LECTURE NOTES
ON

DIGITAL COMMUNICATIONS

(AEC009)

III B.TECH V semester ECE

(AUTONOMOUS-R16)

Dr.Vinoth

(Professor)

Mr.G.Kiran Kumar

(Assistant Professor)

ELECRTONICS AND COMMUNICATION ENGINEERING

INSTITUTE OF AERONAUTICAL ENGINEERING

(Autonomous)

DUNDIGAL, HYDERABAD - 500043

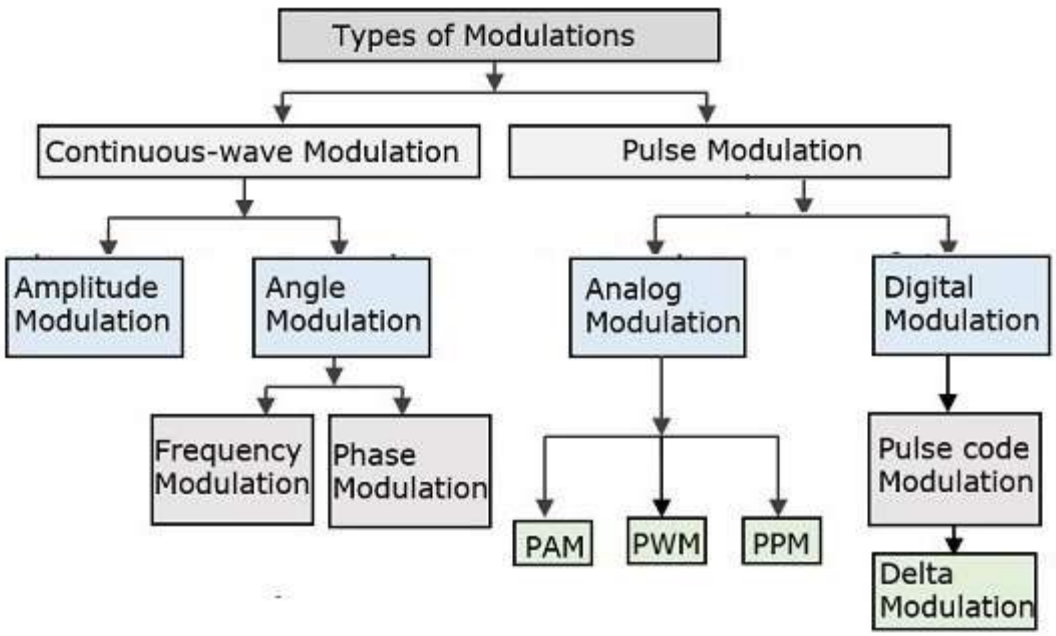
UNIT-I
PULSE DIGITAL MODULATION

Pulse Modulation: Analog pulse modulation, Types of pulse modulation

Many Signals in Modern Communication Systems are digital

- Also, analog signals are transmitted digitally.
- Reduced distortion and improvement in signal to noise ratios.
- PAM, PWM , PPM , PCM and DM.

Types of Modulation – Tree Diagram:



In CW modulation schemes some parameter of modulated wave varies continuously with message. In Analog pulse modulation some parameter of each pulse is modulated by a particular sample value of the message.

- Pulse modulation of two types
- Analog Pulse Modulation
- Pulse Amplitude Modulation (PAM)
- Pulse width Modulation (PWM)
- Pulse Position Modulation (PPM)
- Digital Pulse Modulation
- Pulse code Modulation (PCM)
- Delta Modulation (DM)

Analog Pulse Modulation

Analog pulse modulation results when some attribute of a pulse varies continuously in one-to-one correspondence with a sample value. In analog pulse modulation systems, the amplitude, width, or position of a pulse can vary over a continuous range in accordance with the message amplitude at the sampling instant, as shown in Figure 6.2. These lead to the following
Three types of pulse modulation:

1. Pulse Amplitude Modulation (PAM)
2. Pulse Width Modulation (PWM)
3. Pulse Position Modulation (PPM)

PAM: In this scheme high frequency carrier (pulse) is varied in accordance with sampled value of message signal.

PWM: In this width of carrier pulses are varied in accordance with sampled values of message signal.
Example: Speed control of DC Motors.

PPM: In this scheme position of high frequency carrier pulse is changed in accordance with the sampled values of message signal.

