

M-ary Transmitter and Receiver

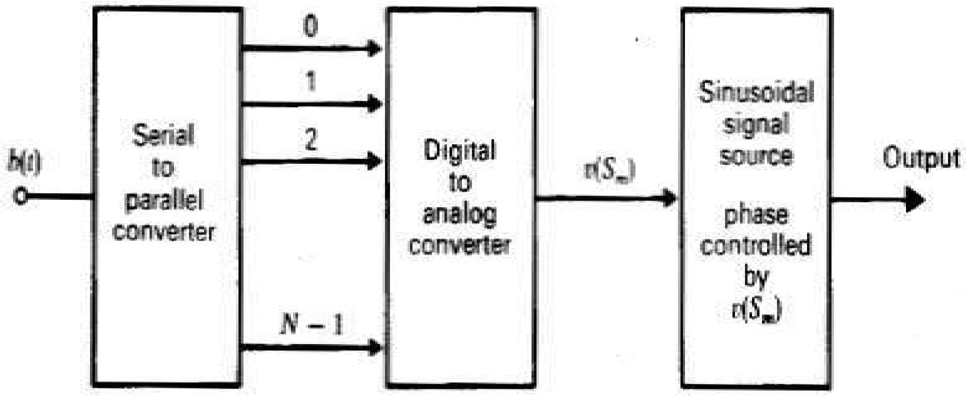


Figure 3.8.2 M Ary Transmitter

The transmitter, the bit stream b(t) is applied to a serial-to-parallel converter. This converter has facility for storing the N bits of a symbol. The N bits have been presented serially, that is, in time sequence, one after another. These N bits, having been assembled, are then presented all at once on N output lines of the converter, that is they are presented in parallel. The converter output remains unchanging for the duration NTb of a symbol during which time the converter is assembling a new group of N bits. Each symbol time the converter output is updated.

The converter output is applied to a D/A converter. This D/A converter generates an output voltage which assumes one of 2N = M different values in a one to-one correspondence to the M possible symbols applied to its input. That is, the D/A output is a voltage v(Sm) which depends on the symbol Sm (m = 0, 1,... ,M - 1). Finally v(Sm) is applied as a control input to a special type of constant amplitude sinusoidal signal source whose phase 4>m is determined by v(Sm). Altogether, then, the output is a fixed amplitude, sinusoidal waveform, whose phase has a one-to-one correspondence to the assembled N-bit symbol. The phase can change once per symbol time.

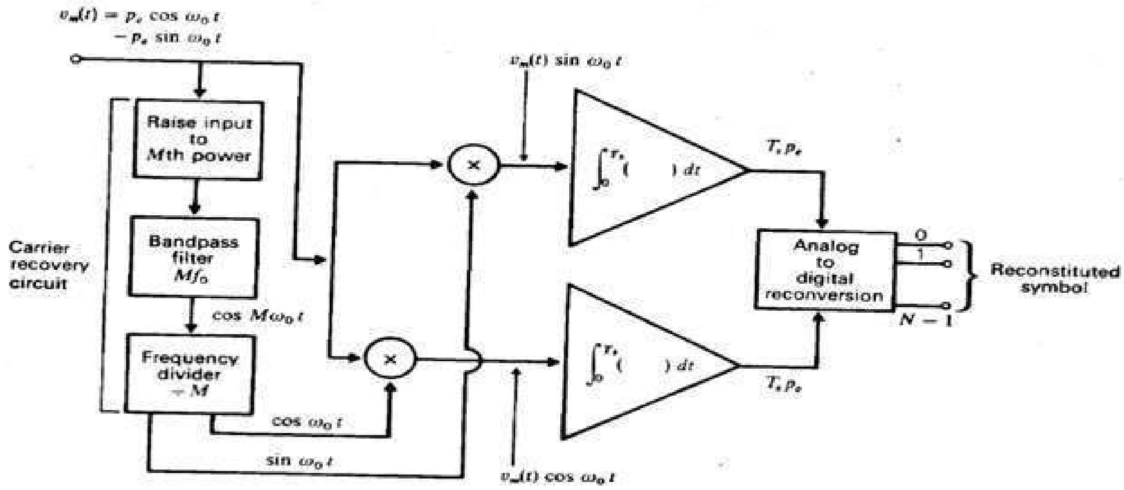


Figure 3.8.3 M Ary Transmitter

The carrier recovery system requires, in the present case a device to raise the received signal to the Mth power, filter to extract the Mfo component and then divide by M.

Since there is no staggering of parts of the symbol, the integrators extend their integration over the same time interval. Of course, again, a bit synchronizer is needed.

The integrator outputs are voltages whose amplitudes are proportional to TsPe and TsPo respectively and change at the symbol rate. These voltages measure the components of the received signal in the directions of the quadrature phasors sinw_0t and cosw_0t. Finally the signals TsPe and TsPo are applied to a device which reconstructs the digital N-bit signal which constitutes the transmitted signal.

Current operating systems are common in which M = 16. In this case the bandwidth is B = 2fh/4 =fb/2 in comparison to B =fb for QPSK. PSK systems transmit information through signal phase and not through signal amplitude. Hence such systems have great merit in situations where, on account of the vagaries of the transmission medium, the received signal varies in amplitude.

3.9 Quadrature Amplitude Shift Keying (QASK)

In BPSK, QPSK, and M-ary PSK we transmit, in any symbol interval, one signal or another which are distinguished from one another in phase but are all of the same amplitude. In each of these individual systems the end points of the signal vectors in signal space falls on the circumference of a circle. Now we have noted that our ability to distinguish one signal vector from another in the presence of noise will depend on the distance between the vector end points. It is hence rather apparent that we shall be able to improve the noise immunity of a system by allowing the signal vectors to differ, not only in their phase but also in amplitude. We now describe such an amplitude and phase shift keying system. Like QPSK it involves direct (balanced) modulation of carriers in quadrature (i.e., $\cos\,\omega_0 t$ and $\sin\,\omega_0 t$) in quadrature and hence abbreviated as QAPSK or simply QASK.

For Example consider to transmit a symbol for every 4 bits. There are then 2^4= 16 different possible symbols and we shall have to be able to generate 16 distinguishable signals. One possible geometrical representation is shown in figure 3.9.1. In this configuration each signal point is equally distant from its neighbors, the distance being d=2a.

Let us assume that all 16 signals are equally likely. Because of the symmetrical placement around the origin, we can determine the average energy associated with a signal, from the four signals in the first quadrant. The average normalized energy of a signal is

$$E_s = \frac{1}{4}\left[(a^2 + a^2) + (9a^2 + a^2) + (a^2 + 9a^2) + (9a^2 + 9a^2)\right] \qquad \text{...3.9.1}$$

$$E_s = 10a^2$$

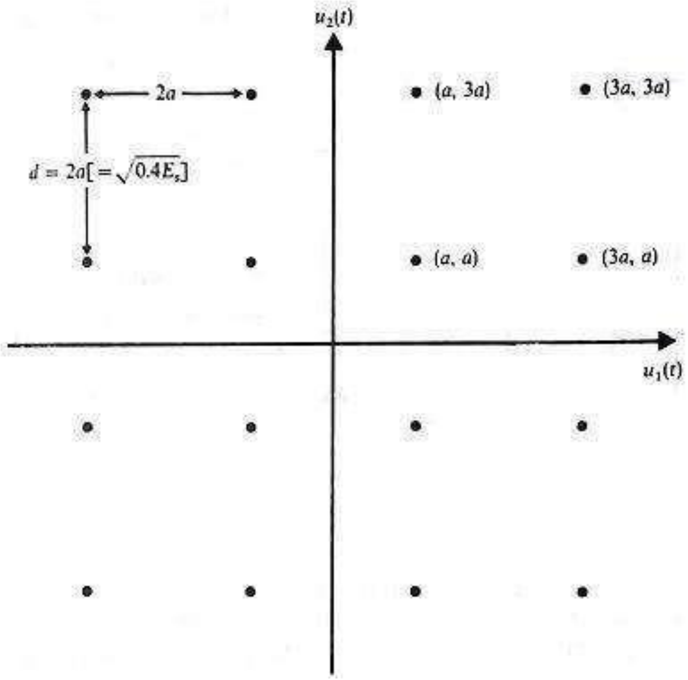


Figure 3.9.1 Geometrical Representation of 16 signals QASK

$$a = \sqrt{0.1E_s} \qquad \text{...3.9.2}$$

$$d = 2\sqrt{0.1E_s} \qquad \text{...3.9.3}$$

In present case each symbol represent 4 bits, the normalized symbol energy is $E_s = 4E_b$ where E_b is the normalized bit energy. Therefore

$$a = \sqrt{0.1E_s} = \sqrt{0.4E_b}$$

$$\text{and } d = 2\sqrt{0.4E_b} \qquad \text{...3.9.4}$$

This distance is significantly less than the distance between adjacent QPSK signals where, $d = 2\sqrt{E_b}$; however the distance is greater than 16 MPSK where

$$d = \sqrt{16E_b \sin^2\frac{\pi}{16}} = 2\sqrt{0.15E_b} \qquad \text{...3.9.5}$$

Thus 16 QASK will have a lower error rate than 16 MPSK, but higher than QPSK.

Generation of QPSK

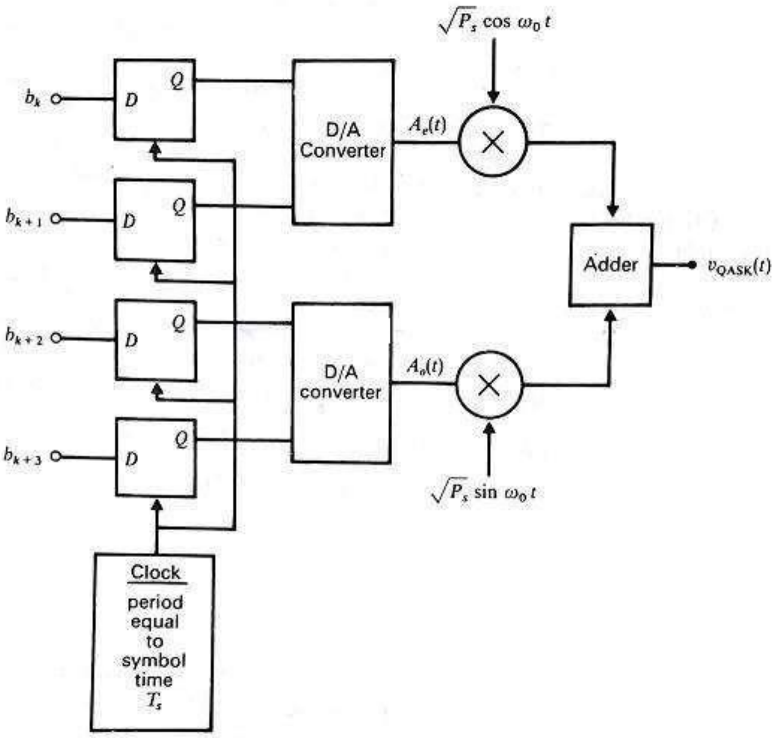


Figure 3.9.2 Generation of QPSK Signal

QASK generator for 4 bit symbol is shown. The 4 bit symbol $b_{k+3}\, b_{k+2}\, b_{k+1}\, b_k$ is stored in the 4 bit register made up of four flip flops. A new symbol is presented once per interval $T_s = 4T_b$ and the content of the register is correspondingly updated at each active edge of the clock which also have a period T_s. Two bits are presented to one D/A converter and two to second converter. The converter output $A_e(t)$ modulates the balanced modulator whose input career is the even function $\sqrt{P_s}\,\cos(\omega_o t)$ and $A_o(t)$ modulates the modulator whose input career is the odd function career. Then the transmitted signal is

$$v_{QASK}(t) = A_e(t)\sqrt{P_s}.\cos(\omega_o t) + A_o(t)\sqrt{P_s}.\sin(\omega_o t) \quad \text{...3.9.6}$$

Bandwidth of QASK

The Bandwidth of the QASK signal is

$$B = 2f_b/N$$

Which is the same as in the case of M ary PSK. With N=4 corresponding to 16 possible distinguishable signals we have $B_{QASK(16)} = f_b/2$ which is one fourth of the bandwidth required for binary PSK.

QASK Receiver

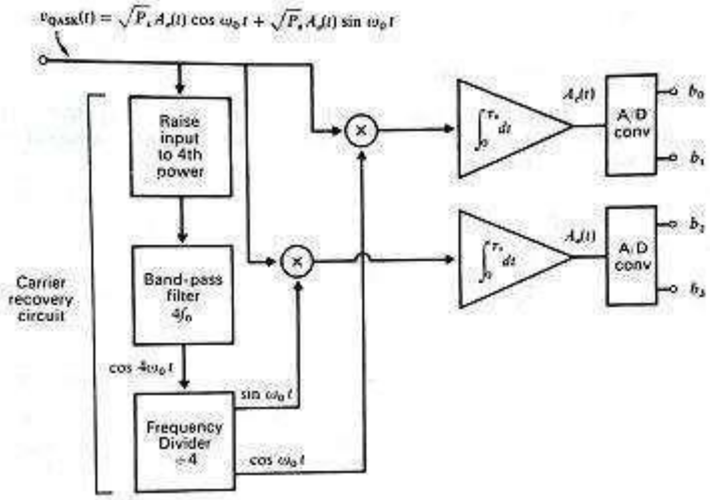


Figure 3.9.3 The QASK Receiver

It is similar to QPSK receiver, where a set of quadrature careers for synchronous demodulation is generated by raising the received signal to the power 4, extracting the components at frequency $4f_o$ and then dividing the frequency by 4.

In present case since the coefficients A_e and A_o are not of fixed value we have to enquire whether the career is still recoverable. We have

$$v^4_{QASK}(t) = P_s^2[A_e(t).\cos(\omega_o t) + A_o(t).\sin(\omega_o t)]^4 \quad \text{...3.9.7}$$

Neglecting all the terms not at the frequency $4f_o$, we have

$$\frac{v^4_{QASK}(t)}{P_s} = \left[\frac{A_e^4(t) + A_o^4(t) - 6A_e^2(t)A_o^2(t)}{8}\right]\cos(4\omega_o t) + \left[\frac{A_e(t)A_o(t)[A_e^2(t) - A_o^2(t)]}{2}\right]\sin(4\omega_o t) \quad \text{...3.9.8}$$

The average value of the coefficient of $\cos 4\omega_o t$ is not zero whereas the average value of the coefficient of $\sin 4\omega_o t$ is zero. Thus a narrow filter centered at $4f_o$ will recover the signal at $4f_o$.

After getting the careers, two balanced modulators together with two integrators recover the signals $A_e(t)$ and $A_o(t)$ as shown in figure. The integrators have an integration time equal to the symbol time T_s. Finally the original input bits are recovered by using A/D Converter.

3.10 Binary Frequency Shift Keying (BFSK)

In binary frequency-shift keying (BFSK) the binary data waveform d(t) generates a binary signal

$$v_{BFSK}(t) = \sqrt{2P_s}\cos[\omega_o t + d(t)\Omega t] \qquad \text{...3.10.1}$$

Here d(t) = + 1 or -1 corresponding to the logic levels 1and 0 of the data waveform. The transmitted signal is of amplitude $\sqrt{2P_s}$ and is either

$$v_{BFSK}(t) = S_H(t) = \sqrt{2P_s}\cos(\omega_o + \Omega)\,t \qquad \text{...3.10.2}$$

$$v_{BFSK}(t) = S_L(t) = \sqrt{2P_s}\cos(\omega_o - \Omega)\,t \qquad \text{...3.10.3}$$

and thus has an angular frequency $\omega_o + \Omega$ or $\omega_o - \Omega$ with Ω a constant offset from the nominal carrier frequency ω_o. We shall call the higher frequency ω_H(= $\omega_o + \Omega$) and the lower frequency ω_L.(= $\omega_o - \Omega$). We may conceive that the BFSK signal is generated in the manner indicated in Fig. 3.7.1. Two balanced modulators are used, one with carrier ω_H and one with carrier ω_L. The voltage values of P_H(t) and of P_L(t) are related to the voltage values of d(t) in the following manner

d(t)	P_H(t)	P_L(t)
+1V	+1V	0V
-1V	0V	+1V

Thus when d(t) changes from +1 to -1 P_H changes from 1 to 0 and P_L from 0 to 1. At any time either P_H or P_L is 1 but not both so that the generated signal is either at angular frequency ω_H or at ω_L.

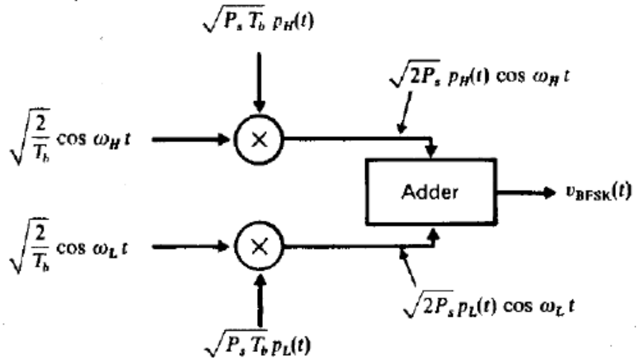


Figure 3.10.1 A representation of a manner in which a BFSK signal can be generated.

Spectrum of BPSK

In terms of variables P_H and P_L the BFSK signal is given by

$$v_{BFSK}(t) = \sqrt{2P_s}P_H\cos[\omega_H t + \theta_H] + \sqrt{2P_s}P_L\cos[\omega_L t + \theta_L] \qquad \text{...3.10.4}$$

where we have assumed that each of the two signals are of independent and random, uniformly distributed phase. Each of the terms in Eq. (3.10.4) looks like the signal $\sqrt{2P_s}b(t)\ \cos\,\omega_o t$ which we encountered in BPSK and for which we have already deduced the spectrum, but there is an important difference. In the BPSK case, b(t) is bipolar, i.e., it alternates between + 1 and – 1 while in the present case P_H and P_L are unipolar, alternating between + 1 and 0. We may, however, rewrite P_H and P_L as the sums of a constant and a bipolar variable, that is

$$P_H(t) = \frac{1}{2} + \frac{1}{2}P'_H(t) \qquad \text{...3.10.5a}$$

$$P_L(t) = \frac{1}{2} + \frac{1}{2}P'_L(t) \qquad \text{...3.10.5b}$$

In above equations $P'_H(t)$ and $P'_L(t)$ are bipolar alternating between +1 and -1 and are complementary i.e. when $P'_H(t) = +1$, $P'_L(t) = -1$ and vice versa. Then from equation 4,

$$v_{BFSK}(t) = \sqrt{\frac{P_s}{2}}\cos[\omega_H t + \theta_H] + \sqrt{\frac{P_s}{2}}\cos[\omega_L t + \theta_L] + \sqrt{\frac{P_s}{2}}P'_H\cos[\omega_H t + \theta_H] + \sqrt{\frac{P_s}{2}}P'_L\cos[\omega_L t + \theta_L] \quad \ldots 3.10.6$$

The first two terms in Eq. (3.10.6) produce a power spectral density which consists of two impulses, one at f_H and one at f_L. The last two terms produce the spectrum of two binary PSK signals one centered about f_H and one about f_L. The individual power spectral density patterns of the last two terms are for the case $f_H - f_L = 2f_b$. For this separation between f_H and f_L we observe that the overlapping between the two parts of the spectra is not large and we may expect to be able, to distinguish the levels of the binary waveform d(t). In any event, with this separation the bandwidth of BFSK is

$$BW_{BFSK} = 4f_b \quad \ldots 3.7.7$$

which is twice the bandwidth of BPSK.

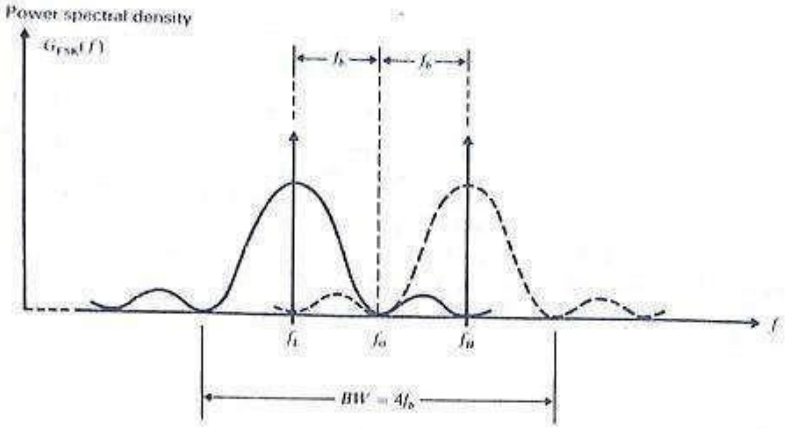


Figure 3.10.2 Power Spectral Densities of Equation 3.7.6

BFSK Receiver:

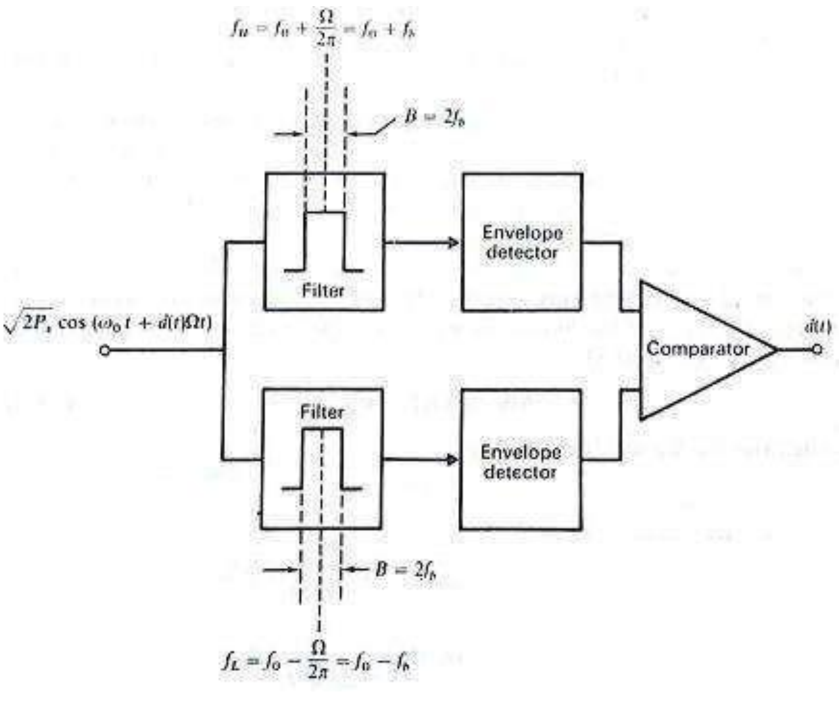


Figure 3.10.3 A Receiver for BFSK Signal

A BFSK signal is typically demodulated by a receiver system as in Fig.3.10.3. The signal is applied to two bandpass filters one with center frequency at f_H the other at f_L. Here we have assumed, that $f_H - f_L = 2(\Omega/2\pi) = 2f_b$. The filter frequency ranges selected do not overlap and each filter has a passband wide enough to encompass a main lobe in the spectrum of Fig. 3.10.2. Hence one filter will pass nearly all the energy in the transmission at f_H the other will perform similarly for the transmission at f_L. The filter outputs are applied to envelope detectors and finally the envelope detector outputs are compared by a comparator. A comparator is a circuit that accepts two input signals. It generates a binary output which is at one level or the other depending on which input is larger. Thus at the comparator output the data d(t) will be reproduced.

