



**srivenkateshwaraa**  
**College of Engineering & Technology**

(Approved by AICTE, New Delhi & Affiliated to Pondicherry University, Puducherry)  
13-A, Pandy - Villupuram Main Road, Ariyur, Puducherry - 605 102.

ASPIRE TO EXCEL



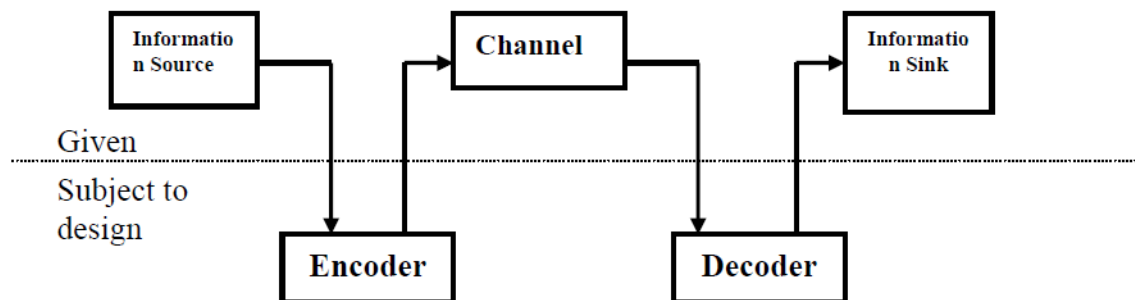
**DEPARTMENT OF ELECTRONICS AND**  
**COMMUNICATION ENGINEERING**

**EC T61 DIGITAL COMMUNICATION**  
**NOTES**

**III YEAR/ VI SEM**

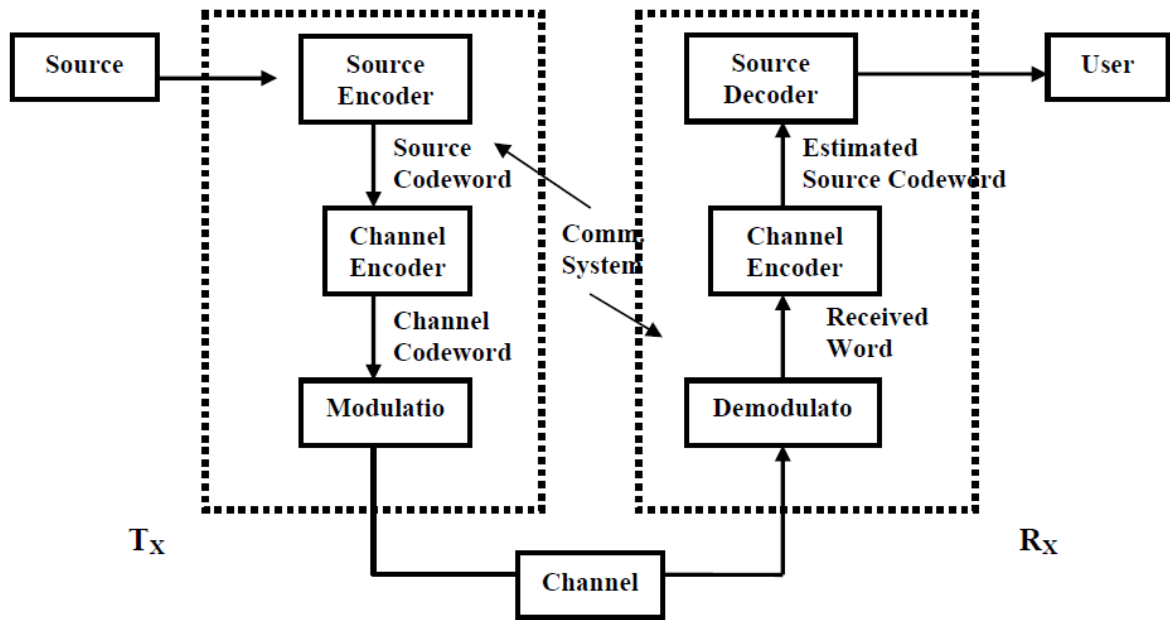
## Block Schematic Description of a Digital Communication System

- Digital communication system is a three-block system, consisting of a) a transmitter, b) a transmission medium and c) a receiver.
- A digital communication system has several distinguishing features when compared with an analog communication system. Both analog (such as voice signal) and digital signals (such as data generated by computers) can be communicated over a digital transmission system.
- When the signal is analog in nature, an equivalent discrete-time-discrete-amplitude representation is possible after the initial processing of sampling and quantization. So, both a digital signal and a quantized analog signal are of similar type, i.e. discrete-time-discrete-amplitude signals.
- A key feature of a digital communication system is that a sense of „information“, with appropriate unit of measure, is associated with such signals. This visualization, credited to Claude E. Shannon, leads to several interesting schematic description of a digital communication system.



**Fig. 1.1.1** *Basic block diagram of a digital communication System*

- It is possible to expand our basic „three-entity“ description of a digital communication system in multiple ways.
- Figure shows a somewhat elaborate block diagram explicitly showing the important processes of „modulation-demodulation“, „source coding-decoding“ and „channel encoding – decoding“. A reader may have multiple queries relating to this kind of abstraction.
- When „information“ has to be sent over a large distance, it is a common knowledge that the signal should be amplified in terms of power and then launched into the physical transmission medium. Diagrams of the type in **Figs. 1.1.1** and **1.1.2** have no explicit reference to such issues.



- Several types of channel can be defined. The „channel“ should more appropriately be called as a „modulation channel“ with an understanding that the actual transmission medium (called „physical channel“), any electromagnetic (or otherwise) transmission- reception operations, amplifiers at the transmission and reception ends and any other necessary signal processing units are combined together to form this „modulation channel“.
- A modulation channel usually accepts modulated signals as analog waveforms at its inputs and delivers another version of the modulated signal in the form of analog waveforms. Such channels are also referred as „waveform channels“.

## Pulse Code Modulation

A schematic diagram for Pulse Code Modulation is shown in **Fig 3.11.1**. The analog voice input is assumed to have zero mean and suitable variance such that the signal samples at the input of A/D converter lie satisfactorily within the permitted single range. As discussed earlier, the signal is band limited to 3.4 KHz by the low pass filter.

Let  $x(t)$  denote the filtered telephone-grade speech signal to be coded. The process of analog to digital conversion primarily involves three operations: (a) Sampling of  $x(t)$ , (b) Quantization (i.e. approximation) of the discrete time samples,  $x(kT_s)$  and (c) Suitable encoding of the quantized time samples  $x_q(kT_s)$ .  $T_s$  indicates the sampling interval where  $R_s = 1/T_s$  is the sampling rate (samples/sec). A standard sampling rate for speech signal, band limited to 3.4 kHz, is 8 Kilo-samples per second ( $T_s = 125\mu$  sec), thus, obeying Nyquist's sampling theorem. We assume instantaneous sampling for our discussion. The encoder in **Fig 3.11.1** generates a group of bits representing one quantized sample. A parallel-to-serial (P/S) converter is optionally used if a serial bit stream is desired at the output of the PCM coder. The PCM coded bit stream may be taken for further digital signal processing and modulation for the purpose of transmission.

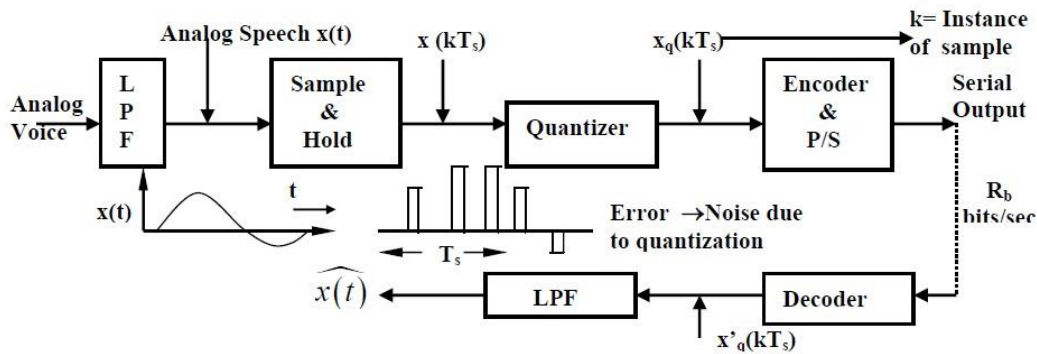


Fig. 3.11.1 Schematic diagram of a PCM coder – decoder

- The PCM decoder at the receiver expects a serial or parallel bit-stream at its input so that it can decode the respective groups of bits (as per the encoding operation) to generate quantized sample sequence  $[x'_q(kT_s)]$ . Following Nyquist's sampling theorem for band limited signals, the low pass filter produces a close replica  $\hat{x}(t)$  of the original speech signal  $x(t)$ .
- If we consider the process of sampling to be ideal (i.e. instantaneous sampling) and if we assume that the same bit-stream as generated by PCM encoder is available at PCM decoder, we should still expect  $\hat{x}(t)$  to be somewhat different from  $x(t)$ . This is solely because of the process of quantization. As indicated, quantization is an approximation process and thus, causes some distortion in the reconstructed analog signal. We say that quantization contributes to "noise".
- The issue of quantization noise, its characterization and techniques for restricting it within an acceptable level are of importance in the design of high quality signal coding and transmission system. We focus a bit more on a performance metric called SQNR (Signal to Quantization Noise power Ratio) for a PCM codec. For simplicity, we consider uniform quantization process. The input-output characteristic for a uniform quantizer is shown in **Fig 3.11.2(a)**.
- The input signal range ( $\pm V$ ) of the quantizer has been divided in eight equal intervals. The width of each interval,  $\delta$ , is known as the step size. While the amplitude of a time sample  $x(kT_s)$  may be any real number between  $+V$  and  $-V$ , the quantizer presents only one of the allowed eight values ( $\pm\delta, \pm3\delta/2, \dots$ ) depending on the proximity of  $x(kT_s)$  to these levels.

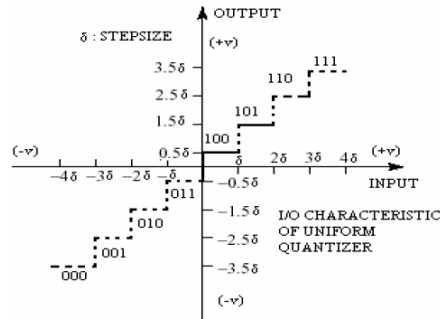


Fig 3.11.2(a) Linear or uniform quantizer

- The quantizer of **Fig 3.11.2(a)** is known as “mid-riser” type. For such a mid-riser quantizer, a slightly positive and a slightly negative values of the input signal will have different levels at output.
- This may be a problem when the speech signal is not present but small noise is present at the input of the quantizer.
- To avoid such a random fluctuation at the output of the quantizer, the “mid-tread” type uniform quantizer [**Fig 3.11.2(b)**] may be used.

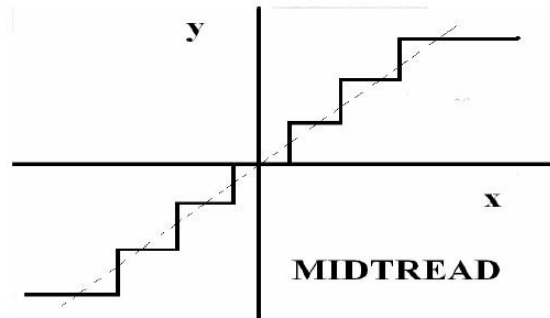


Fig 3.11.2(b) Mid-tread type uniform quantizer characteristics

## SQNR for uniform quantizer

In Fig.3.11.1  $x(kT_s)$  represents a discrete time ( $t = kT_s$ ) continuous amplitude sample of  $x(t)$  and  $x_q(kT_s)$  represents the corresponding quantized discrete amplitude value. Let  $e_k$  represents the error in quantization of the  $k^{\text{th}}$  sample i.e.

$$e_k = x_q(kT_s) - x(kT_s) \quad 3.11.1$$

Let,

$M$  = Number of permissible levels at the quantizer output.

$N$  = Number of bits used to represent each sample.

$\pm V$  = Permissible range of the input signal  $x(t)$ .

Hence,

$$M = 2^N \quad \text{and,}$$

$$M \cdot \delta \cong 2 \cdot V \quad [\text{Considering large } M \text{ and a mid-riser type quantizer}]$$

Let us consider a small amplitude interval  $dx$  such that the probability density function (pdf) of  $x(t)$  within this interval is  $p(x)$ . So,  $p(x)dx$  is the probability that  $x(t)$  lies in the range  $(x - \frac{dx}{2})$  and  $(x + \frac{dx}{2})$ . Now, an expression for the mean square quantization error  $\overline{e^2}$  can be written as:

$$\overline{e^2} = \int_{x_1 - \delta/2}^{x_1 + \delta/2} p(x)(x - x_1)^2 dx + \int_{x_2 - \delta/2}^{x_2 + \delta/2} p(x)(x - x_2)^2 dx + \dots \quad 3.11.2$$

For large  $M$  and small  $\delta$  we may fairly assume that  $p(x)$  is constant within an interval, i.e.  $p(x) = p_1$  in the 1<sup>st</sup> interval,  $p(x) = p_2$  in the 2<sup>nd</sup> interval, ...,  $p(x) = p_k$  in the  $k^{\text{th}}$  interval.

Therefore, the previous equation can be written as

$$\overline{e^2} = (p_1 + p_2 + \dots) \int_{-\delta/2}^{\delta/2} y^2 dy$$

Where,  $y = x - x_k$  for all 'k'.

So,

$$\begin{aligned} \overline{e^2} &= (p_1 + p_2 + \dots) \frac{\delta^3}{12} \\ &= [(p_1 + p_2 + \dots)\delta] \frac{\delta^2}{12} \end{aligned}$$

Now, note that  $(p_1 + p_2 + \dots + p_k + \dots)\delta = 1.0$

$$\therefore \overline{e^2} = \frac{\delta^2}{12}$$

The above mean square error represents power associated with the random error signal. For convenience, we will also indicate it as  $N_Q$ .

## Calculation of Signal Power ( $S_i$ )

After getting an estimate of quantization noise power as above, we now have to find the signal power. In general, the signal power can be assessed if the signal statistics (such as the amplitude distribution probability) is known. The power associated with  $x(t)$  can be expressed as

$$S_i = \overline{x^2(t)} = \int_{-V}^{+V} x^2(t)p(x) dx$$

where  $p(x)$  is the pdf of  $x(t)$ . In absence of any specific amplitude distribution it is common to assume that the amplitude of signal  $x(t)$  is uniformly distributed between  $\pm V$ .

In this case, it is easy to see that

$$S_i = \overline{x^2(t)} = \int_{-V}^{+V} x^2(t) \frac{1}{2V} dx = \left[ \frac{x^3}{3.2V} \right]_{-V}^{+V} = \frac{V^2}{3} = \frac{(M\delta)^2}{12}$$

Now the SNR can be expressed as,

$$\frac{S_i}{N_Q} = \frac{\frac{V^2}{3}}{\frac{\delta^2}{12}} = \frac{(M\delta)^2}{\frac{\delta^2}{12}} = M^2$$

It may be noted from the above expression that this ratio can be increased by increasing the number of quantizer levels  $N$ .

Also note that  $S_i$  is the power of  $x(t)$  at input of the sampler and hence, may not represent the SQNR at the output of the low pass filter in PCM decoder. However, for large  $N$ , small  $\delta$  and ideal and smooth filtering (e.g. Nyquist filtering) at the PCM

decoder, the power  $S_o$  of desired signal at the output of the PCM decoder can be assumed to be almost the same as  $S_i$  i.e.,

$$S_o = S_i$$

With this justification the SQNR at the output of a PCM codec, can be expressed as,

$$SQNR = \frac{S_o}{N_q} = M^2 = (2^N)^2 = 4^N$$

and in dB,

$$\left. \frac{S_o}{N_q} \right|_{dB} = 10 \log_{10} \left( \frac{S_o}{N_q} \right) = 6.02NdB$$

## Logarithmic Pulse Code Modulation (Log PCM) and Companding

In a linear or uniform quantizer, as discussed earlier, the quantization error in the  $k$ -th sample is

$$e_k = x(t) - x_q(kT_s) \quad 3.12.1$$

and the maximum error magnitude in a quantized sample is,

$$\text{Max} |e_k| = \frac{\delta}{2} \quad 3.12.2$$

- If  $x(t)$  itself is small in amplitude and such small amplitudes are more probable in the input signal than amplitudes closer to „ $\pm V$ “, it may be guessed that the quantization noise of such an input signal will be significant compared to the power of  $x(t)$ .
- This implies that SQNR of usually low signal will be poor and unacceptable. In a practical PCM codec, it is often desired to design the quantizer such that the SQNR is almost independent of the amplitude distribution of the analog input signal  $x(t)$ .
- This is achieved by using a non-uniform quantizer. A non-uniform quantizer ensures smaller quantization error for small amplitude of the input signal and relatively larger step size when the input signal amplitude is large. The transfer characteristic of a non – uniform quantizer has been shown in **Fig 3.12.1**.
- A non-uniform quantizer can be considered to be equivalent to an amplitude pre-distortion process [denoted by  $y = c(x)$  in **Fig 3.12.2**] followed by a uniform quantizer with a fixed step size „ $\delta$ “. We now briefly discuss about the characteristics of this pre-distortion or „compression“ function  $y = c(x)$ .



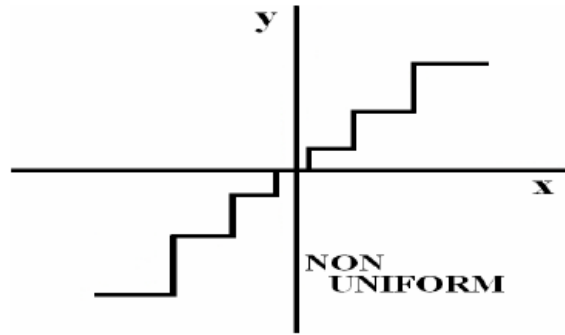


Fig 3.12.1 Transfer characteristic of a non-uniform quantizer

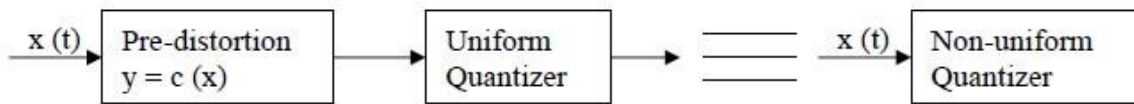


Fig. 3.12.2 An equivalent form of a non-uniform quantizer

Mathematically,  $c(x)$  should be a monotonically increasing function of 'x' with odd symmetry Fig 3.12.3. The monotonic property ensures that  $c^{-1}(x)$  exists over the range of 'x(t)' and is unique with respect to  $c(x)$  i.e.,  $c(x) \times c^{-1}(x) = 1$ .

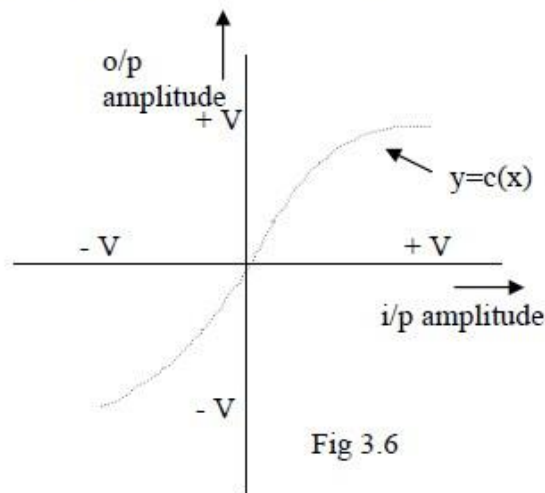


Fig 3.6

Fig. 3.12.3 A desired transfer characteristic for non-linear quantization process

Remember that the operation of  $c^{-1}(x)$  is necessary in the PCM decoder to get back the original signal undistorted. The property of odd symmetry i.e.,  $c(-x) = -c(x)$  simply takes care of the full range  $\pm V$  of  $x(t)$ . The range  $\pm V$  of  $x(t)$  further implies the following:

$$\begin{aligned} c(x) &= +V, & \text{for } x &= +V; \\ &= 0, & \text{for } x &= 0; \\ &= -V, & \text{for } x &= -V; \end{aligned} \quad 3.12.3$$

Let the  $k$ -th step size of the equivalent non-linear quantizer be  $\delta_k$  and the number of signal intervals be  $M$ . Further let the  $k$ -th representation level after quantization when the input signal lies between  $x_k$  and  $x_{k+1}$  be  $y_k$  where

$$y_k = \frac{1}{2}(x_k + x_{k+1}), \quad k = 0, 1, \dots, (M-1) \quad 3.12.4$$

The corresponding quantization error  $e_k$  is

$$e_k = x - y_k; \quad x_k < x \leq x_{k+1}$$

Now observe from Fig 3.12.3 that  $\delta_k$  should be small if  $\frac{dc(x)}{dx}$ , i.e., the slope of  $y = c(x)$  is large.

In view of this, let us make the following simple approximation on  $c(x)$ :

$$\frac{dc(x)}{dx} \approx \frac{2V}{M} \frac{1}{\delta_k}, \quad k = 0, 1, \dots, (M-1) \quad 3.12.5$$

and  $\delta_k = x_{k+1} - x_k$ ,  $k = 0, 1, \dots, (M-1)$

Note that,  $\frac{2V}{M}$  is the fixed step size of the uniform quantizer Fig. 3.12.2.

Let us now assume that the input signal is zero mean and its pdf  $p(x)$  is symmetric about zero. Further for large number of intervals we may assume that in each interval  $I_k$ ,  $k = 0, 1, \dots, (M-1)$ , the  $p(x)$  is constant. So if the input signal  $x(t)$  is between  $x_k$  and  $x_{k+1}$ , i.e.,  $x_k < x \leq x_{k+1}$ ,

$$p(x) = p(y_k)$$

So, the probability that  $x$  lies in the  $k$ -th interval  $I_k$ ,

$$I_k = p_k \triangleq P_r(x_k < x \leq x_{k+1}) = p(y_k) \delta_k \quad 3.12.6$$

where,  $\sum_0^{M-1} P_r(x_k < x \leq x_{k+1}) = 1$

Now, the mean square quantization error  $\overline{e^2}$  can be determined as follows:

$$\begin{aligned} \overline{e^2} &= \int_{-V}^{+V} (x - y_k)^2 p(x) dx \\ &= \sum_{k=0}^{M-1} \int_{x_k}^{x_{k+1}} (x - y_k)^2 p(y_k) dx \\ &= \sum_{k=0}^{M-1} \frac{p_k}{\delta_k} \int_{x_k}^{x_{k+1}} (x - y_k)^2 dx \\ &= \sum_{k=0}^{M-1} \frac{p_k}{\delta_k} \frac{1}{3} \left[ (x_{k+1} - y_k)^3 - (x_k - y_k)^3 \right] \\ &= \sum_{k=0}^{M-1} \frac{1}{3} \left( \frac{p_k}{\delta_k} \right) \left\{ \left[ x_{k+1} - \frac{1}{2}(x_k + x_{k+1}) \right]^3 - \left[ x_k - \frac{1}{2}(x_k + x_{k+1}) \right]^3 \right\} \end{aligned}$$

$$= \frac{1}{3} \sum_{k=0}^{M-1} \frac{P_k}{\delta_k} \frac{1}{4} \delta_k^3 = \frac{1}{12} \sum_{k=0}^{M-1} P_k \delta_k^2 \quad 3.12.7$$

Now substituting

$$\delta_k = \frac{2V}{M} \left[ \frac{dc(x)}{dx} \right]^{-1}$$

in the above expression, we get an approximate expression for mean square error as

$$\overline{e^2} = \frac{V^2}{3M^2} \sum_{k=0}^{M-1} P_k \left[ \frac{dc(x)}{dx} \right]^{-2} \quad 3.12.8$$

The above expression implies that the mean square error due to non-uniform quantization can be expressed in terms of the continuous variable  $x$ ,  $-V < x < +V$ , and having a pdf  $p(x)$  as below:

$$\overline{e^2} = \frac{V^2}{3M^2} \int_{-V}^{+V} p(x) \left[ \frac{dc(x)}{dx} \right]^{-2} dx \quad 3.12.9$$

Now, we can have an expression of SQNR for a non-uniform quantizer as:

$$SQNR = \left( \frac{3M^2}{V^2} \right) \frac{\int_{-V}^{+V} x^2 p(x) dx}{\int_{-V}^{+V} p(x) \left[ \frac{dc(x)}{dx} \right]^{-2} dx} \quad 3.12.10$$

The above expression is important as it gives a clue to the desired form of the compression function  $y = c(x)$  such that the SQNR can be made largely independent of the pdf of  $x(t)$ .

It is easy to see that a desired condition is:

$$\frac{dc(x)}{dx} = \frac{K}{x} \quad \text{where } -V < x < +V \text{ and } K \text{ is a positive constant.}$$

$$\text{i.e.,} \quad c(x) = V + K \ln \left( \frac{x}{V} \right) \quad \text{for } x > 0 \quad 3.12.11$$

$$\text{and} \quad c(x) = -c(x) \quad 3.12.12$$

Let us observe that  $c(x) \rightarrow \pm \infty$  as  $x \rightarrow 0$  from other side. Hence the above  $c(x)$  is not realizable in practice. Further, as stated earlier, the compression function  $c(x)$  must pass through the origin, i.e.,  $c(x) = 0$ , for  $x = 0$ . This requirement is forced in a compression function in practical systems.

There are two popular standards for non-linear quantization known as

- (a) The  $\mu$  - law companding
- (b) The A - law companding.

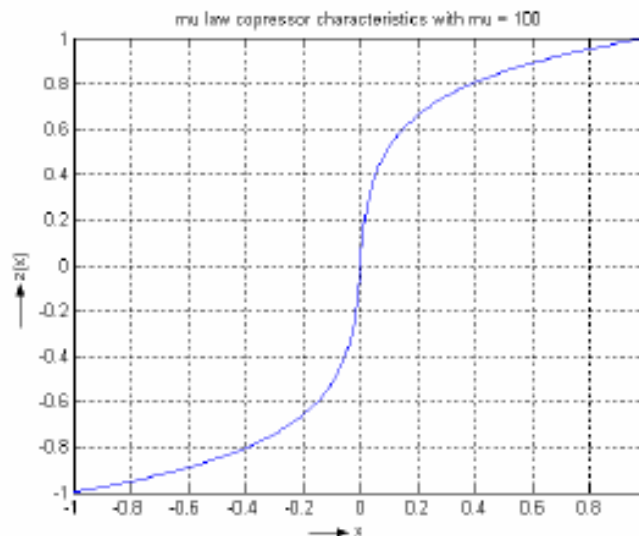
The  $\mu$  - law has been popular in the US, Japan, Canada and a few other countries while the A - law is largely followed in Europe and most other countries, including India, adopting ITU-T standards.

The  $\mu$  - law has been popular in the US, Japan, Canada and a few other countries while the A - law is largely followed in Europe and most other countries, including India, adopting ITU-T standards.

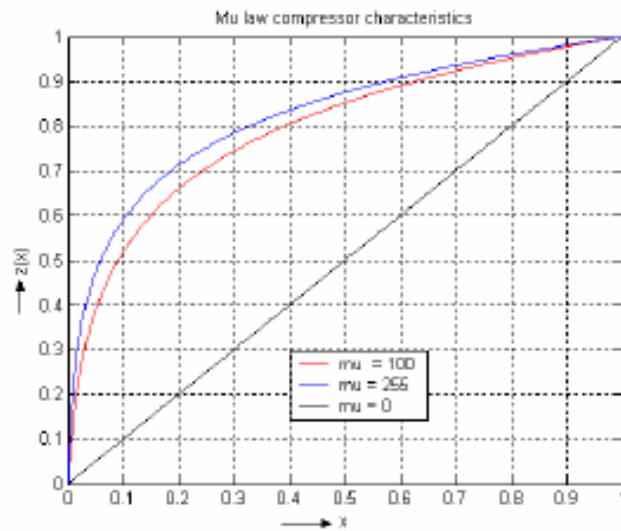
The compression function  $c(x)$  for  $\mu$  - law companding is (Fig. 3.12.4 and Fig. 3.12.5):

$$\frac{c(|x|)}{V} = \frac{\ln\left(1 + \frac{\mu|x|}{V}\right)}{\ln(1 + \mu)}, \quad 0 \leq \frac{|x|}{V} \leq 1.0 \quad 3.12.13$$

' $\mu$ ' is a constant here. The typical value of  $\mu$  lies between 0 and 255.  $\mu = 0$  corresponds to linear quantization.



**Fig. 3.12.4**  $\mu$ -law companding characteristics( $\mu = 100$ )



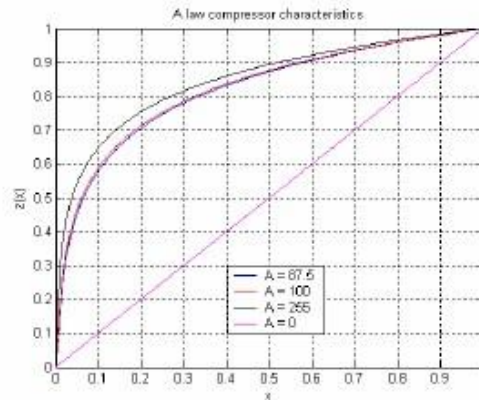
**Fig. 3.12.5**  $\mu$ -law companding characteristics ( $\mu = 0, 100, 255$ )

The compression function  $c(x)$  for A - law companding is (Fig. 3.12.6):

$$\frac{c(|x|)}{V} = \frac{A \frac{|x|}{V}}{1 + \ln A}, \quad 0 \leq \frac{|x|}{V} \leq \frac{1}{A}$$

$$= \frac{1 + \ln \left( A \frac{|x|}{V} \right)}{1 + \ln A}, \quad \frac{1}{A} \leq \frac{|x|}{V} \leq 1.0 \quad 3.12.14$$

'A' is a constant here and the typical value used in practical systems is 87.5.