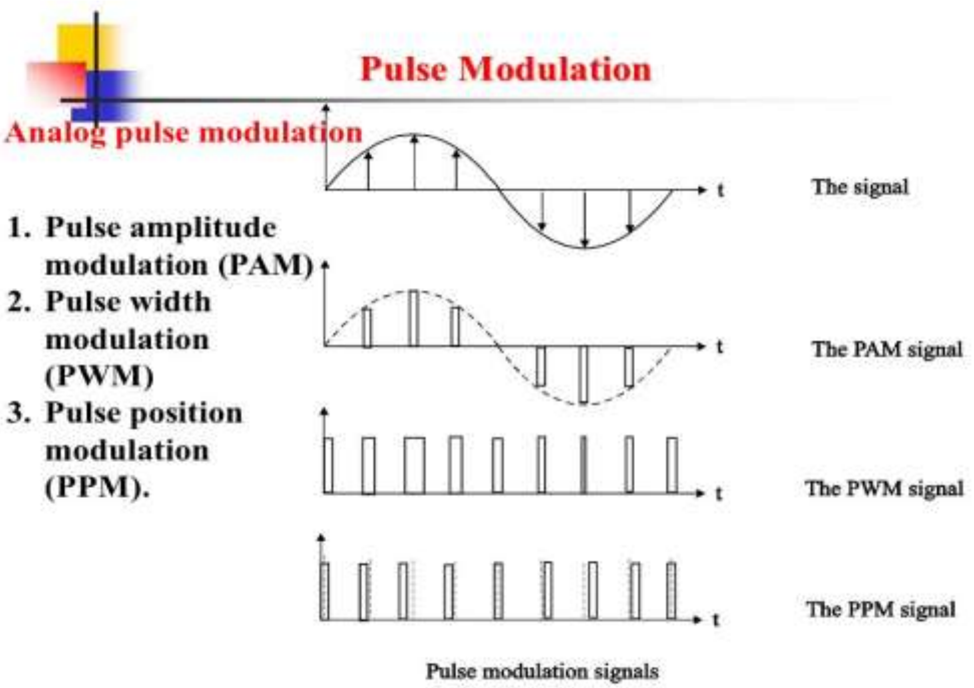


Fig. Representation of Various Analog Pulse Modulations

Digital Pulse Modulation

In systems utilizing digital pulse modulation, the transmitted samples take on only discrete values. Two important types of digital pulse modulation are:

1. Delta Modulation (DM)
2. Pulse Code Modulation (PCM)

PAM (Single polarity, double polarity); Generation & demodulation of PWM

Pulse Amplitude Modulation (PAM):

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

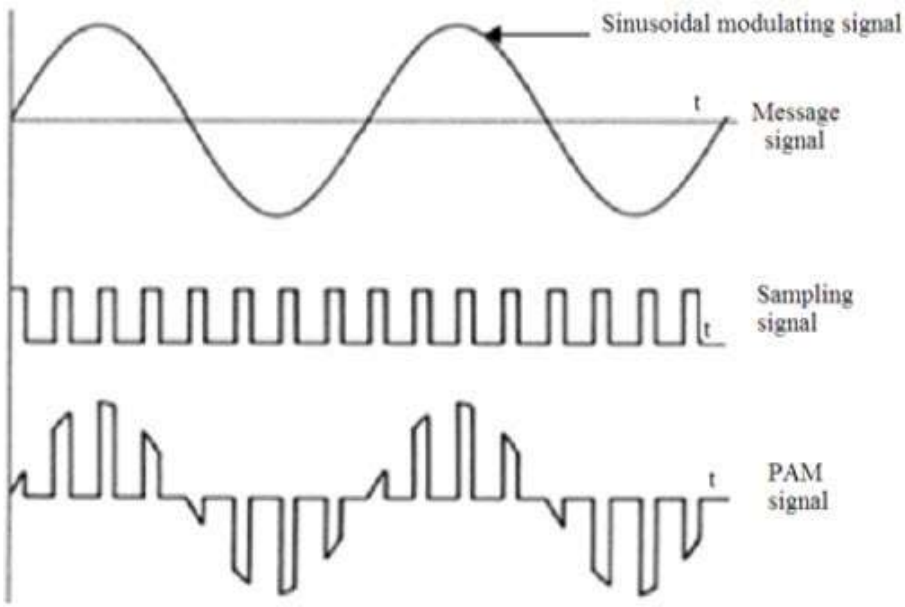


Fig. Pulse Amplitude Modulation Signal

There are two types of sampling techniques for transmitting messages using pulse amplitude modulation, they are