When noise is present, the output of the comparator may vary due to the systems response to the signal and noise. Thus, practical systems use a bit synchronizer and an integrator and sample the comparator output only once at the end of each time interval T_b.

Geometrical Representation of Orthogonal BFSK

In M-ary phase-shift keying and in quadrature-amplitude shift keying, any signal could be represented as $C_1u_1(t) + C_2u_2(t)$. There $u_1(t)$ and $u_2(t)$ are the orthonormal vectors in signal space, that is, $u1(t) = \sqrt{\frac{2}{T_s}}.\cos(\omega_o t)$ and $u2(t) = \sqrt{\frac{2}{T_s}}.\sin(\omega_o t)$. The functions u1 and u2 are orthonormal over the symbol interval T_S. And, if the symbol is a single bit, $T_S = T_b$. The coefficients C1 and C2 are constants. The normalized energies associated with $C_1u_1(t)$ and with $C_2u_2(t)$ are respectively C_1^2 and C_2^2 and the total signal energy is $C_1^2 + C_2^2$. In M-ary PSK and QASK the orthogonality of the vectors u_1 and u_2 results from their phase quadrature. In the present case of BFSK it is appropriate that the orthogonality should result from a special selection of the frequencies of the unit vectors. Accordingly, with m and n integers, let us establish unit vectors

$$u_1(t) = \sqrt{\frac{2}{T_b}}\cos 2\pi m f_b t \quad \text{...3.10.8}$$

and

$$u_2(t) = \sqrt{\frac{2}{T_b}}\cos 2\pi n f_b t \quad \text{...3.10.9}$$

Where $f_b=1/T_b$. The vectors U1 and U2 are the m^{th} and n^{th} harmonics of the (fundamental) frequency f_b. As we are aware, from the principles of Fourier analysis, different harmonics (m ± n) are orthogonal over the interval of the fundamental period $T_b = 1/f_b$.

If now the frequencies f_H and f_L in a BFSK system are selected to be (assuming m > n)

$$f_H = mf_b \quad \text{...3. 10.10a}$$

and

$$f_L = nf_b \quad \text{...3. 10.10b}$$

Then corresponding signal vectors are

$$S_H(t) = \sqrt{E_b}u_1(t) \quad \text{...3. 10.11a}$$

and

$$S_L(t) = \sqrt{E_b}u_2(t) \quad \text{...3. 10.11b}$$

The signal space representation of these signals is shown in Fig. 3. 10.4. The signals, like the unit vectors are orthogonal. The distance between signal end points is therefore

$$d = \sqrt{2E_b}$$

Note that this distance is considerably smaller than the distance separating end points of BPSK signals, which are antipodal.

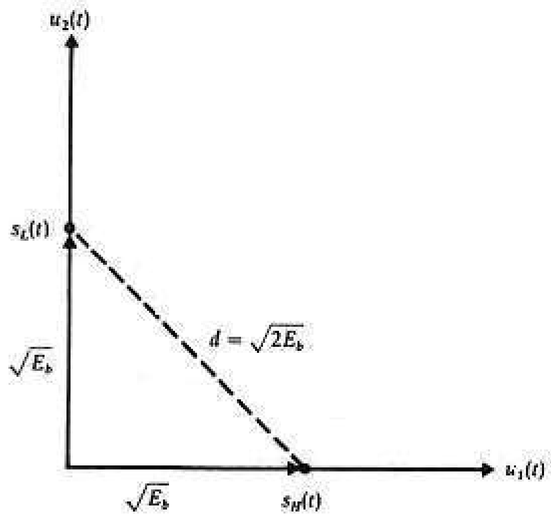


Figure 3.10.4 Signal Space representation of BFSK

Geometrical Representation of Non-Orthogonal BFSK

When the two FSK signals $S_H(t)$ and $S_L(t)$ are not orthogonal, the Gram-Schmidt procedure can still be used to represent the signals of Eqs. 3.10.2 and 3.10.3.

Let us represent the higher frequency signal SH(t) as:

$$S_H(t) = \sqrt{2P_S}\cos\omega_H t = S_{11}u_1(t) \quad 0 \le t \le T_b \qquad \text{...3. 10.12a}$$

and

$$S_L(t) = \sqrt{2P_S}\cos\omega_L t = S_{12}u_1(t) + S_{22}u_2(t) \quad 0 \le t \le T_b \qquad \text{...3. 10.12b}$$

The representation of these two signals in signal space is shown in Fig. 3. 10.5.

Referring to this figure we see that the distance separating S_H and S_L is:

$$d^2_{BFSK} = (S_{11} - S_{12})^2 + S_{22}^2 = S_{11}^2 - 2S_{11}S_{12} + S_{12}^2 + S_{22}^2 \qquad \text{...3. 10.13}$$

In order to determine d^2_{BFSK} when the two signals are not orthogonal we must evaluate S11, S12, and S22 using Eqs. (3. 10.12). From Eq. 3. 10.12a we have:

$$S_{11}^2 = 2P_S\int_0^{T_b}\cos^2\omega_H t\,dt = E_b\left[1 + \frac{\sin 2\omega_H T_b}{2\omega_H T_b}\right] \qquad \text{...3. 10.14}$$

Using Eq. (3. 10.12b) we first determine S_{12} by multiplying both sides of the equation by $u_1(t)$ and integrating from $0 \le t \le T_b$. The result is:

$$S_{12} = \sqrt{2P_S}u_1(t)\ \cos\omega_L t\ dt$$

$$S_{12} = \frac{E_b}{S_{11}}\left[\frac{\mathrm{Sin}(\omega_H - \omega_L)T_b}{(\omega_H - \omega_L)T_b} - \frac{\mathrm{Sin}(\omega_H + \omega_L)T_b}{(\omega_H + \omega_L)T_b}\right] \qquad \text{...3.7.15a}$$

By using equation 3.10.12a where,

$$u_1(t) = (\sqrt{2P_S}/S_{11})\ \cos\omega_H t \qquad \text{...3.7.15b}$$

Finally, S22 is found from Eq. 3.10.12b by squaring both sides of the equation and then integrating from 0 to T_b. Since u_1 and u_2 are orthogonal, the result is:

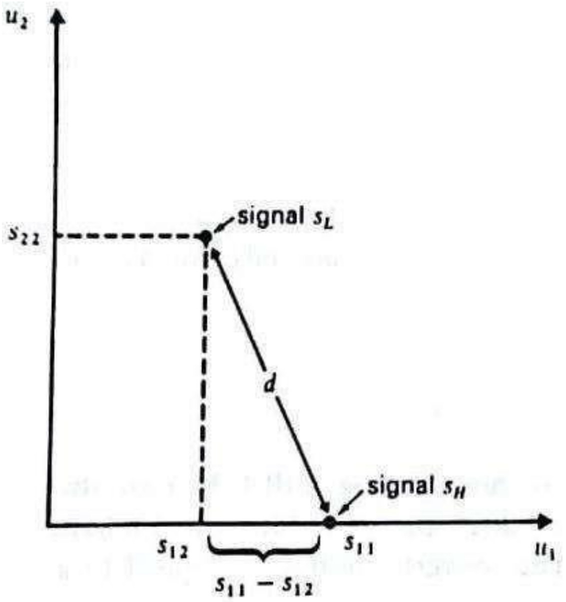


Figure 3.10.5 Signal Space representation of BFSK when $S_H(t)$ and $S_L(t)$ are not orthogonal

$$\int_0^{T_b} S_L^2(t)\,dt = 2P_S\int_0^{T_b} \cos^2\omega_H t\,dt = S_{12}^2 + S_{22}^2 \qquad \text{...3. 10.16a}$$

Therefore

$$S_{12}^2 + S_{22}^2 = E_b\left[1 + \frac{\sin 2\omega_L T_b}{2\omega_L T_b}\right] \qquad \text{...3. 10.16b}$$

The distance d between SH and SL given in Eq. (6.8-14) can now be determined by substituting Eqs. 3. 10.14, 3. 10.15a, and 3. 10.16b into Eq. 3.10.13 The result is:

$$d^2 = E_b\left[1 + \frac{\sin 2\omega_H T_b}{2\omega_H T_b}\right] - 2E_b\left[\frac{\mathrm{Sin}(\omega_H - \omega_L)T_b}{(\omega_H - \omega_L)T_b} + \frac{\mathrm{Sin}(\omega_H + \omega_L)T_b}{(\omega_H + \omega_L)T_b}\right] + E_b\left[1 + \frac{\sin 2\omega_L T_b}{2\omega_L T_b}\right] \qquad \text{...3. 10.18}$$

In above equation,

$$\left|\frac{\sin 2\omega_H T_b}{2\omega_H T_b}\right| \ll 1$$

$$\left|\frac{\sin 2\omega_L T_b}{2\omega_L T_b}\right| \ll 1$$

And

$$\left|\frac{\mathrm{Sin}(\omega_H + \omega_L)T_b}{(\omega_H + \omega_L)T_b}\right| \ll \left|\frac{\mathrm{Sin}(\omega_H - \omega_L)T_b}{(\omega_H - \omega_L)T_b}\right|$$

Then the final result is then

$$d^2 \cong 2E_b\left[1-\frac{\mathrm{Sin}(\omega_H-\omega_L)T_b}{(\omega_H-\omega_L)T_b}\right] \qquad \text{...3.10.19}$$

Here $S_H(t)$ and $S_L(t)$ are orthogonal $(\omega_H\text{-}\omega_L)T_b=2\pi(m-n)f_bT_b=2\pi(m\text{-}n)$ and the above equation given $d = \sqrt{2E_b}$.

Note that if $(\omega_H\text{-}\omega_L)T_b=3\pi/2$, the distance d increases and becomes,

$$d_{opt} = \left[2E_b\left(1+\frac{2}{3\pi}\right)\right]^{1/2} = \sqrt{2.4E_b} \qquad \text{...3. 10.20}$$

d^2 is increased by 20%.

3.11 Comparison of BFSK and BPSK

Let us start with the BFSK signal

$$v_{BFSK}(t) = \sqrt{2P_s}\cos[\omega_o t + d(t)\Omega t]$$

Using the trigonometric identity for the cosine of the sum of two angles and recalling that cos θ = cos (- θ) while sin θ = - sin (-θ) we are led to the alternate equivalent expression

$$v_{BFSK}(t) = \sqrt{2P_s}\cos\Omega t\cos\omega_o t - \sqrt{2P_s}d(t)\sin\Omega t\sin\omega_o t \qquad \text{...3.11.1}$$

Note that the second term in above equation looks like the signal encountered in BPSK i.e., a carrier $\sin\omega_0 t$ multiplied by a data bit d(t) which changes the carrier phase. In the present case however, the carrier is not of fixed amplitude but rather the amplitude is shaped by the factor sin Ωt. We note further the presence of a quadrature reference term $\cos\Omega t\cos\omega_0 t$ which contains no information. Since this quadrature term carries energy, the energy in the information bearing term is thereby diminished. Hence we may expect that BFSK will not be as effective as BPSK in the presence of noise. For orthogonal BFSK, each term has the same energy, hence the information bearing term contains only one-half of the total transmitted energy.

The Generation of BFSK is easier but it has many disadvantages…

- Bandwidth is greater in comparison with BPSK, almost double. (Because we are using two career signals.)
- Error rate of BFSK is higher

3.12 M-ARY FSK

An M-ary FSK communications system is shown in Fig. 3.12.1. It is an obvious extension of a binary FSK system. At the transmitter an N-bit symbol is presented each T_S, to an N-bit D/A converter. The converter output is applied to a frequency modulator, i.e., a piece of hardware which generates a carrier waveform whose frequency is determined by the modulating waveform. The transmitted signal, for the duration of the symbol interval, is of frequency f_0 or f_1 ...or f_{M-1} with M = 2^N. At the receiver, the incoming signal is applied to M paralleled bandpass filters each followed by an envelope detector. The bandpass filters have center frequencies f_0, f_1, ... ,f_{M-1}. The envelope detectors apply their outputs to a device which determines which of the detector indications is the largest and transmits that envelope output to an N-bit AID converter.

The probability of error is minimized by selecting frequencies f_0, f_1, ... ,f_{M-1} so that the M signals are mutually orthogonal. One commonly employed arrangement simply provides that the carrier frequency be successive even harmonics of the symbol frequency $f_S=1/T_S$. Thus the lowest frequency, say f_0, is f_0 = k f_S, while f_1 = (k + 1) f_S, f_2 = (k + 2) f_S etc. In this case, the spectral density patterns of the individual possible transmitted signals overlap in the manner shown in Fig. 3.12.1, which is an extension to M-ary FSK of the pattern of Fig. 3.9.2, which applies to binary FSK. We observe that to pass M-ary FSK the required spectral range is

$$B = 2Mf_S \quad \text{...3.12.1}$$

Since $f_S = f_b/N$ and $M=2^N$, we have

$$B = 2^{N+1}f_b/N \quad \text{...3.12.2}$$

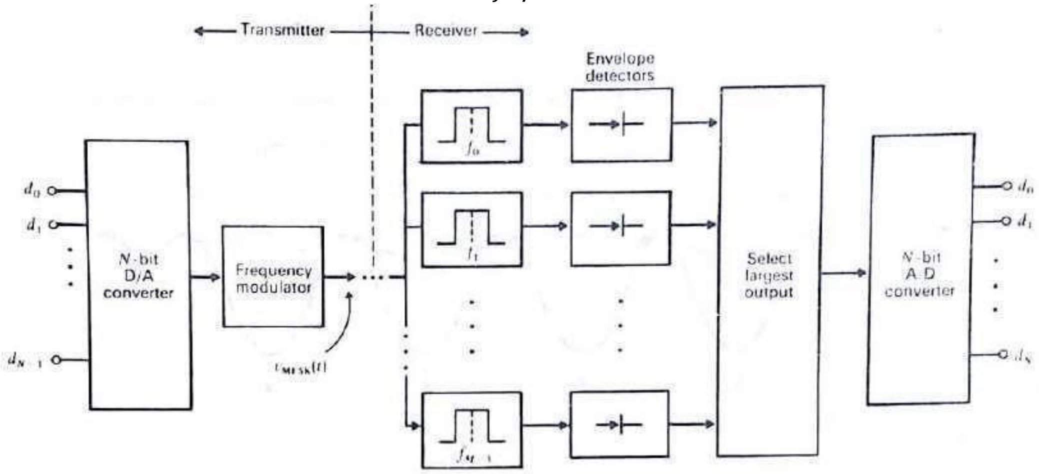


Figure 3.12.1 An M-ARY Communication System

Note that M-ary FSK requires a considerably increased bandwidth in comparison with M-ary PSK. However, as we shall see, the probability of error for M-ary FSK decreases as M increases, while for M-ary PSK, the probability of error increases with M.

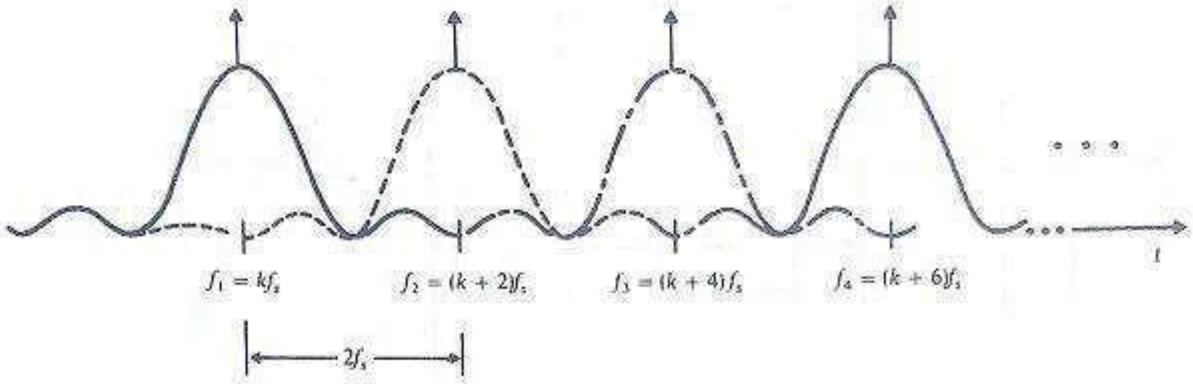


Figure 3.12.2 Power Spectral Density of an M-ARY FSK (Four Frequencies are shown)

Geometrical Representation of an M-ARY FSK

In Fig 3.7.4, we provided a signal space representation for the case of orthogonal binary FSK.

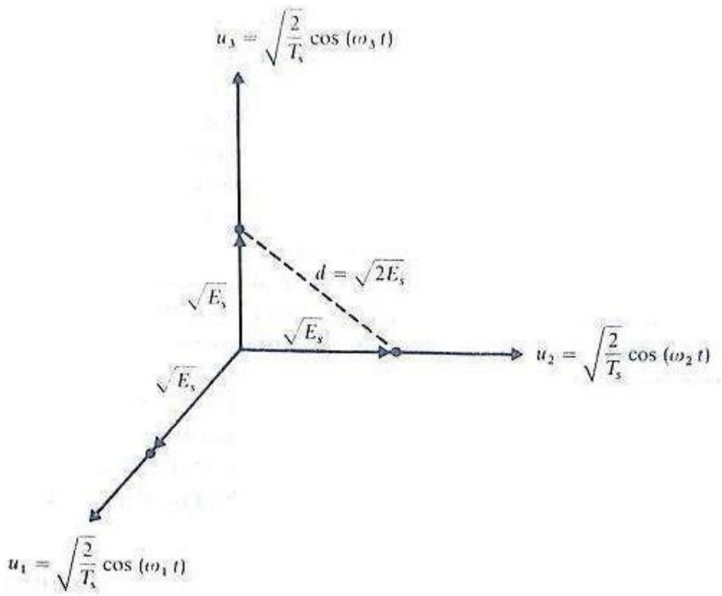


Figure 3.12.3 Geometrical representation of orthogonal M-ary FSK (M = 3) when the frequencies are selected to generate orthogonal signals.

The case of M-ary orthogonal FSK signals is clearly an extension of this figure. We simply conceive of a coordinate system with M mutually orthogonal coordinate axes. The square of the length of the signal vector is the normalized signal energy. Note that, as in Fig. 3.7.4, the distance between signal points is 3.9.3.

Note that this value of d is greater than the values of d calculated for M-ary PSK with the exception of the cases M = 2 and M = 4. It is also greater than d in the case of 16-QASK.

$$d = \sqrt{2E_s} = \sqrt{2NE_b} \qquad ...3.12.3$$

3.13 Minimum Shift Keying (MSK)

There are two important differences between QPSK and MSK:

1. In MSK the baseband waveform, that multiplies the quadrature carrier, is much "smoother" than the abrupt rectangular waveform of QPSK. While the spectrum of MSK has a main center lobe which is 1.5 times as wide as the main lobe of QPSK; the side lobes -in MSK are relatively much smaller in comparison to the main lobe, making filtering much easier.
2. The waveform of MSK exhibits phase continuity, that is, there are not abrupt. phase changes as in QPSK. As a result we avoid the inter-symbol interference caused by nonlinear amplifiers.

The waveforms of MSK are shown in figure 3.13.1. In (a) we start with a typical data bit stream b(t). This bit stream is divided into an odd an even bit stream in (b) and (c), as in the manner of OQPSK. The odd stream $b_o(t)$ consists of the alternate bits b_1, b_3, etc., and the even stream $b_e(t)$ consists of b_2, b_4; etc. Each bit in both streams is held for two bit intervals 2 $T_b = T_s$, the symbol time. The staggering, which is optional in QPSK, is essential in MSK. The staggering is that the changes in the odd and even stream do not occur at the same time.

Also generated at the MSK transmitter are the waveforms sin $2\pi(t/4T_b)$ and cos $2\pi(t/4T_b)$ as in (d). These waveforms, and their phases with respect to the bit streams $b_o(t)$ and $b_e(t)$, meet the essential requirements that sin $2\pi(t/4T_b)$ passes through zero precisely at the end of the symbol time in $b_e(t)$ and cos $2\pi(t/4T_b)$ passes through zero at the end of the symbol time in $b_o(t)$. We now .generate the products $b_e(t)$ sin $2\pi(t/4T_b)$ and $b_o(t)$ cos $2\pi(t/4T_b)$ which are shown in (e) and (f).

In MSK the transmitted signal is

$$v_{MSK}(t) = \sqrt{2P_s}\,[b_e(t)\sin 2\pi\,(\frac{t}{4T_b})]\cos\omega_o t + \sqrt{2P_s}\,[b_0(t)\cos 2\pi\,(\frac{t}{4T_b})]\sin\omega_o t \qquad ...3.13.1$$

In MSK, the carriers are multiplied by the" smoother" waveforms shown in Fig. 3.13.1(e) and (f). As we may expect, the side lobes generated by these smoother waveforms will be smaller than those associated with the rectangular waveforms and hence easier to suppress as is required to avoid interchannel interference.