**Fig. 3.12.6** *A-law* companding characteristics ( $A = 0, 87.5, 100, 255$ )

For telephone grade speech signal with 8-bits per sample and 8-Kilo samples per second, a typical SQNR of 38.4 dB is achieved in practice.

As approximately logarithmic compression function is used for linear quantization, a PCM scheme with non-uniform quantization scheme is also referred as “Log PCM” or “Logarithmic PCM” scheme.

## Differential Pulse Code Modulation (DPCM)

- The standard sampling rate for pulse code modulation (PCM) of telephone grade speech signal is  $f_s = 8$  Kilo samples per sec with a sampling interval of  $125 \mu$  sec. Samples of this band limited speech signal are usually correlated as amplitude of speech signal does not change much within  $125 \mu$  sec. A typical auto correlation function  $R(\tau)$  for speech samples at the rate 8 Kilo samples per sec is shown in **Fig 3.13.1**.  $R(\tau = 125 \mu$  sec) is usually between 0.79 and 0.87.
- This aspect of speech signal is exploited in differential pulse code modulation (DPCM) technique. A schematic diagram for the basic DPCM modulator is shown in **Fig 3.13.2**. Note that a predictor block, a summing unit and a subtraction unit have been strategically added to the chain of blocks of PCM coder instead of feeding the sampler output  $x(kT_s)$  directly to a linear quantizer. An error sample  $e_p(kT_s)$  is fed.

The error sample is given by the following expression:

$$e_p(kT_s) = x(kT_s) - \hat{x}(kT_s) \quad 3.13.1$$

$\hat{x}(kT_s)$  is a predicted value for  $x(kT_s)$  and is supposed to be close to  $x(kT_s)$  such that  $e_p(kT_s)$  is very small in magnitude.  $e_p(kT_s)$  is called as the ‘prediction error for the n-th sample’.

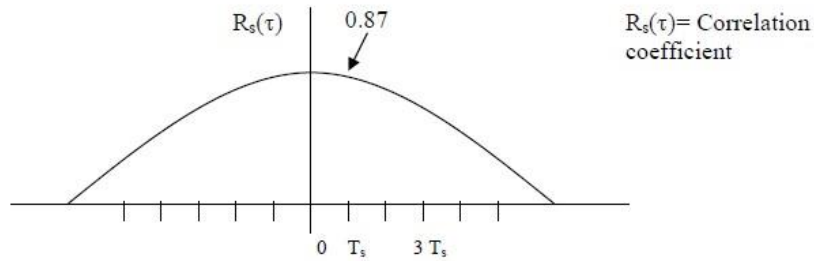


Fig. 3.13.1 Typical normalized auto-correlation coefficient for speech signal

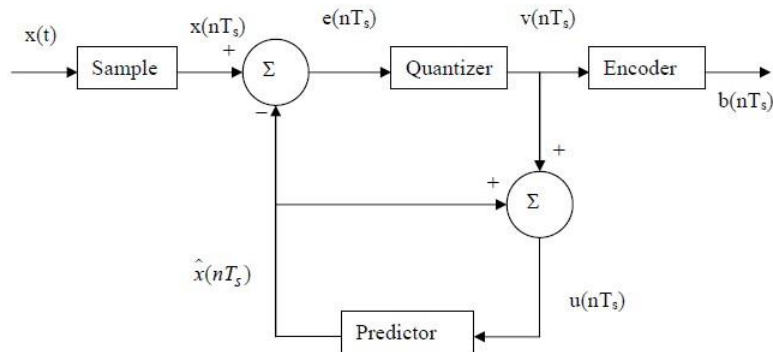


Fig. 3.13.2 Schematic diagram of a DPCM modulator

- If we assume a final enclosed bit rate of 64kbps as of a PCM coder, we envisage smaller step size for the linear quantizer compared to the step size of an equivalent PCM quantizer. As a result, it should be possible to achieve higher SQNR for DPCM codec delivering bits at the same rate as that of a PCM codec.
- There is another possibility of decreasing the coded bit rate compared to a PCM system if an SQNR as achievable by a PCM codec with linear equalizer is sufficient. If the predictor output  $\hat{x}(kT)$  can be ensured sufficiently close to  $x(kT)$  then we can simply encode the quantizer output sample  $v(kT_s)$  in less than 8 bits. For example, if we choose to encode each of  $v(kT_s)$  by 6 bits, we achieve a serial bit rate of 48 kbps, which is considerably less than 64 Kbps. This is an important feature of DPCM, especially when the coded speech signal will be transmitted through wireless propagation channels.



$e_p(kT_s)$  = k-th input to quantizer =  $x(kT_s) - \hat{x}(kT_s)$

$\hat{x}(kT_s)$  = prediction of the k-th input sample  $x(kT_s)$ .

$e_q(kT_s)$  = quantizer output for k-th prediction error.

=  $c[e_p(kT_s)]$ , where  $c[]$  indicates the transfer characteristic of the quantizer

If  $q(kT_s)$  indicates the quantization error for the k-th sample, it is easy to see that

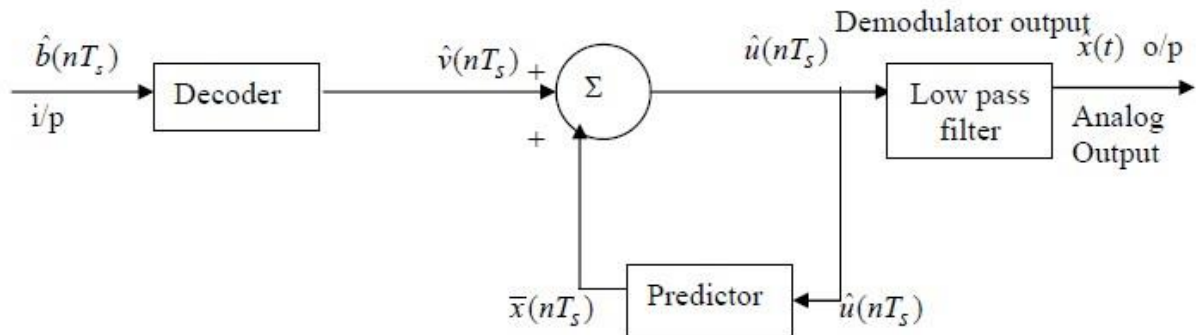
$$e_q(kT_s) = e_p(kT_s) + q(kT_s) \quad 3.13.2$$

Further the input  $u(kT_s)$  to the predictor is,

$$\begin{aligned} u(kT_s) &= \hat{x}(kT_s) + e_q(kT_s) = \hat{x}(kT_s) + e_p(kT_s) + q(kT_s) \\ &= x(kT_s) + q(kT_s) \end{aligned} \quad 3.13.3$$

This equation shows that  $u(kT_s)$  is indeed a quantized version of  $x(kT_s)$ . For a good prediction  $e_p(kT_s)$  will usually be small compared to  $x(kT_s)$  and  $q(kT_s)$  in turn will be very small compared to. Hence, the predictor unit should be so designed that variance of  $q(kT_s) < \text{variance of } e_p(kT_s) \ll \text{variance of } x(kT_s)$ .

A block schematic diagram of a DPCM demodulator is shown in **Fig 3.13.3**. The scheme is straightforward and it tries to estimate  $u(kT_s)$  using a predictor unit identical to the one used in the modulator. We have already observed that  $u(kT_s)$  is very close to  $x(kT_s)$  within a small quantization error of  $q(kT_s)$ . The analog speech signal is obtained by passing the samples  $\hat{u}(kT_s)$  through an appropriate low pass filter. This low pass filter should have a 3 dB cut off frequency at 3.4kHz.



**Fig. 3.13.3** Schematic diagram of a DPCM demodulator; note that the demodulator is very similar to a portion of the modulator

## Calculation of SQNR for DPCM

The expression for signal to quantization noise power ratio for DPCM coding is:

$$\begin{aligned}
 SQNR &= \frac{\text{Variance of } x(kT_s)}{\text{Variance of } q(kT_s)} \\
 &= \left[ \frac{\text{Variance of } x(kT_s)}{\text{Variance of } e_p(kT_s)} \right] \cdot \left[ \frac{\text{Variance of } e_p(kT_s)}{\text{Variance of } q(kT_s)} \right]
 \end{aligned}$$

- As in PCM coding, we are assuming instantaneous sampling and ideal low pass filtering. The first term in the above expression is the „predictor gain ( $G_p$ )“.
- This gain visibly increases for better prediction, i.e., smaller variance of  $e_p(kT_s)$ .
- The second term,  $SNR_p$  is a property of the quantizer. Usually, a linear or uniform quantizer is used for simplicity in a DPCM codec.
- Good and efficient design of the predictor plays a crucial role in enhancing quality of signal or effectively reducing the necessary bit rate for transmission.

### Single-Tap Prediction

A single-tap predictor predicts the next input example  $x(kT_s)$  from the immediate previous input sample  $x([k-1]T_s)$ .

Let,  $\hat{x}(kT_s) = \hat{x}(k|k-1)$  = the k-th predicted sample, given the (k-1)th input sample  
 $= a.u(k-1|k-1)$

Here, 'a' is known as the prediction co-efficient and  $u(k-1|k-1)$  is the (k-1)-th input to the predictor given the (k-1)-th input speech sample, i.e.,  $x(k-1)$ .

Now the k-th prediction error sample at the input of the quantizer may be expressed as

$$\begin{aligned} e_p(kT_s) &\equiv e_p(k) \\ &= x(k) - \hat{x}(k|k-1) \\ &= x(k) - a.u(k-1|k-1) \end{aligned} \quad 3.13.4$$

The mean square error or variance of this prediction error samples is the statistical expectation of  $e_p^2(k)$ .

Now,

$$\begin{aligned} E[e_p^2(k)] &= E[\{x(k) - a.u(k-1|k-1)\}^2] \\ &= E[x(k).x(k) - 2.a.x(k).u(k-1|k-1) + a^2.u(k-1|k-1).u(k-1|k-1)] \\ &= E[x(k).x(k) - 2aE[x(k).u(k-1|k-1)] + a^2.E[u(k-1|k-1).u(k-1|k-1)]] \end{aligned} \quad 3.13.5$$

Let us note that  $E[x(k).x(k)] = R(0)$ . Where ( $R(\tau)$ ) indicates the autocorrelation coefficient. For the second term, let us assume that  $u(k-1|k-1)$  is an unbiased estimate of  $x(k-1)$  and that  $u(k-1|k-1)$  is a satisfactorily close estimate of  $x(k-1)$ , so that we can use the following approximation:

$$\begin{aligned} E[x(k).u(k-1|k-1)] &\simeq E[x(k).x(k-1)] \\ &= R(\tau = 1.T_s) \equiv R(1), \text{ say} \end{aligned}$$

The third term in the expanded form of  $E[e_p^2(k)]$  can easily be identified as:

$$\begin{aligned} a^2.E[u(k-1|k-1).u(k-1|k-1)] &= a^2.R(\tau = 0) = a^2.R(0) \\ \therefore E[e_p^2(k)] &= R(0) - 2.a.R(1) + a^2.R(0) \\ &= R(0)[1 - 2a.\frac{R(1)}{R(0)} + a^2] \end{aligned} \quad 3.13.6$$

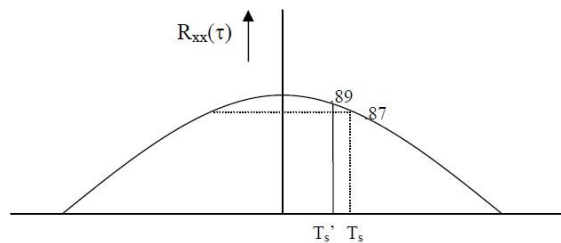
The above expression shows that the mean square error or variance of the prediction error can be minimized if  $a = R(1)/R(0)$ .

## Delta Modulation (DM)

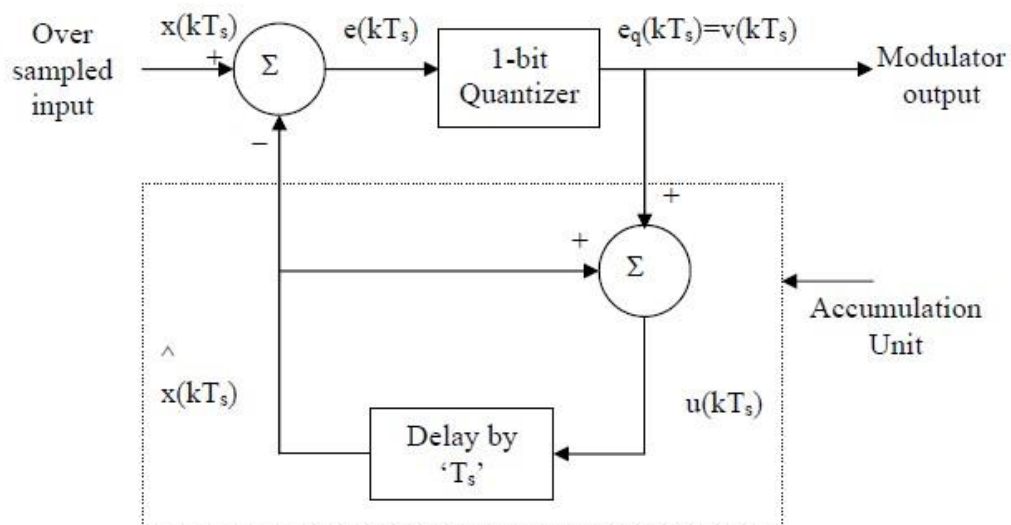
- If the sampling interval „ $T_s$ “ in DPCM is reduced considerably, i.e. if we sample a band limited signal at a rate much faster than the Nyquist sampling rate, the adjacent samples should have higher correlation (**Fig. 3.14.1**).
- The sample-to-sample amplitude difference will usually be very small. So, one may even think of only 1-bit quantization of the difference signal.
- The principle of Delta Modulation (DM) is based on this premise. Delta modulation is also viewed as a 1-bit DPCM scheme.
- The 1-bit quantizer is equivalent to a two-level comparator (also called as a hard limiter). **Fig. 3.14.2** shows the schematic arrangement for generating a delta-modulated signal. Note that,

$$e(kT_s) = x(kT_s) - \hat{x}(kT_s) \quad 3.14.1$$

$$= x(kT_s) - u([k-1]T_s) \quad 3.14.2$$



**Fig. 3.14.1** The correlation increases when the sampling interval is reduced



**Fig. 3.14.2** Block diagram of a delta modulator

- No effective prediction unit – the prediction unit of a DPCM coder (**Fig. 3.13.2**) is eliminated and replaced by a single-unit delay element.
- A 1-bit quantizer with two levels is used. The quantizer output simply indicates whether the present input sample  $x(kT_s)$  is more or less compared to its accumulated approximation  $\hat{x}(kT_s)$ .
- Output  $\hat{x}(kT_s)$  of the delay unit changes in small steps.
- The accumulator unit goes on adding the quantizer output with the previous accumulated version  $\hat{x}(kT_s)$ .
- $u(kT_s)$ , is an approximate version of  $x(kT_s)$ .
- Performance of the Delta Modulation scheme is dependent on the sampling rate. Most of the above comments are acceptable only when two consecutive input samples are very close to each other.

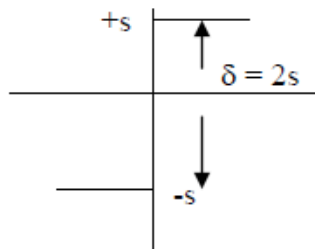
Now, referring back to **Fig. 3.14.2**, we see that,

$$e(kT_s) = x(kT_s) - \{\hat{x}([k-1]T_s) + v([k-1]T_s)\} \quad 3.14.3$$

Further,

$$v(kT_s) = e_q(kT_s) = s \cdot \text{sign}[e(kT_s)] \quad 3.14.4$$

Here, 's' is half of the step-size  $\delta$  as indicated in **Fig. 3.14.3**.



**Fig. 3.14.3** This diagram indicates the output levels of 1-bit quantizer. Note that if  $\delta$  is the step size, the two output levels are  $\pm s$

Now, assuming zero initial condition of the accumulator, it is easy to see that

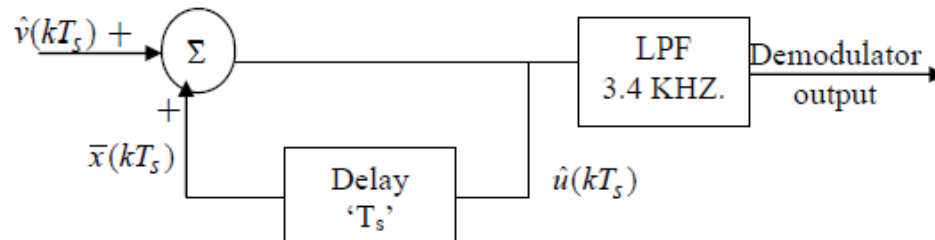
$$u(kT_s) = s \cdot \sum_{j=1}^k \text{sign}[e(jT_s)]$$

$$u(kT_s) = \sum_{j=1}^k v(jT_s) \quad 3.14.5$$

Further,

$$\hat{x}(kT_s) = u([k-1]T_s) = \sum_{j=1}^{k-1} v(jT_s) \quad 3.14.6$$

Eq. 3.14.6 shows that  $\hat{x}(kT_s)$  is essentially an accumulated version of the quantizer output for the error signal  $e(kT_s)$ .  $\hat{x}(kT_s)$  also gives a clue to the demodulator structure for DM. **Fig. 3.14.4** shows a scheme for demodulation. The input to the demodulator is a binary sequence and the demodulator normally starts with no prior information about the incoming sequence.



**Fig. 3.14.4** Demodulator structure for DM

Now, let us recollect from our discussion on DPCM in the previous lesson ( Eq. 3.13.3) that,  $u(kT_s)$  closely represents the input signal with small quantization error  $q(kT_s)$ , i.e.

$$u(kT_s) = x(kT_s) + q(kT_s) \quad 3.14.7$$

Next, from the close loop including the delay-element in the accumulation unit in the Delta modulator structure, we can write

$$u([k-1]T_s) = \hat{x}(kT_s) = x(kT_s) - e(kT_s) = x([k-1]T_s) + q([k-1]T_s) \quad 3.14.8$$

Hence, we may express the error signal as,

$$e(kT_s) = \{x(kT_s) - x([k-1]T_s)\} - q([k-1]T_s) \quad 3.14.9$$

That is, the error signal is the difference of two consecutive samples at the input except the quantization error (when quantization error is small).

### Advantages of a Delta Modulator over DPCM

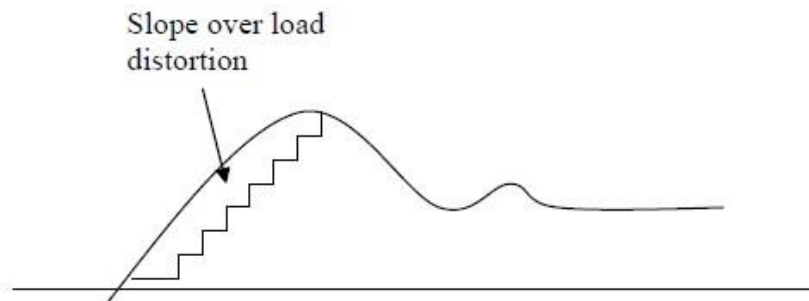
- As one sample of  $x(kT_s)$  is represented by only one bit after delta modulation, no elaborate word-level synchronization is necessary at the input of the demodulator. This reduces hardware complexity compared to a PCM or DPCM demodulator. Bit-timing synchronization is, however, necessary if the demodulator is implemented digitally.
- Overall complexity of a delta modulator-demodulator is less compared to DPCM as the predictor unit is absent in DM.

### Limitations of delta modulation

- **Slope over load distortion:**

If the input signal amplitude changes fast, the step-by-step accumulation process may not catch up with the rate of change (see the sketch in **Fig. 3.14.5**). This happens initially when the demodulator starts operation from cold-start but is usually of negligible effect for speech.

- However, if this phenomenon occurs frequently (which indirectly implies smaller value of auto-correlation co-efficient  $R_{xx}(\tau)$  over a short time interval) the quality of the received signal suffers. The received signal is said to suffer from slope-overload distortion.
- An intuitive remedy for this problem is to increase the step-size  $\delta$  but that approach has another serious lacuna as noted in b).



**Fig. 3.14.5** A sketch indicating slope-overload problem. The horizontal axis represents time. The continuous line represents the analog input signal, before sampling and the stair-case represents the output  $\hat{x}(kT_s)$  of the delay element.

- **Granular noise:**

If the step-size is made arbitrarily large to avoid slope-overload distortion, it may lead to „granular noise“. Imagine that the input speech signal is fluctuating but very close to zero over limited time duration. This may happen due to pauses between sentences or else.

- During such moments, our delta modulator is likely to produce a fairly long sequence of 101010...., reflecting that the accumulator output is close but alternating around the input signal. This phenomenon is manifested at the output of the delta demodulator as a small but perceptible noisy background. This is known as „granular noise“. An expert listener can recognize the crackling sound.
- This noise should be kept well within a tolerable limit while deciding the step-size. Larger step-size increases the granular noise while smaller step size increases the degree of slope-overload distortion. In the first level of design, more care is given to avoid the slope-overload distortion. We will briefly discuss about this approach while keeping the step-size fixed.
- A more efficient approach of adapting the step-size, leading to Adaptive Delta Modulation (ADM) , is excluded.

**Condition for avoiding slope overload:** From Fig. 3.14.3 we may observe that if an input signal changes more than half of the step size (i.e. by 's') within a sampling interval, there will be slope-overload distortion. So, the desired limiting condition on the input signal  $x(t)$  for avoiding slope-overloading is,

$$\left. \frac{dx(t)}{dt} \right|_{\max} \leq \frac{s}{T_s} \quad 3.14.10$$

## Matched Filter

Certain structural modification and simplifications of the correlation receiver are possible by observing that,

- (a) All orthonormal basis functions  $\varphi_j - s$  are defined between  $0 \leq t \leq T_b$  and they are zero outside this range .
- (b) Analog multiplication, which is not always very simple and accurate to implement, of the received signal  $r(t)$  with time limited basis functions may be replaced by some filtering operation.

Let,  $h_j(t)$  represent the impulse response of a linear filter to which  $r(t)$  is applied.

Then, the filter output  $y_j(t)$  may be expressed as:

Then, the filter output  $y_j(t)$  may be expressed as:

$$y_j(t) = \int_{-\infty}^{\infty} r(\tau) h_j(t - \tau) d\tau \quad 4.20.1$$

Now, let,  $h_j(t) = \varphi_j(T - t)$ , a time reversed and time-shifted version of  $\varphi_j(t)$ .

$$\begin{aligned} \text{Now, } y_j(t) &= \int_{-\infty}^{\infty} r(\tau) \cdot \varphi_j[T - (t - \tau)] d\tau \\ &= \int_{-\infty}^{\infty} r(\tau) \cdot \varphi_j(T + \tau - t) d\tau \end{aligned} \quad 4.20.2$$

If we sample this output at  $t = T$ ,

$$y_j(T) = \int_{-\infty}^{\infty} r(\tau) \cdot \varphi_j(\tau) d\tau \quad 4.20.3$$

Let us recall that  $\varphi_j(t)$  is zero outside the interval  $0 \leq t \leq T$ . Using this, the above equation may be expressed as,

$$y_j(T) = \int_0^T r(\tau) \varphi_j(\tau) d\tau$$

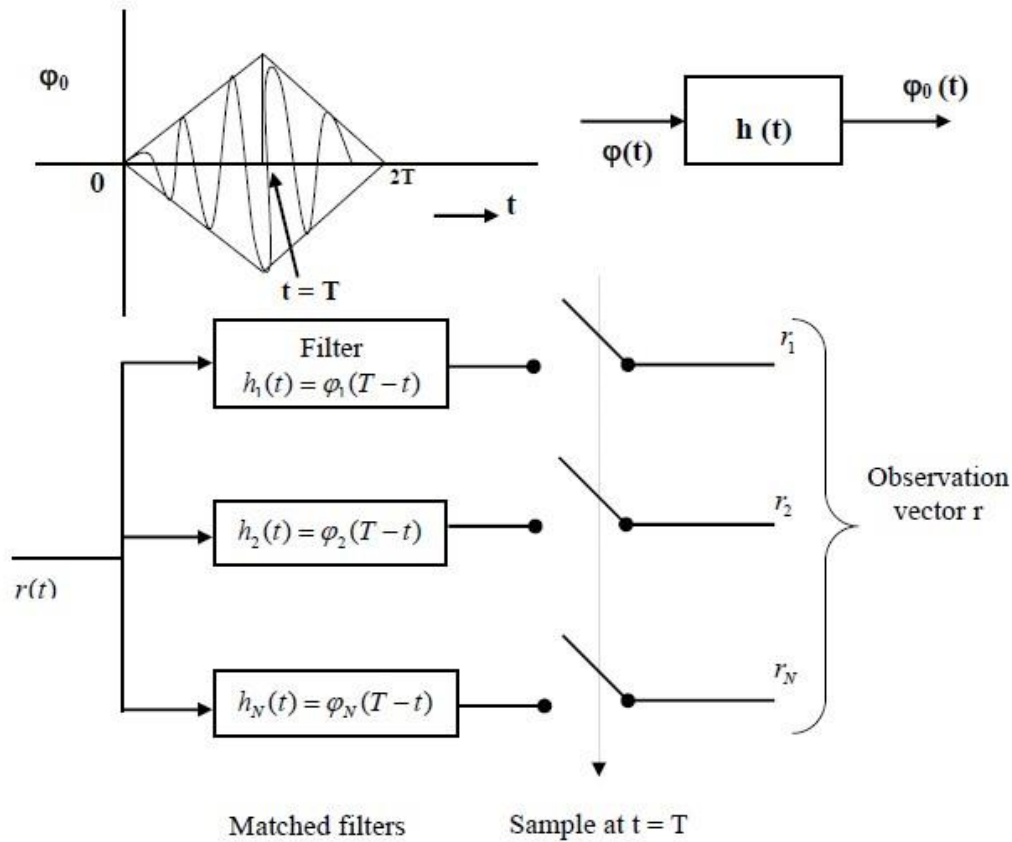


From our discussion on correlation receiver, we recognize that,

$$r_j = \int_0^T r(\tau)\varphi_j(\tau)d\tau = y_j(\tau) \quad 4.20.4$$

The important expression of (Eq.4.20.4) tells us that the  $j$  – th correlation output can equivalently be obtained by using a filter with  $h_j(t) = \varphi_j(T - t)$  and sampling its output at  $t = T$ .

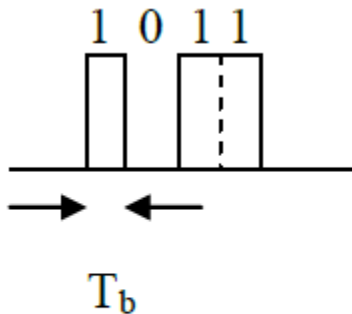
The filter is said to be matched to the orthonormal basis function  $\varphi_j(t)$  and the alternation receiver structure is known as a matched filter receiver. The detector part of the matched filter receiver is shown in [Fig.4.20.1].



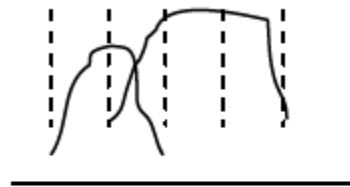
A physically realizable matched filter is to be causal and  $h_j(t) = 0$  for  $t < 0$ . Note that if  $\varphi_j(t)$  is zero outside  $0 \leq t \leq T$ ,  $h_j(t) = \varphi_j(T - t)$  is a causal impulse response.

## Inter Symbol Interference

- Generally, digital data is represented by electrical pulse, communication channel is always band limited. Such a channel disperses or spreads a pulse carrying digitized samples passing through it. When the channel bandwidth is greater than bandwidth of pulse, spreading of pulse is very less.
- But when channel bandwidth is close to signal bandwidth, i.e. if we transmit digital data which demands more bandwidth which exceeds channel bandwidth, spreading will occur and cause signal pulses to overlap.
- This overlapping is called **Inter Symbol Interference**. In short it is called ISI. S
- Similar to interference caused by other sources, ISI causes degradations of signal if left uncontrolled. This problem of ISI exists strongly in Telephone channels like coaxial cables and optical fibers.
- Main objective is to study the effect of ISI, when digital data is transmitted through band limited channel and solution to overcome the degradation of waveform by properly shaping pulse.