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

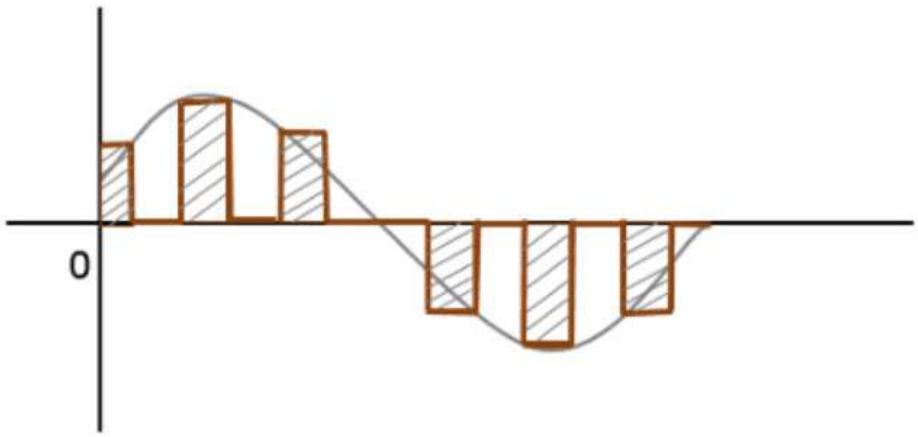


Fig. Flat Top PAM

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

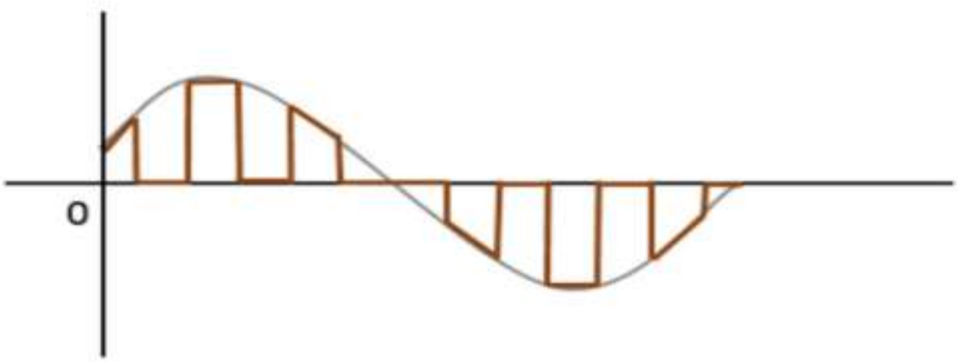


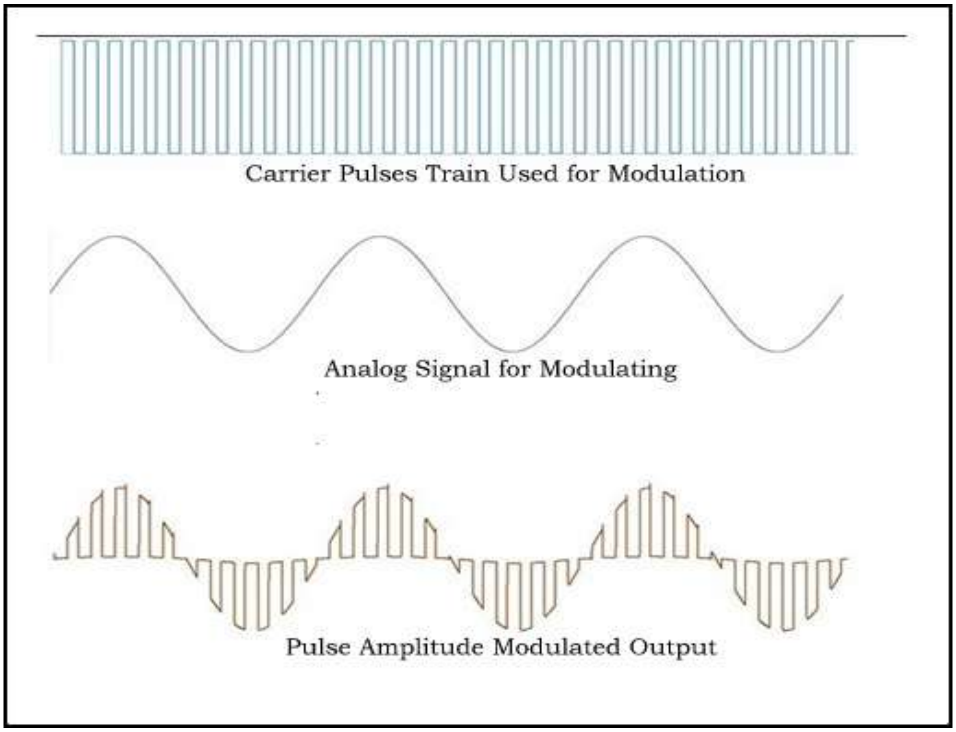
Fig. Natural PAM

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

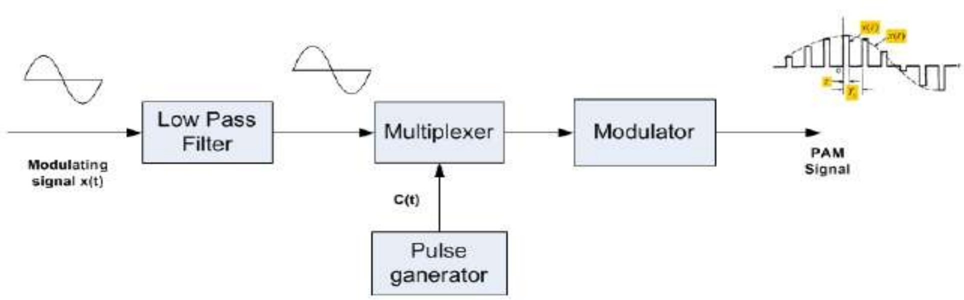
There are two types of pulse amplitude modulation based on signal polarity