In Eq. (3.13.1) MSK appears as a modified form of OQPSK, which we can call" shaped Q'PSK". We can, however, rewrite the equation to make it apparent that MSK is an FSK system. Applying the trigonometric identities for the products of sinusoids we find that Eq. (3.10.1) can be written:

$$v_{MSK}(t) = \sqrt{2P_s}\,[\frac{b_o(t) + b_e(t)}{2}]\sin(\omega_o + \Omega)t + \sqrt{2P_s}[\frac{b_o(t) - b_e(t)}{2}]\cos(\omega_o + \Omega)t \qquad ...3.13.2a$$

$$\text{Where } \Omega = 2\pi/4T_b = 2\pi(fb/4) \qquad ...3.13.2b$$

If we define $C_H = (b_0+b_e)/2$, $C_L==(b_0-b_e)/2$, $\omega_H = \omega_0 + \Omega$, $\omega_H = \omega_0 - \Omega$ then equations 3.10.2 becomes

$$v_{MSK}(t) = \sqrt{2P_s}C_H(t)\sin\omega_H t + \sqrt{2P_s}C_L(t)\omega_L t \qquad ...3.13.3$$

Now b; = + 1 and be = + 1, so that as is easily verified, if b_0, = b_e then CL = 0 while $C_H = b_0 = \pm 1$. Further, if $b_0 = -b_e$ then $C_H = 0$ and $C_L = b_0 = + 1$.

Thus, depending on the value of the bits b_o and b_e in each bit interval, the transmitted signal is at angular frequency ω_H or at ω_L precisely as in FSK and the magnitude of the amplitude is always equal to $\sqrt{2P_s}$.

In MSK, the two frequencies f_H and f_L, are chosen to insure that the two possible signals are orthogonal over the bit interval T_b. That is, we impose the constraint that

$$\int_0^{T_b} \sin \omega_H t \sin \omega_L t \, dt = 0 \qquad \text{...3.13.4}$$

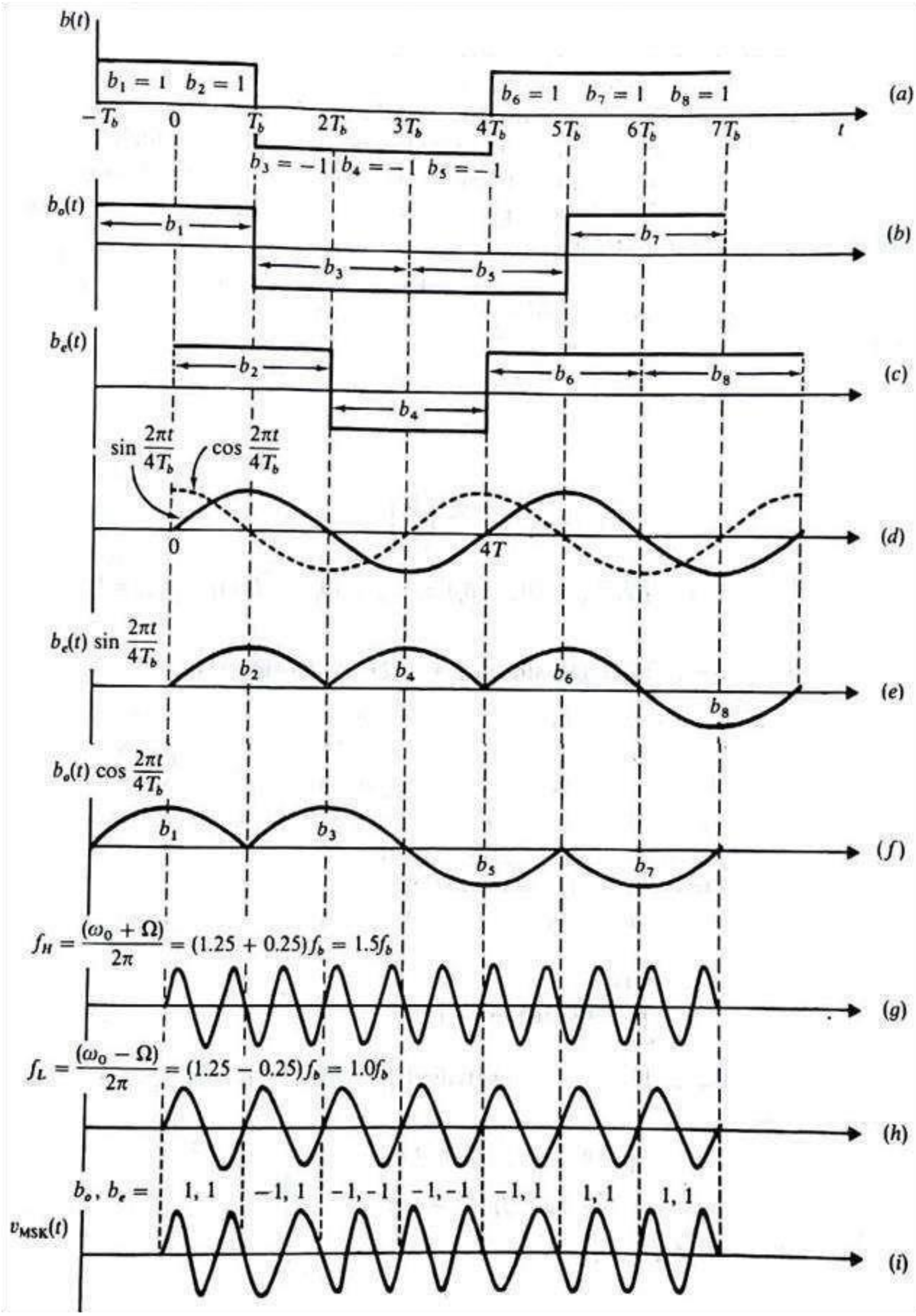


Figure 3.13.1 MSK Waveforms

The equation 3.13.4 will be satisfied provided that it is arranged, with m and n integers, that

$$2\pi(f_H - f_L)T_b = n\pi \qquad \text{...3.13.5a}$$

and

$$2\pi(f_H + f_L)T_b = m\pi \qquad \text{...3.13.5b}$$

Also

$$f_H = f_0 + \frac{f_b}{4} \qquad \text{...3.13.06a}$$

and

$$f_L = f_0 - \frac{f_b}{4} \qquad \text{...3.13.6b}$$

From equations 3.13.5 and 3.13.6

$$f_b T_b = f_b \frac{1}{f_b} = 1 = n \qquad \text{...3.13.7a}$$

and

$$f_0 = \frac{m}{4} f_b \qquad \text{...3.13.7b}$$

Equation (3.13.7a) shows that since n = 1, f_H and f_L are as close together as possible for orthogonality to prevail. It is for this reason that the present system is called" minimum shift keying." Equation (3.13.7b) shows that the carrier frequency f_o is an integral multiple of $f_b/4$. Thus

$$f_H = (m+1)\frac{f_b}{4} \qquad \text{...3.13.8a}$$

and

$$f_L = (m-1)\frac{f_b}{4} \qquad \text{...3.13.8b}$$

Signal Space Representation of MSK

The signal space representation of MSK is shown in Fig. 3.13.2 The orthonormal unit vectors of the coordinate system are given by $u_H(t) = \sqrt{2/T_s}\,\sin\,\omega_H t$ and $u_L(t) = \sqrt{2/T_s}\,\sin\,\omega_L t$. The end points of the four possible signal vectors are indicated by dots. The smallest distance between signal points is

$$d = \sqrt{2E_s} = \sqrt{4E_b} \qquad \text{...3.13.9}$$

just as for the case of QPSK.

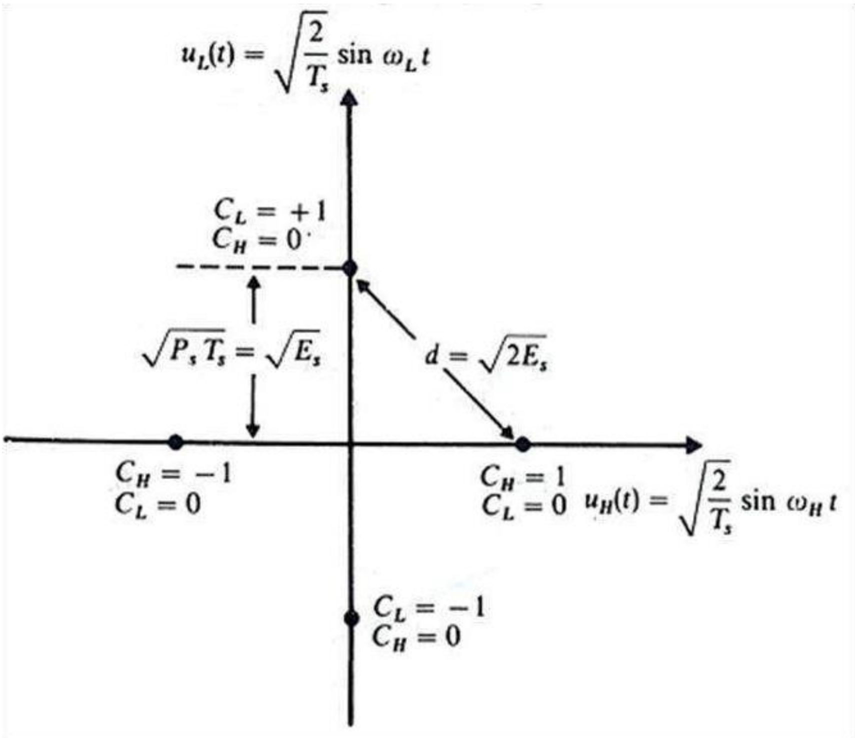


Figure 3.13.2 Signal Space Representation of MSK

We recall that QPSK generates two BPSK signals which are orthogonal to one another by virtue of the fact that the respective carriers are in phase quadrature. Such phase quadrature can also be characterized as time quadrature since, at a carrier frequency f_0 a phase shift of $\pi/2$ is accomplished by a time shift in amount $1/4f_0$, that is $\sin 2\pi f_0(t+1/4f_0) = \sin(2\pi f_0 t + \pi/2) = \cos(2\pi f_0 t)$. It may be noted that in MSK we have again two BPSK signals. Here, however, the respective carriers are orthogonal to one another by virtue of the fact that they are in frequency quadrature.

Generation and Reception of MSK

One way to generate a MSK signal is the following: We start with sin Ωt and sin ω_0t, and use 90° phase shifters to generate sin (Ωt + π/2) = cosΩt and sin (ω_0t + π/2) = cosω_0t. We then use multipliers to form the products sinΩt cosω_0t and cosΩt sin ω_0t. Additional multipliers generate $\sqrt{2P_s}b_e(t)\sin\Omega t\cos\omega_0 t$ and $\sqrt{2P_s}b_0(t)\cos\ \Omega t\ \sin\ \omega_0 t$. Finally an adder is used to form the sum. An alternative and more favored scheme is shown in Fig. 3.13.3a. This technique has the merit that it avoids the need for precise 90° phase shifters at angular frequencies ω_o and Ω.

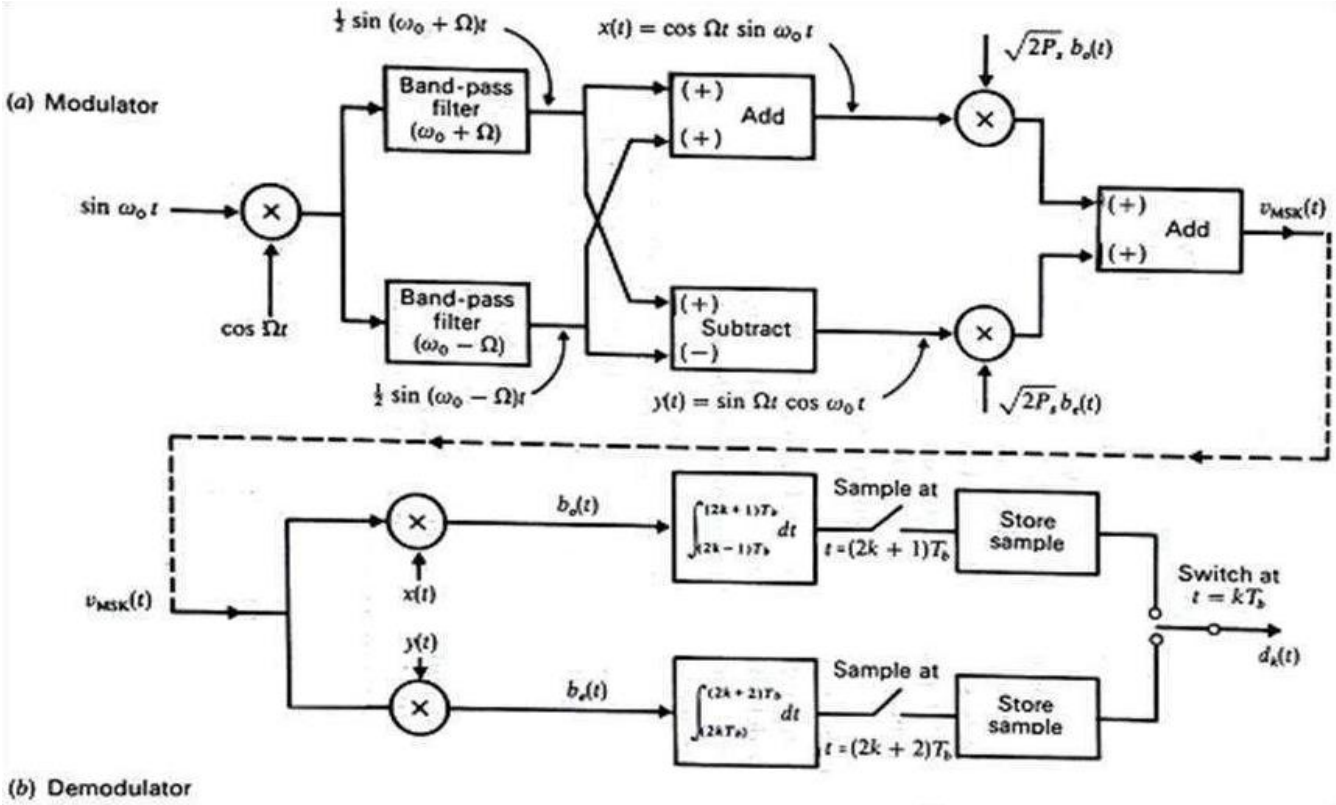


Figure 3.13.3 MSK Modulation and demodulation

The MSK receiver is shown in Fig. 3.13.3b. Detection is performed synchronously, i.e., by determining the correlation of the received signal with the waveform x(t) = cosΩt sin ω_0t to determine bit b_0(t), and with y(t) =sinΩt cosω_0t to determine bit b_e(t). The integration is performed over the symbol interval. The integrators integrate over staggered overlapping intervals of symbol time $T_s = 2T_b$.

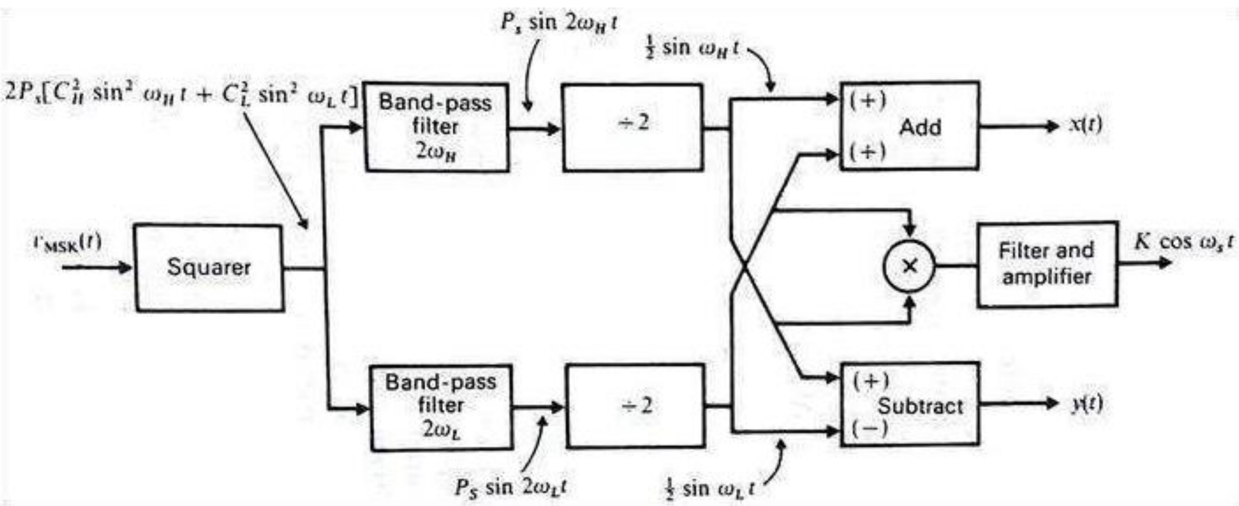


Figure 3.13.4 Regeneration of x(t) and y(t)

At the end of each integration time the integrator output is stored and then the integrator output is dumped. The switch at the output swings back and forth at the bit rate so that finally the output waveform is the original data bit stream $d_k(t)$.

At the receiver we need to reconstruct the waveforms x(t) and y(t). A method for locally regenerating x(t) and y(t) is shown in Fig. 3.10.4. From Eq3.13.3 we see that MSK consists of transmitting one of two possible HPSK signals, the first at frequency $\omega_0 - \Omega$ and the second at frequency $\omega_0 + \Omega$. Thus as in BPSK detection, we first square and filter the incoming signal. The output of the squarer has spectral components at the frequency $2\omega_H = 2(\omega_0 + \Omega)$ and at $2\omega_L = 2(\omega_0 - \Omega)$. These are separated out by band pass filters. Division by 2 yields waveforms $(1/2)\sin \omega_H t$ and $(1/2)\sin \omega_L t$ from which, as indicated, x(t) and y(t) are regenerated by addition and subtraction respectively. Further, the multiplier and low-pass filter shown regenerate a waveform at the symbol rate Is $=f_b/2$ which can 'be used to operate the sampling switches in Fig. 3.13.3b.

Unit 4

Syllabus:

Other Digital Techniques: Pulse shaping to reduce inter channel and inter symbol interference- Duo binary encoding, Nyquist criterion and partial response signaling, Quadrature Partial Response (QPR) encoder decoder, Regenerative Repeater- eye pattern, equalizers

Optimum Reception of Digital Signals: Baseband signal receiver, probability of error, maximum likelihood detector, Bayes theorem, optimum receiver for both baseband and pass band receiver- matched filter and correlator, probability of error calculation for BPSK and BFSK.

4.1 Inter Symbol Interference

This is a form of distortion of a signal, in which one or more symbols interfere with subsequent signals, causing noise or delivering a poor output.

Causes of ISI

The main causes of ISI are −

- Multi-path Propagation
- Non-linear frequency in channels

The ISI is unwanted and should be completely eliminated to get a clean output. The causes of ISI should also be resolved in order to minimize its effect.

To view ISI in a mathematical form present in the receiver output, we can consider the receiver output.

The receiving filter output $y(t)$

is sampled at time $t_i = iT_b$

(with **i** taking on integer values), yielding –

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

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

In the above equation, the first term μa_i is produced by the i^{th} transmitted bit.

The second term represents the residual effect of all other transmitted bits on the decoding of the i^{th} bit. This residual effect is called as Inter Symbol Interference.

In the absence of ISI, the output will be −

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

This equation shows that the i^{th} bit transmitted is correctly reproduced. However, the presence of ISI introduces bit errors and distortions in the output.

While designing the transmitter or a receiver, it is important that you minimize the effects of ISI, so as to receive the output with the least possible error rate.

Correlative Coding

So far, we've discussed that ISI is an unwanted phenomenon and degrades the signal. But the same ISI if used in a controlled manner is possible to achieve a bit rate of 2W bits per second in a channel of bandwidth W Hertz. Such a scheme is called as Correlative Coding or Partial response signaling schemes.

Correlative Level Coding:

Correlative-level coding (partial response signaling) – adding ISI to the transmitted signal in a controlled manner Since ISI introduced into the transmitted signal is known, its effect can be interpreted at the receiver. A practical method of achieving the theoretical maximum signaling rate of 2W symbol per second in a bandwidth of W Hertz.

Using realizable and perturbation-tolerant filters

Since the amount of ISI is known, it is easy to design the receiver according to the requirement so as to avoid the effect of ISI on the signal. The basic idea of correlative coding is achieved by considering an example of Duo-binary Signaling.

4.2 Duo-binary Signaling

If f_M is the frequency of the maximum frequency spectral component of the baseband waveform, then, in AM, the bandwidth is B = $2f_M$. In frequency modulation, if the modulating waveform were a sinusoid of frequency f_M, and if the frequency deviation was Δf, then bandwidth would be

$$B = 2\Delta f + 2f_M \qquad \text{...4.2.1}$$

Altogether, it is apparent that bandwidth decreases with decreasing f_M regardless of the modulation technique employed. We consider now a mode of encoding a binary bit stream, called duobinary encoding which effects a reduction of the maximum frequency in comparison to the maximum frequency of the un-encoded data. Thus, if a carrier is amplitude or frequency modulated by a duobinary encoded waveform, the bandwidth of the modulated waveform will be smaller than if the un-encoded data were used to AM or FM modulate the carrier.

There are a number of methods available for duobinary encoding and decoding. One popular scheme is shown in Fig. 4.2.1. The signal d(k) is the data bit stream with bit duration T_b. It makes excursions between logic 1 and logic 0, and, as had been our custom, we take the corresponding voltage levels to be + 1V and - 1V. The signal b(k), at the output of the differential encoder also makes excursions between + 1V and -1V. The waveform $v_d(k)$ is therefore

$$v_d(k) = b(k) + b(k-1) \qquad \text{...4.2.2}$$

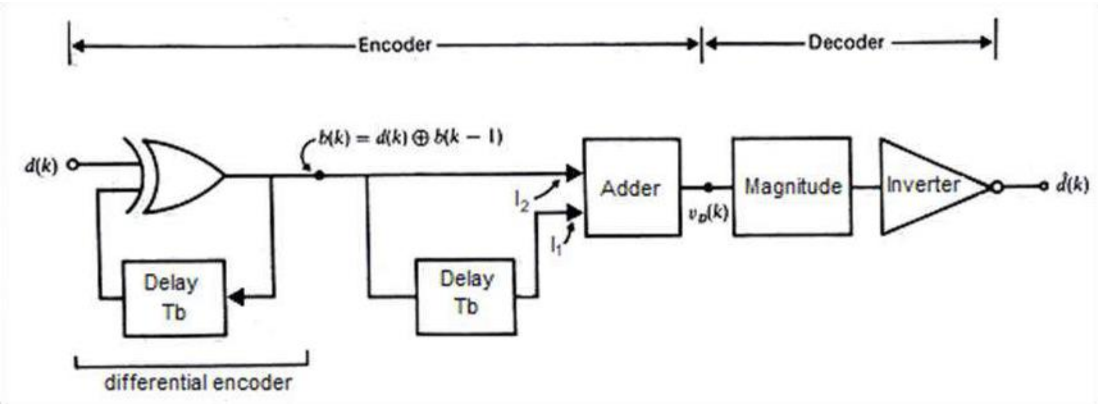


Figure 4.2.1 The Duobinary Encoder Decoder System

which can take on the values $v_D(k)$ = +2V, 0V and - 2V. The value of $v_D(k)$ in any interval k depends on both b(k) and b(k - 1). Hence there is a correlation between the values of $v_D(k)$ in any two successive intervals. For this reason the coding of Fig. 4.2.1 is referred to as correlative coding.