Transmitted Waveform



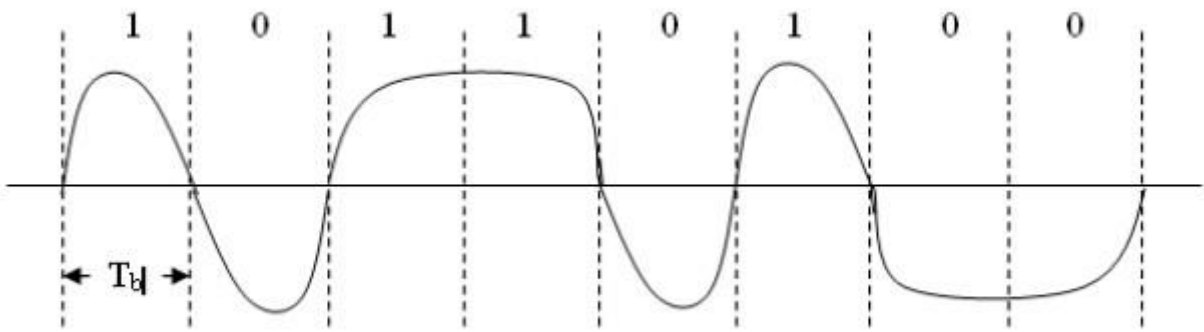
Pulse Dispersion

- The effect of sequence of pulses transmitted through channel is shown in fig. The Spreading of pulse is greater than symbol duration, as a result adjacent pulses interfere. i.e. pulses get completely smeared, tail of smeared pulse enter into adjacent symbol intervals making it difficult to decide actual transmitted pulse
- In base band transmission best way is to map digits or symbols into pulse waveform. This waveform is generally termed as **Line codes**.

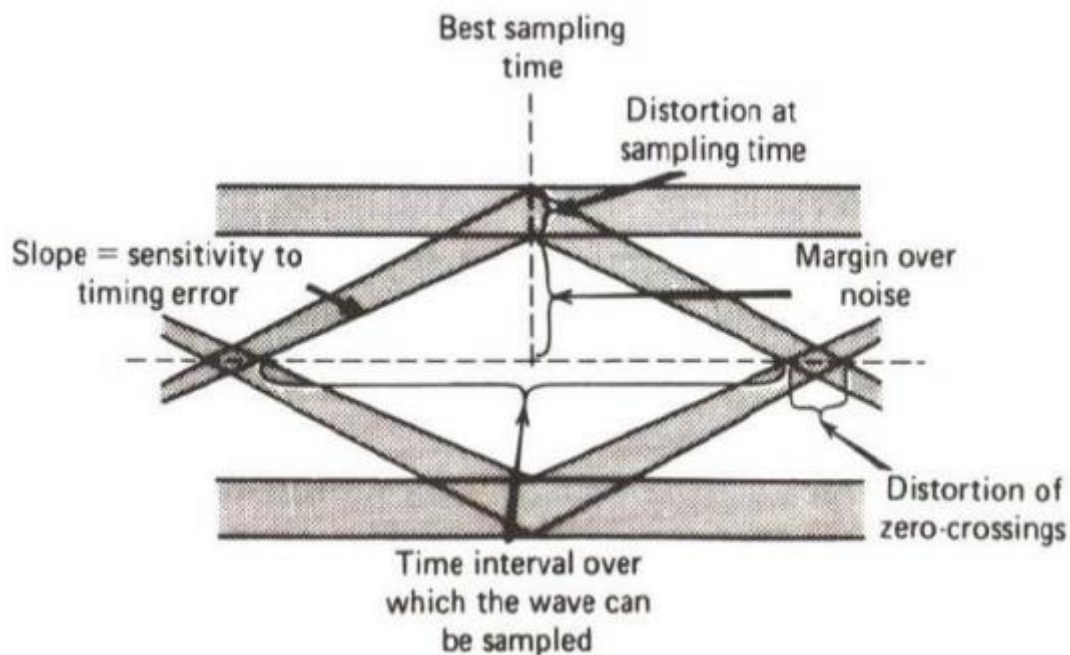
## EYE PATTERN

- The quality of digital transmission systems are evaluated using the bit error rate. Degradation of quality occurs in each process modulation, transmission, and detection.
- The eye pattern is experimental method that contains all the information concerning the degradation of quality. Therefore, careful analysis of the eye pattern is important in analyzing the degradation mechanism.
- Eye patterns can be observed using an oscilloscope. The received wave is applied to the vertical deflection plates of an oscilloscope and the sawtooth wave at a rate equal to transmitted symbol rate is applied to the horizontal deflection plates, resulting display is eye pattern as it resembles human eye.

- The interior region of eye pattern is called eye opening.



We get superposition of successive symbol intervals to produce eye pattern as shown below.



### Interpretation of eye pattern

- The width of the eye opening defines the time interval over which the received wave can be sampled without error from ISI
- The optimum sampling time corresponds to the maximum eye opening
- The height of the eye opening at a specified sampling time is a measure of the margin over channel noise.
- The sensitivity of the system to timing error is determined by the rate of closure of the eye as the sampling time is varied. Any non linear transmission distortion would reveal itself in an asymmetric or squinted eye.
- When the effect of ISI is excessive, traces from the upper portion of the eye pattern cross traces from lower portion with the result that the eye is completely closed.

## UNIT II – BASEBAND TRANSMISSION

### 1. DIGITAL MODULATION

- When the modulating signal is a digital signal, the corresponding modulation is called digital modulation.
- Digital modulation results in discrete changes in the modulated signal.
- The receiver examines the signal at specified times only and the state of the signal at each such time is called a symbol.
- The exact data rate depends upon the number of signal changes per second.
- **Bit Rate:** Is the number of bits transmitted per second, which is the data rate.
- **Baud Rate:** Is defined as the number of signal changes per second or number of symbols per seconds that the line experiences or senses the changes in signal states. It is possible to carry several bits per signal change, giving a higher data rate than the baud rate.
- There is a theoretical limit to the maximum data rate that can be transmitted with a given BW.
- Shannon-Hartley's Law states that:

$$C = 2B \log_2(m)$$

Where  $C$  is the information capacity of the channel in bits/sec

$B$  : Bandwidth.

$m$  : number of possible states per symbol.

Noise also puts limits to the information capacity.

Shannon's Law states that:

$$C = B \log_2 \left( 1 + \frac{S}{N} \right)$$

C can also be expressed as:

$$C = S \log_2(m)$$

Where  $S$  is the baud rate in symbols per second.

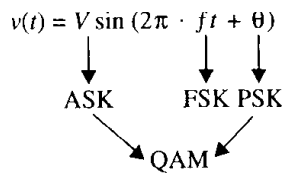
$m$  : number of symbols.

### DIGITAL MODULATION SCHEMES

Types of digital modulation:

- **Amplitude Shift Keying (ASK):**  
If the information signal is digital and the amplitude (IV) of the carrier is varied proportional to the information signal, a digitally modulated signal called *amplitude shift keying* (ASK) is produced.
- **Frequency Shift Keying (FSK):**  
If the frequency (f) is varied proportional to the information signal, frequency shift keying (FSK) is produced.
- **Phase Shift Keying (PSK)**  
If the phase of the carrier ( $\theta$ ) is varied proportional to the information signal, phase shift keying (PSK) is produced.
- **Quadrature Amplitude Modulation (QAM)**

If both the amplitude and the phase are varied proportional to the information signal, quadrature amplitude modulation (QAM) results.



## M-ary Encoding

- *M*-ary is a term derived from the word *binary*.
- *M* simply represents a digit that corresponds to the number of conditions, levels, or combinations possible for a given number of binary variables.
- For example, a digital signal with four possible conditions (voltage levels, frequencies, phases, and so on) is an *M*-ary system where  $M = 4$ . If there are eight possible conditions,  $M = 8$  and so forth.
- The number of bits necessary to produce a given number of conditions is expressed mathematically as

$$N = \log_2 M \quad (1)$$

where  $N$  = number of bits necessary

$M$  = number of conditions, levels, or combinations possible with  $N$  bits

- Equation (1) can be simplified and rearranged to express the number of conditions possible with  $N$  bits as

$$2^N = M \quad (2)$$

- For example, with one bit, only  $2^1 = 2$  conditions are possible. With two bits,  $2^2 = 4$  conditions are possible, with three bits,  $2^3 = 8$  conditions are possible, and so on.

## Baud and Minimum Bandwidth

- Baud refers to the rate of change of a signal on the transmission medium after encoding and modulation have occurred.
- Hence, baud is a unit of transmission rate, modulation rate, or symbol rate and, therefore, the terms symbols per second and baud are often used interchangeably. Mathematically, baud is the reciprocal of the time of one output signaling element, and a signaling element may represent several information bits. Baud is expressed as

$$\text{baud} = \frac{1}{t_s} \quad (3)$$

where baud = symbol rate (baud per second)  
 $t_s$  = time of one signaling element (seconds)

- The minimum theoretical bandwidth necessary to propagate a signal is called the minimum Nyquist bandwidth or sometimes the minimum Nyquist frequency.
- Thus,  $f_b = 2B$ , where  $f_b$  is the bit rate in bps and  $B$  is the ideal Nyquist bandwidth.
- The relationship between bandwidth and bit rate also applies to the opposite situation. For a given bandwidth ( $B$ ), the highest theoretical bit rate is  $2B$ .
- For example, a standard telephone circuit has a bandwidth of approximately 2700 Hz, which has the capacity to propagate 5400 bps through it. However, if more than two levels are used for signaling (higher-than-binary encoding), more than one bit may be transmitted at a time, and it is possible to propagate a bit rate that exceeds  $2B$ .
- Using multilevel signaling, the Nyquist formulation for channel capacity is

$$f_b = B \log_2 M \quad (4)$$

where  $f_b$  = channel capacity (bps)

$B$  = minimum Nyquist bandwidth (hertz)

$M$  = number of discrete signal or voltage levels

- Equation (4) can be rearranged to solve for the minimum bandwidth necessary to pass  $M$ -ary digitally modulated carriers

$$B = \left( \frac{f_b}{\log_2 M} \right) \quad (5)$$

- If  $N$  is substituted for  $\log_2 M$ , Equation (5) reduces to

$$B = \left( \frac{f_b}{N} \right) \quad (6)$$

where  $N$  is the number of bits encoded into each signaling element.

- In addition, since baud is the encoded rate of change, it also equals the bit rate divided by the number of bits encoded into one signaling element. Thus,

$$\text{Baud} = \left( \frac{f_b}{N} \right) \quad (7)$$

- By comparing Equation 6 with Equation 7 the baud and the ideal minimum Nyquist bandwidth have the same value and are equal to the bit rate divided by the number of bits encoded.

## AMPLITUDE-SHIFT KEYING

- The simplest digital modulation technique is *amplitude-shift keying* (ASK), where a binary information signal directly modulates the amplitude of an analog carrier.
- ASK is similar to standard amplitude modulation except there are only two output amplitudes possible. Amplitude-shift keying is sometimes called *digital amplitude modulation* (DAM).
- Mathematically, amplitude-shift keying is

$$v_{(ask)}(t) = [1 + v_m(t)] \left[ \frac{A}{2} \cos(\omega_c t) \right] \quad (8)$$

where

$v_{ask}(t)$  = amplitude-shift keying wave

$v_m(t)$  = digital information (modulating) signal (volts)  
 $A/2$  = unmodulated carrier amplitude (volts)

$\omega_c$  = analog carrier radian frequency (radians per second,  $2\pi f_c t$ )

- In Equation 8, the modulating signal [ $v_m(t)$ ] is a normalized binary waveform, where +1 V = logic 1 and -1 V = logic 0. Therefore, for a logic 1 input,  $v_m(t) = +1$  V, Equation (8) reduces to

$$\begin{aligned} v_{(ask)}(t) &= [1 + 1] \left[ \frac{A}{2} \cos(\omega_c t) \right] \\ &= A \cos(\omega_c t) \end{aligned}$$

and for a logic 0 input,  $v_m(t) = -1$  V, Equation 8 reduces to

- Thus, the modulated wave  $v_{ask}(t)$ , is either  $A \cos(\omega_c t)$  or 0. Hence, the carrier is either "on" or "off," which is why amplitude-shift keying is sometimes referred to as *on-off keying* (OOK).
- Figure 1 shows the input and output waveforms from an ASK modulator.
- From the figure, it can be seen that for every change in the input binary data stream, there is one change in the ASK waveform, and the time of one bit ( $t_b$ ) equals the time of one analog signaling element ( $t_s$ ).

$$B = f_b / 1 = f_b$$

$$\text{baud} = f_b / 1 = f_b$$

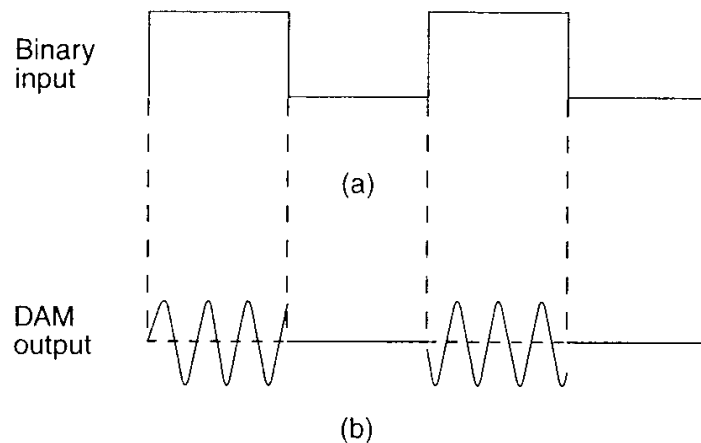


FIGURE 1 Digital amplitude modulation: (a) input binary; (b) output DAM waveform

- The entire time the binary input is high, the output is a constant-amplitude, constant-frequency signal, and for the entire time the binary input is low, the carrier is off.
- The rate of change of the ASK waveform (baud) is the same as the rate of change of the binary input (bps).

---

## FREQUENCY-SHIFT KEYING

- FSK is a form of constant-amplitude angle modulation similar to standard frequency modulation (FM) except the modulating signal is a binary signal that varies between two discrete voltage levels rather than a continuously changing analog



waveform.

- FSK is sometimes called *binary FSK* (BFSK). The general expression for FSK is

$$v_{fsk}(t) = V_c \cos\{2\pi[f_c + v_m(t) \Delta f]t\} \quad (9)$$

where

$v_{fsk}(t)$  = binary FSK waveform

$V_c$  = peak analog carrier amplitude (volts)

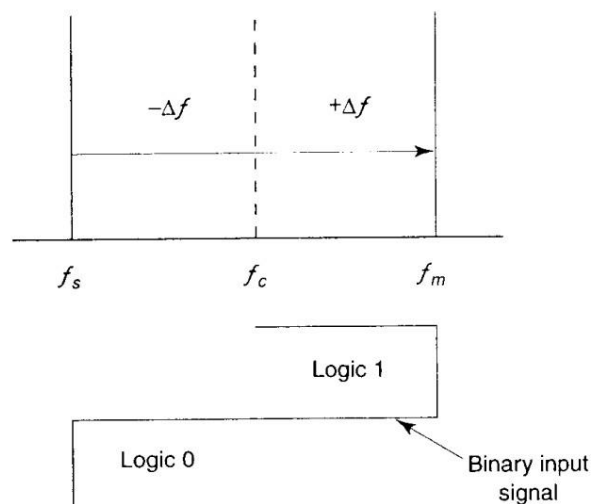
$f_c$  = analog carrier center frequency (hertz)

$\Delta f$  = peak change (shift) in the analog carrier frequency (hertz)  
 $v_m(t)$  = binary input (modulating) signal (volts)

- From Equation 9, it can be seen that the peak shift in the carrier frequency ( $\Delta f$ ) is proportional to the amplitude of the binary input signal ( $v_m[t]$ ), and the direction of the shift is determined by the polarity.
- The modulating signal is a normalized binary waveform where a logic 1 = +1 V and a logic 0 = -1 V. Thus, for a logic 1 input,  $v_m(t) = +1$ , Equation 9 can be rewritten as
- For a logic 0 input,  $v_m(t) = -1$ , Equation (9) becomes

$$v_{fsk}(t) = V_c \cos[2\pi(f_c - \Delta f)t]$$

- With binary FSK, the carrier center frequency ( $f_c$ ) is shifted (deviated) up and down in the frequency domain by the binary input signal as shown in Figure 2.



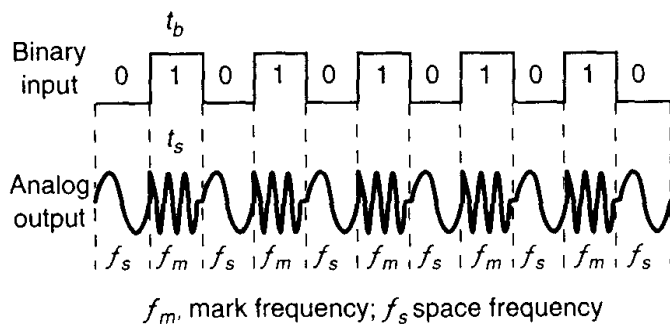
## FIGURE 2. FSK in the frequency domain

- As the binary input signal changes from a logic 0 to a logic 1 and vice versa, the output frequency shifts between two frequencies: a mark, or logic 1 frequency ( $f_m$ ), and a space, or logic 0 frequency ( $f_s$ ). The mark and space frequencies are separated from the carrier frequency by the peak frequency deviation ( $\Delta f$ ) and from each other by  $2 \Delta f$ .
- Frequency deviation is illustrated in Figure 2, and expressed mathematically as

$$\Delta f = |f_m - f_s| / 2 \tag{10}$$

where  $\Delta f$  = frequency deviation (hertz)  
 $|f_m - f_s|$  = absolute difference between the mark and space frequencies (hertz)

- Figure 3a shows in the time domain the binary input to an FSK modulator and the corresponding FSK output.
- When the binary input ( $f_b$ ) changes from a logic 1 to a logic 0 and vice versa, the FSK output frequency shifts from a mark ( $f_m$ ) to a space ( $f_s$ ) frequency and vice versa.
- In Figure 3, the mark frequency is the higher frequency ( $f_c + \Delta f$ ) and the space frequency is the lower frequency ( $f_c - \Delta f$ ), although this relationship could be just the opposite.
- Figure 2-4b shows the truth table for a binary FSK modulator. The truth table shows the input and output possibilities for a given digital modulation scheme.



| binary input | frequency output |
|--------------|------------------|
| 0            | space ( $f_s$ )  |
| 1            | mark ( $f_m$ )   |

(a)

(b)

FIGURE 3 FSK in the time domain: (a) waveform: (b) truth table

### FSK Bit Rate, Baud, and Bandwidth

- In Figure 3a, it can be seen that the time of one bit ( $t_b$ ) is the same as the time the FSK output is a mark of space frequency ( $t_s$ ). Thus, the bit time equals the time of an FSK signaling element, and the bit rate equals the baud.
- The baud for binary FSK can also be determined by substituting  $N = 1$  in Equation 10:

$$\text{baud} = f_b / 1 = f_b$$

- The minimum bandwidth for FSK is given as

$$\begin{aligned} B &= |(f_s - f_b) - (f_m - f_b)| \\ &= |(f_s - f_m)| + 2f_b \end{aligned}$$

and since  $|(f_s - f_m)|$  equals  $2\Delta f$ , the minimum bandwidth can be approximated as  $B =$

$$2(\Delta f + f_b) \quad (11)$$

where

$B$  = minimum Nyquist bandwidth (hertz)  
 $\Delta f$  = frequency deviation  $|(f_m - f_s)|$  (hertz)  
 $f_b$  = input bit rate (bps)

## FSK Transmitter

Figure 4 shows a simplified binary FSK modulator, which is very similar to a conventional FM modulator and is very often a voltage-controlled oscillator (VCO).

The center frequency ( $f_c$ ) is chosen such that it falls halfway between the mark and space frequencies.

- A logic 1 input shifts the VCO output to the mark frequency, and a logic 0 input shifts the VCO output to the space frequency.

Consequently, as the binary input signal changes back and forth between logic 1 and logic 0 conditions, the VCO output shifts or deviates back and forth between the mark and space frequencies.

A VCO-FSK modulator can be operated in the sweep mode where the peak frequency deviation is simply the product of the binary input voltage and the deviation sensitivity of the VCO.

With the sweep mode of modulation, the frequency deviation is expressed mathematically as

$$\Delta f = v_m(t)k_l \quad (12)$$

$v_m(t)$  = peak binary modulating-signal voltage (volts)

$k_l$  = deviation sensitivity (hertz per volt).

### 2-4-3 FSK Receiver

FSK demodulation is quite simple with a circuit such as the one shown in Figure 2-7.

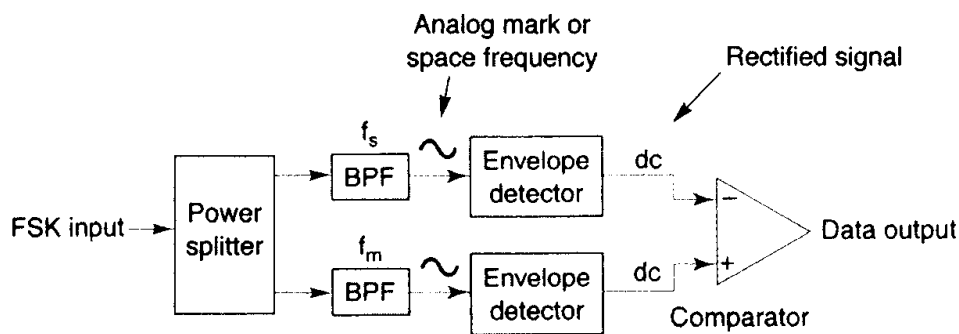


FIGURE 5 Noncoherent FSK demodulator

The FSK input signal is simultaneously applied to the inputs of both bandpass filters (BPFs) through a power splitter. The respective filter passes only the mark or only the space frequency on to its respective envelope detector.

The envelope detectors, in turn, indicate the total power in each passband, and the comparator responds to the largest of the two powers. This type of FSK detection is referred to as noncoherent detection. Figure 6 shows the block diagram for a coherent FSK receiver.

The incoming FSK signal is multiplied by a recovered carrier signal that has the exact same frequency and phase as the transmitter reference.

However, the two transmitted frequencies (the mark and space frequencies) are not generally continuous; it is not practical to reproduce a local reference that is coherent with both of them. Consequently, coherent FSK detection is seldom used.

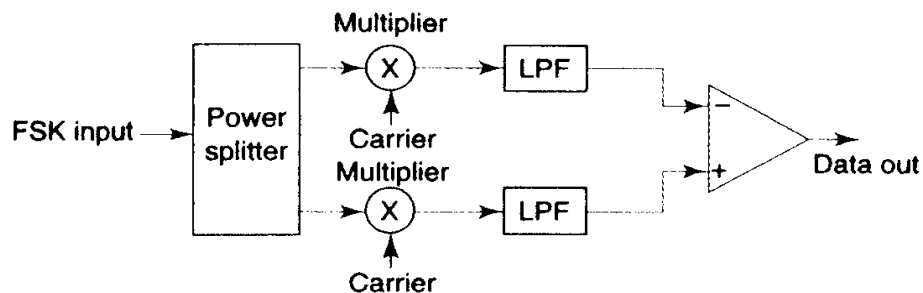


FIGURE 6 Coherent FSK demodulator

The most common circuit used for demodulating binary FSK signals is the *phaselockedloop* (PLL), which is shown in block diagram form in Figure 2-9.

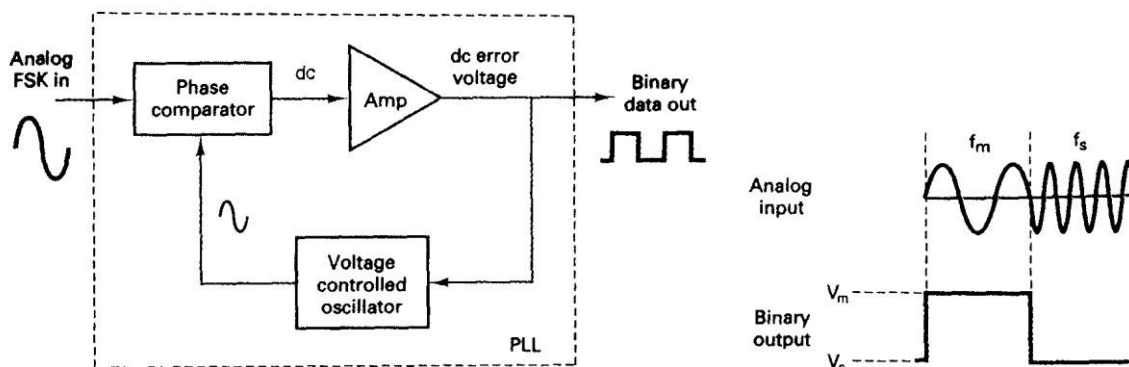


FIGURE 7 PLL-FSK demodulator

As the input to the PLL shifts between the mark and space frequencies, the *dc error voltage* at the output of the phase comparator follows the frequency shift.

Because there are only two input frequencies (mark and space), there are also only two output error voltages. One represents a logic 1 and the other a logic 0.

Binary FSK has a poorer error performance than PSK or QAM and, consequently, is seldom used for high-performance digital radio systems.

Its use is restricted to low-performance, low-cost, asynchronous data modems that are used for data communications over analog, voice-band telephone lines.

## PHASE-SHIFT KEYING

*Phase-shift keying* (PSK) is another form of *angle-modulated, constant-amplitude* digital modulation.

### Binary Phase-Shift Keying

The simplest form of PSK is *binary phase-shift keying* (BPSK), where  $N = 1$  and  $M = 2$ .

Therefore, with BPSK, two phases ( $2^1 = 2$ ) are possible for the carrier.

One phase represents a logic 1, and the other phase represents a logic 0. As the input digital signal changes state (i.e., from a 1 to a 0 or from a 0 to a 1), the phase of the output carrier shifts between two angles that are separated by  $180^\circ$ .

Hence, other names for BPSK are *phase reversal keying* (PRK) and *biphase modulation*. BPSK is a form of square-wave modulation of a *continuous wave* (CW) signal.

## **BPSK transmitter.**

Figure 8 shows a simplified block diagram of a BPSK transmitter.

The balanced modulator acts as a phase reversing switch. Depending on the logic condition of the digital input, the carrier is transferred to the output either in phase or  $180^\circ$  out of phase with the reference carrier oscillator.

Figure 9 shows the schematic diagram of a balanced ring modulator.

The balanced modulator has two inputs: a carrier that is in phase with the reference oscillator and the binary digital data.

For the balanced modulator to operate properly, the digital input voltage must be much greater than the peak carrier voltage.

This ensures that the digital input controls the on/off state of diodes D1 to D4. If the binary input is a logic 1 (positive voltage), diodes D1 and D2 are forward biased and on, while diodes D3 and D4 are reverse biased and off (Figure 9b). With the polarities shown, the carrier voltage is developed across transformer T2 in phase with the carrier voltage across T1. Consequently, the output signal is in phase with the reference oscillator.

If the binary input is a logic 0 (negative voltage), diodes D1 and D2 are reverse biased and off, while diodes D3 and D4 are forward biased and on (Figure 9c). As a result, the carrier voltage is developed across transformer T2  $180^\circ$  out of phase with the carrier voltage across T1.



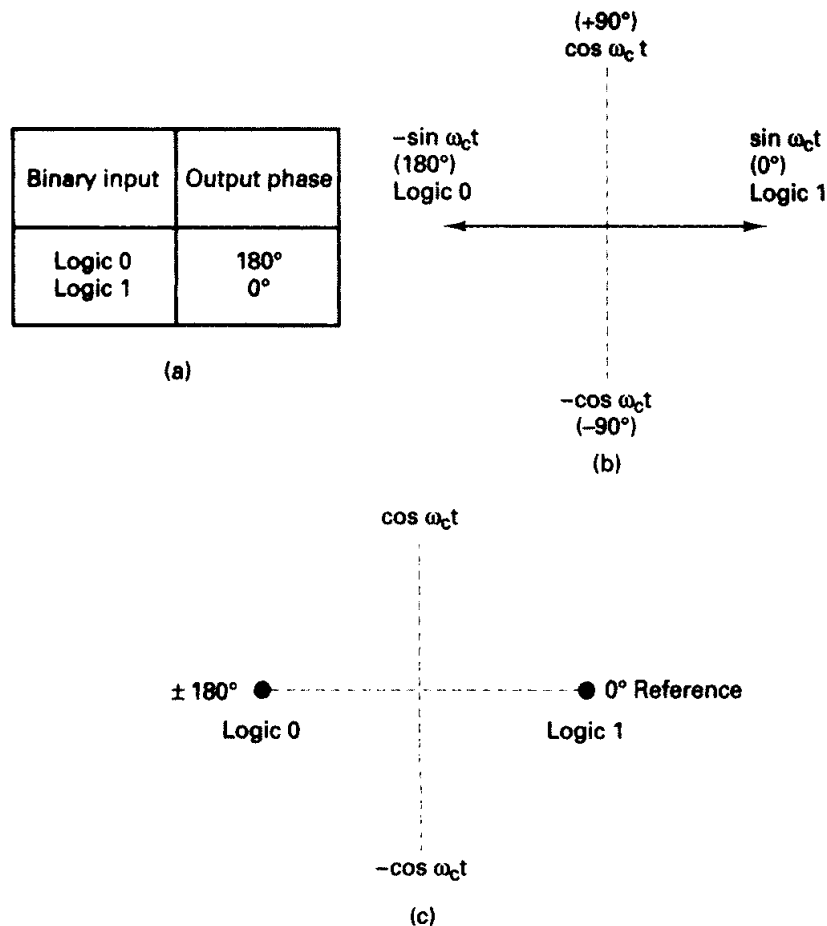


FIGURE 10 BPSK modulator: (a) truth table; (b) phasordiagram; (c) constellation diagram

### Bandwidth considerations of BPSK.

In a BPSK modulator, the carrier input signal is multiplied by the binary data.

If + 1 V is assigned to a logic 1 and -1 V is assigned to a logic 0, the input carrier ( $\sin\omega_c t$ ) is multiplied by either a + or - 1.