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

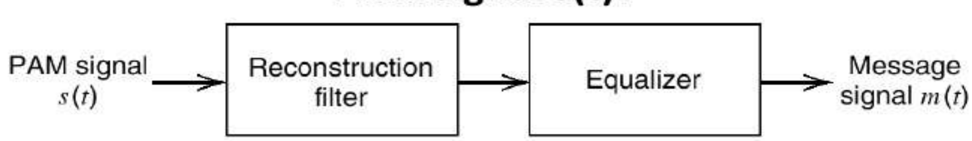
In single polarity pulse amplitude modulation, there is fixed level of DC bias added to the message signal or modulating signal, so the output of modulating signal is always positive. In the double polarity pulse amplitude modulation, the output of modulating signal will have both positive and negative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



Block diagram of PAM generation



System for recovering message signal m(t) from PAM signal s(t).



Advantages of Pulse Amplitude Modulation (PAM):

- It is the base for all digital modulation techniques and it is simple process for both modulation and demodulation technique.
- No complex circuitry is required for both transmission and reception. Transmitter and receiver circuitry is simple and easy to construct.
- PAM can generate other pulse modulation signals and can carry the message or information at same time.

Disadvantages of Pulse Amplitude Modulation (PAM):

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

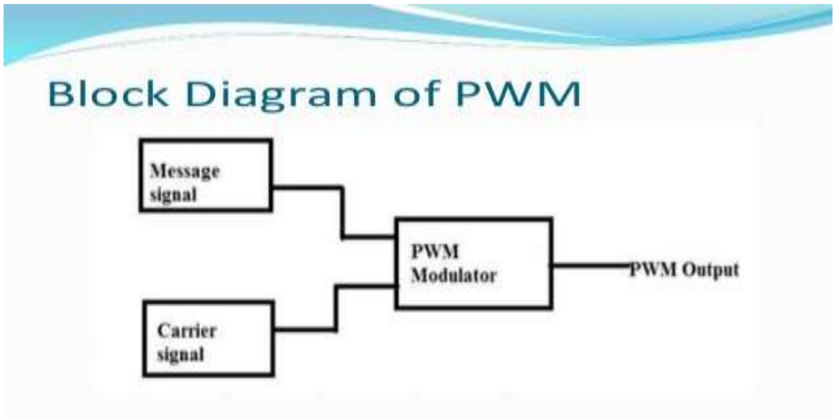
Applications of Pulse Amplitude Modulation (PAM):

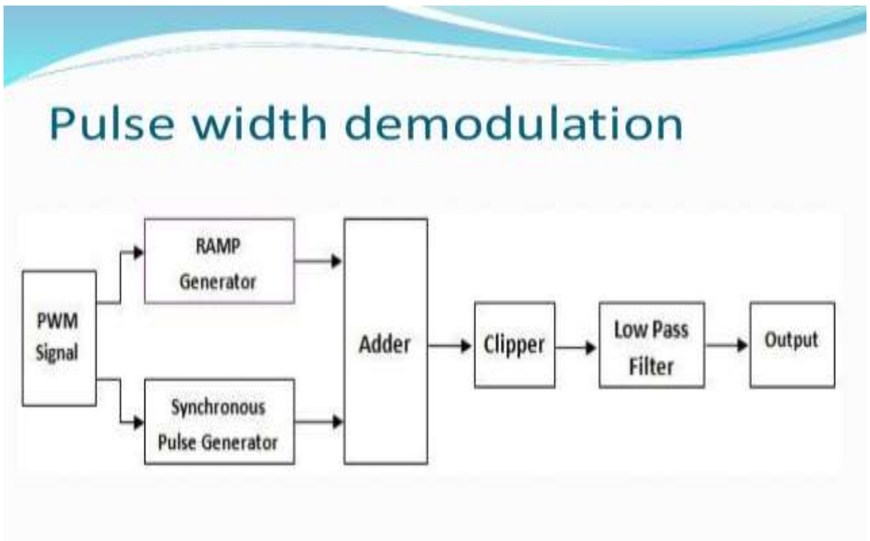
- It is mainly used in Ethernet which is type of computer network communication, we know that we can use Ethernet for connecting two systems and transfer data between the systems. Pulse amplitude modulation is used for Ethernet communications.
- It is also used for photo biology which is a study of photosynthesis.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