The correlation can be made apparent in another way. When the transition is made from one interval to the next, it is not possible for $v_D(k)$ to change from +2V to - 2V or vice versa. In short, in any interval, $v_D(k)$ cannot always assume any of the possible levels independently of its level in the previous intervals. Finally, we note that the term duobinary is appropriate since in each bit interval, the generated voltage $v_D(k)$ results from the combination of two bits.

The decoder, shown in Fig. 4.2.1, consists of a device that provides at its output the magnitude (absolute value) of its input cascaded with a logical inverter. For the inverter we take it that logic 1 is + 1V or greater and logic 0 is 0V. We can now verify that the decoded data $\hat{d}(k)$ is indeed the input data d(k). For this purpose we prepare the following truth table:

Truth Table For Duobinary Signaling

Adder Input 1 ***I_1***		**Adder Input 2** ***I_2***		**Adder output** ***$v_D(k)$***	**Magnitude Output (Inverter output)**		**Inverter output** ***d(k)***
Voltage	Logic	Voltage	Logic	Input Voltage	Voltage	Logic	Logic
-1V	0	-1V	0	-2V	2V	1	0
-1V	0	1V	1	0	0V	0	1
1V	1	-1V	0	0	0V	0	1
1V	1	1V	1	2V	2V	1	0

From the table we see that the inverter output is $I_1 \oplus I_2$.The differential encoder (called a precoder in the present application) output is:

$$I_1 = b(k) = d(k) \oplus b(k-1) \qquad ...4.2.3$$

The input I_2 = b(k - 1) so that the inverter output $\hat{d}(k)$ is:

$$\hat{d}(k) = I_1 \oplus I_2 = d(k) \oplus b(k-1) \oplus b(k-1) = d(k) \qquad ...4.2.4$$

Waveforms of Duobinary Signaling

The more rapidly d(k) switches back and forth between logic levels the higher will be the frequencies of the spectral components generated. When d(k) switches at each time T_b, the switching speed is at a maximum. The waveform d(k), under such circumstances, has the appearance of a square wave of period $2T_b$ and frequency 1/2 T_b as shown in Fig. 4.2.2a.

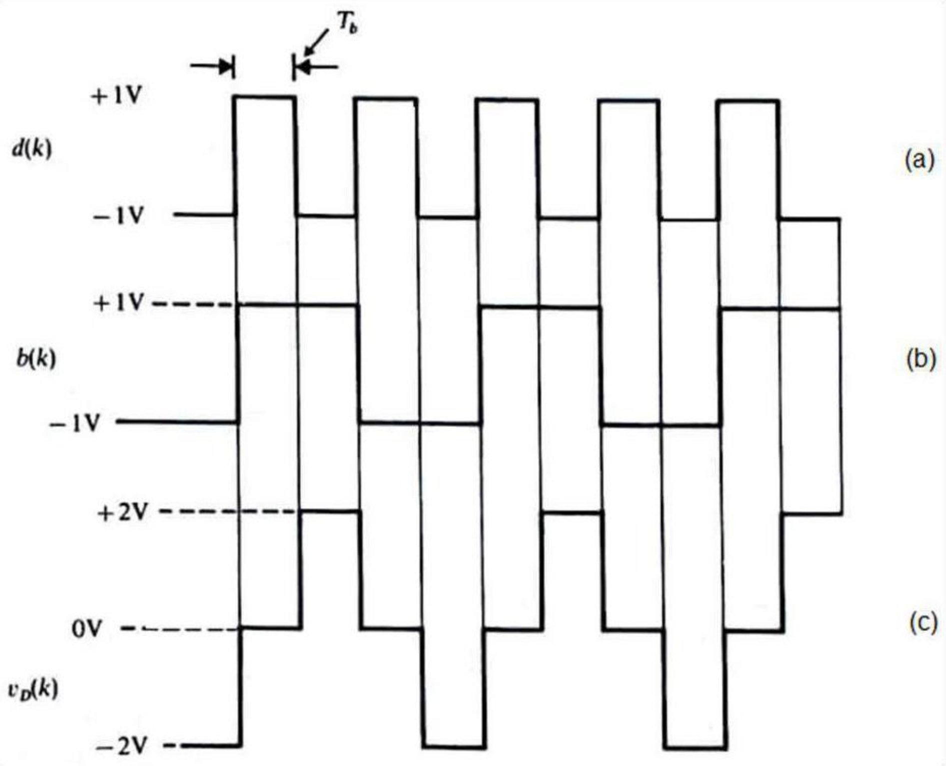


Figure 4.2.2 Waveforms of *d(k), b(k)* and $v_D(k)$

If d(k) is the input to the duobinary encoder of Fig. 4.2.1 then, as can be verified, b(k) appears as in Fig. 4.2.2b and the waveform, $v_D(k)$ which is to be transmitted appears as in Fig. 4.2.2c. Observe that the period of $v_D(k)$ is 4 T_b with corresponding frequency 1/4 T_b. Thus the frequency of $v_D(k)$ is half the frequency of the original unencoded waveform d(k). The waveform d(k) may be a sinusoid of frequency $1/2\ T_b$ and waveform $v_D(k)$ as a sinusoid of frequency $1/4T_b$. If we were free to select either d(k) or $v_D(k)$ as a modulating waveform for a carrier, and if we were interested in conserving bandwidth, we would choose $v_D(k)$. If amplitude modulation were involved, the bandwidth of the modulated waveform would be $2(1/4T_b) = f_b/2$ using $v_D(k)$ since the modulating frequency is $f_M = 1/4T_b$ and would be $2(1/2T_b) = f_b$ using d(k). With frequency modulation, if the peak-to peak carrier frequency deviation were $2\Delta f$, then, the modulated carrier would have a bandwidth $2(\Delta f) + 2(1/2T_b)$ with d(k) as the modulating signal, as in BFSK; and $2(\Delta f) + 2(1/4T_b)$with $v_D(k)$ as the modulating signal.

4.3 Partial Response Signaling

Suppose, that corresponding to each bit of duration Tb, of a data stream we generate a positive impulse of strength +1 whenever the bit is at logic 1 and a negative impulse -1 whenever the bit is at logic 0. Suppose, further, that these impulses are applied to the input of the cosine filter. In Fig. 6.4.3 we have drawn the filter responses individually to five successive positive impulses. For simplicity, we have in each case drawn only the central lobe-and we have indicated by dots all the places where the individual response waveforms pass through zero. Where there is no dot, the waveforms has a finite value. The peaks of the responses are separated by times Tb and the widths of the central lobes are 3Tb.

The total response is, of course, simply the sum of the individual responses.

We can make the following observations from Fig. 4.3.1:

1. If we sample the total response at a time when an individual response is at its peak, the sample will have contributions from all the individual responses.
2. There is no possible time at which a sample of the total response is due only to a single individual response.
3. Importantly, if we sample the total response midway between times when the individual responses are at peak value, i.e., at $t = (2k-1)T_b/2$, then the sample value will have contributions in equal amount from only the two individual responses that straddle the sampling time. These sampling times are indicated in Fig. 4.3.1, by the light vertical lines. One such sampling time, yielding contributions from individual responses 2 and 3 is explicitly marked. It can be calculated that at the sampling time the contribution from each of the straddling individual responses will be a voltage If_b. Note that in sampling at the indicated times, we sample when the individual responses are not at peak value. For this reason, the present signal processing is referred to as Partial Response Signaling.

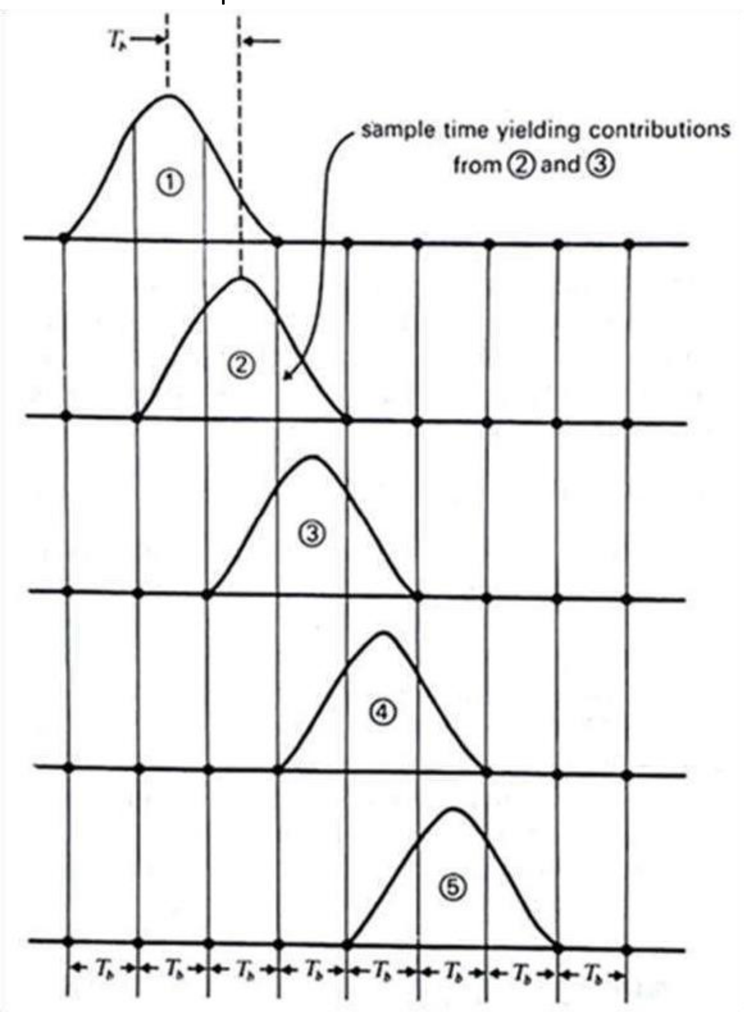


Figure 4.3.1 Filter responses to Five Different Impulses

In partial-response signaling, we shall transmit a signal during each bit interval that has contributions from two successive bits of an original baseband waveform. But this superposition need not prevent us from disentangling the individual original baseband waveform bits. A complete (baseband) partial-response signaling communications system is shown in Fig. 4.3.2.

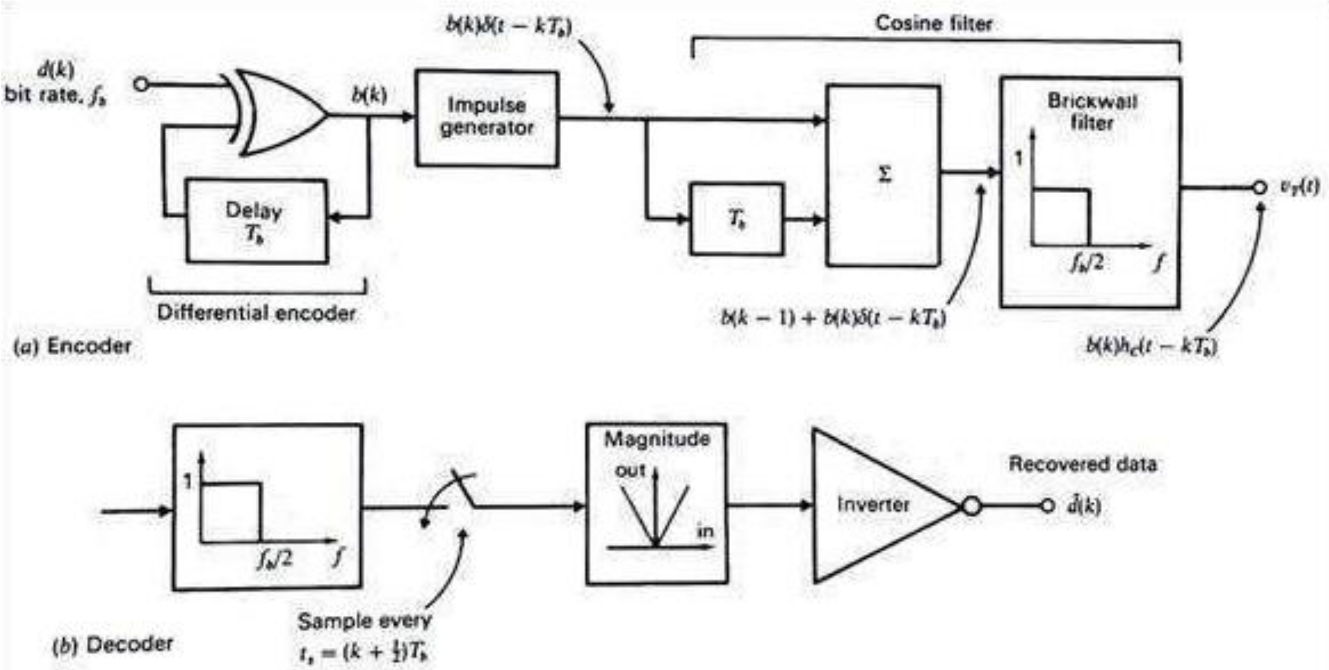


Figure 4.3.2 Duo Binary Encoder and Decoder Using Cosine filter

It is seen to be just an adaptation of duobinary encoding and decoding. The cosine filter employed a delay and an advance of the impulse by amount Tb/2, the total time between delayed and advanced impulses being Tb. Since, in the real world, a time advance is not possible, we have employed only a delay by amount Tb. The brickwall filter at the receiver input serves to remove any out of band noise added to the signal during transmission. It can be shown, that the output data $\hat{d}(\mathrm{k})$ = d(k).

4.4 Quadrature Partial Response (QPR) Encoder and Decoder

Amplitude Modulation of Partial Response Signal

The baseband partial response (duobinary) signal may be used to amplitude or frequency modulate a carrier. If amplitude modulation is employed, either double sideband suppressed carrier DSB/SC or quadrature amplitude modulation QAM can be employed.

For the case of DSB/SC the duobinary signal, $V_T(t)$, shown in Fig. 4.3.2a, is multiplied by the carrier $\sqrt{2}\cos\omega_0\,(t)$. The resulting signal is

$$v_{DSB}(t) = \sqrt{2}v_T(t)\cos\omega_0\,(t) \qquad \text{...4.4.1}$$

The bandwidth required to transmit the signal is twice the bandwidth of the baseband duobinary signal which is $f_b/2$. Hence the bandwidth B_{DSB} of an amplitude modulated duobinary signal is

$$B_{DSB} = 2(f_b/2) = f_b \qquad \text{...4.4.2}$$

If the duobinary signal is to amplitude modulate two carriers in quadrature, the circuit shown in Fig. 4.4.1 is used and the resulting encoder is called a "quadrature partial response" (QPR) encoder.

Figure 4.4.1 shows that the data d(t) at the bit rate fb is first separated into an even and an odd bit stream de(t) and do(t) each operating with the bit rate fb /2, Both de(t) and do(t) are then separately duobinary encoded into signals $V_{Te}(t)$ and $V_{To}(t)$.

Each duobinary encoder is similar to the encoder shown in Fig. 4.3.2a except that each delay is now 2Tb, rather than Tb, the data rate of the input is $f_b/2$ rather than f_b and the bandwidth of the brick wall filter is now $(1/2)(f_b/2)= f_b/4$ rather than $f_b/2$. Thus the bandwidth required to pass $V_{Te}(t)$ and $V_{To}(t)$ is $f_b/4$. Each duobinary signal is then modulated using the quadrature carrier signals cos ω_ot and sin ω_ot.

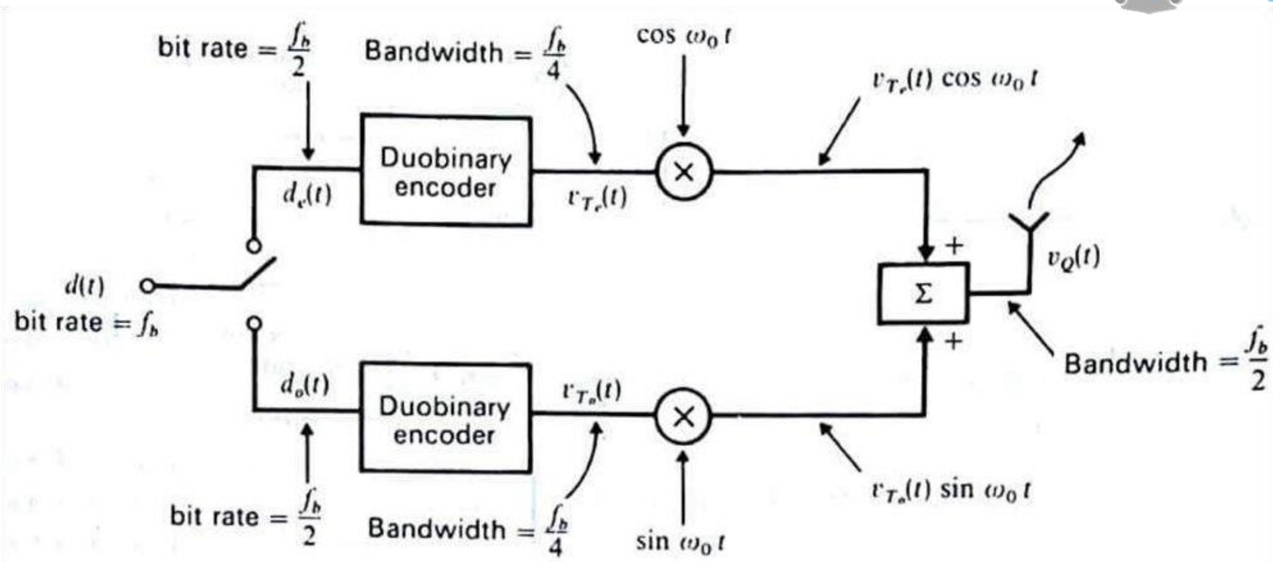


Figure 4.4.1 QPR Encoder

The bandwidth of each of the quadrature amplitude modulated signals is

$$B_{QPR} = 2(f_b/4) = f_b/2 \qquad \text{...4.4.3}$$

Hence the total bandwidth required to pass a QPR signal is also B_{QPR}, since the two quadrature components occupy the same frequency band.

It should be noted that if QPSK, rather than QPR, were used to encode the data d(t), the bandwidth required would be $B_{QPSK} = f_b$. However, if 16 QAM or 16 PSK were used to encode the data the required bandwidth would be $B_{16QAM} = B_{16PSK} = fb/2$. Thus the spectrum required to pass a QPR signal is similar to that required to pass 16 QAM or 16 PSK. However, the QPR signal displays no (or in practice very small) side lobes which makes QPR the encoding system of choice when spectrum width is the major problem. The drawback in using QPR is that the transmitted signal envelope is not a constant but varies with time.

QPR Decoder

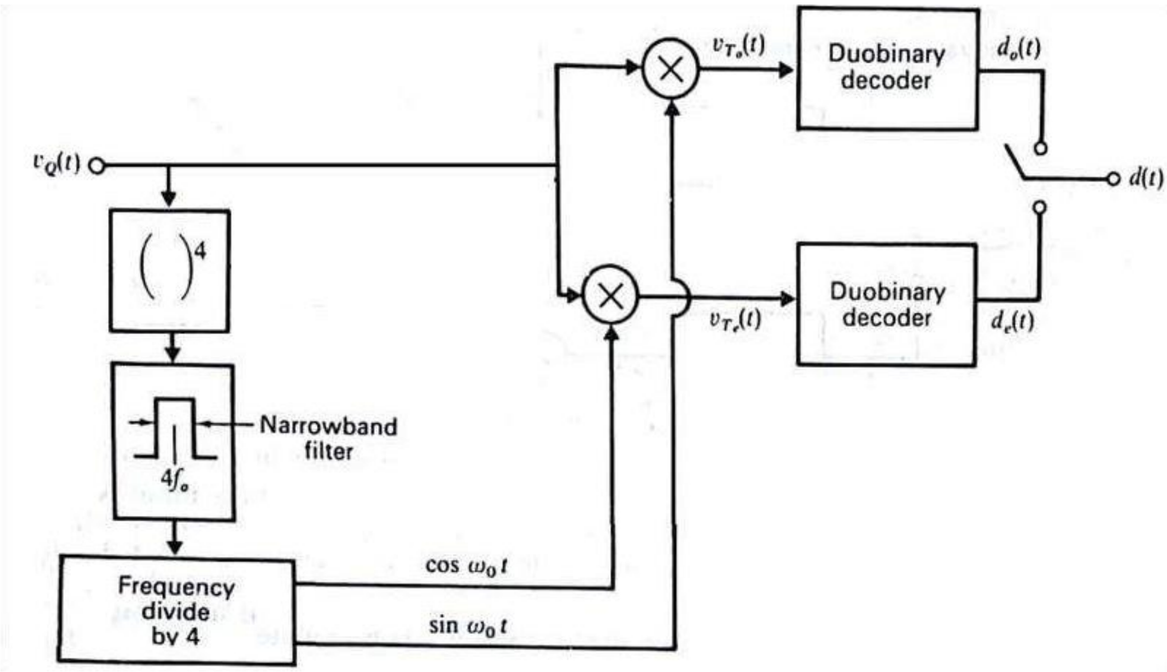


Figure 4.4.2 QPR Decoder

A QPR decoder is shown in Fig. 4.4.2. As in 16-QAM and 16-PSK to decode the input signal, $V_Q(t)$ is first raised to the fourth power, filtered and then frequency divided by 4. The result yields the two quadrature carriers: cos $\omega_o t$ and sin $\omega_o t$. Using the two quadrature carriers we demodulate $V_Q(t)$ and obtain the two baseband duobinary signals $V_{Te}(t)$ and $V_{To}(t)$. Duobinary decoding then takes place; each duobinary decoder being similar to the decoder shown in Fig. 4.3.2b except that they operate at fb/2 rather than at fb. The reconstructed data do(t) and de(t) is then combined to yield the data d(t).

4.5 Eye Pattern

An effective way to study the effects of ISI is the Eye Pattern. The name Eye Pattern was given from its resemblance to the human eye for binary waves. The interior region of the eye pattern is called the eye opening. The following figure shows the image of an eye-pattern.