The output signal is either  $+ 1 \sin \omega_c t$  or  $-1 \sin \omega_c t$ , the first represents a signal that is *in phase* with the reference oscillator, the latter a signal that is 180° out of phase with the reference oscillator.

Each time the input logic condition changes, the output phase changes.

Mathematically, the output of a BPSK modulator is proportional to

$$\text{BPSK output} = [\sin (2\pi f_a t)] \times [\sin (2\pi f_c t)] \quad (13)$$

where

$f_a$  = maximum fundamental frequency of binary input (hertz)  
 $f_c$  = reference carrier frequency (hertz)



Solving for the trig identity for the product of two sine functions,

$$0.5\cos[2\pi(f_c - f_a)t] - 0.5\cos[2\pi(f_c + f_a)t]$$

Thus, the minimum double-sided Nyquist bandwidth ( $B$ ) is

$$-(f_c + f_a) \quad \text{or} \quad \frac{f_c + f_a}{2f_a}$$

and because  $f_a = f_b / 2$ , where  $f_b =$  input bit rate, where  $B$  is the minimum double-sided Nyquist bandwidth.

Figure 2-15 shows the output phase-versus-time relationship for a BPSK waveform.

Logic 1 input produces an analog output signal with a  $0^\circ$  phase angle, and a logic 0 input produces an analog output signal with a  $180^\circ$  phase angle.

As the binary input shifts between a logic 1 and a logic 0 condition and vice versa, the phase of the BPSK waveform shifts between  $0^\circ$  and  $180^\circ$ , respectively.

BPSK signaling element ( $t_s$ ) is equal to the time of one information bit ( $t_b$ ), which indicates that the bit rate equals the baud.

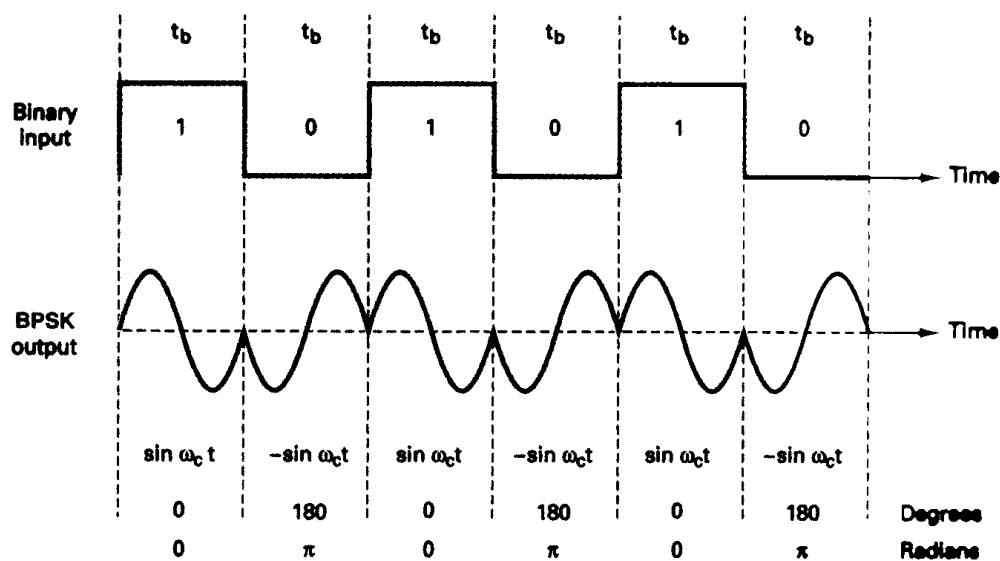


FIGURE 11 Output phase-versus-time relationship for a BPSK modulator

## BPSK receiver.

Figure 12 shows the block diagram of a BPSK receiver. The input signal may be  $+\sin \omega_c t$  or  $-\sin \omega_c t$ .

The coherent carrier recovery circuit detects and regenerates a carrier signal that is both frequency and phase coherent with the original transmit carrier.

The balanced modulator is a product detector; the output is the product of the two inputs (the BPSK signal and the recovered carrier).

The low-pass filter (LPF) operates the recovered binary data from the complex demodulated signal.

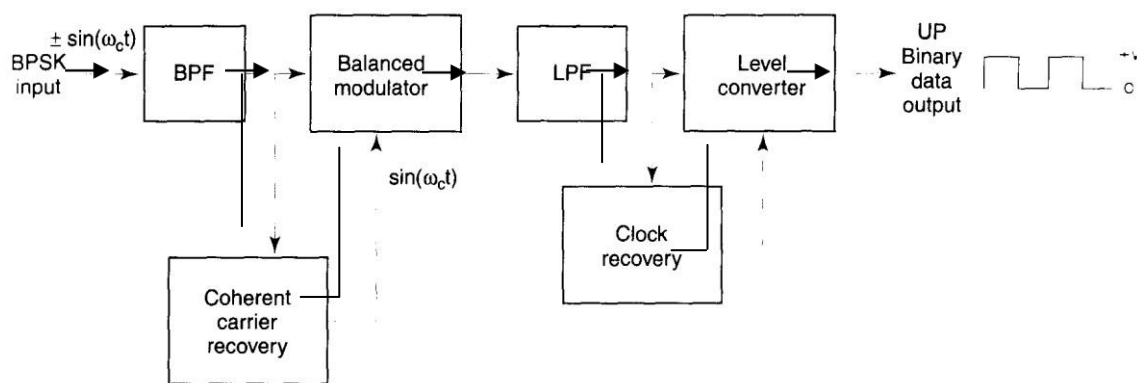


FIGURE 12 Block diagram of a BPSK receiver

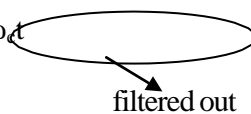
Mathematically, the demodulation process is as follows.

For a BPSK input signal of  $+\sin \omega_c t$  (logic 1), the output of the balanced modulator is

$$\text{output} = (\sin \omega_c t)(\sin \omega_c t) = \sin^2 \omega_c t \quad (14)$$

or

$$\sin^2 \omega_c t = 0.5(1 - \cos 2\omega_c t) = 0.5 - 0.5\cos 2\omega_c t$$


  
 filtered out

leaving

$$\text{output} = +0.5 \text{ V} = \text{logic 1}$$

It can be seen that the output of the balanced modulator contains a positive voltage ( $+0.5 \text{ V}$ ) and a cosine wave at twice the carrier frequency ( $2\omega_c t$ ).

The LPF has a cutoff frequency much lower than  $2\omega_c t$ , and, thus, blocks the second

harmonic of the carrier and passes only the positive constant component. A positive voltage represents a demodulated logic 1.

For a BPSK input signal of  $-\sin \omega_c t$  (logic 0), the output of the balanced modulator is

$$\text{output} = (-\sin \omega_c t)(\sin \omega_c t) = -\sin^2 \omega_c t$$

or

$$\sin^2 \omega_c t = -0.5(1 - \cos 2\omega_c t) = 0.5 - 0.5 \cos 2\omega_c t$$

↑  
filtered out

leaving

$$\text{output} = -0.5 \text{ V} = \text{logic 0}$$

The output of the balanced modulator contains a negative voltage ( $-[1/2]\text{V}$ ) and a cosinewave at twice the carrier frequency ( $2\omega_c t$ ).

Again, the LPF blocks the second harmonic of the carrier and passes only the negative constant component. A negative voltage represents a demodulated logic 0.

## Quaternary Phase-Shift Keying

QPSK is an M-ary encoding scheme where  $N = 2$  and  $M = 4$  (hence, the name "quaternary" meaning "4"). A QPSK modulator is a binary (base 2) signal, to produce four different input combinations, 00, 01, 10, and 11.

Therefore, with QPSK, the binary input data are combined into groups of two bits, called *dibits*. In the modulator, each dibit code generates one of the four possible output phases ( $+45^\circ$ ,  $+135^\circ$ ,  $-45^\circ$ , and  $-135^\circ$ ).

## QPSK transmitter.

A block diagram of a QPSK modulator is shown in Figure 13. Two bits (a dibit) are clocked into the bit splitter. After both bits have been serially inputted, they are simultaneously parallel outputted.

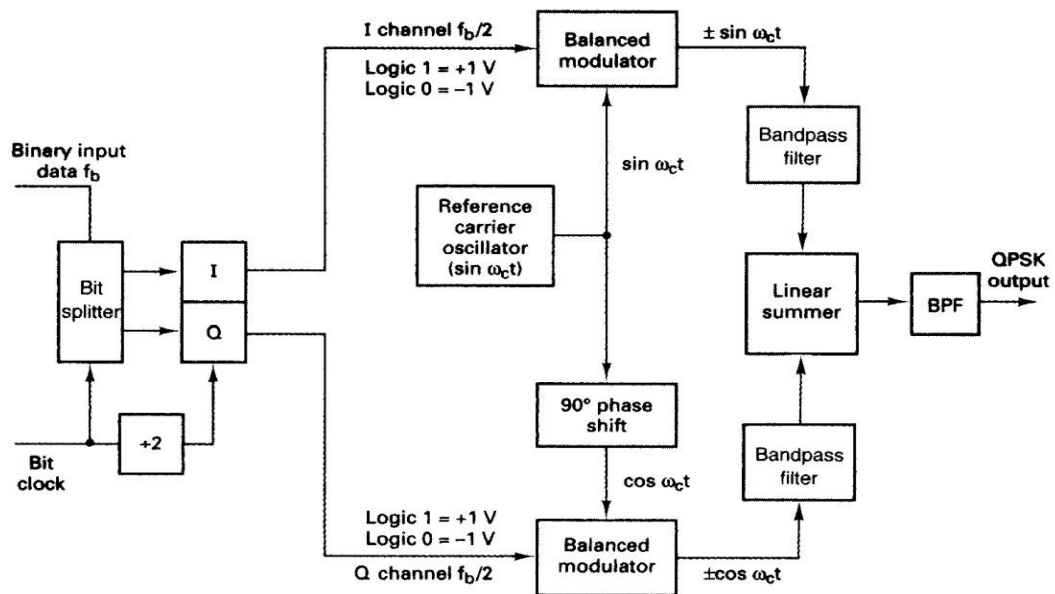
The I bit modulates a carrier that is in phase with the reference oscillator (hence the name "I" for "in phase" channel), and the Q bit modulate, a carrier that is  $90^\circ$  out of phase.

For a logic 1 = + 1 V and a logic 0 = - 1 V, two phases are possible at the output of the I balanced modulator ( $+\sin \omega_c t$  and  $-\sin \omega_c t$ ), and two phases are possible at the

output of the Q balanced modulator ( $+\cos \omega_c t$ ), and ( $-\cos \omega_c t$ ).

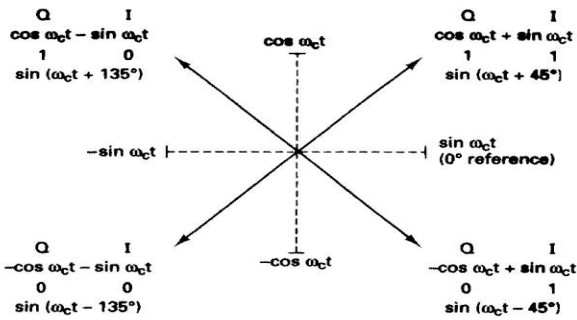
When the linear summer combines the two quadrature ( $90^\circ$  out of phase) signals, there are four possible resultant phasors given by these expressions:  $+\sin \omega_c t + \cos \omega_c t$ ,  $+\sin \omega_c t - \cos \omega_c t$ ,  $-\sin \omega_c t + \cos \omega_c t$ , and  $-\sin \omega_c t - \cos \omega_c t$ .

FIGURE 13. QPSK modulator

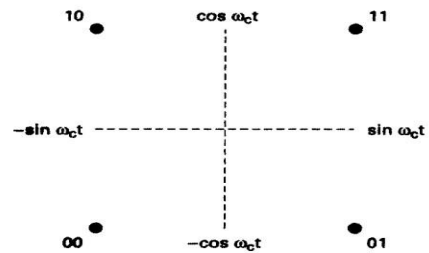


| Binary input |   | QPSK output phase |
|--------------|---|-------------------|
| Q            | I |                   |
| 0            | 0 | -135°             |
| 0            | 1 | -45°              |
| 1            | 0 | +135°             |
| 1            | 1 | +45°              |

(a)



(b)



(c)

FIGURE 14 QPSK modulator: (a) truth table; (b) phasor diagram; (c) constellation diagram

In Figures 14b and c, it can be seen that with QPSK each of the four possible output phasors has exactly the same amplitude. Therefore, the binary information must be encoded entirely in the phase of the output signal.

Figure 14b, it can be seen that the angular separation between any two adjacent phasors in QPSK is 90°.

Therefore, a QPSK signal can undergo almost a +45° or -45° shift in phase during transmission and still retain the correct encoded information when demodulated at the receiver.

Figure 14 shows the output phase-versus-time relationship for a QPSK modulator.



FIGURE 14 Output phase-versus-time relationship for a PSK modulator.

## Bandwidth considerations of QPSK

With QPSK, because the input data are divided into two channels, the bit rate in either the I or the Q channel is equal to one-half of the input data rate ( $f_b/2$ ) (one-half of  $f_b/2 = f_b/4$ ).

This relationship is shown in Figure 15.

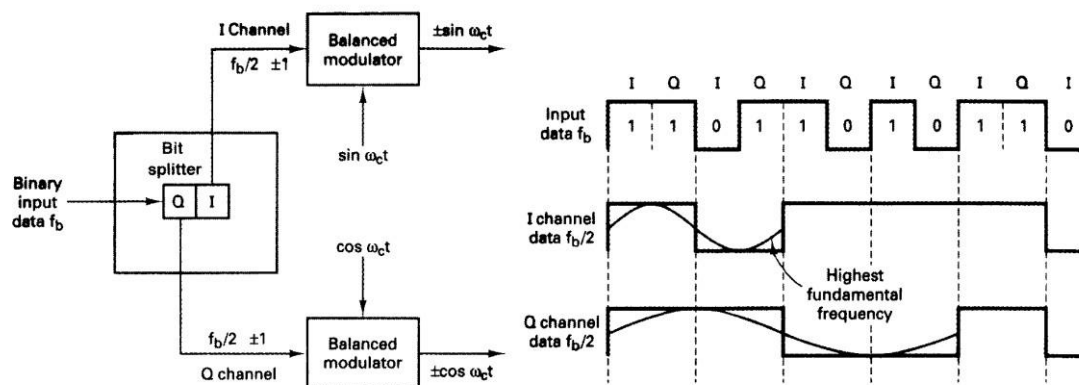


FIGURE 15 Bandwidth considerations of a QPSK modulator

In Figure 15, it can be seen that the worst-case input condition to the I or Q balanced modulator is an alternative 1/0 pattern, which occurs when the binary input data have a 1100 repetitive pattern. One cycle of the fastest binary transition (a 1/0 sequence in the I or Q channel) takes the same time as four input data bits.

Consequently, the highest fundamental frequency at the input and fastest rate of change at the output of the balanced modulators is equal to one-fourth of the binary input bit rate.

The output of the balanced modulators can be expressed mathematically as

$$\text{output} = (\sin \omega_u t)(\sin \omega_c t) \quad (15)$$

where

$$\omega_u = 2\pi f_b \quad \text{and} \quad \omega_c = 2\pi f_c$$

$$\text{output} = \left( \sin 2\pi \frac{f_b}{4} t \right) (\sin 2\pi f_c t)$$

$$\frac{1}{2} \cos 2\pi \left( f_c - \frac{f_b}{4} \right) t - \frac{1}{2} \cos 2\pi \left( f_c + \frac{f_b}{4} \right) t$$

The output frequency spectrum extends from  $f_c + f_b / 4$  to  $f_c - f_b / 4$  and the minimum band-width ( $f_N$ ) is

$$\left( f_c - \frac{f_b}{4} \right) \left( f_c + \frac{f_b}{4} \right) = 2f_b = f_b$$

### QPSK receiver

The block diagram of a QPSK receiver is shown in Figure 16. The power splitter directs the input QPSK signal to the I and Q product detectors and the carrier recovery circuit. The carrier recovery circuit reproduces the original transmit carrier oscillator signal. The recovered carrier must be frequency and phase coherent with the transmit reference carrier. The QPSK signal is demodulated in the I and Q product detectors, which generate the original I and Q data bits. The outputs of the product detectors are fed to the bit combining circuit, where they are converted from parallel I and Q data channels to a single binary output data stream.

The incoming QPSK signal may be any one of the four possible output phases shown in Figure 2-18. To illustrate the demodulation process, let the incoming QPSK signal be  $-\sin \omega_c t + \cos \omega_c t$ . Mathematically, the demodulation process is as follows.

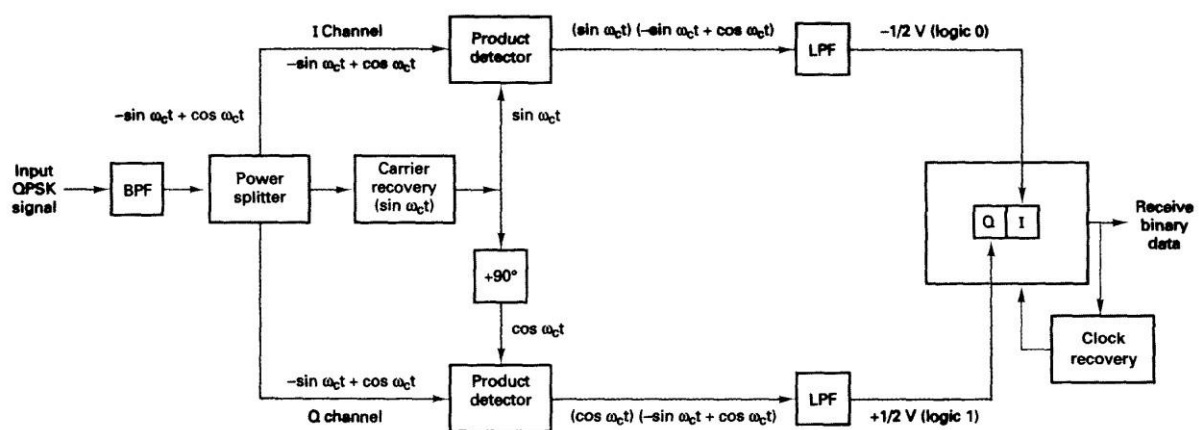


FIGURE 16 QPSK receiver

The receive QPSK signal  $(-\sin \omega_c t + \cos \omega_c t)$  is one of the inputs to the I product detector. The other input is the recovered carrier  $(\sin \omega_c t)$ . The output of the I product de-

Again, the receive QPSK signal  $(-\sin \omega_c t + \cos \omega_c t)$  is one of the inputs to the Q product detector. The other input is the recovered carrier shifted  $90^\circ$  in phase  $(\cos \omega_c t)$ . The output of the Q product detector is

The demodulated I and Q bits (0 and 1, respectively) correspond to the constellation diagram and truth table for the QPSK modulator shown in Figure 14.

### Offset QPSK.

*Offset QPSK (OQPSK)* is a modified form of QPSK where the bit waveforms on the I and Q channels are offset or shifted in phase from each other by one-half of a bit time.

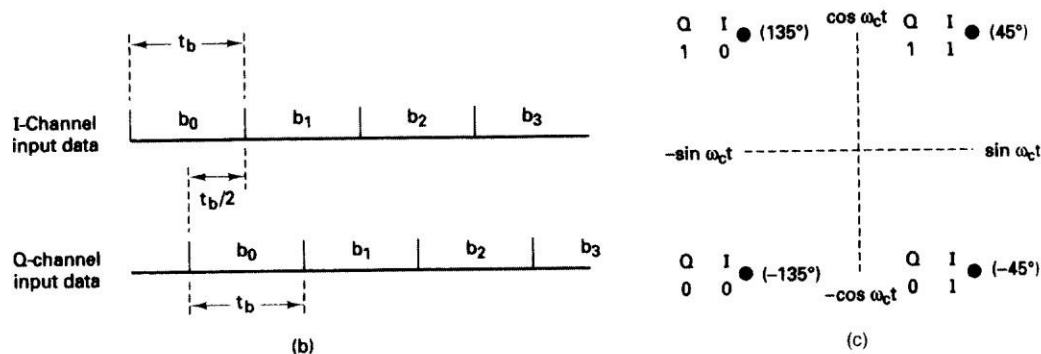


FIGURE 16: Offset keyed (OQPSK): (a) block diagram; (b) bit alignment; (c) constellation diagram

Because changes in the I channel occur at the midpoints of the Q channel bits and vice versa, there is never more than a single bit change in the dibit code and, therefore, there is never more than a  $90^\circ$  shift in the output phase. In conventional QPSK, a change in the input dibit from 00 to 11 or 01 to 10 causes a corresponding  $180^\circ$  shift in the output phase.

Therefore, an advantage of OQPSK is the limited phase shift that must be imparted during modulation.

A disadvantage of OQPSK is that changes in the output phase occur at twice the data rate in either the I or Q channel".

Consequently, with OQPSK the baud and minimum bandwidth are twice that of conventional QPSK for a given transmission bit rate. OQPSK is sometimes called OKQPSK (*offset-keyed QPSK*).

## QUADRATURE – AMPLITUDE

### MODULATION 8-QAM



8-QAM is an M-ary encoding technique where  $M = 8$ . Unlike 8-PSK, the output signal from an 8-QAM modulator is not a constant-amplitude signal.

### **8-QAM transmitter.**

Figure 2-30a shows the block diagram of an 8-QAM transmitter. As you can see, the only difference between the 8-QAM transmitter and the 8PSK transmitter shown in Figure 2-23 is the omission of the inverter between the C channel and the Q product modulator. As with 8-PSK, the incoming data are divided into groups of three bits (tribits): the I, Q, and C bit streams, each with a bit rate equal to one-third of the incoming data rate. Again, the I and

Q bits determine the polarity of the PAM signal at the output of the 2-to-4-level converters, and the C channel determines the magnitude. Because the C bit is fed uninverted to both the I and the Q channel 2-to-4-level converters, the magnitudes of the I and Q PAM signals are always equal. Their polarities depend on the logic condition of the I and Q bits and, therefore, may be different. Figure 2-30b shows the truth table for the I and Q channel 2-to-4-level converters; they are identical.

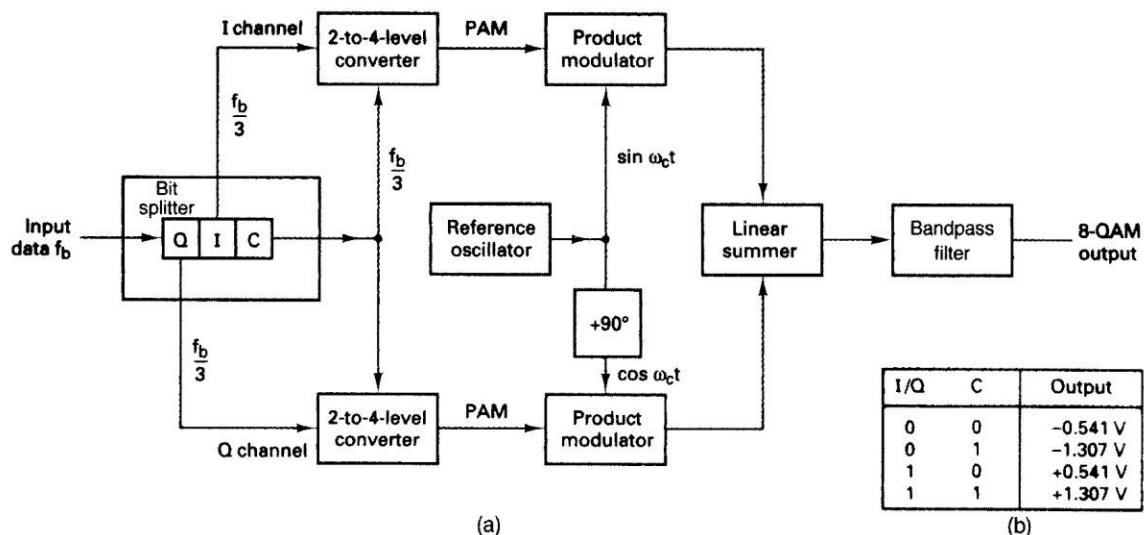


FIGURE 17: 8-QAM transmitter: (a) block diagram; (b) truth table 2-4 level converters

For a tritbit input of  $Q = 0$ ,  $I = 0$ , and  $C = 0$  (000), determine the output amplitude and phase for the 8-QAM transmitter shown in Figure 2-30a.

The inputs to the I channel 2-to-4-level converter are  $I = 0$  and  $C = 0$ . From Figure 2-30b, the output is  $-0.541$  V. The inputs to the Q channel 2-to-4-level converter are  $Q = 0$  and  $C = 0$ . Again from Figure 17b, the output is  $-0.541$  V.

Thus, the two inputs to the I channel product modulator are  $-0.541$  and  $\sin \omega_c t$ . The output is  $I =$

$$(-0.541)(\sin \omega_c t) = -0.541 \sin \omega_c t.$$

The two inputs to the Q channel product modulator are  $-0.541$  and  $\cos \omega_c t$ . The output is  $Q =$

$$(-0.541)(\cos \omega_c t) = -0.541 \cos \omega_c t.$$

The outputs from the I and Q channel product modulators are combined in the linear summer and produce a modulated output of

$$\begin{aligned} \text{summer output} &= -0.541 \sin \omega_c t - 0.541 \cos \omega_c t \\ &= 0.765 \sin(\omega_c t - 135^\circ) \end{aligned}$$

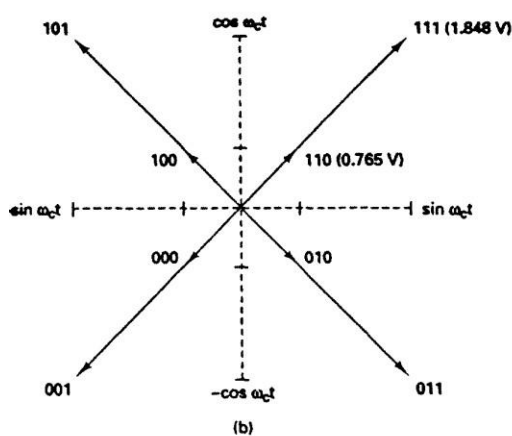
For the remaining tritbit codes (001, 010, 011, 100, 101, 110, and 111), the procedure is the same.

The results are shown in Figure 18.

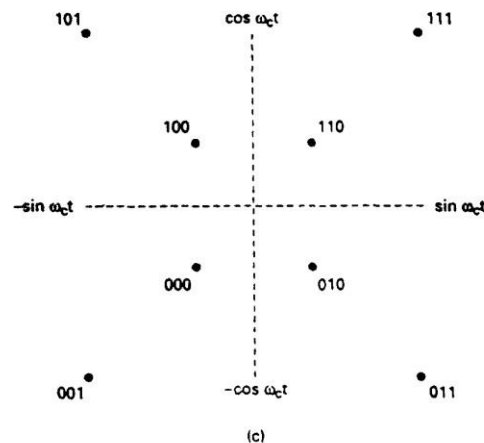
Figure 2-32 shows the output phase-versus-time relationship for an 8-QAM modulator. Note that there are two output amplitudes, and only four phases are possible.

| Binary input |   |   | 8-QAM output |       |
|--------------|---|---|--------------|-------|
| Q            | I | C | Amplitude    | Phase |
| 0            | 0 | 0 | 0.765 V      | -135° |
| 0            | 0 | 1 | 1.848 V      | -135° |
| 0            | 1 | 0 | 0.765 V      | -45°  |
| 0            | 1 | 1 | 1.848 V      | -45°  |
| 1            | 0 | 0 | 0.765 V      | +135° |
| 1            | 0 | 1 | 1.848 V      | +135° |
| 1            | 1 | 0 | 0.765 V      | +45°  |
| 1            | 1 | 1 | 1.848 V      | +45°  |

(a)



(b)



(c)

FIGURE 18:8-QAM modulator: (a) truth table; (b) phasor diagram; (c) constellation diagram

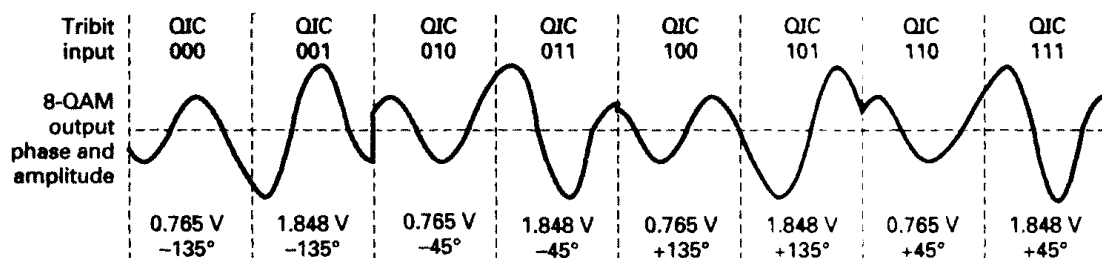


FIGURE 19 Output phase and amplitude-versus-time relationship for 8-QAM

## BANDWIDTH EFFICIENCY

*Bandwidth efficiency* (sometimes called *information density* or *spectral efficiency*, often used to compare the performance of one digital modulation technique to another.