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

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

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





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

Advantages of Pulse Width Modulation (PWM):

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

Disadvantages of Pulse Width Modulation (PWM):

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

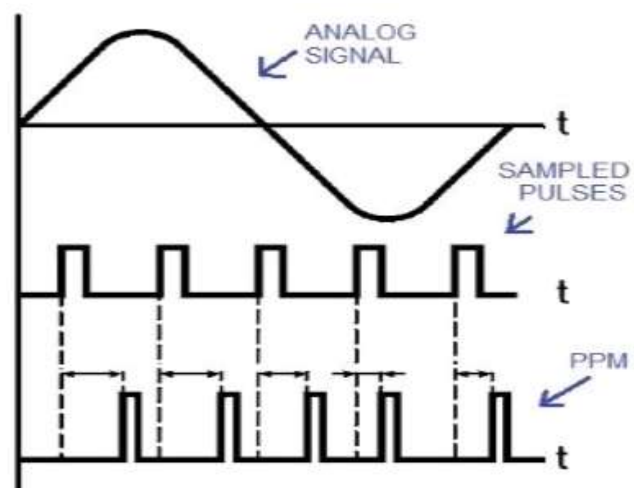
Applications of Pulse Width Modulation (PWM):

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

Generation and demodulation of PPM

Pulse Position Modulation (PPM):

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



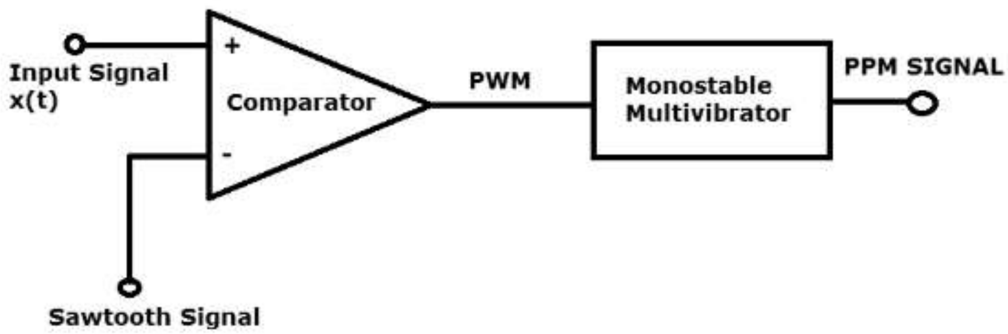
Pulse Position Modulation (PPM)

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

.

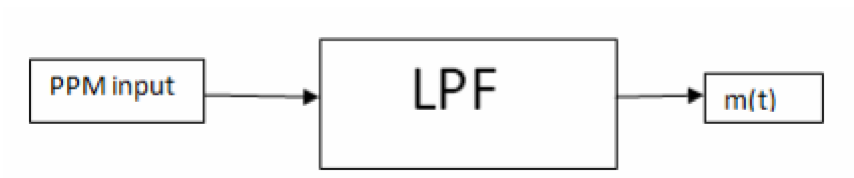
Generation of PPM Signal

1. The PPM Signal can be generated from PWM signal. The PWM pules obtained at the comparator output are applied to a mono stable multivibrator
2. Hence corresponding to each trailing edge of PWM signal, the mono stable output goes high. It remains high for a fixed time decided by its own RC comparator.
3. Thus as the trailing edges of the PWM signal keep shifting in proportion with the modulating signal x(t), the PWM pulses also keep shifting
4. All the PPM pulses have the same with and amplitude. The information is conveyed via changing the portion of pulses.



Generation of PPM Signal

Demodulation of PPM



Advantages of Pulse Position Modulation (PPM):

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

Disadvantages of Pulse Position Modulation (PPM):

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

Applications of Pulse Position Modulation (PPM):

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

Introduction: Elements of digital communication systems, advantages and disadvantages of digital communication systems, applications:

The communication that occurs in our day-to-day life is in the form of signals. These signals, such as sound signals, generally, are analog in nature. When the communication needs to be established over a distance, then the analog signals are sent through wire, using different techniques for effective transmission.

The Necessity of Digitization

The conventional methods of communication used analog signals for long distance communications, which suffer from many losses such as distortion, interference, and other losses including security breach.

In order to overcome these problems, the signals are digitized using different techniques. The digitized signals allow the communication to be more clear and accurate without losses.

The following figure indicates the difference between analog and digital signals. The digital signals consist of **1s** and **0s** which indicate High and Low values respectively.