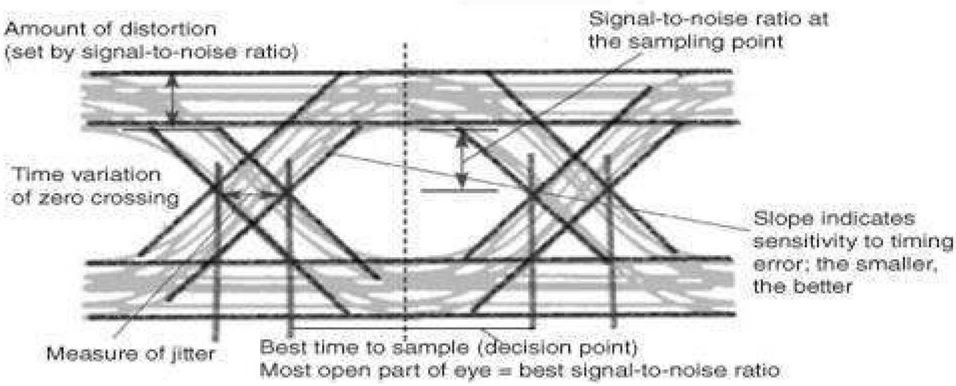


Figure 4.6 Image of eye pattern

Jitter is the short-term variation of the instant of digital signal, from its ideal position, which may lead to data errors.

When the effect of ISI increases, traces from the upper portion to the lower portion of the eye opening increases and the eye gets completely closed, if ISI is very high.

An eye pattern provides the following information about a particular system.

- Actual eye patterns are used to estimate the bit error rate and the signal-to-noise ratio.
- The width of the eye opening defines the time interval over which the received wave can be sampled without error from ISI.
- The instant of time when the eye opening is wide, will be the preferred time for sampling.
- The rate of the closure of the eye, according to the sampling time, determines how sensitive the system is to the timing error.
- The height of the eye opening, at a specified sampling time, defines the margin over noise.

Hence, the interpretation of eye pattern is an important consideration.

4.6 Equalization

For reliable communication to be established, we need to have a quality output. The transmission losses of the channel and other factors affecting the quality of the signal have to be treated. The most occurring loss, as we have discussed, is the ISI.

To make the signal free from ISI, and to ensure a maximum signal to noise ratio, we need to implement a method called Equalization. The following figure shows an equalizer in the receiver portion of the communication system.

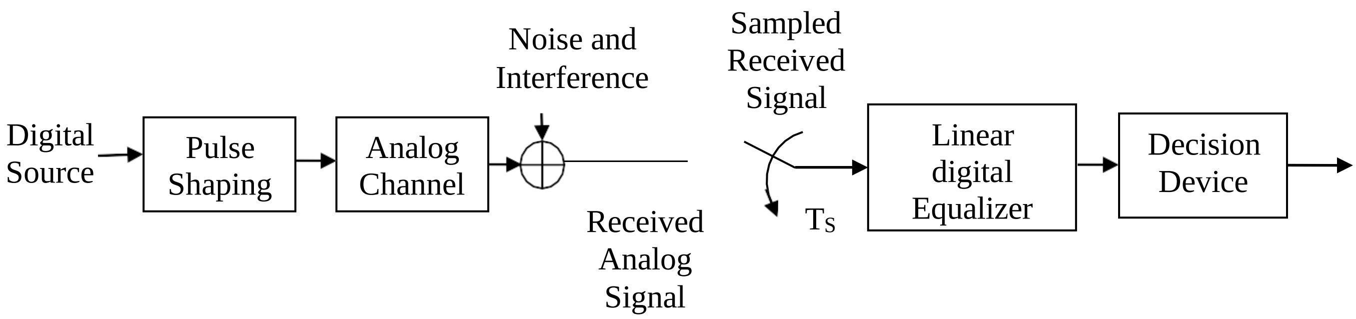


Figure 4.7 Equalization

The noise and interferences which are denoted in the figure are likely to occur, during transmission. The regenerative repeater has an equalizer circuit, which compensates the transmission losses by shaping the circuit. The Equalizer is feasible to get implemented.

Error Probability and Figure-of-merit

The rate at which data can be communicated is called the data rate. The rate at which error occurs in the bits, while transmitting data is called the Bit Error Rate (BER).

The probability of the occurrence of BER is the Error Probability. The increase in Signal to Noise Ratio (SNR) decreases the BER, hence the Error Probability also gets decreased.

In an Analog receiver, the figure of merit at the detection process can be termed as the ratio of output SNR to the input SNR. A greater value of figure-of-merit will be an advantage.

Regenerative Repeater

For any communication system to be reliable, it should transmit and receive the signals effectively, without any loss. A PCM wave, after transmitting through a channel, gets distorted due to the noise introduced by the channel.

The regenerative pulse compared with the original and received pulse, will be as shown in the following figure.

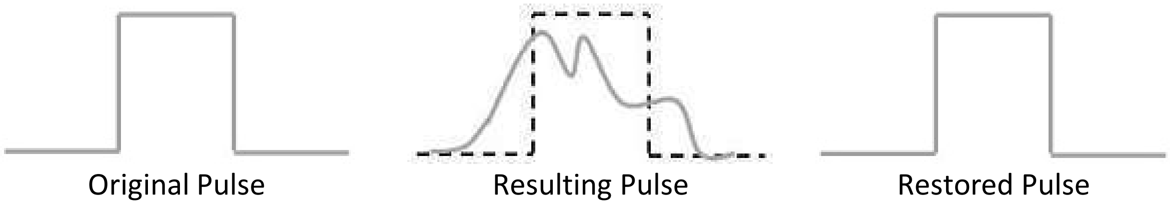


Figure 4.8 Regenerative Repeater

For a better reproduction of the signal, a circuit called as regenerative repeater is employed in the path before the receiver. This helps in restoring the signals from the losses occurred. Following is the diagrammatical representation.

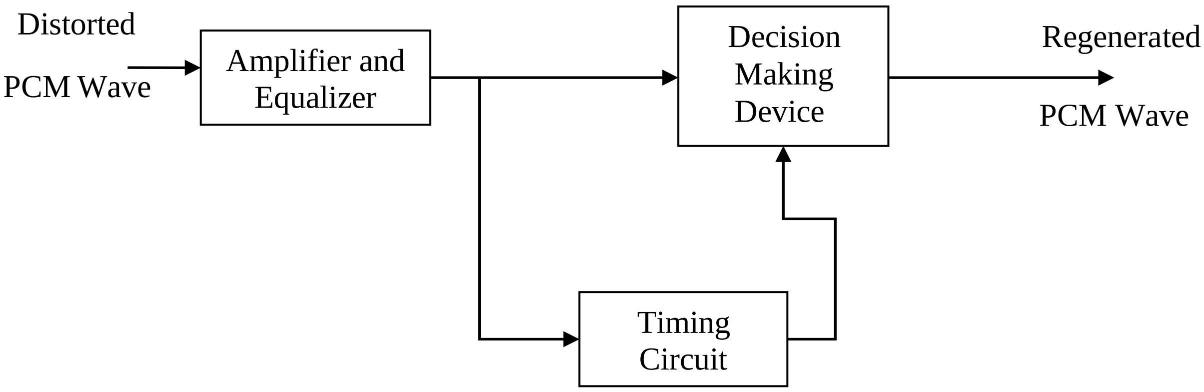


Figure 4.9 Block Diagram of Regenerative Repeater

This consists of an equalizer along with an amplifier, a timing circuit, and a decision making device. Their working of each of the components is detailed as follows.

Equalizer

The channel produces amplitude and phase distortions to the signals. This is due to the transmission characteristics of the channel. The Equalizer circuit compensates these losses by shaping the received pulses.

Timing Circuit

To obtain a quality output, the sampling of the pulses should be done where the signal to noise ratio (SNR) is maximum. To achieve this perfect sampling, a periodic pulse train has to be derived from the received pulses, which is done by the timing circuit.

Hence, the timing circuit allots the timing interval for sampling at high SNR, through the received pulses.

Decision Device

The timing circuit determines the sampling times. The decision device is enabled at these sampling times. The decision device decides its output based on whether the amplitude of the quantized pulse and the noise, exceeds a pre-determined value or not.

4.7 Baseband Signal Receiver:

Consider that a binary-encoded signal consists of a time sequence of voltage levels +V or-V. If there is a guard interval between the bits, the signal forms a sequence of positive and negative pulses. In either case there is no particular interest in preserving the waveform of the signal after reception. We are interested only in knowing within each bit interval whether the transmitted voltage was + V or - V. With noise present, the received signal and noise together will yield sample values generally different from ± V. In this case, what deduction shall we make from the sample value concerning the transmitted bit?

Suppose that the noise is Gaussian and therefore the noise voltage has a probability density which is entirely symmetrical with respect to zero volts. Then the probability that the noise has increased the sample value is the same as the probability that the noise has decreased the sample value. It then seems entirely reasonable that we can do no better than to assume that if the sample value is positive the transmitted level was + V, and if the sample value is negative the transmitted level was - V. It is, of course, possible that at the sampling time the noise voltage may be of magnitude larger than V and of a polarity opposite to the polarity assigned to the transmitted bit. In this case an error will be made as indicated in Fig. 4.7.1. Here the transmitted bit is represented by the voltage + V which is sustained over an interval T from t_1 to t_2. Noise has been superimposed on the level + V so that the voltage *v* represents the received signal and noise. If now the sampling should happen to take place at a time $t=t_1+\Delta t$; an error will have been made.

We can reduce the probability of error by processing the received signal plus noise in such a manner that we are then able to find a sample time where the sample voltage due to the signal is emphasized relative to the sample voltage due to the noise. Such a processer (receiver) is shown in Fig. 4.7.2. The signal input during a bit interval is indicated. As a matter of convenience we have set t = q at the beginning of the interval. The waveform of the signal s(t) before t =0 and after t = T has not been indicated since, as will appear, the operation of the receiver during each bit interval is independent of the waveform during past and future bit intervals.

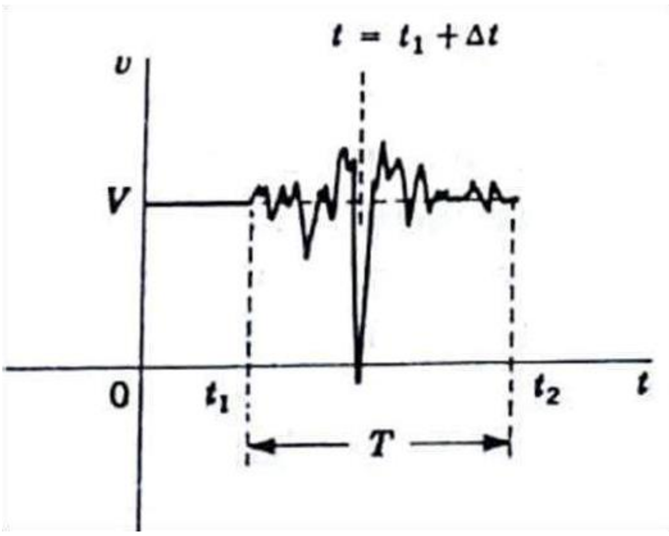


Figure 4.7.1 Illustration that noise may cause an error in determination of transmitted voltage level

The signal s(t) with added white gaussian noise n(t) of power spectral density η/2 is presented to an integrator. At time t = 0 + we require that capacitor C be uncharged. Such a discharged condition may be ensured by a brief closing of switch SW_1 at time t = 0-, thus relieving C of any charge it may have acquired

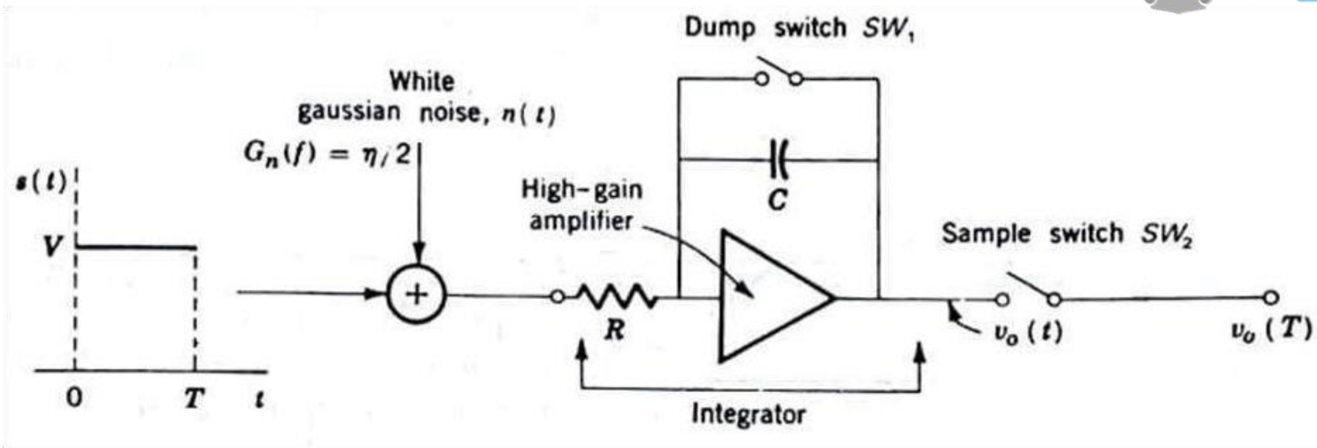


Figure 4.7.2 A Receiver for Binary Coded Signal

during the 'previous interval. The sample is taken at the output of the integrator by closing this sampling switch SW_2. This sample is taken at the end of the bit interval, at t = T. The signal processing indicated in Fig. 4.7.2 is described by the phrase integrate and dump, the term dump referring to the abrupt discharge of the capacitor after each sampling.

Peak Signal to RMS Noise Output Voltage Ratio

The integrator yields an output which is the integral of its input multiplied by 1/RC. Using τ = RC, we have

$$v_o(T) = \frac{1}{r}\int_0^T [s(t) + n(t)]dt = \frac{1}{r}\int_0^T s(t)dt + \frac{1}{r}\int_0^T n(t)dt \qquad \text{...4.7.1}$$

The sample voltage due to the signal is

$$s_o(T) = \frac{1}{r}\int_0^T V dt = \frac{VT}{r} \qquad \text{...4.7.2}$$

The sample voltage due to the noise is

$$s_n(T) = \frac{1}{r}\int_0^T n(t)dt \qquad \text{...4.7.3}$$

This noise-sampling voltage $n_o(T)$ is a Gaussian random variable in contrast with n(t) which is a Gaussian random process.

The variance of $n_o(T)$ is given by

$$\sigma_0^2 = \overline{n_0^2(T)} = \frac{nT}{2r^2} \qquad \text{...4.7.4}$$

It has a Gaussian probability density.

The output, of the integrator, before the sampling switch, is $v_o(t) = S_o(t) + n_o(t)$. As shown in Fig. 4.7.3a, the signal output So(t) is a ramp, in each bit interval, of duration T. At the end of the interval the ramp attains the voltage $S_0(t)$ which is + VT/τ or - VT/τ, depending on whether the bit is a 1 or a 0. At the end of each interval the switch SW_1 in Fig. 4.7.2 closes momentarily to discharge the capacitor so that so(t) drops to zero. The noise $n_0(t)$ shown in Fig. 4.7.3b, also starts each interval with no(0) = 0 and has the random value $n_o(t)$ at the end of each interval. The sampling switch SW_2 closes briefly just before the closing of SW_1 and hence reads the voltage

$$v_o(T) = s_o(T) + n_o(T) \qquad \text{...4.7.5}$$

We would naturally like the output signal voltage to be as large as possible in comparison with the noise voltage. Hence a figure of merit of interest is the signal-to-noise ratio

$$\frac{[s_o(T)]^2}{\overline{n_o^2(T)}} = \frac{2}{\eta}V^2T \qquad \text{...4.7.6}$$

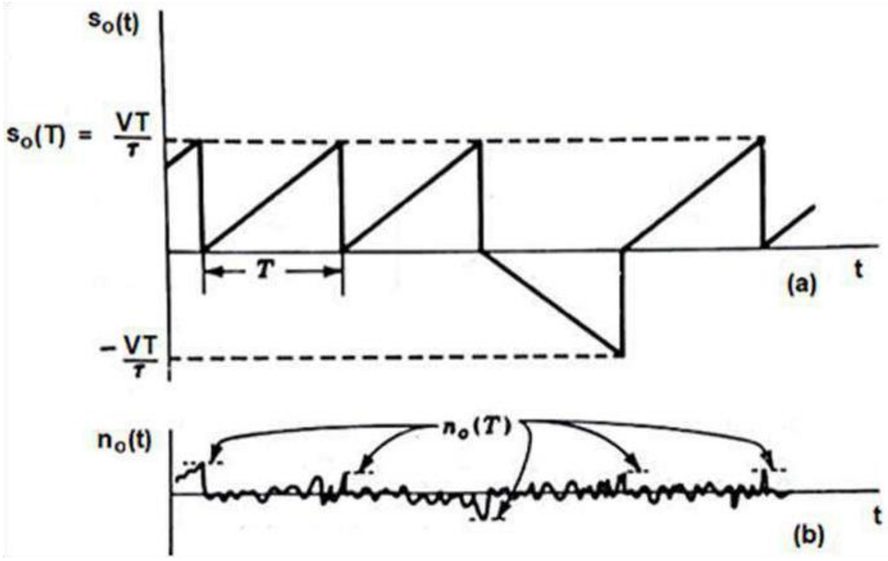


Figure 4.7.3 (a) The Signal Output and (b) the Noise Output of the integrator

This result is calculated from Eqs. (4.7.2) and (4.7.4). Note that the signal-to noise ratio increases with increasing bit duration T and that it depends on V^2T which is the normalized energy of the bit signal. Therefore, a bit represented by a narrow, high amplitude signal and one by a wide, low amplitude signal are equally effective, provided V^2T is kept constant. It is instructive to note that the integrator filters the signal and the noise such that the signal voltage increases linearly with time, while the standard deviation (rms value) of the noise increases more slowly, as $\sqrt{T}$. Thus, the integrator enhances the signal relative to the noise, and this enhancement increases with time as shown in Eq. (4.7.6).

4.8 Probability of Error:

Since the function of a receiver of a data transmission is to distinguish the bit 1 from the bit 0 in the presence of noise, a most important characteristic is the probability that an error will be made in such a determination. We now calculate this error probability P, for the integrate-and-dump receiver of Fig. 4.7.2. We have seen .that the probability density of the noise sample *no(T)* is Gaussian and hence appears as in Fig. 4.7.1.The density is therefore given by

$$f[n_o(T)] = \frac{e^{-n_0^2(T)/2\sigma_0^2}}{\sqrt{2\pi\sigma_0^2}} \qquad \text{...4.8.1}$$

Where σ_0^2 the variance is given by $\sigma_0^2 = \overline{n_0^2(t)}$. Suppose, then, that during some bit interval the input-signal voltage is held at, say, - V. Then, at the sample time, the signal sample voltage is So(T)= -VT/τ, while the noise sample is no(T). If no(T) is positive and larger in magnitude than VT/ τ, the total sample voltage vo(T) = so(T) + no(T) will be positive. Such a positive sample voltage will result in an error, since as noted earlier, we have instructed the receiver to interpret such a positive sample voltage to mean that the signal voltage was + V during the bit interval. The probability of such a misinterpretation, that is, the probability that no(T) > VT/ τ, is given by the area of the shaded region in Fig. 4.8.1. The probability of error is, using Eq. (4.8.1).

$$P_e = \int_{VT/c}^{\infty} f[n_o(T)]\, dn_o(T) = \int_{VT/c}^{\infty} \frac{e^{-n_0^2(T)/2\sigma_0^2}}{\sqrt{2\pi\sigma_0^2}}\, dn_o(T) \qquad \text{...4.8.2}$$

Defining $x \equiv \frac{n_o(T)}{\sqrt{2}\sigma_0}$ and using equations 4.7.4 equation 4.8.2 may be written as

$$P_e = \frac{1}{2}\frac{2}{\sqrt{\pi}}\int_{x=VT/c}^{\infty} e^{-x^2}\, dx = \frac{1}{2} erfc\left(V\sqrt{\frac{T}{\eta}}\right) = \frac{1}{2} erfc\left(\frac{VT}{\eta}\right)^{1/2} \frac{1}{2} erfc\left(\frac{E_S}{\eta}\right)^{1/2} \qquad \text{...4.8.3}$$

In which $E_s = V^2T$ is the signal energy of a bit.

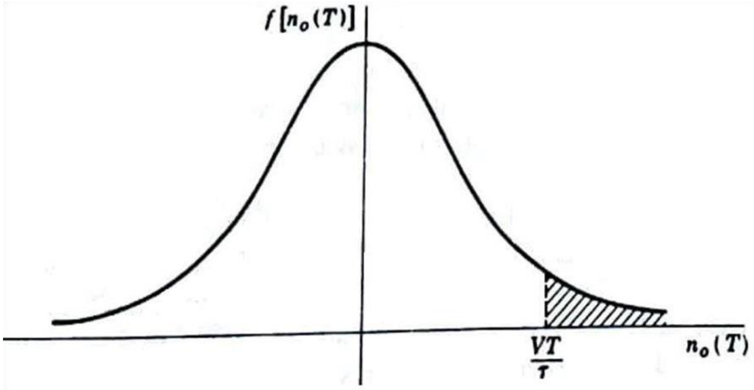


Figure 4.8.1 The Gaussian Probability Density of the noise sample $n_0(T)$

If the signal voltage were held instead at + V during some bit interval, then it is clear from the symmetry of the situation that the probability of error would again be given by P, in Eq. (4.8.3). Hence Eq. (4.8.3) gives P, quite generally.

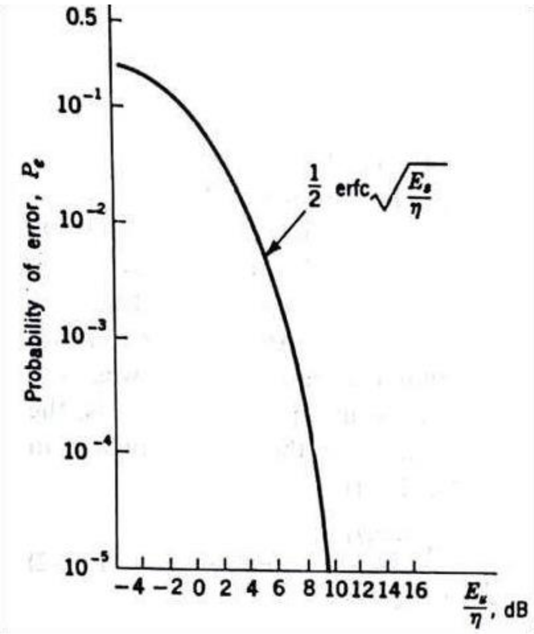


Figure 4.8.2 Variation of P_e versus Es/η

The probability of error Pe as given in Eq. (4.8.3), is plotted in Fig. 4.8.2. Note that Pe decreases rapidly as Es/η increases. The maximum value of Pe is 1/2. Thus, even if the signal is entirely lost in the noise so that any determination of the receiver is a sheer guess, the receiver cannot be wrong more than half the time on the average.