Mathematical bandwidth efficiency is

$$B\eta = \frac{\text{transmission bit rate (bps)}}{\text{minimum bandwidth (Hz)}} = \frac{\text{bits / s}}{\text{Hertz}} \quad (18)$$

Where  $B\eta$  = bandwidth efficiency

---

## DIFFERENTIAL PHASE-SHIFT KEYING

*Differential phase-shift keying (DPSK)* is an alternative form of digital modulation where the binary input information is contained in the difference between two successive signaling elements rather than the absolute phase.

### Differential PSK

#### DPSK transmitter.

Figure 2-37a shows a simplified block diagram of a *differential binary phase-shift keying (DBPSK)* transmitter. An incoming information bit is XNORed with the preceding bit prior to entering the BPSK modulator (balanced modulator).

For the first data bit, there is no preceding bit with which to compare it. Therefore, an initial reference bit is assumed. Figure 2-37b shows the relationship between the input data, the XNOR output data, and the phase at the output of the balanced modulator. If the initial reference bit is assumed a logic 1, the output from the XNOR circuit is simply the complement of that shown.

In Figure 2-37b, the first data bit is XNORed with the reference bit. If they are the same, the XNOR output is a logic 1; if they are different, the XNOR output is a logic 0. The balanced modulator operates the same as a conventional BPSK modulator; a logic 1 produces  $+\sin \omega_c t$  at the output, and a logic 0 produces  $-\sin \omega_c t$  at the output.

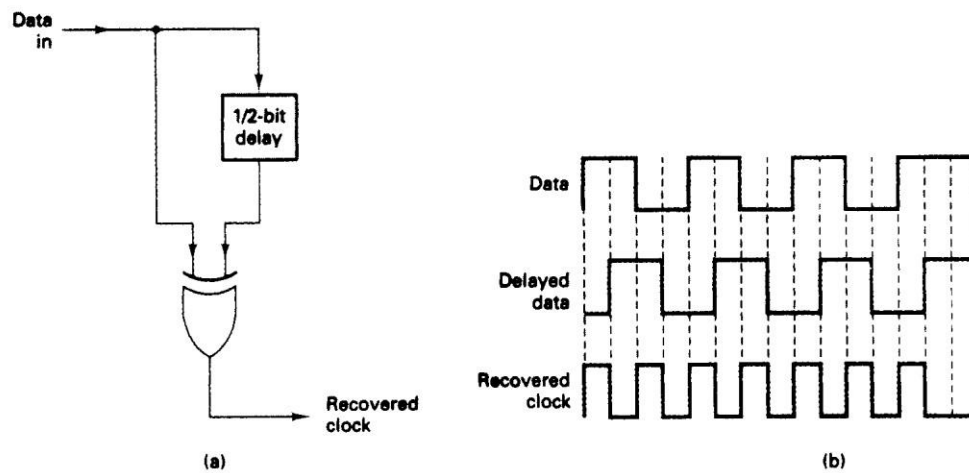


FIGURE 9-40 (a) Clock recovery circuit; (b) timing diagram

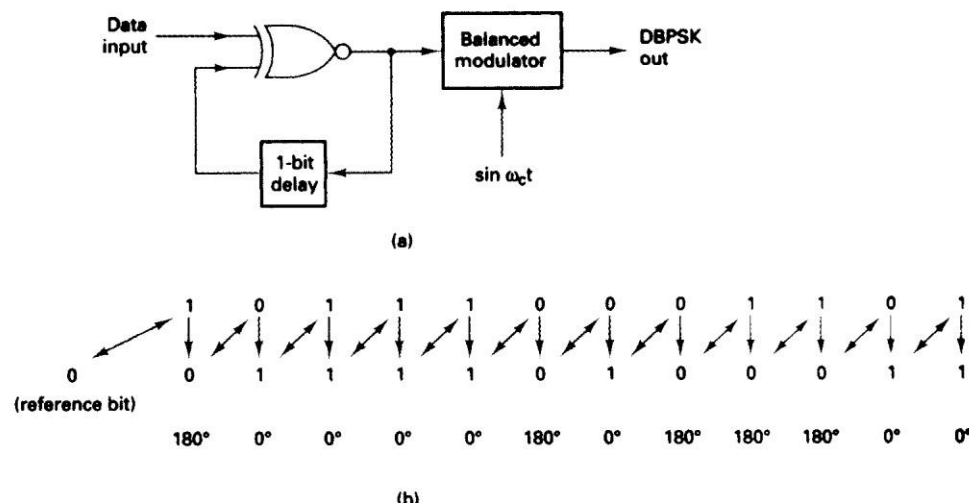


FIGURE 20 DBPSK modulator (a) block diagram (b) timing diagram

### DPSK receiver.

Figure 20 shows the block diagram and timing sequence for a DBPSK receiver. The received signal is delayed by one bit time, then compared with the next signaling element in the balanced modulator. If they are the same, a logic 1 (+ voltage) is generated. If they are different, a logic 0 (- voltage) is generated. [If the reference phase is incorrectly assumed, only the first demodulated bit is in error. Differential encoding can be implemented with higher-than-binary digital modulation schemes, although the differential algorithms are much more complicated than for DBPSK.]

The primary advantage of DBPSK is the simplicity with which it can be implemented. With DBPSK, no carrier recovery circuit is needed. A disadvantage of DBPSK is, that it requires between 1 dB and 3 dB more signal-to-noise ratio to achieve the same bit error rate

as that of absolute PSK.

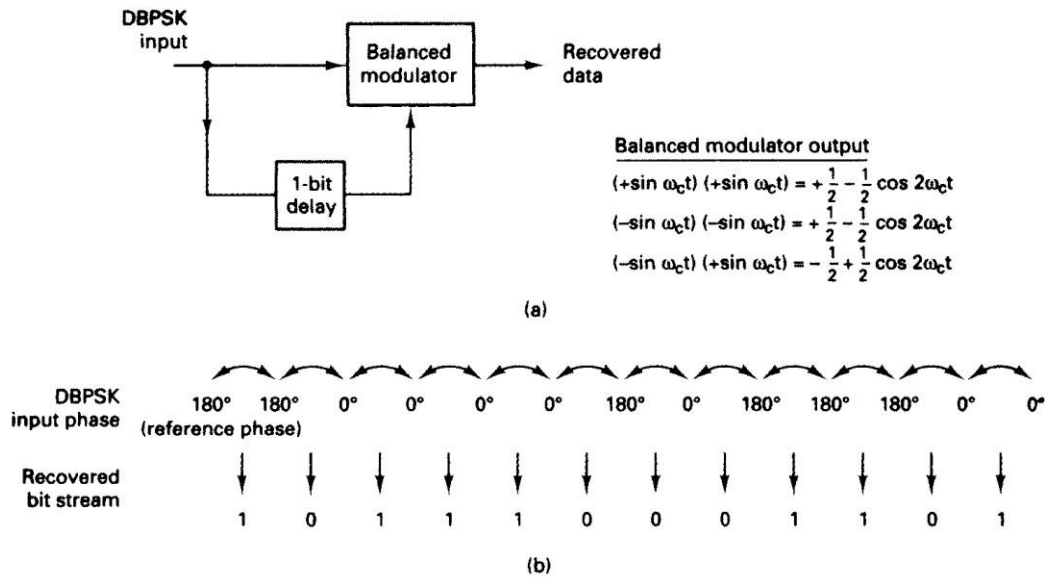


FIGURE 21:DBPSK demodulator: (a) block diagram; (b) timing sequence

## ERROR PERFORMANCE

### PSK Error Performance

The bit error performance is related to the distance between points on a signal state-space diagram.

For example, on the signal state-space diagram for BPSK shown in Figure 21, it can be seen that the two signal points (logic 1 and logic 0) have maximum separation ( $d$ ) for a given power level ( $D$ ).

The figure shows, a noise vector ( $V_N$ ), when combined with the signal vector ( $V_s$ ), effectively shifts the phase of the signaling element ( $V_{SE}$ ) alpha degrees.

If the phase shift exceeds  $+90^\circ$ , the signal element is shifted beyond the threshold points into the error region.

For BPSK, it would require a noise vector of sufficient amplitude and phase to produce more than a  $\pm 90^\circ$  phase shift in the signaling element to produce an error.

For PSK systems, the general formula for the threshold points is

$$TP = \pm \pi / M \quad (19)$$

where  $M$  is the number of signal states.

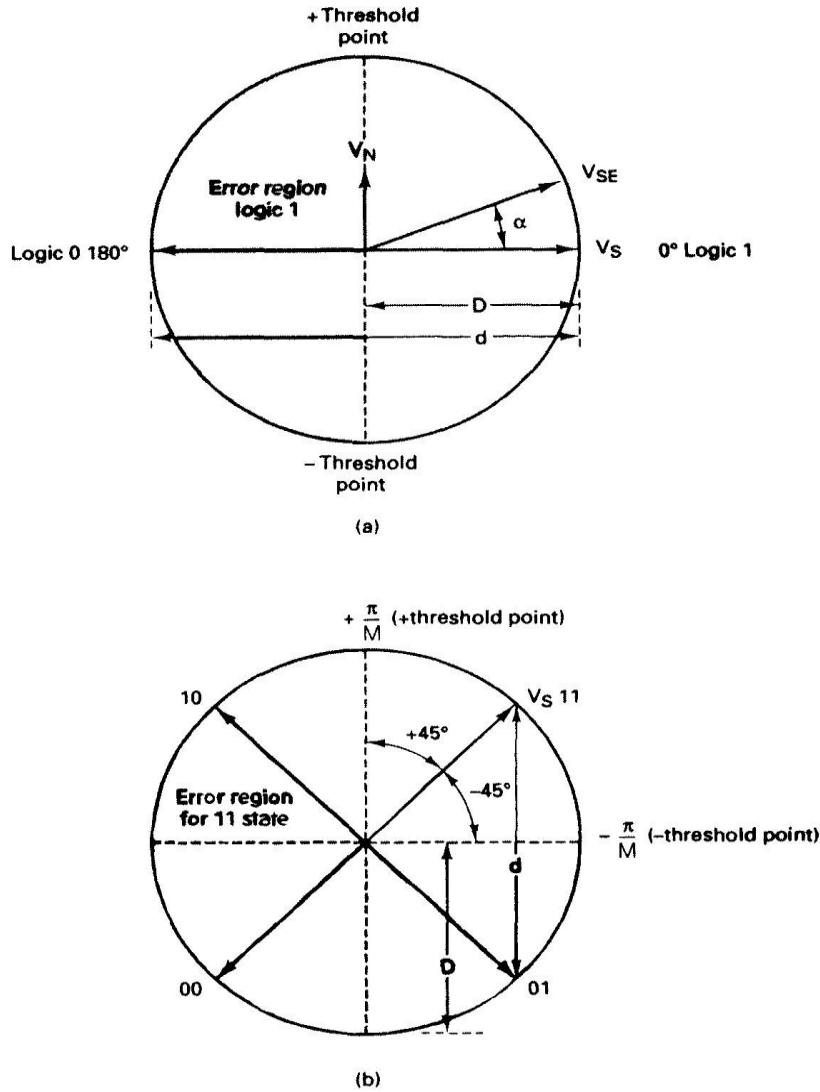


FIGURE 21: PSK error region: (a) BPSK; (b) QPSK

The phase relationship between signaling elements for BPSK (i.e.,  $180^\circ$  out of phase) is the optimum signaling format, referred to as *antipodal signaling*, and occurs only when two binary signal levels are allowed and when one signal is the exact negative of the other. Because no other bit-by-bit signaling scheme is any better, antipodal performance is often used as a reference for comparison.

The error performance of the other multiphase PSK systems can be compared with that of BPSK simply by determining the relative decrease in error distance between points on a signal state-space diagram.

For PSK, the general formula for the maximum distance between signaling points is given by

$$\sin \theta = \sin 360^\circ / 2M = (d/2) / D \quad (20)$$

$d$  = error distance  
 $M$  = number of phases

$D$  = peak signal amplitude

Rearranging equation 20 and solving for  $d$  yields

$$d = \left( 2 \sin \frac{180^\circ}{M} \right) \times D \quad (21)$$

Figure 21b shows the signal state-space diagram for QPSK. From Figure 21 and Equation 21, it can be seen that QPSK can tolerate only a  $\pm 45^\circ$  phase shift.

From Equation 21 the maximum phase shift for 8-PSK and 16-PSK is  $\pm 22.5^\circ$  and  $\pm 11.25^\circ$ , respectively.

The higher the level of modulation, the smaller the angular separation between signal points and the smaller the error distance.

The general expression for the bit error probability of an  $M$ -phase PSK system is

$$P(e) = (1 / \log_2 M) \operatorname{erf}(z) \quad (22)$$

where  $\operatorname{erf}$  = error function

$$z = \sin(\pi/M) \left( \sqrt{\log_2 M} \right) \left( \sqrt{E_b / N_0} \right)$$

By substituting into Equation 22, it can be shown that QPSK provides the same error performance as BPSK. This is because the 3-dB reduction in error distance for QPSK is offset by the 3-dB decrease in its bandwidth (in addition to the error distance, the relative widths of the noise bandwidths must also be considered).

Thus, both systems provide optimum performance. Figure 2-40 shows the error performance for 2-, 4-, 8-, 16-, and 32-PSK systems as a function of  $E_b / N_0$ .

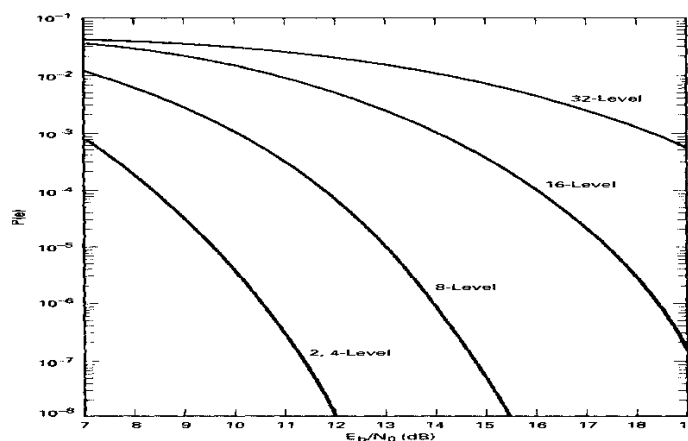


FIGURE 22 Error rates of PSK modulation systems



## QAM Error Performance

For a large number of signal points (i.e., M-ary systems greater than 4), QAM outperforms PSK. This is because the distance between signaling points in a PSK system is smaller than the distance between points in a comparable QAM system. The general expression for the distance between adjacent signaling points for a QAM system with L levels on each axis is

$$d = \frac{\sqrt{2}}{L-1} D \quad (23)$$

where  $d$  = error distance

$L$  = number of levels on each axis

$D$  = peak signal amplitude

In comparing Equation 23 to Equation 22, it can be seen that QAM systems have an advantage over PSK systems with the same peak signal power level. The general expression for the bit error probability of an L-level QAM system is

$$P(e) = \frac{1}{\log_2 L} \left( \frac{L-1}{L} \right) \text{erfc}(z) \quad (24)$$

Where  $\text{erfc}(z)$  is the complementary error function

Figure 23 shows the error performance for 4-, 16-, 32-, and 64-QAM systems as a function of  $E_b/N_0$ .

Table lists the minimum carrier-to-noise power ratios and energy per bit-to-noise power density ratios required for a probability of error  $10^{-6}$  for several PSK and QAM modulation schemes.

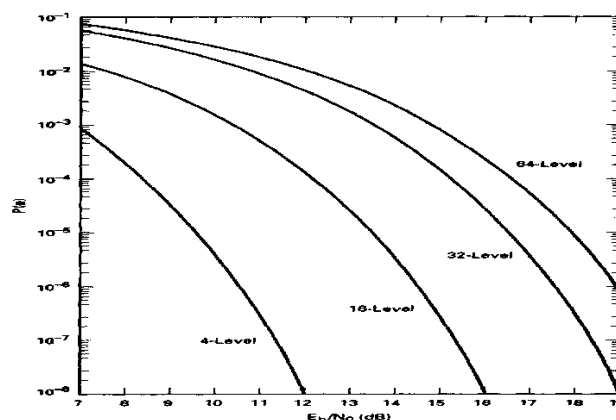


FIGURE 23 Error rates of QAM modulation systems.

Performance Comparison of various digital modulation schemes (BER = 10<sup>-6</sup>)

| Modulation Technique | C/N Ratio (dB) | E <sub>b</sub> /N <sub>0</sub> Ratio (dB) |
|----------------------|----------------|---|
| BPSK                 | 10.6           | 10.6                                      |
| QPSK                 | 13.6           | 10.6                                      |
| 4-QAM                | 13.6           | 10.6                                      |
| 8-QAM                | 17.6           | 10.6                                      |
| 8-PSK                | 18.5           | 14  |
| 16-PSK               | 24.3           | 18.3                                      |
| 16-QAM               | 20.5           | 14.5                                      |
| 32-QAM               | 24.4           | 17.4                                      |
| 64-QAM               | 26.6           | 18.8                                      |

### FSK Error Performance

With noncoherent FSK, the transmitter and receiver are not frequency or phase synchronized. With coherent FSK, local receiver reference signals are in frequency and phase lock with the transmitted signals. The probability of error for noncoherent FSK is

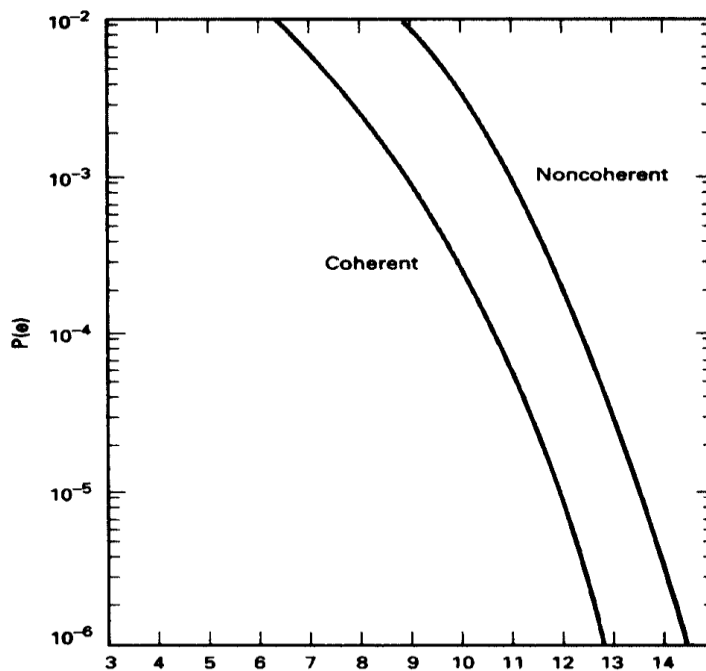


FIGURE 24 Error rates for FSK modulation systems

$$P(e) = 1/2 \exp\left(\frac{-E_b}{2N_o}\right) \tag{25}$$

The probability of error for coherent FSK is

$$P(e) = \text{erfc} \sqrt{\frac{E_b}{N_o}} \tag{26}$$

Figure 24 shows probability of error curves for both coherent and noncoherent FSK for several values of  $E_b/N_o$ . From Equations 25 and 26, it can be determined that the probability of error for noncoherent FSK is greater than that of coherent FSK for equal energy per bit-to-noise power density ratios.

## UNIT-III

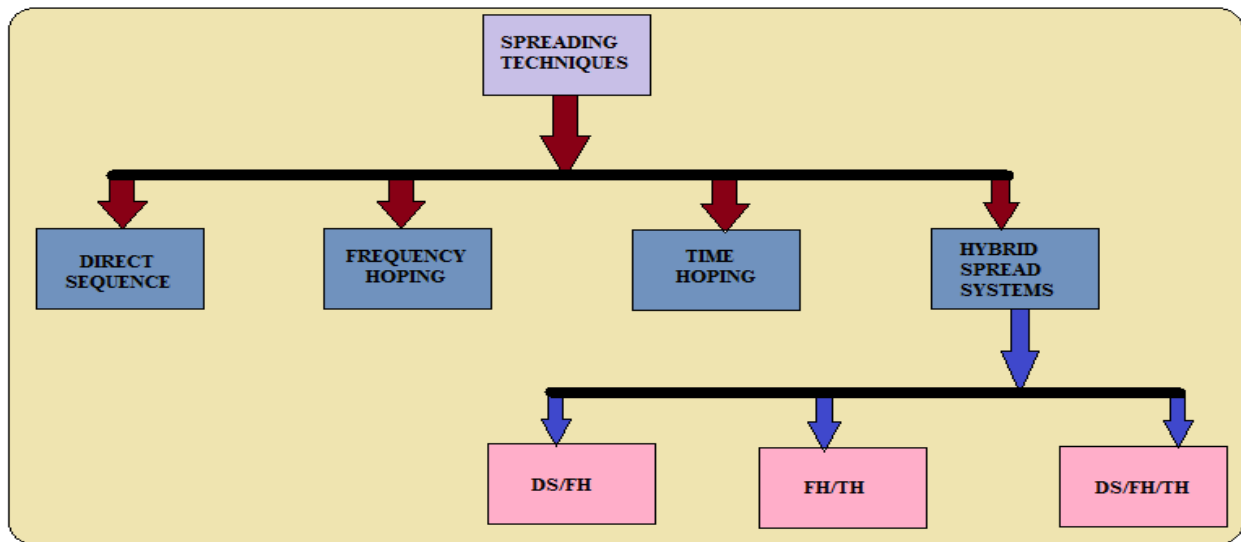
### SPREAD SPECTRUM COMMUNICATIONS

#### 1. Explain the Spread-Spectrum Techniques?

- ❖ Spread spectrum techniques were initially developed for military applications (guidance and communication systems), where resistance to jamming (interference) is of major concern.
- ❖ Spread spectrum investigation was motivated primarily by the desire to achieve highly jam-resistance communication systems.
- ❖ The primary advantage of the spread spectrum communication system is its ability to reject interference whether it be the unintentional interference by another user simultaneously attempting to transmit through the channel, or the intentional interference by a hostile transmitter attempting to jam the transmission.
- ❖ In spread spectrum technique, the transmission bandwidth employed is much greater than the minimum bandwidth required to send the information.
- ❖ A system is defined to be spread spectrum system if it fulfills the following requirements.
  1. The signal occupies a bandwidth much in excess of the minimum bandwidth necessary to send the information.
  2. The spectrum spreading is accomplished before transmission through the use of a code, called spreading signal or code signal which is independent of the data sequence.
  3. The same code is used in the receiver to despread the received signal so that the original data sequence may be recovered.
- ❖ Spread spectrum techniques can be mainly classified into four groups.
  - Direct Sequence
  - Frequency Hopping
  - Time Hopping
  - Hybrid spread spectrum techniques
- ❖ In a **direct sequence** spread spectrum technique, two stages of modulation are used, First, the incoming data sequence is used to modulate a wideband code. This code transforms

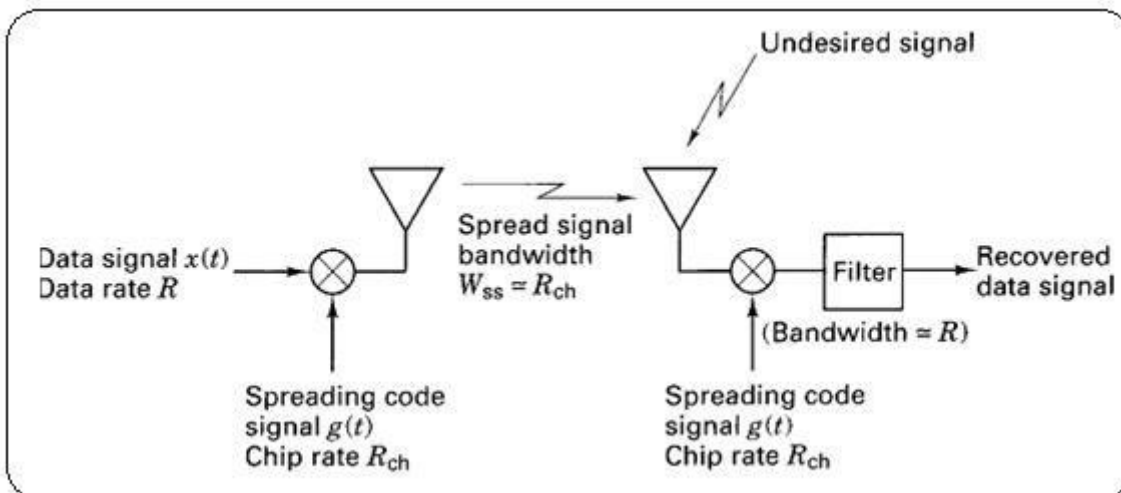
the narrow band data sequence into a noise like wideband signal. The resulting wide band signal undergoes a second modulation using a phase shift keying technique.

- ❖ In **frequency – hop**(FH) spread spectrum technique, the spectrum of data modulated carrier is widened by changing the carrier frequency in a pseudo random manner.
- ❖ In **Time – hop** spread spectrum technique, a message with data rate  $R$  is allocated a longer transmission time duration than would be used with a conventional modulation scheme. In time hopping, the signal is spread in time domain.



SPREAD SPECTRUM TECHNIQUES

**BASIC SPREAD SPECTRUM TECHNIQUE:**



BASIC SPREAD SPECTRUM TECHNIQUE

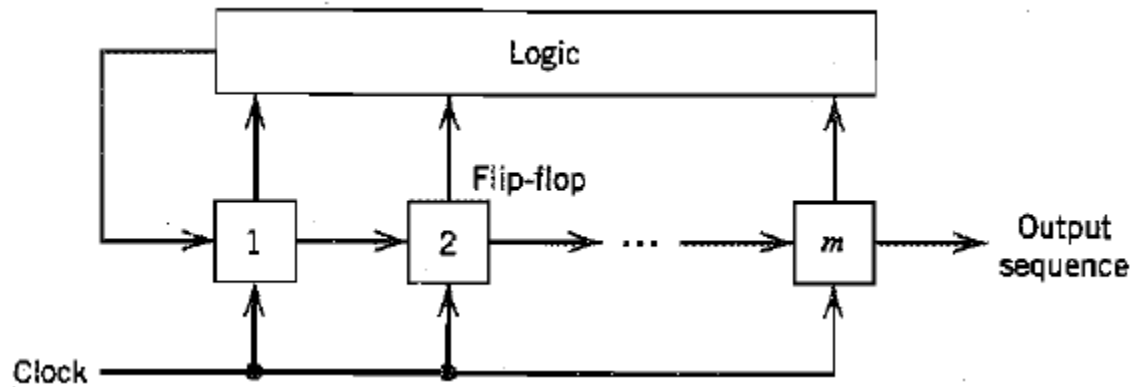
- ❖ At the modulator, the information signal  $x(t)$ , with a data rate of  $R$  bits/seconds is multiplied by a spreading signal  $g(t)$ , having a code symbol rate usually called the code chip rate  $R_{ch}$  chips/seconds.
- ❖ Assume the transmission bandwidths for  $x(t)$  and  $g(t)$  are  $R$  hertz and  $R_{ch}$  hertz respectively.
- ❖ Multiplication in time domain transforms to convolution in the frequency domain.

$$x(t) g(t) = x(w) * g(w)$$

- ❖ At the demodulator, the received signal is ideally multiplied by a synchronized replica of the spreading signal  $g(t)$ , which results in the despreading of the signal.
- ❖ A filter with bandwidth  $R$  is used to remove any spurious frequency components.
- ❖ Multiplication by the spreading signal twice followed by filtering recovers the original signal.

## 2. PSEUDO NOISE SEQUENCES (PN-Sequences)?

- ❖ Pseudo noise sequence is a binary sequence used in spread spectrum communication for spreading the message signal.
- ❖ These coded sequences are generated independently at two or more sites, and this sequence must be deterministic, even though it should appear random to unauthorized listeners.
- ❖ Such random appearing deterministic signals are called pseudo noise or pseudo random signals.
- ❖ A pseudo noise sequence is a periodic binary sequence with a noise like waveform that is usually generated by means of a feedback shift register, a general block diagram of which is shown below.



- ❖ The flip flops in the shift register are regulated by a single timing clock.
- ❖ At each clock pulse, the state of the flip flop is shifted to the next one down the line.

### ❖ Properties of pseudo noise signals

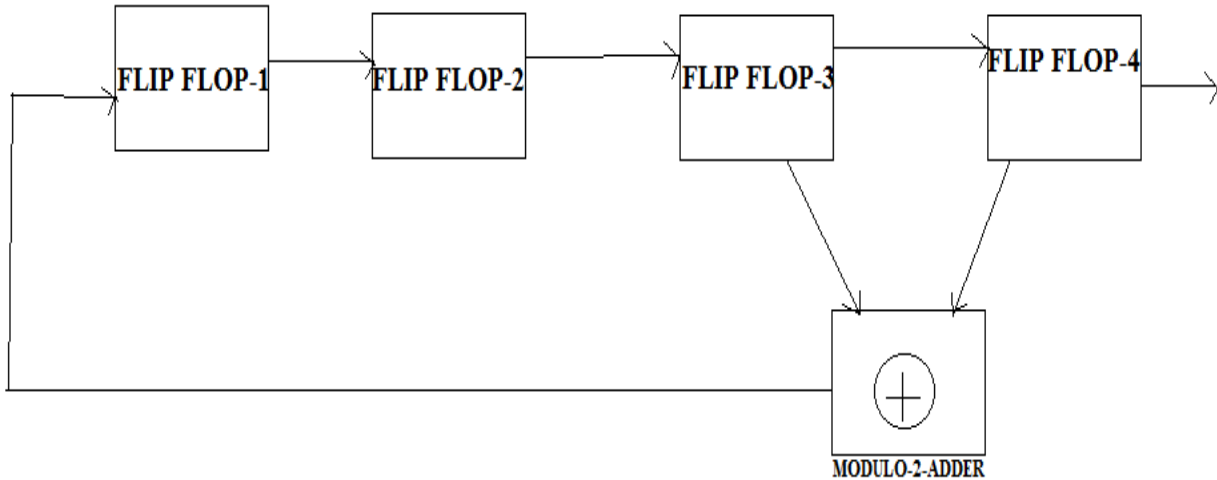
The randomness properties that make a pseudo random signal appear truly random are

- **BALANCE PROPERTY:** Good balance requires that in each period of the sequence the number of binary ones differs from the binary zeros by at most one digit.
- **RUN PROPERTY:** a run is defined as a sequence of a single type of binary digits. The appearance of alternate digit in a sequence starts a new run. The length of the run is the number of digits in the run.
- **CORRELATION PROPERTY:** if a period of a sequence is compared term by term with any cyclic shift of itself, it is best if the number of agreements differs from the number of disagreements by not more than one count.

$$\mathbf{Re(x)} = \frac{(Na - Nd)}{N} = -1/p$$

## SHIFT REGISTER BASED PN SEQUENCE EXAMLE:

Consider a linear feedback shift register which is shown below.