Representation of Signals

Advantages of Digital Communication

As the signals are digitized, there are many advantages of digital communication over analog communication, such as −

1. The effect of distortion, noise, and interference is much less in digital signals as they are less affected.
2. Digital circuits are more reliable.
3. Digital circuits are easy to design and cheaper than analog circuits.
4. The hardware implementation in digital circuits is more flexible than analog.
5. The occurrence of cross-talk is very rare in digital communication.

6. The signal is un-altered as the pulse needs a high disturbance to alter its properties, which is very difficult.
7. Signal processing functions such as encryption and compression are employed in digital circuits to maintain the secrecy of the information.
8. The probability of error occurrence is reduced by employing error detecting and error correcting codes.
9. Spread spectrum technique is used to avoid signal jamming.
10. Combining digital signals using Time Division Multiplexing (TDM) is easier than combining analog signals using Frequency Division Multiplexing (FDM).
11. The configuring process of digital signals is easier than analog signals.
12. Digital signals can be saved and retrieved more conveniently than analog signals.
13. Many of the digital circuits have almost common encoding techniques and hence similar devices can be used for a number of purposes.
14. The capacity of the channel is effectively utilized by digital signals.

Disadvantages of digital communication:

1). Generally, more bandwidth is required than that for analog systems.

2). Synchronization is required.

3). High power consumption (Due to various stages of conversion).

4). Complex circuit, more sofisticated device making is also drawbacks of digital system.

5). Introduce sampling error

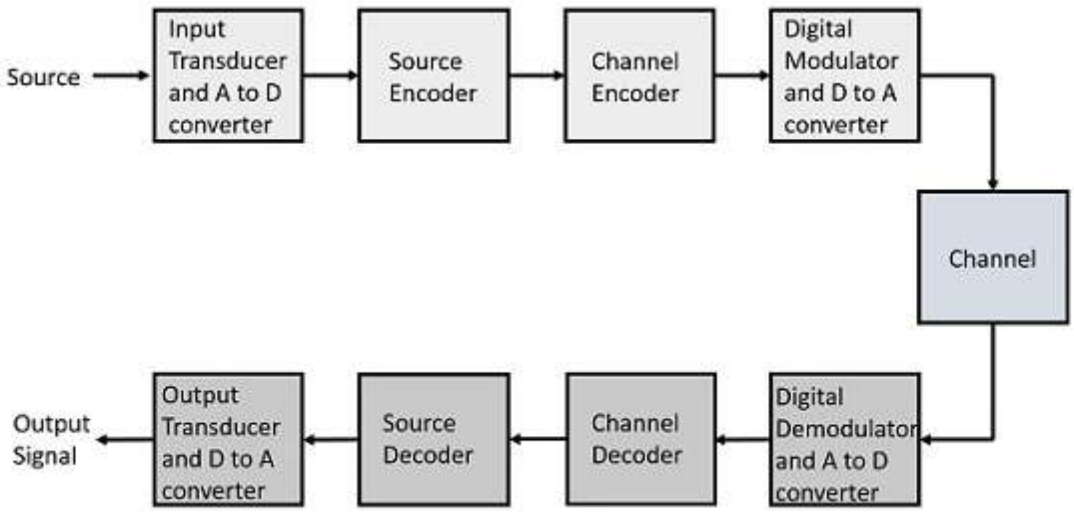
6). As square wave is more affected by noise, That's why while communicating through channel we send sin waves but while operating on device we use squire pulses.

Application of digital communication systems

1. It is used in military application for secure communication and missile guidance.
2. It is used in image processing for pattern recognition, robotic vision and image enhancement.
3. It is used in digital signal processing.
4. The digital communication systems used in telephony for text messaging, etc
5. It is used in space communication where a spacecraft transmits signals to earth.
6. It is used in video compression.
7. It is used in speech processing.
8. It is used in digital audio transmission.
9. It is used in data compression.

Elements of Digital Communication

The elements which form a digital communication system is represented by the following block diagram for the ease of understanding.



Basic Elements of a Digital Communication System

Following are the sections of the digital communication system.

Source

The source can be an **analog** signal. **Example**: A Sound signal

Input Transducer

This is a transducer which takes a physical input and converts it to an electrical signal (**Example**: microphone). This block also consists of an **analog to digital** converter where a digital signal is needed for further processes. A digital signal is generally represented by a binary sequence.

Source Encoder

The source encoder compresses the data into minimum number of bits. This process helps in effective utilization of the bandwidth. It removes the redundant bits (unnecessary excess bits, i.e., zeroes).

Channel Encoder

The channel encoder, does the coding for error correction. During the transmission of the signal, due to the noise in the channel, the signal may get altered and hence to avoid this, the channel encoder adds some redundant bits to the transmitted data. These are the error correcting bits.