4.9 The Optimum Receiver

In the receiver system of Fig. 4.7.2, the signal was passed through a filter (i.e. the integrator), so that at the sampling time the signal voltage might be emphasized in comparison with the noise voltage. We are naturally led to ask whether the integrator is the optimum filter for the purpose of minimizing the probability of error. We shall find that for the received signal contemplated in the system of Fig. 4.7.2 the integrator is indeed the optimum filter.

We assume that the received signal is a binary waveform. One binary digit (bit) is represented by a signal waveform $S_1(t)$ which persists for time T, while the other bit is represented by the waveform $S_2(t)$ which also lasts for an interval T. For example, in the case of transmission at baseband, as shown in Fig. 4.7.2, $S_1(t) = +V$, while $S_2(t) = -V$; for other modulation systems, different waveforms are transmitted. For example, for PSK signalling, $S_1(t) = A \cos \omega_0 t$ and $S_2(t) = -A \cos \omega_0 t$; while for FSK, $S_1(t) = A \cos (\omega_0+\Omega)t$ and $S_2(t) = A \cos (\omega_0-\Omega)t$.

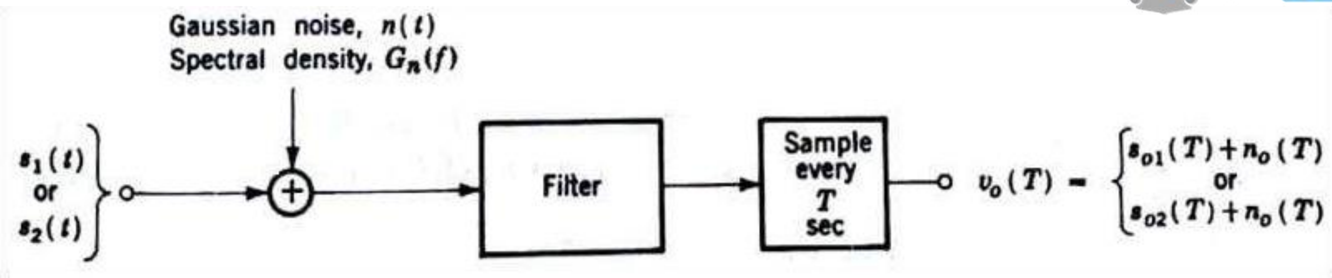


Figure 4.9.1 A Receiver for binary coded signaling

As shown in Fig. 4.9.1 the input, which is $S_1(t)$ or $S_2(t)$, is corrupted by the addition of noise n(t). The noise is gaussian and has a spectral density G(f). [In most cases of interest the noise is white, so that G(J) = η/2. However, we shall assume the more general possibility, since it introduces no complication to do so.] The signal and noise are filtered and then sampled at the end of each bit interval. The output sample is either vo(T) = $S_{01}(T) + n_0(T)$ or $v_o(T) = S_02(T) + n_o(T)$. We assume that immediately after each sample, every energy-storing element in the filter has been discharged.

We note that in the absence of noise the output sample would be $v_o(T) = S_{o1}(T)$ or $S_{02}(T)$. When noise is present we have shown that to minimize the probability of error one should assume that $S_1(t)$ has been transmitted if $v_0(T)$ is closer to $S_{01}(T)$ than to $S_{02}(T)$. Similarly, we assume $S_2(t)$ has been transmitted if $v_o(T)$ is closer to $S_{02}(T)$. The decision boundary is therefore midway between $S_{o1}(T)$ and $S_{02}(T)$. For example, in the baseband system of Fig. 4.7.2, where $S_{01}(T) = VT/\tau$ and $S_{02}(T) = - VT/\tau$, the decision boundary is $v_o(T) = 0$. In general. we shall take the decision boundary to be

$$v_o(T) = \frac{s_{o1}(T) + s_{o2}(T)}{2} \qquad ...4.9.1$$

The probability of error for this general case may be deduced as an extension of the considerations used in the baseband case. Suppose that $S_{01}(T) > S_{02}(T)$ and that $S_2(t)$ was transmitted. If, at the sampling time the noise $n_o(T)$ is positive and larger in magnitude than the voltage difference (1/2)[$S_{01}(T)$+ $S_{02}(T)$] - $S_{02}(T)$, an error will have' been made. That is, an error will result if

$$n_o(T) \geq \frac{s_{o1}(T) - s_{o2}(T)}{2} \qquad ...4.9.2$$

Hence the probability of error is

$$P_e = \int_{[s_{o1}(T)-s_{o2}(T)]/2}^{\infty} \frac{e^{-n_0^2(T)/2\sigma_0^2}}{\sqrt{2\pi\sigma_0^2}} dn_o(T) \qquad ...4.9.3$$

If we make the substitution $x \equiv \frac{n_o(T)}{\sqrt{2}\sigma_0}$, then above equation becomes,

$$P_e = \frac{1}{2}\frac{2}{\sqrt{\pi}}\int_{[s_{o1}(T)-s_{o2}(T)]/2\sqrt{2}\sigma_0}^{\infty} e^{-x^2}\, dx \qquad ...4.9.4a$$

$$P_e = \frac{1}{2} erfc\left[\frac{s_{o1}(T) - s_{o2}(T)}{2\sqrt{2}\sigma_0}\right] \qquad ...4.9.4b$$

Note that for the case $S_{01}(T) = VT/\tau$ and $S_{02}(T) = - VT/\tau$, and, using Eq. (4.7.4), Eq. (4.9.4b) reduces to Eq. (4.8.3) as expected.

The complementary error function is a monotonically decreasing function of its argument. (See Fig. 4.8.2.) Hence, as is to be anticipated, Pe decreases as the difference $S_{01}(T) - S_{02}(T)$ becomes larger and as the rms noise voltage σ_0 becomes smaller. The optimum filter, then, is the filter which maximizes the ratio

$$\gamma = \frac{s_{o1}(T) - s_{o2}(T)}{\sigma_0} \qquad ...4.9.5$$

We now calculate the transfer function H(f) of this optimum filter.

Optimum Filter Transfer Function *H(f)*

The fundamental requirement we make of a binary encoded data receiver is that it distinguishes the voltages $S_1(t) + n(t)$ and $S_2(t) + n(t)$. We have seen that the ability of the receiver to do so depends on how large a particular receiver can make γ. It is important to note that it is proportional not to $S_1(t)$ nor to $S_2(t)$, but rather to the difference between them. For example, in the baseband system we represented the signals by voltage levels + V and - V. But clearly, if our only interest was in distinguishing levels, we would do just as well to use +2 volts and 0 volt, or +8 volts and +6 volts, etc, (The + V and - V levels, however, have the advantage of requiring the least average power to be transmitted.) Hence, while $S_1(t)$ or $S_2(t)$ is the received signal, the signal which is to be compared with the noise, i.e., the signal which is relevant in all our error-probability calculations, is the difference signal

$$p(T) \equiv s_1(T) - s_2(T) \qquad \text{...4.9.6}$$

Thus, for the purpose of calculating the minimum error probability, we shall assume that the Input Signal to the optimum filter is p(t). The corresponding output signal of the filter is then

$$p_0(T) \equiv s_{01}(T) - s_{02}(T) \qquad \text{...4.9.7}$$

Let *P(f)* and *P₀(f)* ne the Fourier transform of *p(f)* and *P₀(f)* respectively. If H(f) is the transfer function of the filter

$$P_0(f) = H(f)P(f) \qquad \text{...4.9.8}$$

And

$$p_0(T) = \int_{-\infty}^{\infty} P_0(f)\, e^{j2\pi fT} df = \int_{-\infty}^{\infty} H(f)P(f)\, e^{j2\pi fT} df \qquad \text{...4.9.9}$$

The input noise to the optimum filler is n(t). The output noise is $n_0(t)$ which has a power spectral density $Gn_0(f)$ and is related to the power spectral density of the input noise Gn(f) by

$$G_{n0}(f) = |H(f)|^2 G_n(f) \qquad \text{...4.9.10}$$

Using Parseval's theorem, we find that the normalized output noise power, i.e., the noise varianceσ_0^2, is

$$\sigma_0^2 = \int_{-\infty}^{\infty} G_{n0}(f)\, df = \int_{-\infty}^{\infty} |H(f)|^2 G_n(f)\, df \qquad \text{...4.9.11}$$

From equation 9 and 11,

$$\gamma^2 = \frac{p_0^2(T)}{\sigma_0^2} = \frac{\left[\int_{-\infty}^{\infty} H(f)P(f)\, e^{j2\pi fT} df\right]^2}{\int_{-\infty}^{\infty} |H(f)|^2 G_n(f)\, df} \qquad \text{...4.9.12}$$

Equation 4.9.12 is unaltered by the inclusion or deletion of the absolute value sign in the numerator since the quantity within the magnitude sign $p_0(T)$ is a positive real number. The sign has been included, however, in order to allow further development of the equation through the use of the Schwarz inequality. The Schwarz inequality state that given arbitrary complex functions X(f) and Y(f) of a common variable f, then

$$\left|\int_{-\infty}^{\infty} X(f)Y(f)\, df\right|^2 \leq \int_{-\infty}^{\infty} |X(f)|^2\, df \int_{-\infty}^{\infty} |Y(f)|^2\, df \qquad \text{...4.9.13}$$

The equal sign applies when

$$X(f) = KY^*(f) \qquad \text{...4.9.14}$$

where K is an arbitrary constant and Y*(f) is the complex conjugate of Y(f).

We now apply the Schwarz inequality to Eq. (4.9.12) by making the identification

$$X(f) \equiv \sqrt{G_n(f)}\, H(f) \qquad \text{...4.9.15}$$

and

$$Y(f) \equiv \frac{1}{\sqrt{G_n(f)}} P(f) e^{j2\pi Tf} \qquad \text{...4.9.16}$$

Using equation 15 and 16 and using Schwarz inequality equation 13, we may write equation 12 as,

$$\frac{p_0^2(T)}{\sigma_0^2} = \frac{\left[\int_{-\infty}^{\infty} X(f)Y(f)\,df\right]^2}{\int_{-\infty}^{\infty} |H(f)|^2\,df} \le \int_{-\infty}^{\infty} |Y(f)|^2\,df \qquad \text{...4.9.17}$$

Using equation 16,

$$\frac{p_0^2(T)}{\sigma_0^2} \le \int_{-\infty}^{\infty} |Y(f)|^2\,df = \int_{-\infty}^{\infty} \frac{[P(f)]^2}{G_n(f)}\,df \qquad \text{...4.9.18}$$

The ratio $p_0^2(T)/\sigma_0^2$; will attain its maximum value when the equal sign in Eq. (4.9.18) may be employed as is the case when X(f) = K Y*(f). We then find from Eqs. (4.9.15) and (4.9.16) that the optimum filter which yields such a maximum ratio $p_0^2(T)/\sigma_0^2$; has a transfer function

$$H(f) = K\frac{P^*(f)}{G_n(f)}e^{-j2\pi fT} \qquad \text{...4.9.19}$$

Correspondingly, the maximum ratio is, from Eq. (4.9.18),

$$\left[\frac{p_0^2(T)}{\sigma_0^2}\right]_{max} = \int_{-\infty}^{\infty} \frac{[P(f)]^2}{G_n(f)}\,df \qquad \text{...4.9.20}$$

4.10 White Noise: The Matched Filter

An optimum filter which yields a maximum ratio $p_0^2(T)/\sigma_0^2$is called a matched filler when the input noise is white. In this case $G_n(f) = \eta/2$, and Eq. (4.9.19) becomes

$$H(f) = K\frac{P^*(f)}{\eta/2}e^{-j2\pi fT} \qquad \text{...4.10.1}$$

The impulsive response of this filter, i.e., the response of the filter to a unit strength impulse applied at t = 0, is

$$h(t) = F^{-1}[H(f)] = \frac{2K}{\eta}\int_{-\infty}^{\infty} P^*(f)e^{-j2\pi fT}\,e^{-j2\pi ft}df \qquad \text{...4.10.2(a)}$$

$$= \frac{2K}{\eta}\int_{-\infty}^{\infty} P^*(f)e^{j2\pi f(t-T)}\,df \qquad \text{...4.10.2(b)}$$

A physically realizable filter will have an impulse response which is real, i.e., not complex. Therefore h(t) = h*(t). Replacing the right-hand member of Eq. (4.10.2b) by its complex conjugate, an operation which leaves the equation unaltered, we have

$$h(t) = \frac{2K}{\eta}\int_{-\infty}^{\infty} P(f)e^{j2\pi f(t-T)}\,df \qquad \text{...4.10.3(a)}$$

$$= \frac{2K}{\eta}p(T-t) \qquad \text{...4.10.3(b)}$$

Finally since $p(t)=s_1(t) - s_2(t)$, we have

$$h(t) = \frac{2K}{\eta}[s_1(T-t) - s_2(T-t)] \qquad \text{...4.10.4}$$

As shown in Fig. 4.10.1a, the $s_1(t)$ is a triangular waveform of duration T, while $s_2(t)$, (Fig. 4.10.1b), is of identical form except of reversed polarity. Then p(t) is as shown in Fig. 4.10.1c, and p(-t) appears in Fig. 4.10.1d. The waveform p(-t) is the waveform p(t) rotated around the axis t =0. Finally, the waveform p(T - t) called for as the impulse response of the filter in Eq. (4.10.3b) is this rotated waveform p(-t) translated in the positive t direction by amount T. This last translation ensures that h(t) = 0 for t < 0 as is required for a causal filter.

In general, the impulsive response of the matched filter consists of p(t) rotated about t = 0 and then delayed long enough (i.e., a time T) to make the filter realizable. We may note in passing, that any additional delay that a filter might introduce would in no way interfere with the performance of the filter, for both signal and noise would be delayed by the same amount, and at the sampling the ratio of signal to noise would remain unaltered.

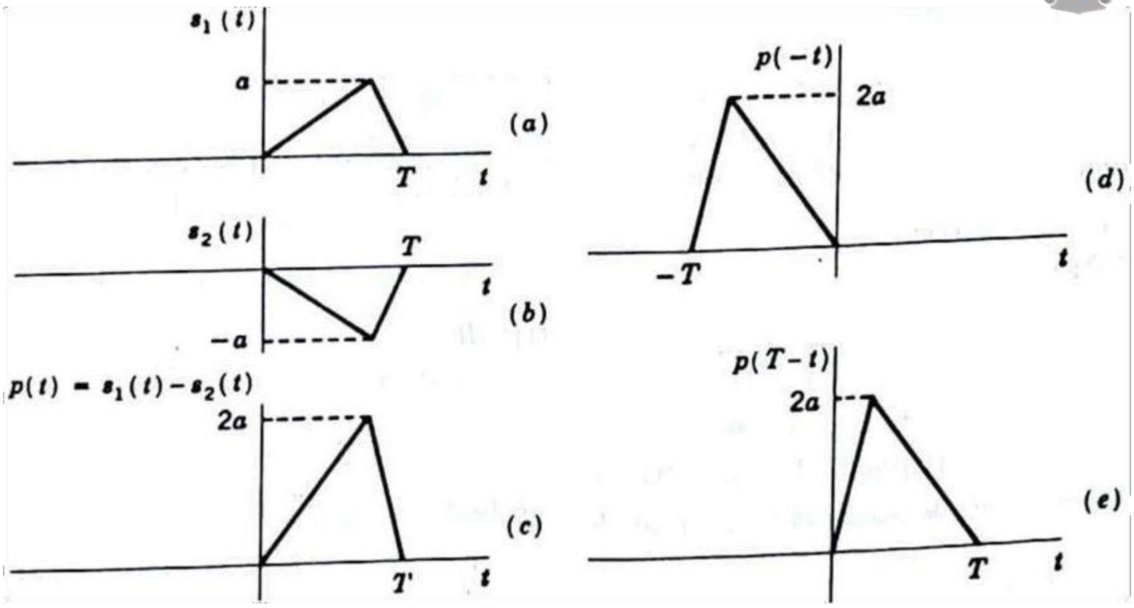


Figure 4.10.1 The signals (a) $s_1(t)$, (b) $s_2(t)$, (c) $p(t)=s_1(t)- s_2(t)$, (d) p(t) rotated about the axis t=0, (e) The waveform of (d) translated to right by amount T.

4.11 Correlator

Coherent Detection: Correlation

Coherent detection is an alternative type of receiving system, which is identical in performance with the matched filter receiver. Again, as shown in Fig. 4.11.1, the input is a binary data waveform $S_1(t)$ or $S_2(t)$ corrupted by noise n(t). The bit length is T. The received signal plus noise $v_i(t)$ is multiplied by a locally generated waveform $S_1(t)$ - $S_2(t)$. The output of the multiplier is passed through an integrator whose output is sampled at t = T. As before, immediately after each sampling, at the beginning of each new bit interval, all energy-storing elements in the integrator are discharged. This type of receiver is called a correlator, since we are correlating the received signal and noise with the waveform $S_1(t)$- $S_2(t)$.

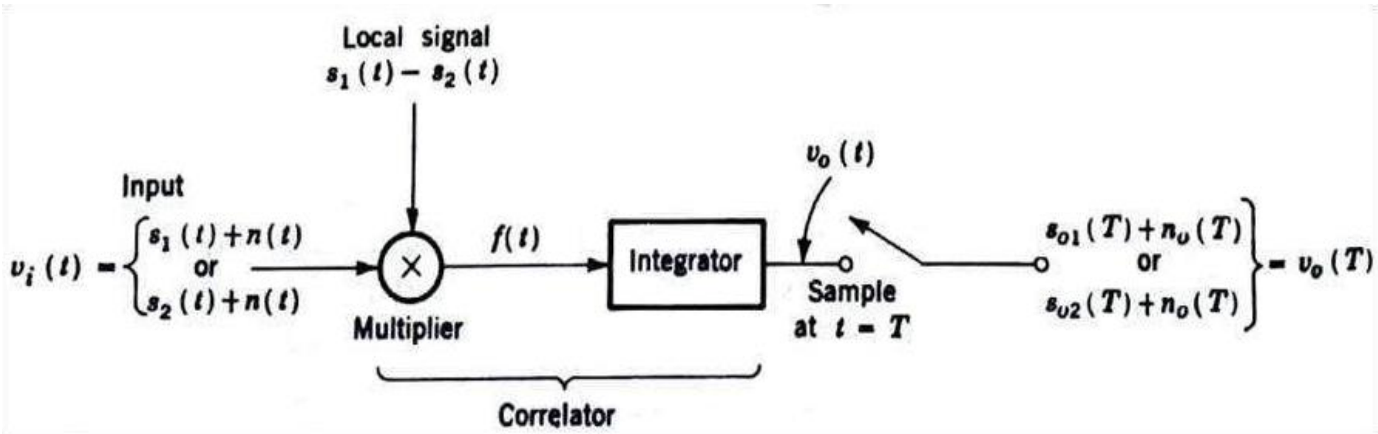


Figure 4.11.1 A Coherent System of Signal Reception

The output signal and noise of the correlator shown in Fig. 4.11.1 are

$$s_0(t) = \frac{1}{\tau}\int_0^T S_i(t)[S_1(t) - S_2(t)]\, dt \qquad \text{...4.11.1}$$

$$n_0(t) = \frac{1}{\tau}\int_0^T n(t)[S_1(t) - S_2(t)]\, dt \qquad \text{...4.11.2}$$

where $S_i(t)$ is either $S_1(t)$ or $S_2(t)$, and where τ is the constant of the integrator (i.e., the integrator output is l/τ times the integral of its input). We now compare these outputs with the matched filter outputs.

If h(t) is the impulsive response of the matched filter, then the output of the matched filter $v_o(t)$ can be found using the convolution integral. We have

$$v_0(t) = \int_{-\infty}^{\infty} v_i(\lambda)h(t-\lambda)\, d\lambda = \int_{0}^{T} v_i(\lambda)h(t-\lambda)\, d\lambda \qquad ...4.11.3$$

The limits on the integral have been changed to 0 and T since we are interested in the filter response to a bit which extends only over that interval. Using Eq. (4.10.4) which gives h(t) for the matched filter, we have

$$h(t) = \frac{2K}{\eta}[s_1(T-t) - s_2(T-t)] \qquad ...4.11.4$$

So that $$h(t-\lambda) = \frac{2K}{\eta}[s_1(T-t+\lambda) - s_2(T-t+\lambda)] \qquad ...4.11.5$$

Submitting equation 4.11.5, in equation 4.11.3

$$v_0(t) = \frac{2K}{\eta}\int_0^T v_i(\lambda)[s_1(T-t+\lambda) - s_2(T-t+\lambda)]\, d\lambda \qquad ...4.11.6$$

Since $v_i(\lambda) = s_i(\lambda)+n(\lambda)$, and $v_0(t) = s_0(t)+n_0(t)$, setting t=T yields,

$$s_0(t) = \frac{2K}{\eta}\int_0^T s_i(\lambda)[s_1(\lambda) - s_2(\lambda)]d\lambda \qquad ...4.11.7$$

Where $s_i(\lambda)$ is equal to $s_1(\lambda)$ or $s_2(\lambda)$. Simillarly,

$$n_0(t) = \frac{2K}{\eta}\int_0^T n(\lambda)[s_1(\lambda) - s_2(\lambda)]d\lambda \qquad ...4.11.8$$

Thus as we can see from above equations $s_o(T)$ and $n_o(T)$, are identical. Hence the performances of the two systems are identical.

The matched filter and the correlator are not simply two distinct, independent techniques which happen to yield the same result. In fact they are two techniques of synthesizing the optimum filter h(t).

4.12 Probability of error calculation for BPSK and BFSK

(i) BPSK

The synchronous detector for BPSK is shown in figure 4.12.1(b). Since the BPSK signal is one dimensional, The only relevant noise in the present case is

$$n(t) = n_0 u(t) = n_0\sqrt{2/T_b}\cos\omega_0 t \qquad ...4.12.1$$

where n_o is a Gaussian random variable of variance $\sigma_0^2 = \eta/2$. Now let us suppose that S_2 was transmitted.