- ❖ A feedback shift register is said to be linear when the feedback logic consist entirely of modulo -2 adders.
- ❖ The shift register operation is controlled by a sequence of clock pulses, which is not shown in the circuit diagram.
- ❖ At each clock pulse, the content of each flip flop is shifted one stage to the right.
- ❖ Also at each clock pulse, the contents of flip flops 3 and 4 are modulo 2- added and the result is feedback to the flip flop 1.
- ❖ Assume the flip flop-1 is initially filled with a '1' and the remaining stages are filled with zeros. That is the initial state of the shift register is 1000.
- ❖ The next states are given below.
- ❖ 1000 0100 0010 1001 1100 0110 1011 0101 1010 1101 1110 1111  
0111 0011 0001 1000
- ❖ The shift register repeats the forgoing sequence after 15 clock pulses.
- ❖ The output sequence is obtained by noting the contents of flip flop 4 at each clock pulse.
- ❖ The output sequence is 000100110101111.



## TESTING THE RANDOM PROPERTIES OF THIS SEQUENCE

- **BALANCE PROPERTY:** There are seven zeros and eight ones, it meets the balance condition.
- **RUN PROPERTY:** there are four zero and one runs.
- **CORRELATION PROPERTY:**

|          |          |          |          |          |          |          |          |          |          |          |          |          |          |          |
|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|
| 0        | 0        | 0        | 1        | 0        | 0        | 1        | 1        | 0        | 1        | 0        | 1        | 1        | 1        | 1        |
| <b>1</b> | <u>0</u> | <u>0</u> | <u>0</u> | <u>1</u> | <u>0</u> | <u>0</u> | <u>1</u> | <u>1</u> | <u>0</u> | <u>1</u> | <u>0</u> | <u>1</u> | <u>1</u> | <u>1</u> |
|          | d        | a        | a        | d        | d        | a        | d        | a        | d        | ddd      | a        | aa       |          |          |

$$\text{Re}(x) = \frac{(Na - Nd)}{N} = -1/p$$

$$= \frac{1(7-8)}{15} = -1/15 \text{ it meets the correlation property.}$$

### (3). DIRECT SEQUENCE SPREAD SPECTRUM?

- ❖ Direct sequence is the name given to the spectrum spreading technique whereby a carrier wave is first modulated with a data signal  $x(t)$ , then the data modulated signal is again modulated with a high speed wide band spreading signal  $g(t)$ .
- ❖ Consider a constant envelop data modulated carrier having power,  $P$ , angular frequency  $w_0$  and data phase modulation  $\theta_x(t)$ , given by

$$S_x(t) = \sqrt{2p} \cos[w_0 t + \theta_x(t)] \quad (1)$$

- ❖ After modulating the signal,  $S_x(t)$  by the spreading signal  $g(t)$ , the transmitted waveform can be expressed as

$$S(t) = \sqrt{2p} \cos[w_0(t) + \theta_x(t) + \theta_g(t)] \quad (2)$$

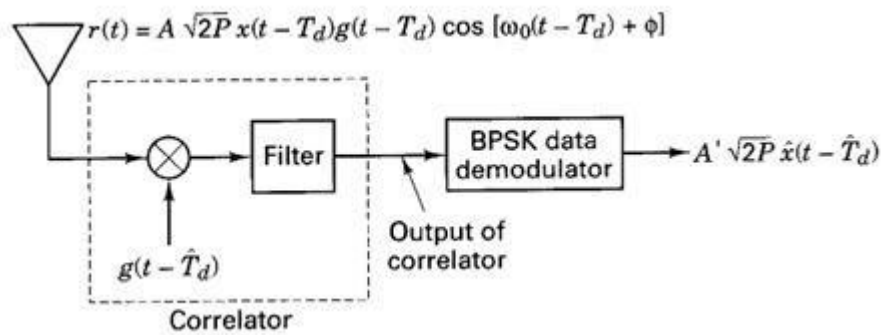
- ❖ On the other hand equations (1) and (2) can be written as

$$S_X(t) = \sqrt{2px(t)} \cos \omega_0 t \quad (3)$$

$$S(t) = \sqrt{2p}x(t)g(t) \cos\omega_0 t \quad (4)$$

| Pulse value | Binary value |
|-------------|--------------|
| 1           | 0            |
| -1          | 1            |

### BPSK DIRECT SEQUENCE TRANSMITTER (MODULATOR)



### BPSK DIRECT SEQUENCE RECEIVER (DEMODULATOR)

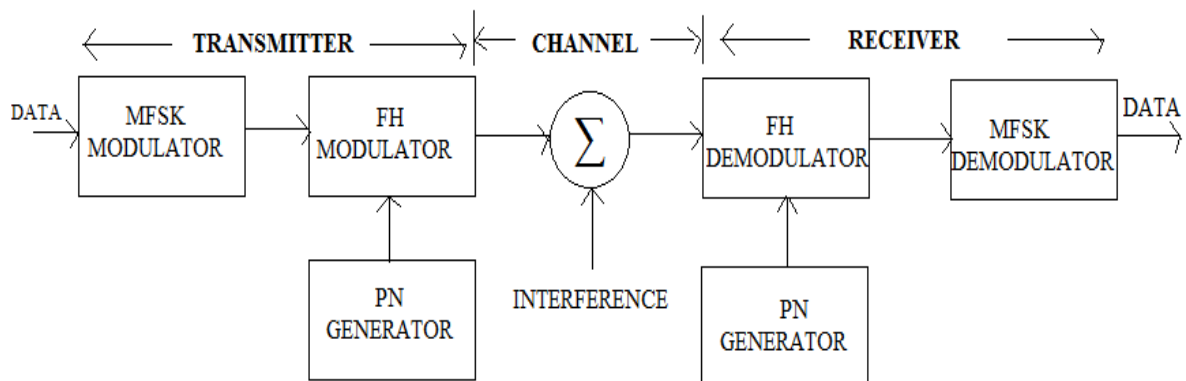
- ❖ In the absence of noise and interference, the output signal from the correlator can be written as

$$r(t) = A\sqrt{2p}x(t - T_d)g(t - \hat{T}_d)\cos[\omega_0(t - T_d) + \phi]$$

Where A is the system gain parameter and  $\phi$  is the random phase angle,  $\hat{T}_d$  system's estimate of propagation delay  $T_d$  from the transmitter to the receiver.

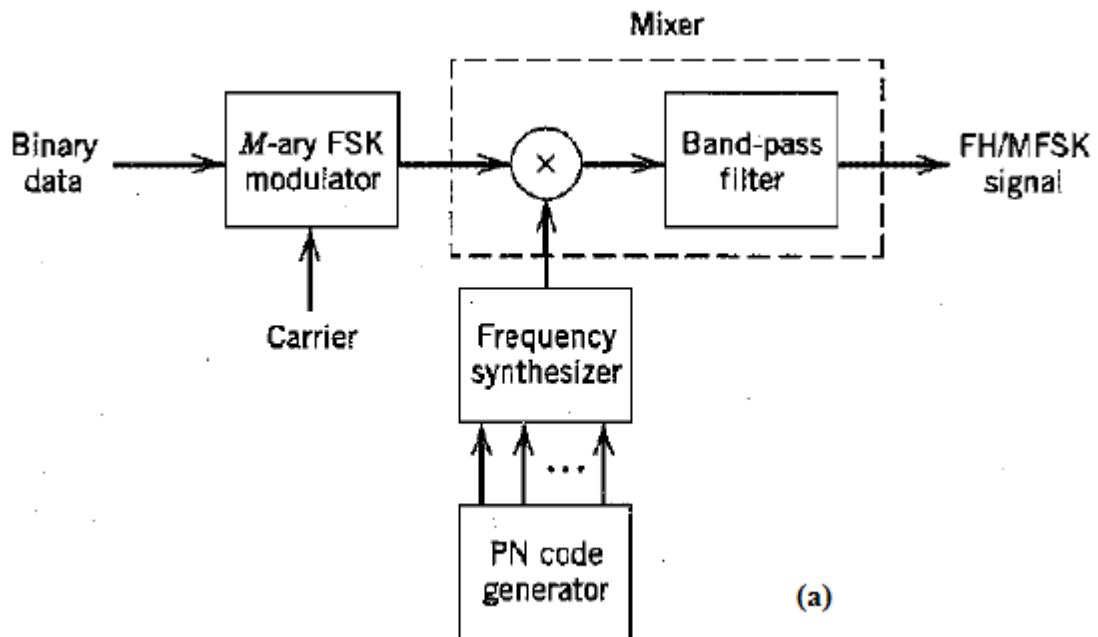
#### 4. FREQUENCY HOPPING SPREAD SPECTRUM SYSTEMS

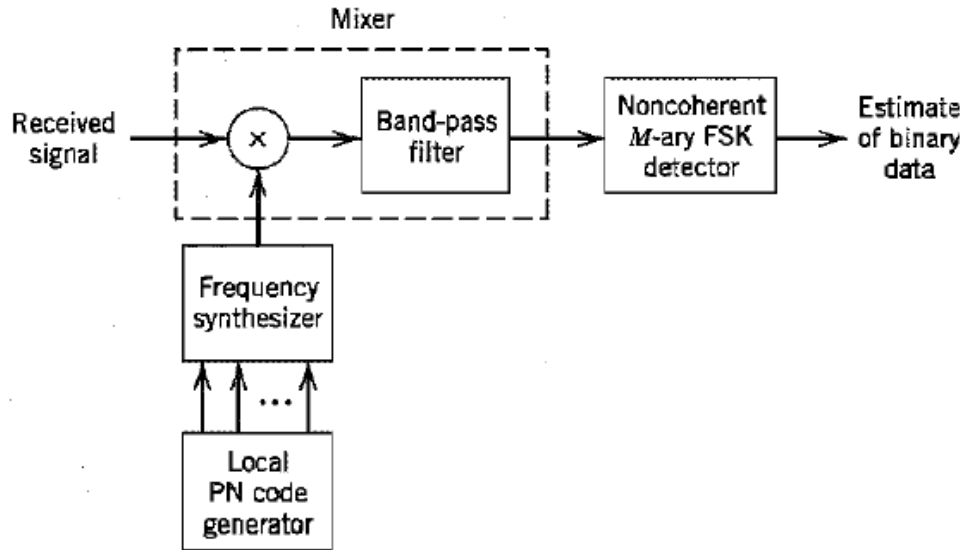
- ❖ The type of spread spectrum in which the carrier hops randomly from one frequency to another is called frequency-hop spread spectrum.
- ❖ A common modulation format for FH systems is that of M-ary frequency shift keying (MFSK).
- ❖ The combination of these two techniques is referred as FH/MFSK, where  $k = \log_2 M$  information bits are used to determine which one of M-frequencies is to be transmitted.
- ❖ The position of the M-ary signal set is shifted pseudo randomly by the frequency synthesizer over a hopping bandwidth  $W_{ss}$ .
- ❖ A typical FH/MFSK system block diagram is shown below.



## FH/MFSK SYSTEM-(BLOCK DIAGRAM)

- ❖ In a conventional MFSK system, the data symbol modulates a fixed frequency carrier.
- ❖ In a FH/MFSK system, the data symbol modulates a carrier whose frequency is pseudo randomly determined.
- ❖ An FH/MFSK system has two step modulation processes.
  1. Data modulation
  2. Frequency hopping modulation
- ❖ The detailed block diagram of a FH/MFSK transmitter and receiver are shown below.
- ❖ In FH/MFSK transmitter, which involves frequency modulation followed by mixing, first, the incoming binary data are applied to an M-ary FSK modulator.
- ❖ The resulting modulated wave and the output from a digital frequency synthesizer are then applied to a mixer that consists of a multiplier followed by a band pass filter.
- ❖ The filter is designed to select the sum frequency component resulting from the multiplication process as the transmitted signal.



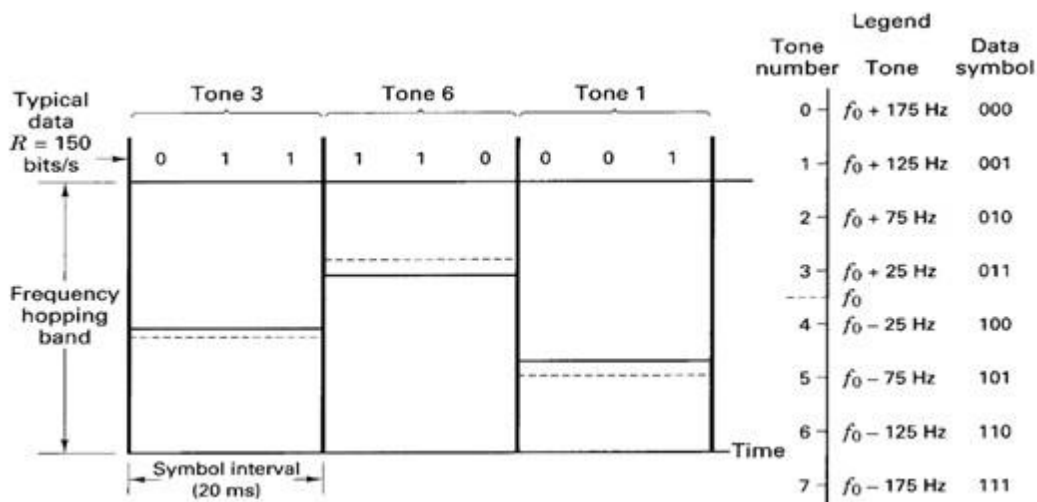


(b)

**Frequency-hop spread  $M$ -ary frequency-shift keying. (a) Transmitter. (b) Receiver.**

- ❖ In the receiver, the frequency hopping is first removed by mixing the received signal with the output of a local frequency synthesizer that is synchronously controlled in the same manner as that in the transmitter.
- ❖ The resulting output is then band pass filtered, and subsequently processed by non coherent matched filters, each of which is matched to one of the MFSK tones.

**❖ Frequency hopping example is shown below**

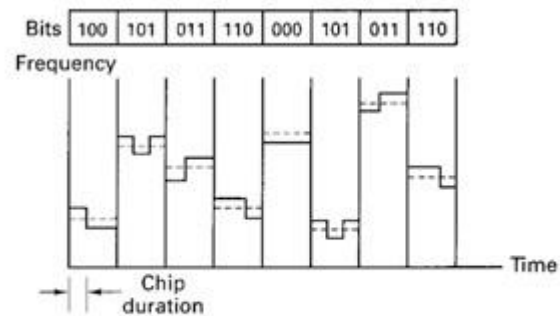


Frequency Hopping systems are classified into two groups.

1. Slow frequency hoping
2. Fast frequency hoping

❖ **Slow frequency hoping:**

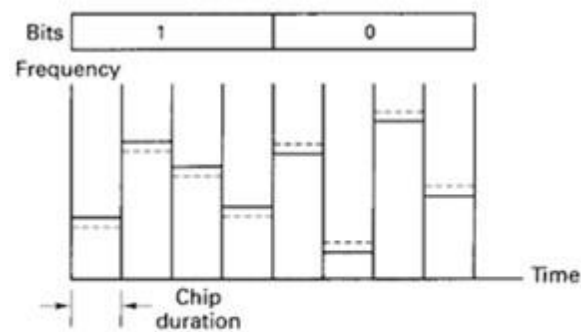
In **slow frequency hoping**, the symbol rate  $R_s$  of the MFSK signal is integer multiple of the hop rate  $R_b$ . That is, several symbols are transmitted on each frequency hop.



**Slow frequency hoping : 3Bits/hop**

❖ **Fast frequency hoping (FFH/MFSK):**

In fast frequency hoping, the hop rate  $R_b$  is an integer multiple of the MFSK symbol rate  $R_s$ . That is the carrier frequency will change or hop several times during the transmission of one symbol. FFH/MFSK process is illustrated in figure shown below



**FFH/MFSK 4hops/Bit**

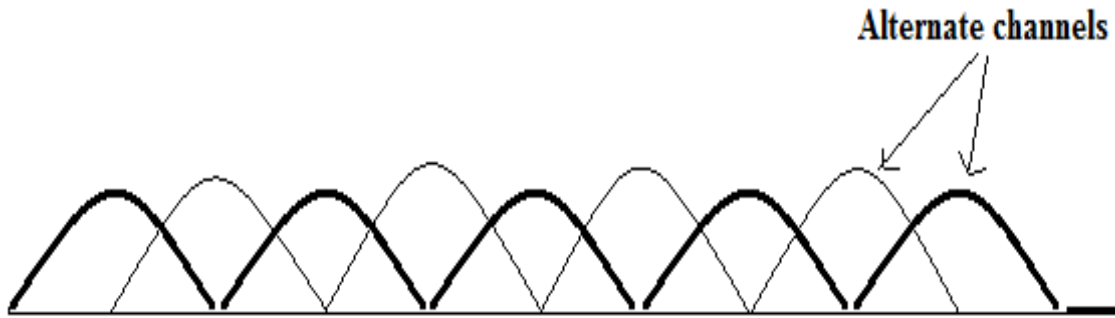
### HYBRID SPREAD SPECTRUM SYSTEMS

- ❖ Hybrid systems use the combinations of modulation that offer certain advantages over direct sequence and frequency hoping techniques.
- ❖ The most commonly used hybrid spread spectrum techniques are
  1. Frequency hopping/ Direct sequence modulation (FH/DS)

2. Time & Frequency hopping
3. Time hopping & Direct sequence modulation

**FREQUENCY HOPED/DIRECT SEQUENCE MODULATION**

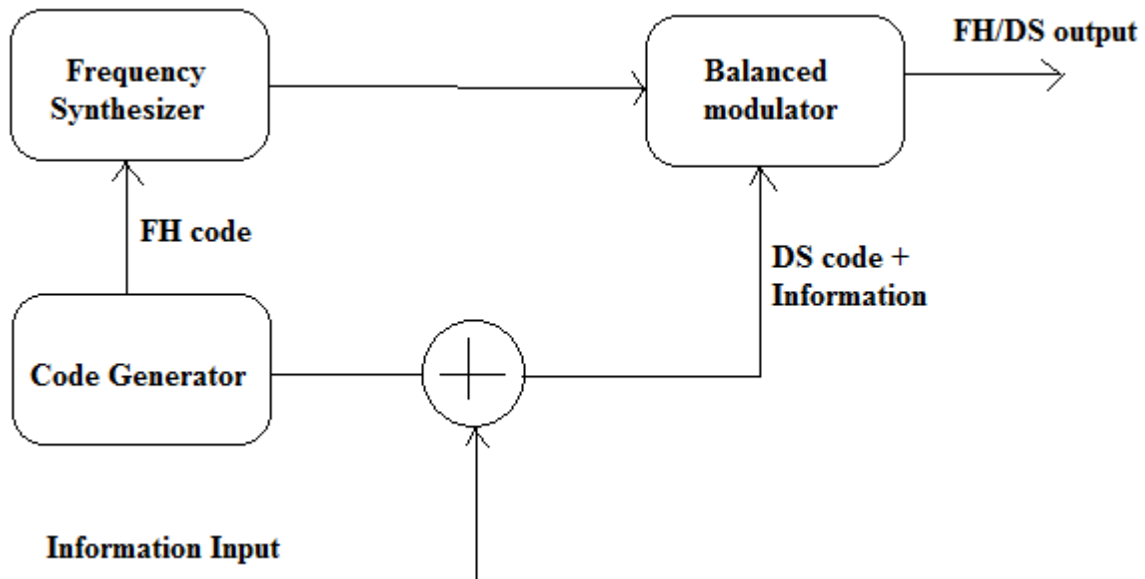
- ❖ Frequency hopped/Direct sequence modulation (FH/DS), consists of a direct sequence modulated signal whose center frequency hops periodically.



**Frequency spectrum of hybrid FH/DS system**

Hybrid FH/DS signals are used for various reasons such as

1. To extend spectrum spreading capability
2. For multiple access and discrete address
3. For multiplexing

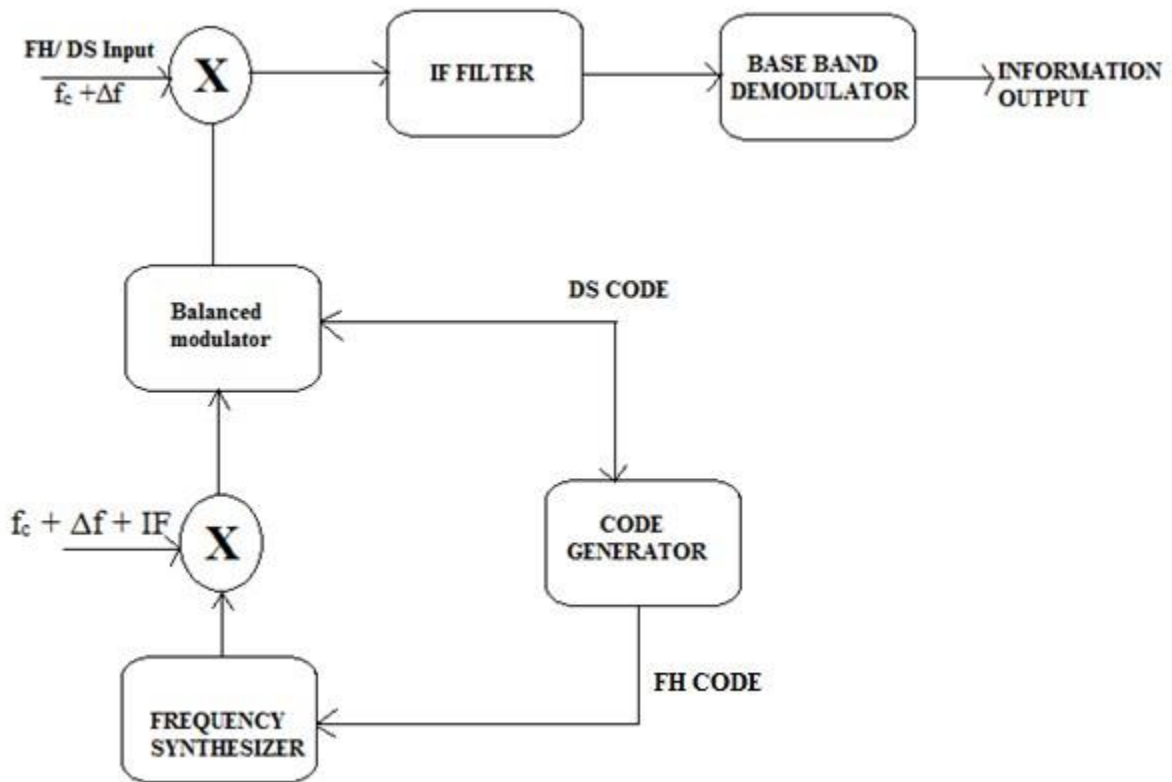


**HYBRID FH/DS MODULATOR**

- ❖ FH/DS transmitters are combinations of direct sequence modulation on a frequency hopping carrier.

- ❖ This modulator differ from a simple direct sequence modulator mainly in that the carrier frequency is varying (hopped) rather than being at a constant frequency, as for simple DS modulation.

$$\frac{BW_{RF}}{R_{INFO}} = FH / DS$$



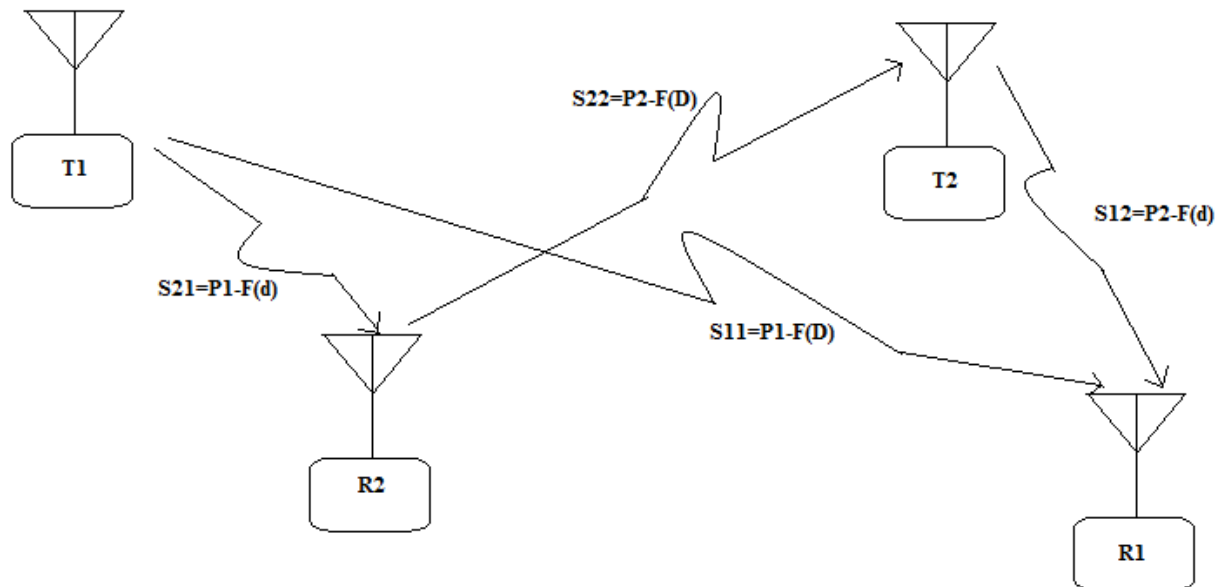
HYBRID FH/DS RECEIVER BLOCK DIAGRAM

### TIME FREQUENCY HOPING

- ❖ Time – Frequency hopping modulation has found its greatest application in those systems in which a large number of users with widely variable distances or transmitted power are to operate simultaneously in a single link.
- ❖ Such systems tend to employ simple coding.
- ❖ It can eliminate near far problem.
- ❖ The problem in this system lies in the difference in distance from a receiver to its desired transmitter and to the near by transmitter.



- ❖ A close transmitter would produce signals attenuated by 40 –plus dB the far transmitter would produce signals attenuated by 63 plus dB.
- ❖ A much better solution is to time all transmissions so that the desired and undesired transmitters are never transmitting at the same time.