Digital Modulator

The signal to be transmitted is modulated here by a carrier. The signal is also converted to analog from the digital sequence, in order to make it travel through the channel or medium.

Channel

The channel or a medium, allows the analog signal to transmit from the transmitter end to the receiver end.

Digital Demodulator

This is the first step at the receiver end. The received signal is demodulated as well as converted again from analog to digital. The signal gets reconstructed here.

Channel Decoder

The channel decoder, after detecting the sequence, does some error corrections. The distortions which might occur during the transmission are corrected by adding some redundant bits. This addition of bits helps in the complete recovery of the original signal.

Source Decoder

The resultant signal is once again digitized by sampling and quantizing so that the pure digital output is obtained without the loss of information. The source decoder recreates the source output.

Output Transducer

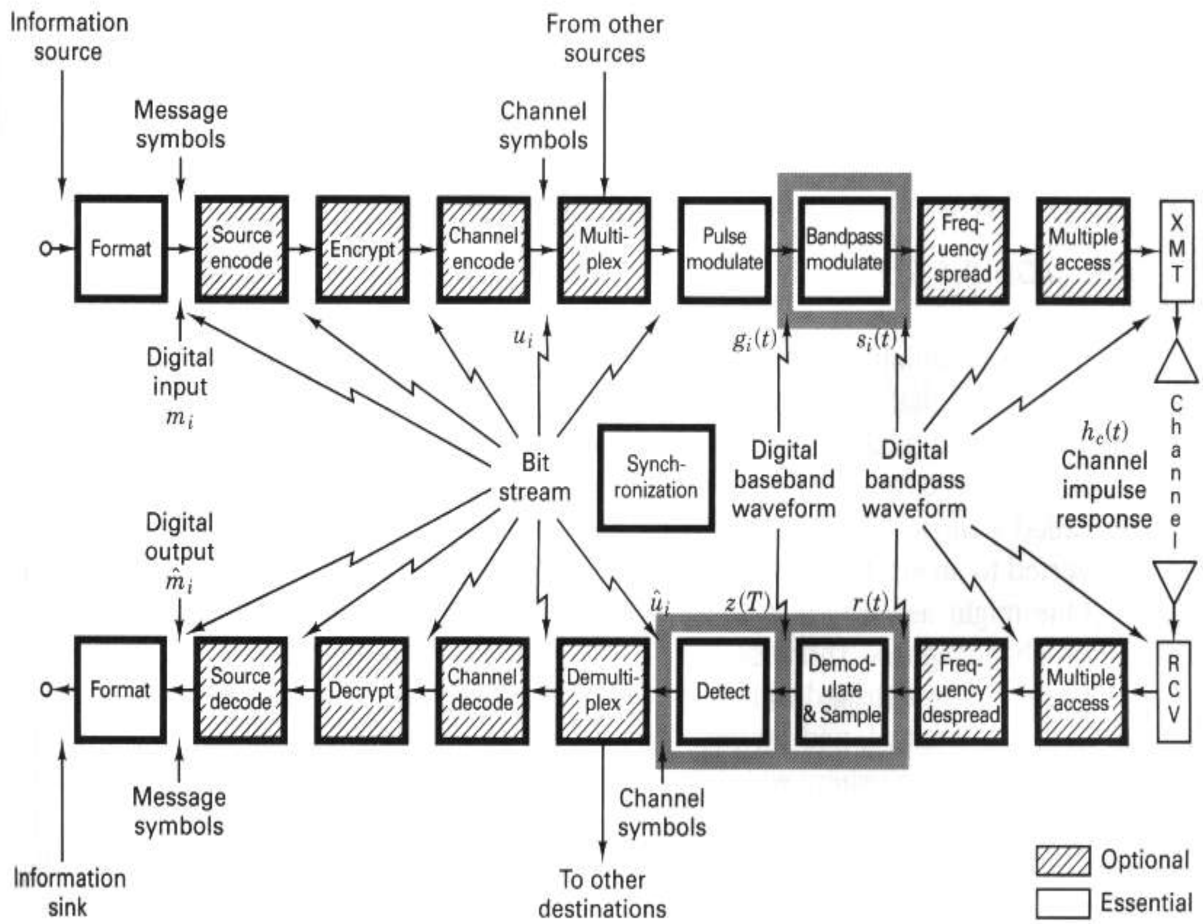
This is the last block which converts the signal into the original physical form, which was at the input of the transmitter. It converts the electrical signal into physical output (**Example**: loud speaker).

Output Signal

This is the output which is produced after the whole process. **Example** − The sound signal received.

This unit has dealt with the introduction, the digitization of signals, the advantages and the elements of digital communications.