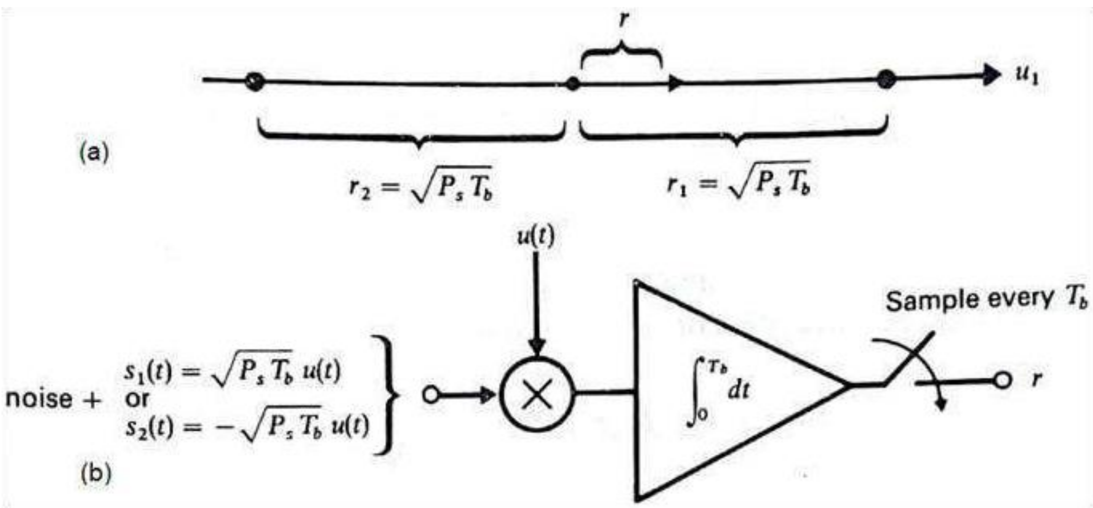


Figure 4.12.1 (a) BPSK representation in signal space showing r_1 and r_2 (b)Correlator receiver for BPSK showing that $r=r_1+n_0$ or $r_2+ n_0$

The error probability, i.e., the probability that the signal is mistakenly judged to be S_1 is the probability that $n_0 > \sqrt{P_s T_b}$. Thus the error probability P_e, is

$$P_e = \frac{1}{\sqrt{2\pi\sigma^2}}\int_{\sqrt{P_s T_b}}^{\infty} e^{-n_0^2/2\sigma_0^2}\, dn_0 = \frac{1}{\sqrt{\pi\eta}}\int_{\sqrt{P_s T_b}}^{\infty} e^{-n_0^2/\gamma}\, dn_0 \qquad ...4.12.2$$

Let $y^2 = n_0^2/2\,\sigma_0^2$, then

$$P_e = \frac{1}{\sqrt{\pi}}\int_{\sqrt{P_sT_b/y}}^{\infty} e^{-y^2}\,dy = \frac{1}{2}\frac{2}{\sqrt{\pi}}\int_{\sqrt{P_sT_b/y}}^{\infty} e^{-y^2}\,dy = \frac{1}{2}\,erfc\,\sqrt{P_sT_b/\eta} \qquad \text{...4.12.3}$$

The signal energy is $E_b = P_sT_b$ and the distance between end points of the signal vectors in Fig. 4.12.1 is $= 2\sqrt{P_sT_b}$. Accordingly we find that

$$P_e = \frac{1}{2}\,erfc\,\sqrt{E_b/\eta} = \frac{1}{2}\,erfc\,\sqrt{d^2/4\eta} \qquad \text{...4.12.4}$$

The error probability is thus seen to fall off monotonically with an increase in distance between signals.

(ii) BFSK

The case of synchronous detection of orthogonal binary FSK is represented in Fig. 4.12.2. The signal space is shown in (a). The unit vectors are

$$u_1(t) = \sqrt{2/T_b}\cos\omega_1 t \qquad \text{...4.12.5a}$$

and

$$u_2(t) = \sqrt{2/T_b}\cos\omega_2 t \qquad \text{...4.12.5b}$$

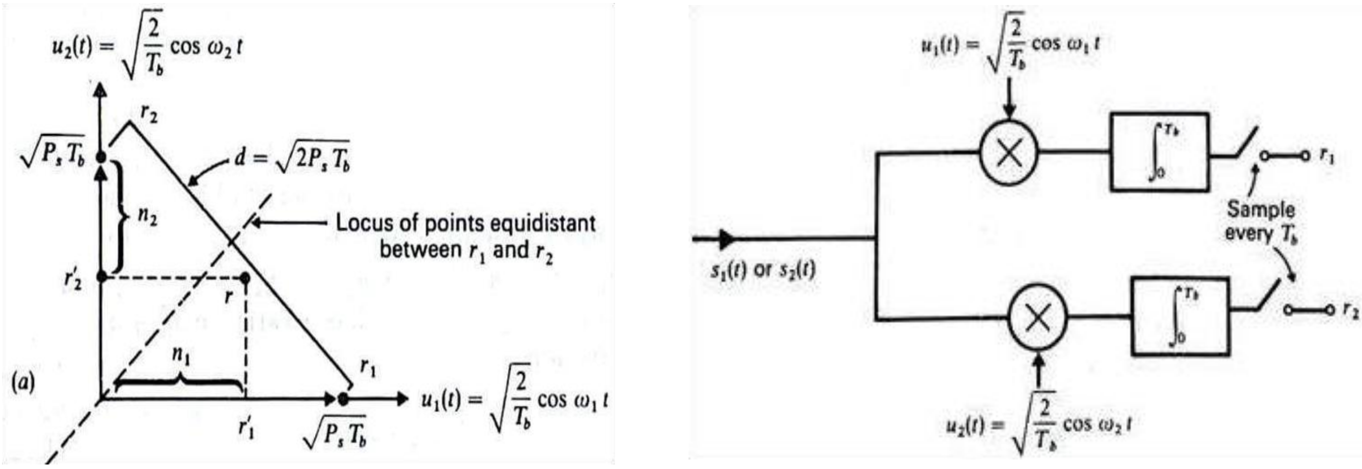


Figure 4.12.2 (a) Signal Space representation of BFSK (b) Correlator Receiver for BFSK

Orthogonality over the interval T_b having been insured by the selection of ω_1 and ω_2. The transmitted signals s_1 and s_2 are of power P_s, and are

$$s_1(t) = \sqrt{2P_s}\cos\omega_1 t = \sqrt{P_sT_b}\sqrt{2/T_b}\cos\omega_1 t = \sqrt{P_sT_b}\,u_1(t) \qquad \text{...4.12.6a}$$

and

$$s_2(t) = \sqrt{2P_s}\cos\omega_2 t = \sqrt{P_sT_b}\sqrt{2/T_b}\cos\omega_2 t = \sqrt{P_sT_b}\,u_2(t) \qquad \text{...4.12.6b}$$

Detection is accomplished in the manner shown in Fig. 4.12.2 (b). The outputs are r_1 and r_2. In the absence of noise when $s_1(t)$ is received, $r_2 = 0$ and $r_1 = \sqrt{P_sT_b}$. For $S_2(t)$, $r_1 = 0$ and $r_2 = \sqrt{P_sT_b}$. Hence the vectors representing r_1 and r_2 are of length $\sqrt{P_sT_b}$ as shown in Fig. 4.12.2(a).

Since the signal is two dimensional the relevant noise in the present case is

$$n(t) = n_1 u_1(t) + n_2 u_2(t) \qquad \text{...4.12.7}$$

In which n_1 and n_2 are Gaussian random variables each of variance $\sigma_1^2 = \sigma_2^2 = \eta/2$. Now let us suppose that $S_2(t)$ is transmitted and that the observed voltages at the output of the processor are r'_1 and r'_2 as shown in Fig. 4.12.2a. We find that $r'_2 \neq r_2$ because of the noise n_2 and $r'_1 \neq 0$ because of the noise n_1. We have drawn the locus of points equidistant from r_1 and r_2 and suppose, that the received voltage r, is closer to r_1 than to r_2. Then we shall have made an error in estimating which signal was transmitted. It is readily apparent that such an error will occur whenever $n_1 > r_2 - n_2$ or $n_1 + n_2 > \sqrt{P_sT_b}$. Since n_1 and n_2 are uncorrelated, the random variable $n_0 = n_1 + n_2$ has a variance $\sigma_0^2 = \sigma_1^2 + \sigma_2^2 = \eta$ and its probability density function is

$$f(n_0) = \frac{1}{\sqrt{2\pi n}}e^{-n_0^2/2y} \qquad \text{...4.12.8}$$

The probability of error is

$$P_e = \frac{1}{\sqrt{2\pi\eta}} \int_{\sqrt{PsTb}}^{\infty} e^{-n_0^2/2y}\, dn_0 \qquad \text{...4.12.9}$$

Again we have $E_b = P_sT_b$ and in the present case the distance between r_1 and r_2 is $d = \sqrt{2}\sqrt{P_sT_b}$. Accordingly, proceeding as in Eq. (4.12.2) we find that

$$P_e = \frac{1}{2} erfc\sqrt{E_b/2\eta} \qquad \text{...4.12.10a}$$

$$= \frac{1}{2} erfc\sqrt{d^2/2\eta} \qquad \text{...4.12.10b}$$

Comparing Eqs. (4.12.10b) and (4.12.4) we see that when expressed in terms of the distance d, the error probabilities are the same for BPSK and BFSK.

............X-............

Unit 5

Syllabus:

Information Theory Source Coding: Introduction to information theory, uncertainty and information, average mutual information and entropy, source coding theorem, Huffman coding, Shannon-Fano-Elias coding, Channel Coding: Introduction, channel models, channel capacity, channel coding, information capacity theorem, Shannon limit.

Encoding is the process of converting the data or a given sequence of characters, symbols, alphabets etc., into a specified format, for the secured transmission of data.
Decoding is the reverse process of encoding which is to extract the information from the converted format.

Data Encoding

Encoding is the process of using various patterns of voltage or current levels to represent 1s and 0s of the digital signals on the transmission link.
The common types of line encoding are Unipolar, Polar, Bipolar, and Manchester.

Encoding Techniques

The data encoding technique is divided into the following types, depending upon the type of data conversion.

- **Analog data to Analog signals** − The modulation techniques such as Amplitude Modulation, Frequency Modulation and Phase Modulation of analog signals, fall under this category.
- **Analog data to Digital signals** − This process can be termed as digitization, which is done by Pulse Code Modulation (PCM). Hence, it is nothing but digital modulation. As we have already discussed, sampling and quantization are the important factors in this. Delta Modulation gives a better output than PCM.
- **Digital data to Analog signals** − The modulation techniques such as Amplitude Shift Keying (ASK), Frequency Shift Keying (FSK), Phase Shift Keying (PSK), etc., fall under this category. These will be discussed in subsequent chapters.
- **Digital data to Digital signals** − These are in this section. There are several ways to map digital data to digital signals. Some of them are −

Information is the source of a communication system, whether it is analog or digital. Information theory is a mathematical approach to the study of coding of information along with the quantification, storage, and communication of information.

Conditions of Occurrence of Events

If we consider an event, there are three conditions of occurrence.

- If the event has not occurred, there is a condition of uncertainty.
- If the event has just occurred, there is a condition of surprise.
- If the event has occurred, a time back, there is a condition of having some information.

These three events occur at different times. The differences in these conditions help us gain knowledge on the probabilities of the occurrence of events.

5.01 Information Theory:

Information theory provides a quantitative measure of the information contained in a message signal and allow us to determine the capacity of a communication system to transfer this information from source to destination.

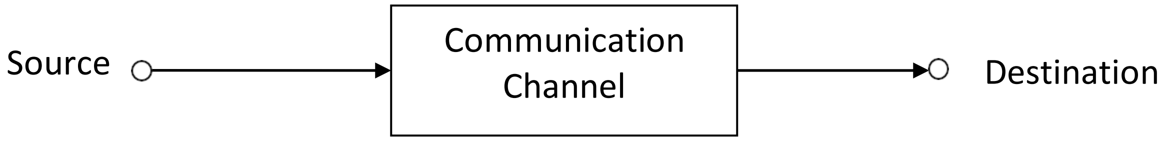


Figure 5.01 A Communication System

1. The information associated with any event depends upon the probability with which it exists.
2. Higher the probability of occurring, lower the information associated with it and vice versa.
3. If the probability of occurring is 1, the information associated with that event is zero since we are certain that a particular bit actually exists at the input of the system.

5.02 Information Source:

An information source is an object that produces an event, the outcome of which is selected at random according to a probability distribution. A discrete information source has only a finite set of symbols as possible outputs.

A source with memory is one for which a current symbol depends on the previous one.

A memory less source is one for which each symbol produced is independent of the previous symbol.

Information: Anything to which some meaning or sense can be attached is called the information. Example, a written message, a spoken word, a picture etc.

5.03 Unit of Information:

Consider a DMS(discrete memory less source), denoted by X, with alphabet $\{x_1, x_2, \ldots x_m\}$. The information content of a symbol x_i, denoted by $I(x_i)$, is defined by

$$I(x_i) = \log_b \frac{1}{P(x_i)} = -\log_b P(x_i) \qquad \ldots 5.1.1$$

Where *$P(x_i)$* is the probability of occurrence of symbol x_i. The *$I(x_i)$* satisfy the following conditions i.e. properties

(1) $I(x_i) = 0 \; for \; P(x_i) = 1$

(2) $I(x_i) \geq 0$

(3) $I(x_i) \geq I(x_j)$ if $P(x_i) \geq P(x_j)$

(4) $I(x_i.x_j) = I(x_i) + I(x_j)$ if x_i and x_j are independent

The unit of *$I(x_i)$* is bit if b=2, Hartley or decit if b=10 and nat (natural unit) if b=e.

$I(x_i) = -\log_2 P(x_i)$ bits

$I(x_i) = -\log_{10} P(x_i)$ hartley or decit

$I(x_i) = -\log_e P(x_i)$ nats

It is standard to use b=2.

The unit bit (b) is a measure of information content.

Conversion of Units

$$\log_2 a = \frac{\ln a}{\ln 2} = \frac{\log a}{\log 2}$$

$$\log_2 6 = \frac{\log_{10} 6}{\log_{10} 2} = \frac{1}{\log_{10} 2} \log_{10} 6$$

5.03 Average Information or Entropy

In a practical Communication system, we usually transmit long sequences of symbol from an information source. Hence it is more important to find the average information that a source produces than the information content of a single symbol.

The mean value of *$I(x_i)$* over the alphabet of source X, with in different symbol is given by-

$$H(x) = E[I(x_i)] = \sum_{i=1}^{m} P(x_i).\, I(x_i)$$

$$H(x) = E[I(x_i)] = -\sum_{i=1}^{m} P(x_i).\log_b P(x_i) \quad \text{b/symbol}$$

The quantity H(x) is called the entropy of source X. It is a measure of the average information content per source symbol. The source entropy H(x) can be considered as the average amount of uncertainty with in the source X.

For a binary source X, that generates independent symbols 0 & 1, with equal probability, the source entropy H(x) is

$$H(x) = -\frac{1}{2} log_2 \left(\frac{1}{2}\right) - \frac{1}{2} log_2 \left(\frac{1}{2}\right)$$

$$H(x) = \frac{1}{2} log_2 2 + \frac{1}{2} log_2 2 \qquad 1 \text{ b/symbol}$$

The source entropy H(x) satisfy the following relation

$$0 \leq H(x) \leq log_2 m$$

Where m is the number of symbols (also called size of alphabet of source (X)

Case 1 If in input only one digit is occurring i.e. m=1; P(i)=1

$$H(x) = 0$$

Case 2 If all m digits are equal probable i.e. P(i) = 1/m

$$H(x) = -\sum_{i=1}^{m} P(i).\log_2 P(i) = -\sum_{i=1}^{m} \frac{1}{m}.\log_2 \frac{1}{m}$$

$$= -\log_2 \frac{1}{m} = \log_2 m$$

This is the maximum value, therefore

$$H(x)|_{max} = \log_2 m$$

$$0 \leq H(x) \leq log_2 m$$

Entropy

When we observe the possibilities of the occurrence of an event, how surprising or uncertain it would be, it means that we are trying to have an idea on the average content of the information from the source of the event.

Entropy can be defined as a measure of the average information content per source symbol. Claude Shannon, the “father of the Information Theory”, provided a formula for it as –

$$H = -\sum_i p_i \log_b p_i$$

Where p_i is the probability of the occurrence of character number i from a given stream of characters and b is the base of the algorithm used. Hence, this is also called as Shannon’s Entropy.

The amount of uncertainty remaining about the channel input after observing the channel output, is called as Conditional Entropy. It is denoted by $H(X \mid Y)$.

5.04 Mutual Information

Let us consider a channel whose output is Y and input is X

Let the entropy for prior uncertainty be X = H(x)

(This is assumed before the input is applied)

To know about the uncertainty of the output, after the input is applied, let us consider Conditional Entropy, given that Y = y_k

$$H(x \mid y_k) = \sum_{j=0}^{j-1} p(x_j \mid y_k)\, log_2\left[\frac{1}{p(x_j \mid y_k)}\right]$$

$$H(X \mid y = y_0) \ldots\ldots\ldots\ldots\ldots\ldots\ldots. H(X \mid y = y_k)$$

This is a random variable for $p(y_0) \ldots\ldots\ldots\ldots\ldots\ldots\ldots. p(y_{k-1})$ with probabilities respectively. The mean value of $H(X / y = y_k)$ for output alphabet y is –

$$H(X \mid Y) = \sum_{k=0}^{k-1} H(X \mid y = y_k)\, p(y_k)$$

$$= \sum_{k=0}^{k-1} \sum_{j=0}^{j-1} p(x_j \mid y_k)\, p(y_k) log_2[\frac{1}{p(x_j \mid y_k)}]$$

$$= \sum_{k=0}^{k-1} \sum_{j=0}^{j-1} p(x_j, y_k)\, log_2[\frac{1}{p(x_j \mid y_k)}]$$

Now, considering both the uncertainty conditions (before and after applying the inputs), we come to know that the difference, i.e.

$$H(x) - H(x \mid y)$$

must represent the uncertainty about the channel input that is resolved by observing the channel output. This is called as the Mutual Information of the channel.

Denoting the Mutual Information as *I*(*x*;*y*), we can write the whole thing in an equation, as follows

$$I(x; y) = H(x) - H(x \mid y)$$

Hence, this is the equational representation of Mutual Information.

Properties of Mutual information

These are the properties of Mutual information.

1. Mutual information of a channel is symmetric.

$$I(x; y) = I(y; x)$$

2. Mutual information is non-negative.

$$I(x; y) \geq 0$$

3. Mutual information can be expressed in terms of entropy of the channel output.

$$I(x; y) = H(y) - H(y \mid x)$$

Where $H(\, y \mid x\,)$is a conditional entropy.

4. Mutual information of a channel is related to the joint entropy of the channel input and the channel output.

$$I(x; y) = H(x) + H(y) - H(x, y)$$

Where the joint entropy $H(x, y)$is defined by

$$H(x, y) = \sum_{j=0}^{j-1} \sum_{k=0}^{k-1} p(x_j, y_k)\, \log_2[\frac{1}{p(x_j,\, y_k)}]$$

5.05 Conditional Entropy

Using the input probabilities $P(x_i)$, output probabilities $P(y_i)$, transition probabilities $P(x_i/ y_i)$ and joint probabilities $P(x_i, y_i)$, we can define the following various entropy functions for a channel with m input and n outputs,

Source Entropy
$$H(X) = -\sum_{i=1}^{m} P(x_i) \log_2 P(x_i)$$

Destination Entropy
$$H(Y) = -\sum_{j=1}^{n} P(y_j) \log_2 P(y_j)$$

The conditional entropy H(X/Y) is a measure of the average uncertainty about the channel input after the channel output has been observed.

$$H(X/Y) = -\sum_{j=1}^{n}\sum_{i=1}^{m} P(x_i, y_j) \log_2 P(x_i/y_j)$$

The conditional entropy H(Y/X) is the average uncertainty of the channel output given that X was transmitted.

$$H(Y/X) = -\sum_{j=1}^{n}\sum_{i=1}^{m} P(x_i, y_j) \log_2 P(y_j/x_i)$$

The joint entropy H(X,Y) is the average uncertainty of the communication channel as a whole

$$H(X,Y) = -\sum_{j=1}^{n}\sum_{i=1}^{m} P(x_i, y_j) \log_2 P(x_i, y_j)$$

5.06 Efficiency

The transmission efficiency or the channel efficiency is defined as

$$\eta = \frac{actual\ transinformation}{maximum\ transinformation}$$

$$\eta = \frac{I(X;Y)}{\max I(X;Y)} = \frac{I(X;Y)}{C_s}$$

Case I If I(X;Y) = max I(X;Y)
Then η = 100%
The channel is fully utilized.

Case II If I(X;Y) < max I(X;Y)
Then η < 100%

Case III If I(X;Y) > max I(X;Y)
Then η > 100%
The situation is avoided.
In the second case, to increase η, we code the input data using

1. Shannon Fano Coding
2. Huffman Coding Procedure

5.07 Discrete Memory less Source

A source from which the data is being emitted at successive intervals, which is independent of previous values, can be termed as discrete memory less source.

This source is discrete as it is not considered for a continuous time interval, but at discrete time intervals. This source is memory less as it is fresh at each instant of time, without considering the previous values.
The Code produced by a discrete memory less source, has to be efficiently represented, which is an important problem in communications. For this to happen, there are code words, which represent these source codes.

5.08 Source Coding

A conversion of the output of a DMS into a sequence of binary symbols is called source coding. The device that performs this conversion is called the source encoder.

Let us take a look at the block diagram.

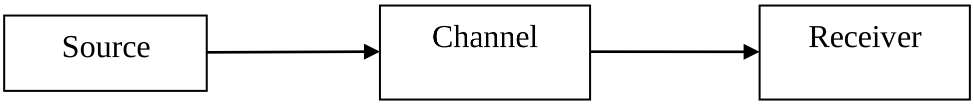


Figure 5.08.01 (a) Without Source Encoder

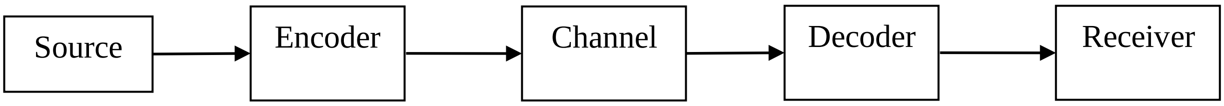


Figure 5.08.01 (b) With Encoder- Decoder

An objective of the source coding is to minimize the average bit rate required for representation of the source by reducing the redundancy (i.e. increasing the efficiency) of the information source.