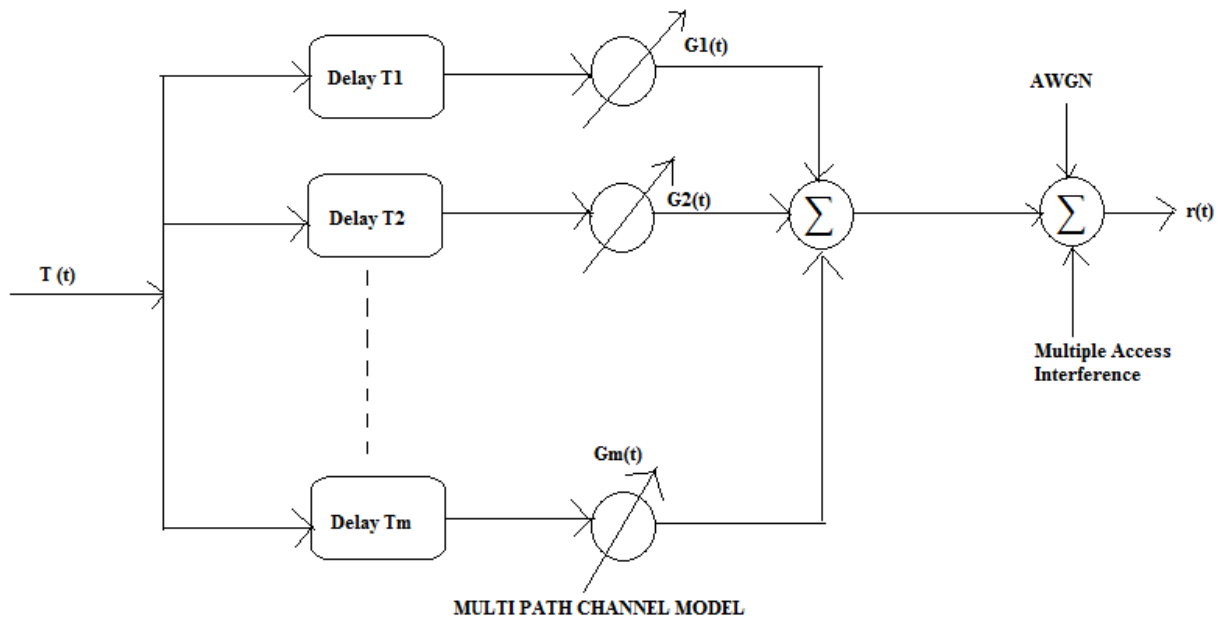
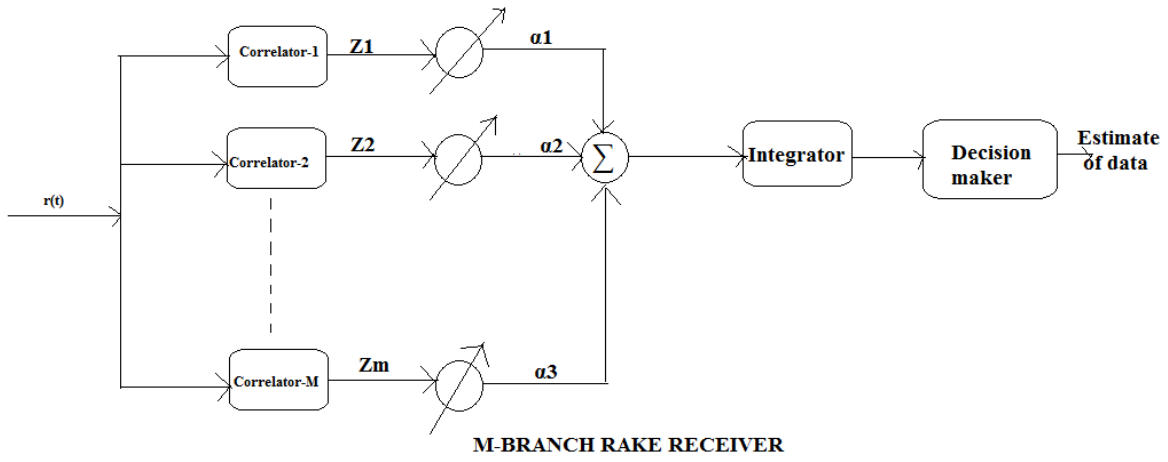
**TWO LINK COMMUNICATION SYSTEM ILLUSTRATING THE NEAR FAR PROBLEM**

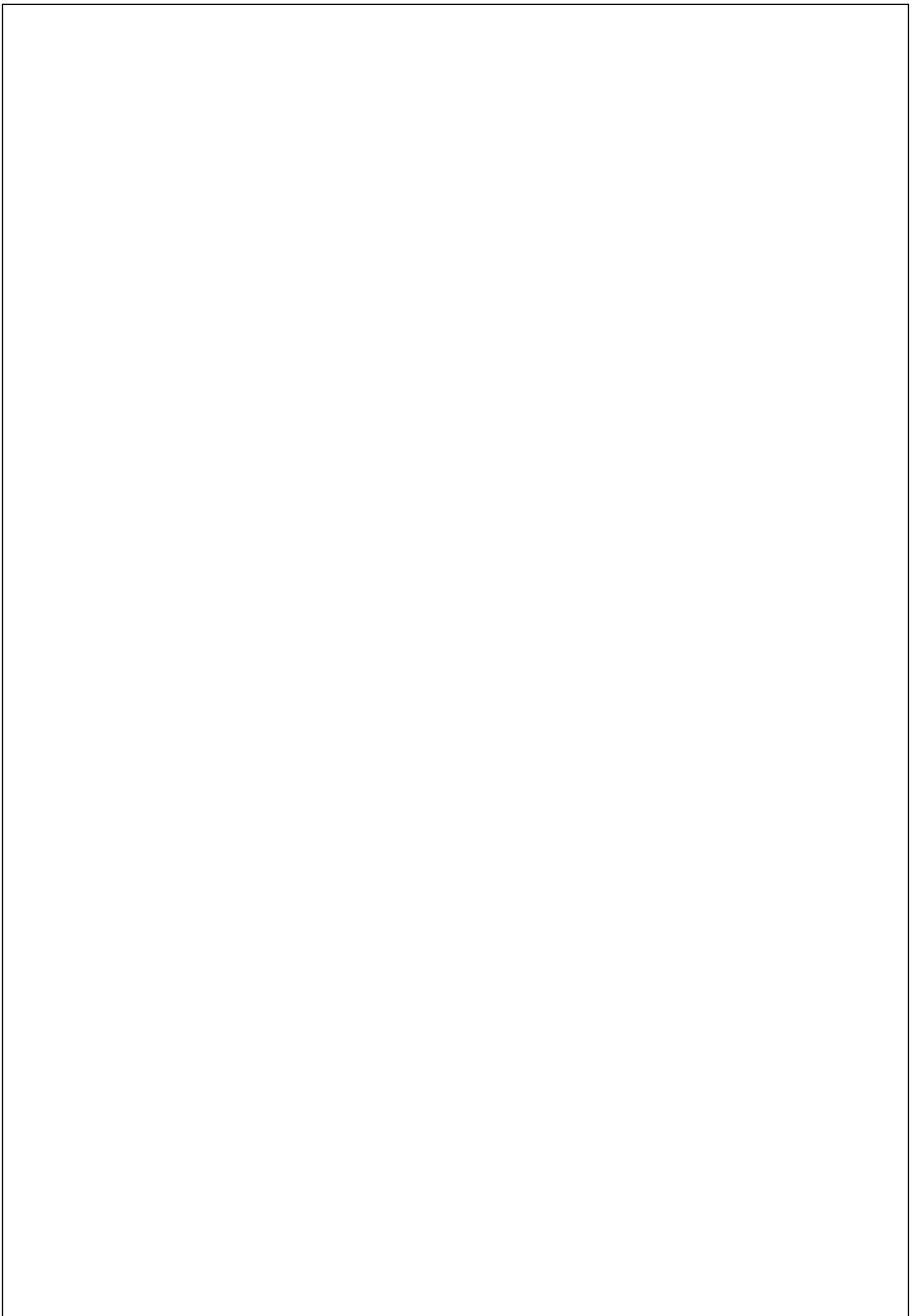
## **RAKE RECEIVERS**

- ❖ A rake receiver is a radio receiver designed to counter the effects of multipath fading.
- ❖ A rake receiver utilizes multiple correlators to separately detect M-strongest multipath components.
- ❖ Rake receiver is used in CDMA based systems and can combine multipath components, which are time delayed versions of the original signal transmission.
- ❖ Combining is done in order to improve the signal to noise ratio at the receiver.
- ❖ Due to reflections from the obstacles, a radio channel can consist of many copies of originally transmitted signals having different amplitudes, phases, and delays.
- ❖ Rake receiver attempts to collect the time shifted versions of the original signals by providing a separate correlation receiver for each of the multipath signals.
- ❖ The outputs of each correlator are weighed to provide better estimate of the transmitted signal than is provided by a single component.

- ❖ Demodulation and bit decisions are then based on the weighed outputs of the M-correlators.

❖





# UNIT IV – SYNCHRONIZATION

## Fundamentals of Synchronization

The analysis in previous chapters assumed that the modulator and demodulator are synchronized. That is, both modulator and demodulator know the exact symbol rate and the exact symbol phase, and where appropriate, both also know the exact carrier frequency and phase. In practice, the common (receiver/transmitter) knowledge of the same timing and carrier clocks rarely occurs unless some means is provided for the receiver to synchronize with the transmitter. Such synchronization is often called phase-locking.

### 1. Phase-error generation

This operation, sometimes also called “phase detection,” derives a phase difference between the received signal’s phase  $\theta(t)$  and the receiver estimate of this phase,  $\hat{\theta}(t)$ . The actual signals are  $s(t) = \cos(\omega t + \theta(t))$  and  $\hat{s}(t) = \cos(\omega t + \hat{\theta}(t))$ , but only their phase difference is of interest in synchronization. This difference is often called the phase error,  $\varphi(t) = \theta(t) - \hat{\theta}(t)$ .

### 2. Phase-error processing

This operation, sometimes also called “loop filtering” extracts the essential phase difference trends from the phase error by averaging. Phase-error processing typically rejects random noise and other undesirable components of the phase error signal. Any gain in the phase detector is assumed to be absorbed into the loop filter. Both analog and digital phase-error processing.

### 3. Local phase reconstruction

This operation, which in some implementations is known as a “voltage-controlled oscillator” (VCO), regenerates the local phase from the processed phase error in an attempt to match the incoming phase,  $\theta(t)$ . That is, the phase reconstruction tries to force  $\varphi(t) = 0$  by generation of a local phase  $\hat{\theta}(t)$  so that  $\hat{s}(t)$  matches  $s(t)$ .

## Symbol Timing Synchronization

Generally in data transmission, a sinusoid synchronized to the symbol rate is not supplied to the receiver. The receiver derives this sinusoid from the received data. Thus, the unabetted PLL’s studied so far would not be sufficient for recovering the symbol rate. The recovery of this symbol-rate sinusoid from the received channel signal, in combination with the PLL, is called timing recovery. There are two types of timing recovery. The first type is called open loop timing recovery and does not use the receiver’s decisions. The second type is called decision-directed or decision-aided and uses the receiver’s decisions. Such methods are an approximation to the ML synchronization. Since the recovered symbol rate is used to sample the incoming waveform in most systems, care must be exerted in the higher-performance decision-directed methods that not too much delay appears between the sampling device and the decision device. Such delay can seriously degrade the performance of the receiver or even render the phase-lock loop unstable.

Synchronization is one of the most critical functions performed at the receiver of a synchronous communication system. To some extent, it is the basis of a synchronous communication system.

Three broad types of synchronization:

**1. Carrier synchronization (2)**

To recover the signal without distortion, receiver needs to estimate and compensate for frequency and phase differences between a received signal's carrier wave and the receiver's local oscillator for the purpose of coherent demodulation.

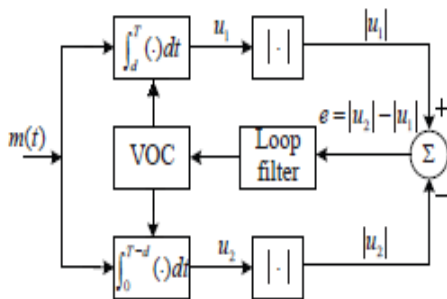
**2. Symbol & Bit synchronization (2)**

The output of the receiving filter must be sampled at the symbol rate and at the precise sampling time instants. Hence, we require a clock signal. The process of extracting such a clock signal at the receiver is called symbol/bit synchronization.

**3. Frame synchronization (2)**

Receiver can proceed by every group of symbols instead of every single symbol, such as a frame in TDM system. Similar with symbol/bit synchronization, the process of extracting such a clock signal is called frame synchronization.

**Early-Late Gate Synchronization:**



(2)

The synthesiser operates by performing two separate integrations of the incoming signal energy over two different portions of the symbol interval.

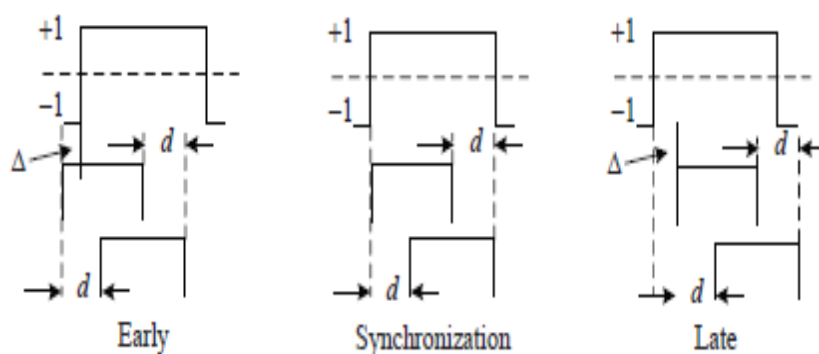
The first integration (early gate) begins at the loop's best estimate of the beginning of a symbol period and integrates for the next  $(T_s - d)$  where  $d$  is the delay between the onset of the two integrator. Obviously  $d$  is less than  $T_s$ . (2)

The second integrator (late gate) starts its integration  $d$  times after the first starts and integrates for the next  $(T_s-d)$  time i.e. upto the end of the period. The difference between the integrated values of the early and late gate is  $-2\Delta A$ .

(2)

This error signal will lower the input voltage to the VCO and as a consequence the VCO output frequency gets lowered. The net effect is the retardation of the receiver's timing been late so that it falls in line with the incoming signal's transitions.

Similarly had the receiver's timing been late, the error signal would have been positive and VCO frequency increased to advance the receiver timing towards that of the incoming signal. (2)



(2)

## Tracking and acquisition in spread spectrum system

### Code Synchronization

- Code synchronization is the process of achieving and maintaining proper alignment between the reference code in a spread spectrum receiver and the spreading sequence that has been used in the transmitter to spread the information bits.
- Code synchronization is achieved in two stages: a) code acquisition and b) code tracking.
- Acquisition is the process of initially attaining coarse alignment (typically within  $\pm$  half of the chip duration),
- Tracking ensures that fine alignment within a chip duration is maintained.

### Code Acquisition Schemes

- Acquisition is basically a process of searching through an uncertainty region, which may be one-dimensional, e.g. in time alone or two-dimensional, viz. in time and

frequency (if there is drift in carrier frequency due to Doppler effect etc.) – until the correct code phase is found.

- The uncertainty region is divided into a number of cells. In the one-dimensional case, one cell may be as small as a fraction of a PN chip interval.
- The acquisition process has to be reliable and the average time taken to acquire the proper code phase also should be small
- A basic operation, often carried out during the process of code acquisition is the correlation of the incoming spread signal with a locally generated version of the spreading code sequence (obtained by multiplying the incoming and local codes and accumulating the result) and comparing the correlator output with a set threshold to decide whether the codes are in phase or not.
- The correlator output is usually less than the peak autocorrelation of the spreading code due to a) noise and interference in the received signal or b) time or phase misalignment or c) any implementation related imperfections. If the threshold is set too high (equal or close to the autocorrelation peak), the correct phase may be missed i.e. probability of missed detection ( $M P$ ) is high. On the other hand, if the threshold is set too low, acquisition may be declared at an incorrect phase (or time instant), resulting in high probability of false acquisition ( $FAP$ ). A tradeoff between the two is thus desired.

- A code acquisition scheme may also be active or passive.
- In a receiver employing active code acquisition (sometimes called code detection), portions of the incoming and the local codes (a specific phase shifted version) are multiplied bit by bit and the product is accumulated over a reasonable interval before comparison is made against a decision threshold.
- If the accumulated value does not exceed the threshold, the process is repeated with new samples of the incoming code and for another offset version of the local code.
- A passive detector, on the other hand, is much like a matched filter with the provision to store the incoming code samples in a shift register.
- With every incoming chip, a decision is made (high decision rate) based on a correlation interval equal to the length of the matched filter (MF) correlator.
- A disadvantage of this approach is the need for more hardware when large correlation intervals are necessary.
- Code acquisition schemes are also classified based on the criterion used for deciding the threshold. For example, a Bayes' code detector minimizes the average probability of missed detection while a Neyman Pearson detector minimizes the probability of missed detection for a particular value of  $FAP$ .

### **Classification based on search strategy**

As noted earlier, the acquisition schemes are also classified on the basis of the search strategy in the region of uncertainty (Fig 7.39.2). Following the maximum likelihood estimation

technique, the incoming signal is simultaneously (in parallel) correlated with all possible time-shifted versions of the local code and the local code phase that yields the highest output is declared as the phase of the incoming code sequence. This method requires a large number of correlators. However, the strategy may also be implemented (somewhat approximately) in a serial manner, by correlating the incoming code with each phase-shifted version of the local code, and taking a decision only after the entire code length is scanned. While one correlator is sufficient for this approach, the acquisition time increases linearly with the correlation length. Further, since the noise conditions are usually not the same for all code phases, this approach is not strictly a maximum likelihood estimation algorithm.

Proposed in 1984, this is a rapid acquisition scheme for CDMA systems, that employs Subsequence Matched Filtering (SMF). The detector consists of several correlator based matched filters, each matched to a subsequence of length  $M$ . The subsequence may or may not be contiguous, depending on the length of the code and the hardware required to span it. With every incoming chip of the received waveform, the largest of the SMF value is compared against a pre-selected threshold. Threshold exceedance leads to loading that subsequence in the local PN generator for correlation over a longer time interval. If the correlation value does not exceed a second threshold, a negative feedback is generated, and a new state estimate is loaded into the local generator. Otherwise, the PN generation and correlation process continues.

### **Frame Synchronization in Data Transmission**

- While carrier and symbol clock may have been well established in a data transmission system, the boundary of long symbols or packets may not be known to a receiver.
- Such synchronization requires searching for some known pattern or known characteristic of the transmitted waveform to derive a phase error (which is typically measured in the number of sample periods or dimensions in error).
- Such synchronization, once established, is only lost if some kind of catastrophic failure of other mechanisms has occurred (or dramatic sudden change in the channel)



because it essentially involves counting the number of samples from the recovered timing clock.

### Autocorrelation Methods

- Autocorrelation methods form the discrete channel output autocorrelation

$$R_{yy}(l) = E [y_k y_{k-l}^*] = h_l * h_{-l}^* * R_{xx}(l) + R_{nn}(l)$$

from channel output samples by time averaging over some interval of M samples

$$\hat{R}_{yy}(l) = \frac{1}{M} \sum_{m=1}^M \sum_{m=1}^M y_m y_{m-l}^*$$

- In forming such a sum, the receiver implies a knowledge of sampling times 1, ..., M. If those times are not coincident with the corresponding (including channel delay) positions of the packet in the transmitter, then the autocorrelation function will look shifted.
- For instance, if a peak was expected at position 1 and occurs instead at position 10, then the difference is an indication of a packet timing error.
- The phase error used to drive a PLL is that difference. Typically, the VCO in this case supplies only discrete integer number- of-sample steps. If noise is relatively small and  $h_l$  is known, then one error may be sufficient to find the packet boundary.
- If not, many repeated estimates of  $\hat{R}_{yy}$  at different phase adjustments may be tried. There are 3 fundamental issues that affect the performance of such a scheme:
  1. The knowledge of or choice of the input sequence  $x_k$ ,
  2. The channel response's ( $h_k$ 's) effect upon the autocorrelation of the input,
  3. The noise autocorrelation.

### Synchronization Patterns

- A synchronization pattern or "synch sequence" is some known transmitted signal, typically with properties that create a large peak in the autocorrelation function at the channel output at a known lag  $l$ . Typically with small (or eliminated) ISI, such a sequence corresponds to a white input.
- Simple random data selection may often not be sufficient to guarantee a high autocorrelation peak for a short averaging period  $M$ . Thus, special sequences are often selected that have a such a short-term peaked time-averaged autocorrelation.
- A variety of sequences exist. The most common types are between 1 and 2 cycles of a known period pattern. Pseudorandom binary sequences of degree  $p$  have period of  $2^p - 1$  samples are those specified by their basic property that cyclic shifts of the sequence have each length- $p$  binary pattern (except all 0's) once and only once.
  - A simple linear-feedback register implementation appears there. The important property is that the autocorrelation (with binary antipodal modulation) of such sequences is

$$\hat{R}_{xx}(l) = \begin{cases} 1 & l = 0 \\ -\frac{1}{2^p - 1} & l = 1, \dots, 2^p - 1 \end{cases}$$

- For reasonably long  $p$ , the peak is easily recognizable if the channel output has little or no distortion. In fact, the entire recognition of the lag can be positioned as a basic

Chapter 1 detection problem, where each of the possible input shifts of the same known training sequence is to be detected.

- Because all these patterns have the same energy and they're almost orthogonal, a simple largest matched filter output (the matched filter outputs are the computation of the autocorrelation at different time lags) with the probability of error
- Another synchronization pattern that can have appeal is the so-called "chirp" sequence

$$x_k = e^{j2\pi k^2/M}$$

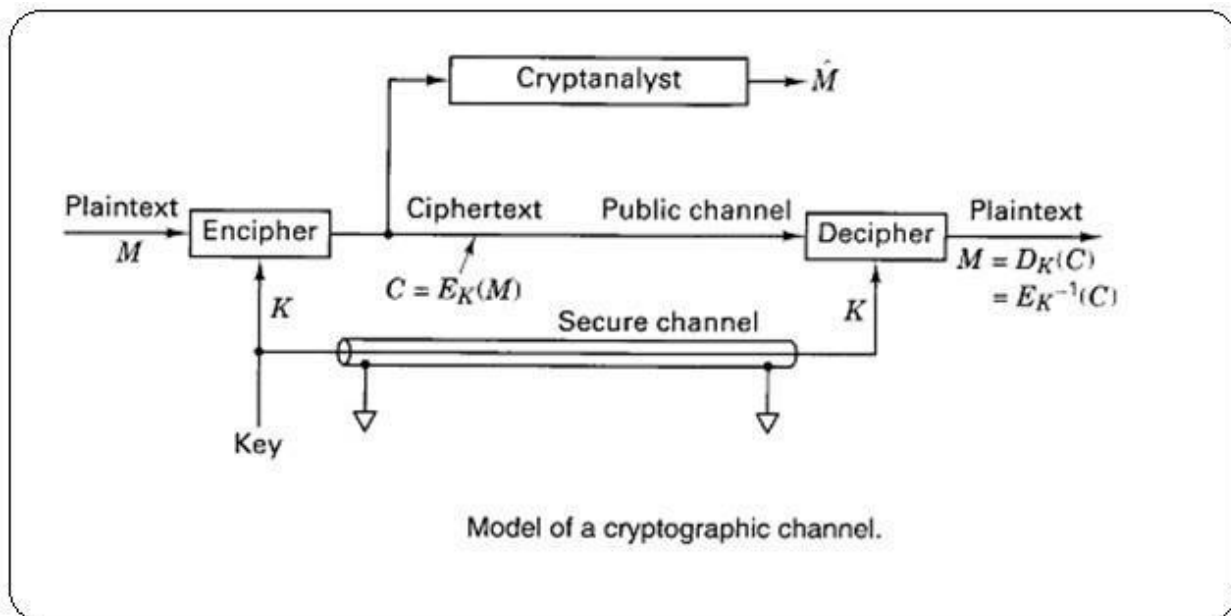
which also has period M. It also has a single peak at time zero of size 1 and zero autocorrelation. It is harder to generate and requires a complex QAM baseband system, but in some sense is a perfect synch pattern.

- A third alternative are the so-called "Barker codes" that are very short and not repeated and designed to still have peakiness in the face of unknown preceding and succeeding data surrounding the pattern.
- Such patterns may be periodically inserted in a transmission stream so that if the receiver for any reason lost packet synchronization (or a new received came on after a first had already acquired the signal if more than receiver scan the same or related channels), then the calculation of autocorrelation would immediately recommence until the peak was found.
- An example of a good short synchronization pattern is the 7-symbol Barker code, which is for instance used with 2B1Q transmission in some symmetric DSL systems that employ PAM modulation.

# ENCRYPTION AND DECRYPTION

## MODEL ENCRYPTOR-DECRYPTOR (MODEL OF A CRYPTOGRAPHIC CHANNEL)

- ❖ **CRYPTOGRAPHY**: is the study of different ways to protect messages from unauthorized interception.
- ❖ Encipher or encrypt is performed at transmitter.
- ❖ Encryption is the transformation of messages at the transmitter.
- ❖ Decipher or decrypt refers to the inverse transformation performed at the receiver.
- ❖ The two primary reasons for using cryptosystems in communications are
  1. **PRIVACY**-To prevent unauthorized persons from extracting the information from the channel (**EAVES DROPPING**).
  2. **AUTHENTICATION**-To prevent unauthorized persons from injecting information into the channel (**SPOOFING**).
- ❖ The model of a cryptographic channel is shown below



- ❖ **PLAIN TEXT**: A message before encryption is called plain text (M).
- ❖ **CIPHER TEXT**: A message (plain text) M is encrypted by the use of an invertible transformation,  $E_k$ , that produces cipher text.
- ❖ **CIPHER TEXT,  $C = E_K(M)$**
- ❖ The cipher text is transmitted over an insecure or public channel.
- ❖ When an authorized receiver obtains C, he decrypts it with the inverse transformation  $D_K = E_K^{-1}$ , to obtain original plain text.

$$D_K(C) = E_K^{-1}[E_K(M)]$$

- ❖ “K” is a set of symbols or characters called key, which identify a specific encryption transformation  $E_K$  from a family of cryptographic transformation.

- ❖ Originally, the security of crypto systems depends on the secrecy of the entire encryption process.

- ❖ Encryption schemes are of two types.
  1. BLOCK ENCRYPTION
  2. DATA STREAM OR SIMPLY STREAM ENCRYPTION
- ❖ **BLOCK ENCRYPTION:** with block encryption, the plain text is segmented into blocks of fixed size, each block is encrypted independently from others.
- ❖ **DATA STREAM ENCRYPTION:** with data stream encryption, there is no fixed block size. Each plain text bit,  $M_i$  is encrypted with the  $i^{\text{th}}$  element  $K_i$  of a sequence of symbols generated with the key.
- ❖ **PERIODIC & NON PERIODIC ENCRYPTION:** The encryption is periodic if the key stream repeats itself after P-characters for some fixed P. Otherwise it is non periodic.
- ❖ Successful cryptosystems are classified into two groups.
  1. UNCONDITIONALLY SECURE
  2. COMPUTATIONALLY SECURE
- ❖ A system is said to be unconditionally secure when the amount of information available to the cryptanalyst is insufficient to determine the encryption and decryption transformations, no matter how much computing power a cryptanalyst has available.
- ❖ One such system called a “one time pad” involves encrypting a message with a random key that is used one time only. The key is never reused.
- ❖ Computational security for x years, which means that under circumstances favorable to the cryptanalyst, the system security could be broken in a period of x- years, but could not be broken in less than x-years.

### ❖ **CLASSIC THREATS:**

1. **CIPHER TEXT ONLY ATTACK:** It is the weakest threat on a system. In this attack, the cryptanalyst might have some knowledge of the general system and the language used in the message, but the only significant data available to him is the encrypted transmission intercepted from the public channel.
2. **PLAIN TEXT ATTACK:** It is a serious threat to a system. It involves the knowledge of the plain text and knowledge of its cipher text counterpart.
3. **CHOSEN PLAIN TEXT ATTACK:** When the cryptanalyst is in the position of selecting the plain text, the threat is termed a chosen plain text attack. Such an attack was used by the united states to learn more about the Japanese crypto systems during world war II.

## **CLASSICAL ENCRYPTION TECHNIQUES {CLASSIC CIPHERS}:**

1. **CEASER CIPHER:**It is used by Julius Ceasar during the Gallic wars. Each plain text letter is replaced with a new letter obtained by an alphabetic shift. This type of encryption transformation is shown below. It uses a 3-end- around shift of the alphabet. The decryption key is simply the number of alphabetic shifts. The code is changed by choosing a new key.

|             |   |
|-------------|---|
| Plaintext:  | A B C D E F G H I J K L M N O P Q R S T U V W X Y Z |
| Ciphertext: | D E F G H I J K L M N O P Q R S T U V W X Y Z A B C |

|             |                         |
|-------------|-------------------------|
| Plaintext:  | N O W I S T H E T I M E |
| Ciphertext: | Q R Z L V W K H W L P H |

2. **POLYBIUS SQUARE:** It is an another classic cipher system which is shown below. Letters I and J are first combined and treated as a single character. Here the alphabet is arranged in a 5 x 5 array. Encryption of any character is accomplished by choosing the appropriate row-column or column-row number pair. The code is changed by a rearrangement of the letters in the 5x5 array.

**POLYBIUS SQUARE**

|   |   |   |   |   |   |
|---|---|---|---|---|---|
|   | 1 | 2 | 3 | 4 | 5 |
| 1 | A | B | C | D | E |
| 2 | F | G | H | I | K |
| 3 | L | M | N | O | P |
| 4 | Q | R | S | T | U |
| 5 | V | W | X | Y | Z |

**EXAPLE: POLYBIUS SQUARE**

|             |    |    |    |    |    |    |    |    |    |    |    |    |
|-------------|----|----|----|----|----|----|----|----|----|----|----|----|
| Plaintext:  | N  | O  | W  | I  | S  | T  | H  | E  | T  | I  | M  | E  |
| Ciphertext: | 33 | 43 | 25 | 42 | 34 | 44 | 32 | 51 | 44 | 42 | 23 | 51 |

3. **TRITHEMIUS PROGRESSIVE KEY:** It is an example of a poly alphabetic cipher. The row labeled shift 0 is identical to the usual arrangement of the alphabet.

**EXAMPLE:** one method of using such an alphabet is to select the first cipher character from firstrow, the second cipher character from second row, and so on.

|             |   |   |   |   |   |   |   |   |   |   |   |   |
|-------------|---|---|---|---|---|---|---|---|---|---|---|---|
| Plaintext:  | N | O | W | I | S | T | H | E | T | I | M | E |
| Ciphertext: | O | Q | Z | M | X | Z | O | M | C | S | X | Q |

There are several ways to select the Thrithimius progressive key. One way is called Vigenere keymethod, employs a key word. For example, the key word is “TYPE”. The key indicates the row choices for encryption and decryption of each successive character of the message.

|    |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|----|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
|    | a | b | c | d | e | f | g | h | i | j | k | l | m | n | o | p | q | r | s | t | u | v | w | x | y | z |
| 0  | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | R | U | V | W | X | Y | Z |
| 1  | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | O | R | S | T | U | V | W | X | Y | Z | A |
| 2  | C | D | E | F | G | H | I | J | K | L | M | N | O | P | O | R | S | T | U | V | W | X | Y | Z | A | B |
| 3  | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | A | B | C |
| 4  | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | A | B | C | D |
| 5  | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | A | B | C | D | E |
| 6  | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | A | B | C | D | E | F |
| 7  | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | A | B | C | D | E | F | G |
| 8  | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | A | B | C | D | E | F | G | H |
| 9  | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | A | B | C | D | E | F | G | H | I |
| 10 | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | A | B | C | D | E | F | G | H | I | J |
| 11 | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | A | B | C | D | E | F | G | H | I | J | K |
| 12 | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | A | B | C | D | E | F | G | H | I | J | K | L |
| 13 | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | A | B | C | D | E | F | G | H | I | J | K | L | M |
| 14 | O | P | Q | R | T | U | V | W | X | Y | Z | A | B | C | D | E | F | G | H | I | J | K | L | M | N |   |
| 15 | P | Q | R | S | S | U | V | W | X | Y | Z | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O |
| 16 | Q | R | S | T | U | V | W | X | Y | Z | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P |
| 17 | R | S | T | U | V | W | X | Y | Z | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q |
| 18 | S | T | U | V | W | X | Y | Z | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R |
| 19 | T | U | V | W | X | Y | Z | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S |
| 20 | U | V | W | X | Y | Z | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T |
| 21 | V | W | X | Y | Z | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U |
| 22 | W | X | Y | Z | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V |
| 23 | X | Y | Z | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W |
| 24 | Y | Z | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X |
| 25 | Z | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y |

Trithemius progressive key.

|             |   |   |   |   |   |   |   |   |   |   |   |   |
|-------------|---|---|---|---|---|---|---|---|---|---|---|---|
| Key:        | T | Y | P | E | T | Y | P | E | T | Y | P | E |
| Plaintext:  | N | O | W | I | S | T | H | E | T | I | M | E |
| Ciphertext: | G | M | L | M | L | R | W | I | M | G | B | I |

**Name four factors needed for a secure network?**

**Privacy:** The sender and the receiver expect confidentiality.

**Authentication:** The receiver is sure of the sender’s identity and that an imposter hasnot sent the message.

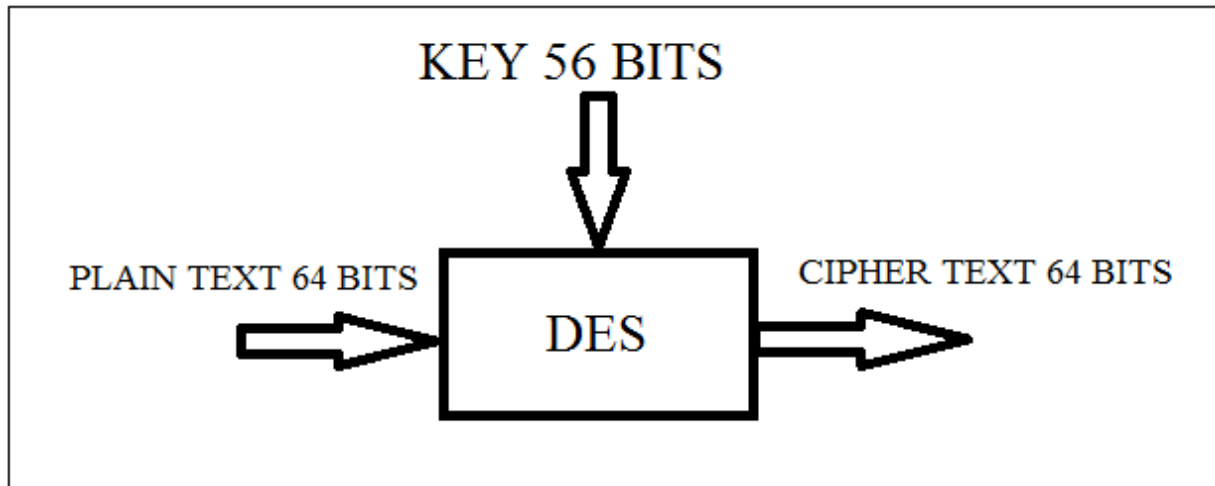
**Integrity:** The data must arrive at the receiver exactly as it was sent.

**Non-Reputation:** The receiver must able to prove that a received message came from a specific sender.

**DATA ENCRYPTION STANDARD (DES):**

- ❖ The data encryption standard (DES) is certainly the best known, and the most widely used, secret- key crypto algorithm; the term algorithm is used to describe a sequence of computations.
- ❖ The basic DES algorithm can be used for both data encryption and data authentication.

- ❖ It is the standard crypto algorithm for data storage and mail systems, electronic fund transfers, and electronic business data interchange.
- ❖ The DES algorithm is a strong block cipher that operates on 64 bit blocks of plain text data and uses a 56 bit key-it is designed in accordance with shannon's methods of diffusion and confusion.
- ❖ Essentially the same algorithm is used for encryption and decryption.
- ❖ From a system input – output point of view, DES can be regarded as a block encryption system with an alphabet size of  $2^{64}$  symbols, which is shown below.

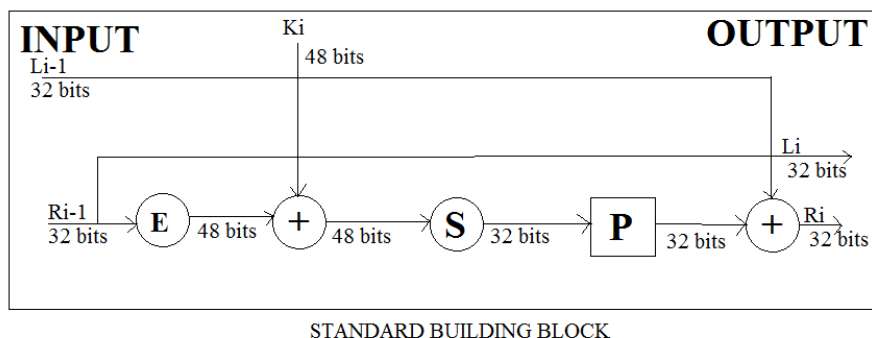


- ❖ The encryption algorithm starts with an initial permutation (IP) of the 64 plain text bits.

Initial Permutation (IP) table

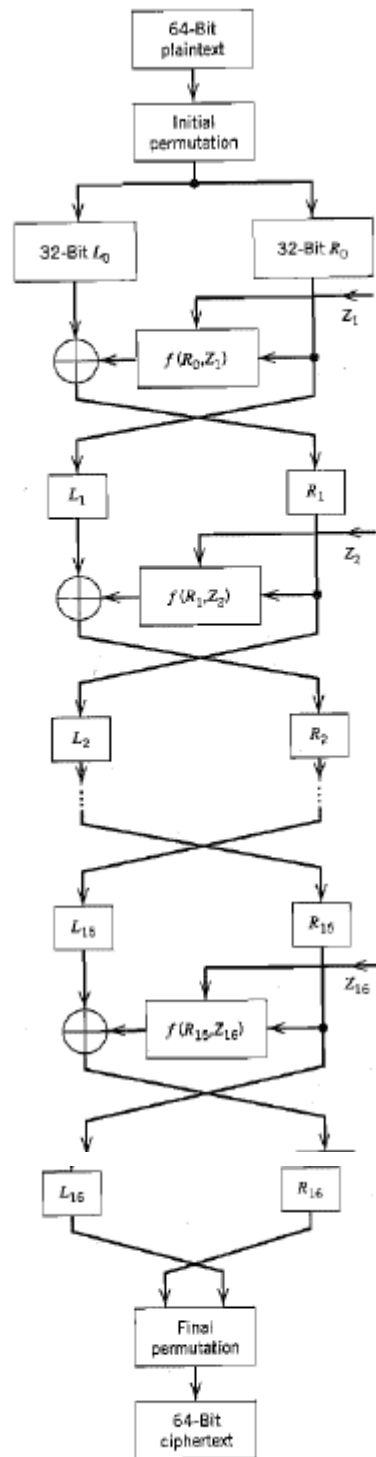
|    |    |    |    |    |    |    |   |
|----|----|----|----|----|----|----|---|
| 58 | 50 | 42 | 34 | 26 | 18 | 10 | 2 |
| 60 | 52 | 44 | 36 | 28 | 20 | 12 | 4 |
| 62 | 54 | 46 | 38 | 30 | 22 | 14 | 6 |
| 64 | 56 | 48 | 40 | 32 | 24 | 16 | 8 |
| 57 | 49 | 41 | 33 | 25 | 17 | 9  | 1 |
| 59 | 51 | 43 | 35 | 27 | 19 | 11 | 3 |
| 61 | 53 | 45 | 37 | 29 | 21 | 13 | 5 |
| 63 | 55 | 47 | 39 | 31 | 23 | 15 | 7 |

- ❖ After this initial permutation, the heart of the encryption algorithm consists of 16 iterations using the standard building block (SBB).





- ❖ The standard building block uses 48 bits of key to transform the 64 bit input data bits into 64 output data bits, designated as 32 left half bits and 32 right half bits.
- ❖ The output of each building block becomes the input to the next building block.
- ❖ The input right half 32 bits ( $R_{i-1}$ ) are copied unchanged to become the output left half 32 bits ( $L_i$ ).
- ❖ The  $R_{i-1}$  bits are also extended and transformed into 48 bits with the E-Table, then added with the 48 bits key using the summer.

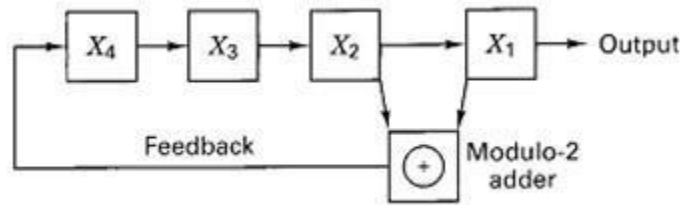


DATA ENCRYPTION STANDARD

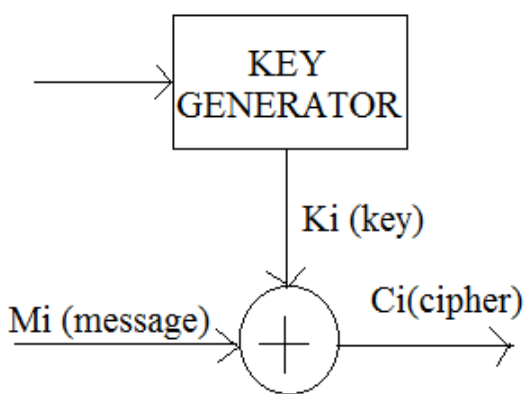
## STREAM ENCRYPTION:

- ❖ A stream encryption system uses a random key stream, ie the key sequence never repeats.
- ❖ Thus perfect secrecy can be achieved for an infinite number of messages since each message would be encrypted with a different portion of the random key stream.

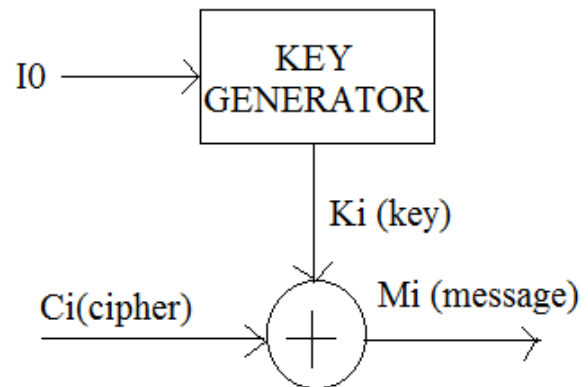
- ❖ Stream encryption techniques use pseudo random (PN) sequences. However these sequences are deterministic.
- ❖ These techniques are popular because encryption and decryption algorithms are readily implemented with feedback shift registers.
- ❖ Key generation using a linear feedback shift register is shown below.



- ❖ A shift register can be converted into a pseudo random sequence generator by including a feedback loop.
- ❖ The initial state of the stages ( $X_4, X_3, X_2, X_1$ ) is 1000, the next stage is triggered by clock pulses would be 1000, 0100, 0010, 1001, 1100, and so on.
- ❖ **SYNCHRONOUS AND SELF SYNCHRONOUS STREAM ENCRYPTION SYSTEMS:** In the synchronous stream encryption systems, the key stream is generated independently of the message.

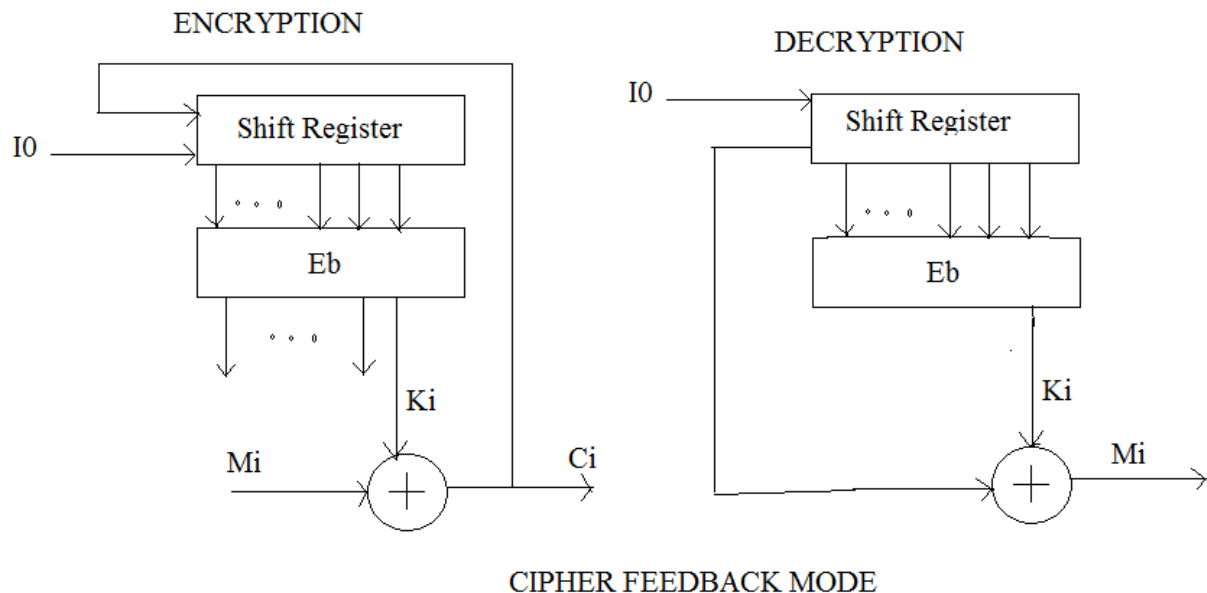


(A) ENCRYPTOR



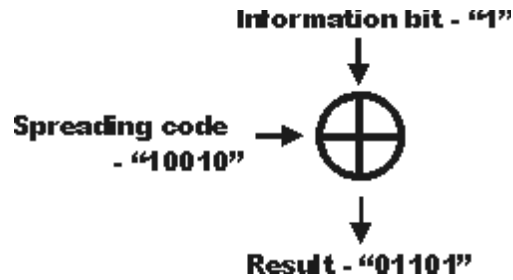
(B) DECRYPTOR

- ❖ In this type a lost character during transmission necessitates a resynchronization of the transmitter and receiver key generators.
- ❖ In a self synchronous stream cipher each key character is derived from a fixed number,  $N$ , of the preceding cipher text characters giving rise to the name cipher feedback.
- ❖ In this system the cipher text character is lost during transmission, the error propagates forward for  $n$  characters but the system resynchronizes itself after  $n$  correct cipher text characters are received.



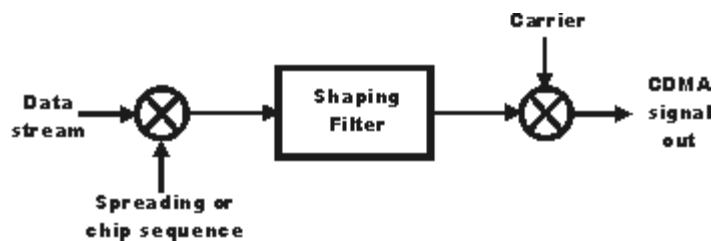
## CDMA SPREAD SPECTRUM BASICS

- CDMA is based around the use of direct sequence spread spectrum techniques.
- Essentially CDMA is a form of spread spectrum transmission which uses spreading codes to spread the signal out over a wider bandwidth than would normally be required.
- By using CDMA spread spectrum technology, many users are able to use the same channel and gain access to the system without causing undue interference to each other.
- Although as the number of users increases care has to be taken to ensure that interference levels do not rise to the extent that performance falls, it is still possible to provide access to a large number of different users and allow them access.
- The key element of code division multiple access CDMA is its use of a form of transmission known as direct sequence spread spectrum, DSSS.
- Direct sequence spread spectrum is a form of transmission that looks very similar to white noise over the bandwidth of the transmission.
- However once received and processed with the correct descrambling codes, it is possible to extract the required data.
- When transmitting a CDMA spread spectrum signal, the required data signal is multiplied with what is known as a spreading or chip code data stream. The resulting data stream has a higher data rate than the data itself. Often the data is multiplied using the XOR (exclusive OR) function.



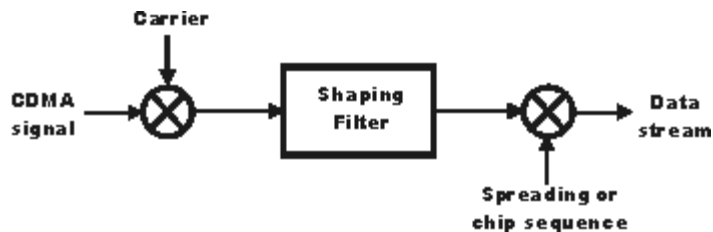
### CDMA spreading

- Each bit in the spreading sequence is called a chip, and this is much shorter than each information bit. The spreading sequence or chip sequence has the same data rate as the final output from the spreading multiplier. The rate is called the chip rate, and this is often measured in terms of a number of M chips / sec.
- The baseband data stream is then modulated onto a carrier and in this way the overall signal is spread over a much wider bandwidth than if the data had been simply modulated onto the carrier. This is because, signals with high data rates occupy wider signal bandwidths than those with low data rates.



### CDMA spread spectrum generation

- To decode the signal and receive the original data, the CDMA signal is first demodulated from the carrier to reconstitute the high speed data stream. This is multiplied with the spreading code to regenerate the original data. When this is done, then only the data with that was generated with the same spreading code is regenerated, all the other data that is generated from different spreading code streams is ignored.



### CDMA spread spectrum decoding

- The use of CDMA spread spectrum is a powerful principle and using this CDMA technique, it is possible to transmit several sets of data independently on the same carrier and then reconstitute them at the receiver without mutual interference. In this way a base station can communicate with several mobiles on a single channel. Similarly several mobiles can communicate with a single base station, provided that in each case an independent spreading code is used.

## CDMA spread spectrum encode / decode process

- In order to visualise how the CDMA spread spectrum process operates, the easiest method is to show an example of how the system actually operates in terms of data bits, and how the data is recovered from the

CDMA spread spectrum signal.

- The first part of the process is to generate the CDMA spread spectrum signal. Take as an example that the data to be transmitted is 1001, and the chip or spreading code is 0010. For each data bit, the complete spreading code is used to multiple the data, and in this way, for each data bits, the spread or expanded signal consists of four bits.

|      |      |      |      |                              |
|------|------|------|------|------------------------------|
| 1    | 0    | 0    | 1    | Data to be transmitted       |
| 0010 | 0010 | 0010 | 0010 | Chip or spreading code       |
| 1101 | 0010 | 0010 | 1101 | Resultant spread data output |

With the signal obtained and transmitted, it needs to be decoded within the remote receiver:

|      |      |      |      |                        |
|------|------|------|------|------------------------|
| 1101 | 0010 | 0010 | 1101 | Incoming CDMA signal   |
| 0010 | 0010 | 0010 | 0010 | Chip or spreading code |
| 1111 | 0000 | 0000 | 1111 | Result of de-spreading |
| 1    | 0    | 0    | 1    | Integrated output      |

**NB:**  $1 \times 1 = 0$     $1 \times 0 = 1$

- In this way it can be seen that the original data is recovered exactly by using the same spreading or chip code. Had another code been used to regenerate the CDMA spread spectrum signal, then it would have resulted in a random sequence after de-spreading. This would have appeared as noise in the system.
- The spreading code used in this example was only four bits long. This enabled the process to be visualized more easily. Commonly spreading codes may be 64 bits, or even 128 bits long to provide the required performance.

## CDMA spreading gain

- The bandwidth of the CDMA spread spectrum signal will be much wider than the original data stream. To quantify the increase in bandwidth, a term known as the spreading gain is used. If the bandwidth of the CDMS spread spectrum signal is **W** and the input data bit length or period **1/R** then the CDMA spreading gain can be defined:

$$\text{Spreading gain} = W / R$$

- It is found that the larger the spreading gain of the CDMA spread spectrum signal, the more effective the performance of the system is. This is because the wanted signal becomes larger. In the example shown above, the spreading gain is four, as seen by the fact that four "1"s are generated for each required data bit. Data produced by other disspreading codes would appear as noise and can be discarded as it would be lower in value.
- The principle behind CDMA spread spectrum communications is relatively straightforward. The same code must be sued within generation and decoding of the CDMA spread spectrum signal to enable the data to pass unchanged through the system. The use of a different code in transmission and reception results in a signal similar in character to noise being generated and this can be discarded.

## RSA ALGORITHM

- **RSA** is one of the first practicable public-key cryptosystems and is widely used for secure data transmission.
- In such acryptosystem, the encryption key is public and differs from the decryption key which is kept secret.
- In RSA, this asymmetry is based on the practical difficulty of factoring the product of two large prime numbers, the factoring problem.
- RSA stands for Ron Rivest, Adi Shamir and Leonard Adleman, who first publicly described the algorithm in 1977. Clifford Cocks, an English mathematician, had developed an equivalent system in 1973, but it was not declassified until 1997.
- A user of RSA creates and then publishes a public key based on the two large prime numbers, along with an auxiliary value.
- The prime numbers must be kept secret. Anyone can use the public key to encrypt a message, but with currently published methods, if the public key is large enough, only someone with knowledge of the prime numbers can feasibly decode the message.
- Breaking RSA encryption is known as the RSA problem. It is an open question whether it is as hard as the factoring problem.
- RSA involves a *public key* and a *private key*.
- The public key can be known by everyone and is used for encrypting messages. Messages encrypted with the public key can only be decrypted in a reasonable amount of time using the private key.
- The keys for the RSA algorithm are generated the following way:
  1. Choose two distinct prime numbers  $p$  and  $q$ .
    - For security purposes, the integers  $p$  and  $q$  should be chosen at random, and should be of similar bit-length. Prime integers can be efficiently found using a primality test.
  2. Compute  $n = pq$ .
    - $n$  is used as the modulus for both the public and private keys. Its length, usually expressed in bits, is the key length.
  3. Compute  $\phi(n) = \phi(p)\phi(q) = (p - 1)(q - 1) = n - (p + q - 1)$ , where  $\phi$  is Euler's totient function.
  4. Choose an integer  $e$  such that  $1 < e < \phi(n)$  and  $\text{gcd}(e, \phi(n)) = 1$ ; i.e.,  $e$  and  $\phi(n)$  are co-prime.
    - $e$  is released as the public key exponent.
    - $e$  having a short bit-length and small Hamming weight results in more efficient encryption – most commonly  $2^{16} + 1 = 65,537$ . However, much smaller values of  $e$  (such as 3) have been shown to be less secure in some settings.<sup>[5]</sup>
  5. Determine  $d$  as  $d \equiv e^{-1} \pmod{\phi(n)}$ ; i.e.,  $d$  is the multiplicative inverse of  $e$  (modulo  $\phi(n)$ ).
    - This is more clearly stated as: solve for  $d$  given  $d \cdot e \equiv 1 \pmod{\phi(n)}$
    - This is often computed using the extended Euclidean algorithm. Using the pseudocode in the *Modular integers* section, inputs  $a$  and  $n$  correspond to  $e$  and  $\phi(n)$ , respectively.
    - $d$  is kept as the private key exponent.



The *public key* consists of the modulus  $n$  and the public (or encryption) exponent  $e$ . The *private key* consists of the modulus  $n$  and the private (or decryption) exponent  $d$ , which must be kept secret.  $p$ ,  $q$ , and  $\phi(n)$  must also be kept secret because they can be used to calculate  $d$ .

- An alternative, used by PKCS#1, is to choose  $d$  matching  $de \equiv 1 \pmod{\lambda}$  with  $\lambda = \text{lcm}(p-1, q-1)$ , where  $\text{lcm}$  is the least common multiple. Using  $\lambda$  instead of  $\phi(n)$  allows more choices for  $d$ .  $\lambda$  can also be defined using the Carmichael function,  $\lambda(n)$ .
- The ANSI X9.31 standard prescribes, IEEE 1363 describes, and PKCS#1 allows, that  $p$  and  $q$  match additional requirements: being strong primes, and being different enough that Fermat factorization fails.

### **Encryption**[edit]

Alice transmits her public key  $(n, e)$  to Bob and keeps the private key  $d$  secret. Bob then wishes to send message  $M$  to Alice.

He first turns  $M$  into an integer  $m$ , such that  $0 \leq m < n$  by using an agreed-upon reversible protocol known as a padding scheme. He then computes the ciphertext  $c$  corresponding to

This can be done efficiently, even for 500-bit numbers, using Modular exponentiation. Bob then transmits  $c$  to Alice.

Note that at least nine values of  $m$  will yield a ciphertext  $c$  equal to  $m$ ,<sup>[note 1]</sup> but this is very unlikely to occur in practice.

### **Decryption**[edit]

Alice can recover  $m$  from  $c$  by using her private key exponent  $d$  via computing

Given  $m$ , she can recover the original message  $M$  by reversing the padding scheme.

(In practice, there are more efficient methods of calculating  $c^d$  using the precomputed values below.)

### **A worked example**[edit]

Here is an example of RSA encryption and decryption. The parameters used here are artificially small, but one can also use OpenSSL to generate and examine a real keypair.

1. Choose two distinct prime numbers, such as  
and

2. Compute  $n = pq$  giving

$$n = 61 \times 53 = 3233$$

3. Compute the totient of the product as  $\phi(n) = (p-1)(q-1)$  giving

$$\phi(3233) = (61 - 1)(53 - 1) = 3120$$

4. Choose any number  $1 < e < 3120$  that is coprime to 3120. Choosing a prime number for  $e$  leaves us only to check that  $e$  is not a divisor of 3120.

Let  $e = 17$

5. Compute  $d$ , the modular multiplicative inverse of  $e \pmod{\phi(n)}$  yielding,

$$d = 2753$$

Worked example for the modular multiplicative inverse:

$$e \times d \pmod{\phi(n)} \Rightarrow \text{choice of public exponent } e$$

$$17 \times 2753 \pmod{3120} = 1$$

The **public key** is  $(n = 3233, e = 17)$ . For a padded plaintext message  $m$ , the encryption function is

$$c(m) = m^{17} \pmod{3233}$$

$$m(c) = c^{2753} \pmod{3233}$$

The **private key** is  $(n = 3233, d = 2753)$ . For an encrypted ciphertext  $c$ , the decryption function is

For instance, in order to encrypt  $m = 65$ , we calculate

$$c = 65^{17} \pmod{3233} = 2790$$

$$m = 2790^{2753} \pmod{3233} = 65$$

To decrypt  $c = 2790$ , we calculate

Both of these calculations can be computed efficiently using the square-and-multiply algorithm for modular exponentiation. In real-life situations the primes selected would be much larger; in our example it would be trivial to factor  $n$ , 3233 (obtained from the freely available public key) back to the primes  $p$  and  $q$ . Given  $e$ , also from the public key, we could then compute  $d$  and so acquire the private key.

Practical implementations use the Chinese remainder theorem to speed up the calculation using modulus of factors ( $\pmod{pq}$  using  $\pmod{p}$  and  $\pmod{q}$ ).

The values  $d_p$ ,  $d_q$  and  $q_{\text{inv}}$ , which are part of the private key are computed as follows:

$$d_p = d \pmod{p-1} = 2753 \pmod{61-1} = 53$$

$$d_q = d \pmod{q-1} = 2753 \pmod{53-1} = 49$$

$$q_{\text{inv}} = q^{-1} \pmod{p} = 53^{-1} \pmod{61} = 38$$

$$\Rightarrow (q_{\text{inv}} \times q) \pmod{p} = 38 \times 53 \pmod{61} = 1$$

Here is how  $d_p$ ,  $d_q$  and  $q_{\text{inv}}$  are used for efficient decryption. (Encryption is efficient

$$m_1 = c^{d_p} \bmod p = 2790^{53} \bmod 61 = 4$$

$$m_2 = c^{d_q} \bmod q = 2790^{49} \bmod 53 = 12$$

$$h = (q_{\text{inv}} \times (m_1 - m_2)) \bmod p = (38 \times -8) \bmod 61 = 1$$

$$m = m_2 + h \times q = 12 + 1 \times 53 = 65$$

### **RSA Algorithm Example**

- Choose  $p = 3$  and  $q = 11$
- Compute  $n = p * q = 3 * 11 = 33$
- Compute  $\phi(n) = (p - 1) * (q - 1) = 2 * 10 = 20$
- Choose  $e$  such that  $1 < e < \phi(n)$  and  $e$  and  $n$  are coprime. Let  $e = 7$
- Compute a value for  $d$  such that  $(d * e) \% \phi(n) = 1$ . One solution is  $d = 3$  [ $(3 * 7) \% 20 = 1$ ]
- Public key is  $(e, n) \Rightarrow (7, 33)$
- Private key is  $(d, n) \Rightarrow (3, 33)$
- The encryption of  $m = 2$  is  $c = 2^7 \% 33 = 29$
- The decryption of  $c = 29$  is  $m = 29^3 \% 33 = 2$