Code Length

Let X be a DMS with finite entropy H(X) and an alphabet $\{x_1, x_2, \ldots, x_m\}$ with the corresponding probabilities of occurrence $P(x_i)$ $\{i=1,2,\ldots,m\}$.
Let the binary code word assigned to symbol x_i have n_i length, measured in bits. Then the average code word length L per source symbol is given by:

$$\boxed{L = \sum_{i=1}^{m} P(x_i)\, n_i}$$

L represent the average number of bits per source symbol.
The code efficiency η can be defined as

$$\boxed{\eta = \frac{L_{min}}{L}}$$

Where L_{min} is the minimum possible value of L.

Code redundancy γ

$$\boxed{\gamma = 1 - \eta}$$

5.09 Source Coding Theorem

It states that for a DMS-X, with entropy H(X), the average code word length L per symbol is bounded as

$$L \geq H(X)$$

And further L can be made as close to H(X) as desired for some suitable chosen code.
Thus with

$$L_{min} = H(X)$$

$$\boxed{\eta = \frac{H(X)}{L}}$$

Example 5.01 A source delivers six digits with the following probabilities:

i	A	B	C	D	E	F
P(i)	1/2	1/4	1/8	1/16	1/32	1/32

Find (1) H(X) (2) H'(X) = r.H(X) (3) H'(X)$_{max}$=C (4) Efficiency

Solution:

(1)

$$H(X) = -\sum_{i=1}^{m} P(x_i).\log_2 P(x_i) \qquad \text{Bits/ Symbol}$$

$$H(X) = -\sum_{i=1}^{6} P(i).\log_2 P(i)$$

$$= -[\frac{1}{2}log_2\frac{1}{2} + \frac{1}{4}log_2\frac{1}{4} + \frac{1}{8}log_2\frac{1}{8} + \frac{1}{16}log_2\frac{1}{16} + \frac{1}{32}log_2\frac{1}{32} + \frac{1}{32}log_2\frac{1}{32}]$$

$$= -[\frac{1}{2}\times(-1) + \frac{1}{4}\times(-2) + \frac{1}{8}\times(-3) + \frac{1}{16}\times(-4) + \frac{2}{32}\times(-5)]$$

$$= 1.938\ bits/symbol$$

(2)

$$H'(X) = r.H(X)$$

$$H(x) = -\sum_{i=1}^{6} P(i).\log_2 P(i)$$

$$H'(X) = 1.938\ bits/sec$$

(3)

$$H(X)|max = \log_2 m = \log_2 6 = \frac{\log_{10} 6}{\log_{10} 2} = 2.58\ bits/symbol$$

$$H'(X)|max = 2.58\ bits/symbol \qquad \text{r=1 symbol/sec}$$

(4)

$$\eta = \frac{H'(X)}{H'(X)|max}\times 100\% = \frac{1.938}{2.58}\times 100\%$$

$$\boxed{\eta = 75.11\%}$$

5.10 Shannon-Fano Coding

An efficient code can be obtained by the following simple procedure, known as Shannon-Fano coding:

1. First write the source symbols in order of decreasing probability,
2. Partition the set into two most equi-probable sub sets and assign a '0' to the upper set and '1' to the lower one,
3. Continue this procedure, each time partitioning the sets with as nearly as equal probabilities as possible until further partitioning is not possible.

Example 5.02 Apply the Shannon Fano coding procedure for the following message ensemble:

X	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
P	1/4	1/8	1/16	1/16	1/16	1/4	1/16	1/8

Take M=2. Find the Code Efficiency.

Solution:

As per the procedure explained in section 5.10, The code can be obtained as under,

Message	Prob.	Step 1	Step 2	Step 3	Step 4	Code	Code Length
x_1	0.25	0	0			00	2
x_6	0.25	0	1			01	2
x_2	0.125	1	0	0		100	3
x_8	0.125	1	0	1		101	3
x_3	0.0625	1	1	0	0	1100	4
x_4	0.0625	1	1	0	1	1101	4
x_5	0.0625	1	1	1	0	1110	4
x_7	0.0625	1	1	1	1	1111	4

$$H(X) = -\sum_{i=1}^{m} P(i).\log_2 P(i)$$

$$= -[\frac{1}{4}log_2\frac{1}{4} + \frac{1}{8}log_2\frac{1}{8} + \frac{1}{16}log_2\frac{1}{16} + \frac{1}{16}log_2\frac{1}{16} + \frac{1}{16}log_2\frac{1}{16} + \frac{1}{4}log_2\frac{1}{4} + \frac{1}{16}log_2\frac{1}{16} + \frac{1}{8}log_2\frac{1}{8}]$$

$$= -[\frac{1}{4}\times(-2) + \frac{1}{8}\times(-3) + \frac{1}{16}\times(-4)\times 3 + \frac{1}{4}\times(-2) + \frac{1}{16}\times(-4) + \frac{1}{8}\times(-3)]$$

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

$$= 2.75\ bits/symbol$$

Now the average code word length L per source symbol is

$$L = \sum_{i=1}^{m} P(x_i).n_i$$

$$= \frac{1}{4}\times 2 + \frac{1}{4}\times 2 + \frac{1}{8}\times 3 + \frac{1}{8}\times 3 + \frac{1}{16}\times 4\times 4$$

$$= 2.75\ bits/symbol$$

Then Efficiency

$$\eta = \frac{H(X)}{L} = \frac{2.75}{2.75}100\% = 100\%\ (\boldsymbol{Answer})$$

Example 5.03 A DMS has seven messages with probabilities

X	**x_1**	**x_2**	**x_3**	**x_4**	**x_5**	**x_6**	**x_7**
P	**0.4**	**0.2**	**0.12**	**0.08**	**0.08**	**0.08**	**0.04**

Apply the Shannon Fano coding procedure and calculate the efficiency of the code. Take M=2

Solution:

Message	Prob.	Step 1	Step 2	Step 3	Step 4	Code	Code Length
X_1	0.4	0	0			0	1
X_2	0.2	1	0	0		100	3
X_3	0.12	1	0	1		101	3
X_4	0.08	1	1	0	0	1100	4
X_5	0.08	1	1	0	1	1101	4
X_6	0.08	1	1	1	0	1110	4
X_7	0.04	1	1	1	1	1111	4

Then Entropy is

$$H(X) = -\sum_{i=1}^{m} P(x_i).\log_2 P(x_i)$$
$$= -[(0.4)log_2(0.4) + (0.2)log_2(0.2) + (0.12)log_2(0.12) + (0.08)log_2(0.08) + (0.08)log_2(0.08) + (0.08)log_2(0.08) + (0.04)log_2(0.04)]$$
$$= \quad 2.42\ letters/symbol$$

Now the average code word length L per source symbol is

$$L = \sum_{i=1}^{m} P(x_i).n_i$$
$$= 0.4 \times 1 + 0.2 \times 3 + 0.12 \times 3 + 0.08 \times 4 + 0.08 \times 4 + 0.08 \times 4 + 0.04 \times 4$$
$$= \quad 2.48\ bits/message$$

Then Efficiency

$$\eta = \frac{H(X)}{Lavg} \times 100\% = \frac{2.42}{2.48} 100\% = 97.58\%\ (\boldsymbol{Answer})$$

5.10 Huffman Encoding

Huffman Encoding results in a code that has the highest efficiency. The Huffman Coding procedure is as under:

1. List the source symbol in decreasing probability,
2. Combine the probabilities of the two symbols having the least probabilities and reorder the resultant probabilities. This step is called reduction. The same procedure is repeated until there are two ordered probabilities remaining,
3. Start reduction with the last reduction, which consists of exactly two ordered probabilities. Assign '0' to the first probabilities and a '1' to the second probability.
4. Now assign '0' and '1' for the probabilities that were combined in the previous reduction step, until the first reduction step.

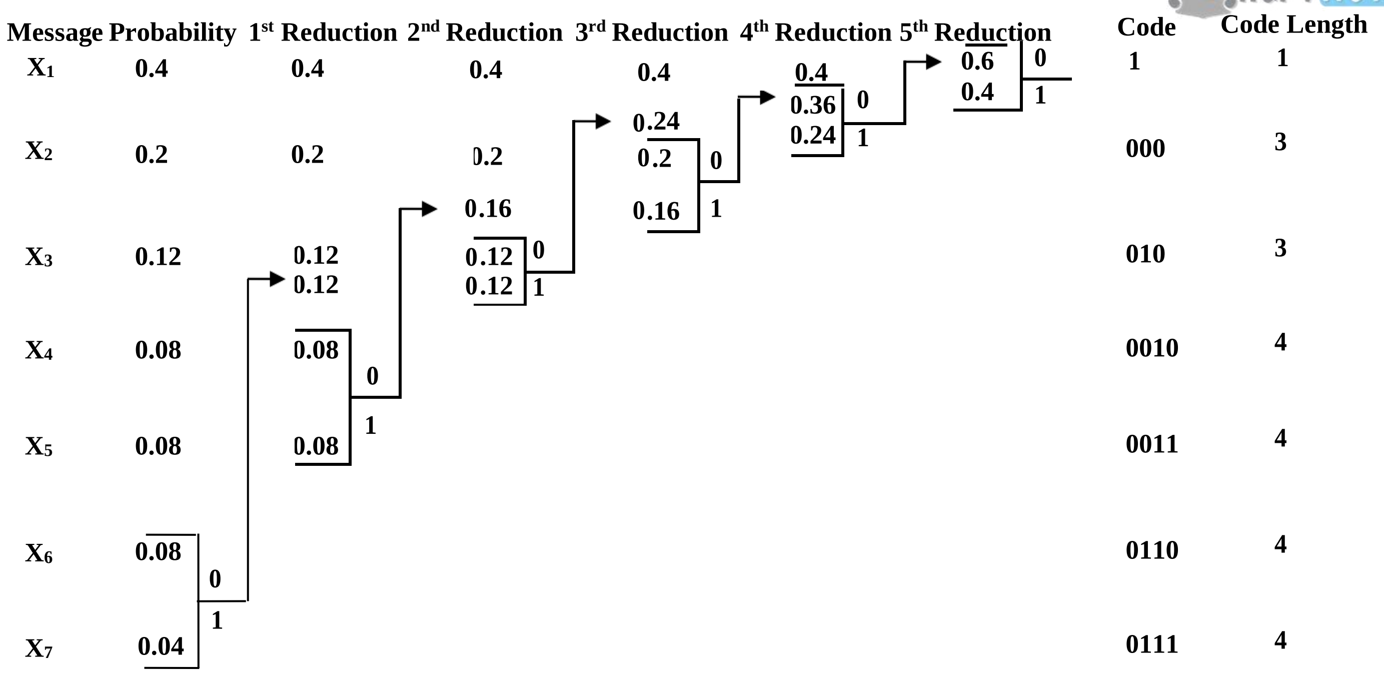
Example 5.04 A DMS has seven messages with probabilities

X	x_1	x_2	x_3	x_4	x_5	x_6	x_7
P	0.4	0.2	0.12	0.08	0.08	0.08	0.04

Apply Huffman coding procedure and calculate the efficiency of the code.

Solution:

Applying the Huffman's Coding procedure and determining the codes as under



Now, the Entropy

$$H(X) = -\sum_{i=1}^{m} P(x_i).\log_2 P(x_i)$$

$$= -[(0.4)log_2(0.4) + (0.2)log_2(0.2) + (0.12)log_2(0.12) + (0.08)log_2(0.08) + (0.08)log_2(0.08) + (0.08)log_2(0.08) + (0.04)log_2(0.04)]$$

$$= \quad 2.42\ letters/symbol$$

Now the average code word length L per source symbol is

$$L = \sum_{i=1}^{m} P(x_i).n_i$$

$$= 0.4 \times 1 + 0.2 \times 3 + 0.12 \times 3 + 0.08 \times 4 + 0.08 \times 4 + 0.08 \times 4 + 0.04 \times 4$$

$$= \quad 2.48\ bits/message$$

Then Efficiency

$$\eta = \frac{H(X)}{Lavg} \times 100\% = \frac{2.42}{2.48} 100\% = 97.58\%\ (\boldsymbol{Answer})$$

5.11 Shannon's Theorem

Given a source of equally likely messages, with M>>1, which is generating information at a rate R. Given a channel with channel capacity C. Then if R≤C, there exists a coding technique such that the output of the source may be transmitted over the channel with the probability of error of receiving the message signal very small.

Thus according to the theorem if R≤C, then the noise free transmission is possible in the presence of noise. The negative theorem states that if the information rate R exceeds C (R>C), error probability approaches to unity as M increases.

5.12 Channel Capacity

We have so far discussed mutual information. The maximum average mutual information, in an instant of a signaling interval, when transmitted by a discrete memory less channel, the probabilities of the rate of maximum reliable transmission of data, can be understood as the channel capacity.

It is denoted by C and is measured in bits per channel use.

5.13 Shannon Limit

Shannon-Hartley equation relates the maximum capacity (transmission bit rate) that can be achieved over a given channel with certain noise characteristics and bandwidth. For an AWGN the maximum capacity is given by

$$C = B\log_2\left(1+\frac{S}{N}\right)$$

$$C = B\log_2\left(1+\frac{S}{n\beta}\right)$$

Here C is the maximum capacity of the channel in bits/second otherwise called Shannon's capacity limit for the given channel, B is the bandwidth of the channel in Hertz, S is the signal power in Watts and N is the noise power, also in Watts. The ratio S/N is called Signal to Noise Ratio (SNR). It can be ascertained that the maximum rate at which we can transmit the information without any error, is limited by the bandwidth, the signal level, and the noise level. It tells how many bits can be transmitted per second without errors over a channel of bandwidth B Hz, when the signal power is limited to S Watt and is exposed to Gaussian White Noise of additive nature.

Example 5.05 Calculate the channel capacity of a channel with BW 3kHz & S/N ratio given as 10^3, assuming that the noise is white Gaussian noise.

Solution:

We know that

$$C = B\log_2\left(1+\frac{S}{N}\right)$$

$$= 3000\log_2(1+10^3)$$

$$= 30000\ bits/sec$$

5.14 Binary Symmetric Channel (BSC)

A symmetric channel is defined as the one for which

(i) $H(Y/x_j)$ is independent of j: the entropy corresponding to each row of P(Y/X) is the same. &

(ii) $\sum_{j=1}^{m} P(y_k/x_j)$ is independent of k i.e. the sum of all columns of P(Y/X) is the same.

For a symmetric channel

$$I(X{:}Y) = H(Y) - H(Y/X)$$

$$= H(Y) - A\sum_{j=1}^{m} H\left(\frac{Y}{x_j}\right)P(x_j)$$

$$= H(Y) - A\sum_{j=1}^{m} P(x_j)$$

Where $A=H(Y/x_j)$ is independent of j and hence taken out of the summation sign. Also

$$\sum_{j=1}^{m} P(x_j) = 1$$

therefore

$$I(X{:}Y) = H(Y) - A$$

Hence the channel capacity

$$C = \max I(x{:}y)$$

$$= \max[H(Y) - A]$$

$$= \max[H(Y)] - A$$
$$C = \log n - A$$

Where n is the total number of receiver symbols, and max[H(Y)] = logn.

Binary Symmetric Channel;

The most important case of a symmetric channel is BSC. In this case m=n=2, and the channel matrix is

$$D = \left[P\left(\frac{Y}{X}\right)\right] = \begin{bmatrix} P & 1-P \\ 1-P & P \end{bmatrix} = \begin{bmatrix} P & q \\ q & P \end{bmatrix}$$

BSC is shown graphically as shown

The channel is symmetric because the probability of receiving a 1, if 0 was transmitted is same as the probability of receiving a 0 if a 1 was transmitted. This common transition probability is given by P.

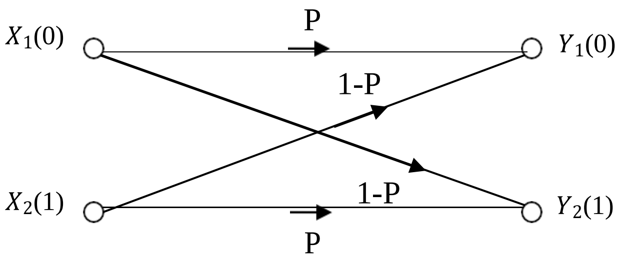
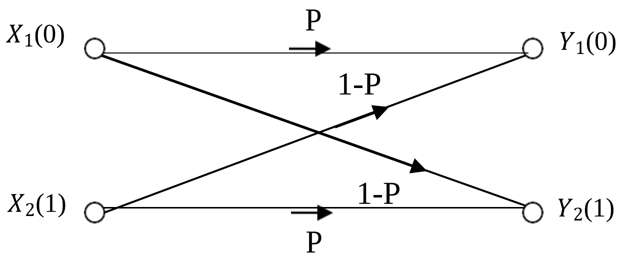


Figure 5.14.1 BSC

Example 5.06 For the Binary Symmetric Channel Calculate the channel capacity for (i) P=0.9 and (ii) P= 0.6



Solution:

We know that

$$C = \log n - A$$
$$C = \log 2 - H\left(\frac{Y}{x_j}\right)$$
$$C = \log_2 2 - \left[-\sum_{j=1}^{2} P\left(\frac{y_k}{x_j}\right)\log_2 P\left(\frac{y_k}{x_j}\right)\right]$$
$$C = \log 2 + P\log P + (1-P)\log(1-P)$$
$$C = \log 2 + (P\log P + q\log q)$$
$$C = 1 + (P\log P + q\log q)$$
$$C = 1 + H(P) = 1 - H(q)$$

(i) for p=0.9

$$C = 1 + (0.9\log 0.9 + 0.1\log 0.1)$$
$$C = 0.531 \text{ bits / message}$$

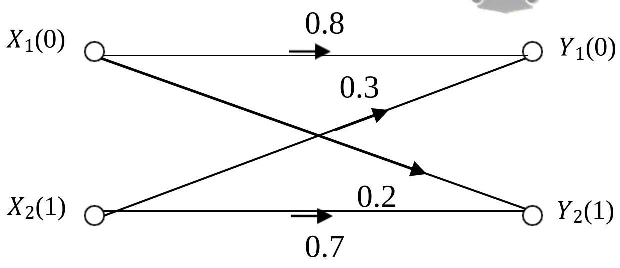
(ii) for p=0.6

$$C = 1 + (0.6\log 0.6 + 0.4\log 0.4)$$
$$C = 0.029 \text{ bits / message}$$

Example 5.07 Find the entropy of the source, information rate, average length and efficiency if the rate of message generation is 300message per seconds and if the symbols and the probabilities are as under

X	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
P	0.1	0.13	0.01	0.04	0.08	0.29	0.06	0.22	0.05	0.02

Example 5.08 For the Channel shown in figure, Calculate the channel capacity.



Solution:

The channel matrix is given by

$$[P(Y/X)] = \begin{bmatrix} 0.8 & 0.2 \\ 0.3 & 0.7 \end{bmatrix}$$

Since this is an un symmetric channel therefore the channel capacity of a binary channel is

$$C = \log(2^{Q_1} + 2^{Q_2})$$

Where Q_1 and Q_2 are defined by [P].[Q]=[H], therefore

$$\begin{bmatrix} 0.8 & 0.2 \\ 0.3 & 0.7 \end{bmatrix} \begin{bmatrix} Q_1 \\ Q_2 \end{bmatrix} = \begin{bmatrix} P_{11}\log P_{11} + P_{12}\log P_{12} \\ P_{21}\log P_{21} + P_{22}\log P_{22} \end{bmatrix}$$

$$= \begin{bmatrix} 0.8\log 0.8 + 0.2\log 0.2 \\ 0.3\log 0.3 + 0.7\log 0.7 \end{bmatrix}$$

$$= \begin{bmatrix} -0.2576 - 0.4644 \\ -0.5211 - 0.3602 \end{bmatrix} = \begin{bmatrix} -0.722 \\ -0.8813 \end{bmatrix}$$

Therefore $0.8Q_1 + 0.2Q_2 = -0.722$

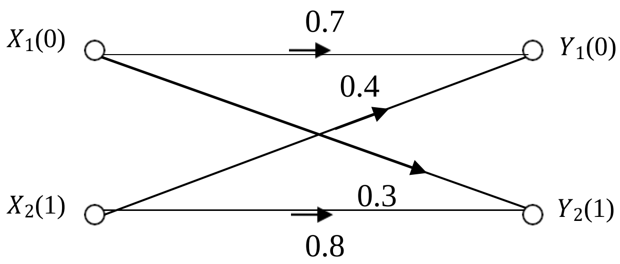
and $0.3Q_1 + 0.7Q_2 = -0.8813$

After solving $Q_1 = -0.6568$ and $Q_2 = -0.9764$

Then $C = \log(2^{Q_1} + 2^{Q_2}) = \log(2^{-0.6558} + 2^{-0.9764})$

$$C = \log(0.6343 + 0.5082) = \log_2(1.14255) = 0.192\ bit/message\ \boldsymbol{Answer}$$

Example 5.09 For the Channel shown in figure, Calculate the channel capacity.



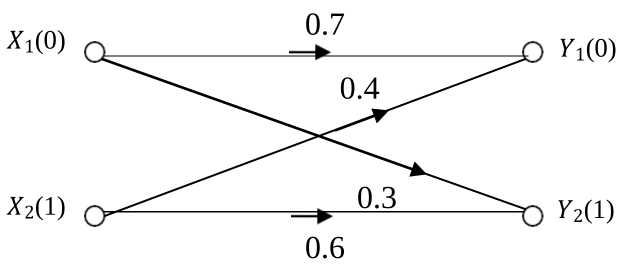
Solution

After solving $Q_1 = -1.061$ and $Q_2 = -0.456$

Then $C = \log(2^{-1.061} + 2^{-0.456})$

$$C = \log(0.4793 + 0.729) = 0.273\ bit/symbol\ \boldsymbol{Answer}$$

Example 5.10 For the Channel shown in figure, Calculate the channel capacity.



Solution

After solving $Q_1 = -0.79$ and $Q_2 = -1.09$

$$C = 0.067\ bit/symbol\ \boldsymbol{Answer}$$