Functional block diagram of digital communiation systems


Information source
Message symbols
Channel symbols
From other sources
Format
Source encode
Encrypt
Channel encode
Multi-plex
Pulse modulate
Bandpass modulate
Freq-uency spread
Multiple access
XMT
Digital input m_i
u_i
$g_i(t)$
$s_i(t)$
Bit stream
Synch-ronization
Digital baseband waveform
Digital bandpass waveform
$h_c(t)$ Channel impulse response
Channel
Digital output $\hat{m}_i$
$\hat{u}_i$
$z(T)$
$r(t)$
Format
Source decode
Decrypt
Channel decode
Demulti-plex
Detect
Demod-ulate & Sample
Freq-uency despread
Multiple access
RCV
Message symbols
Channel symbols
Information sink
To other destinations
Optional
Essential

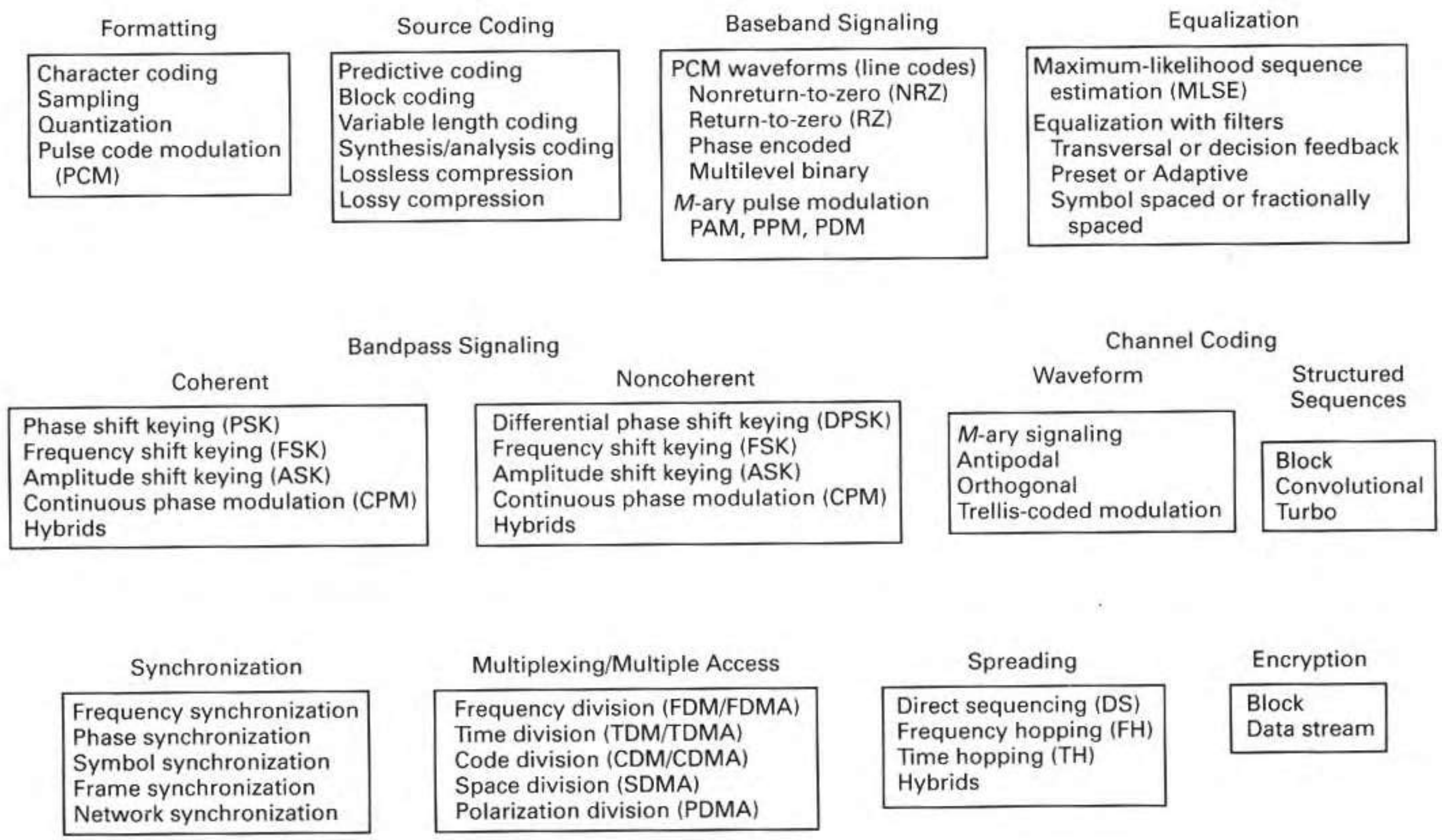


Figure 1.3 Basic digital communication transformations.

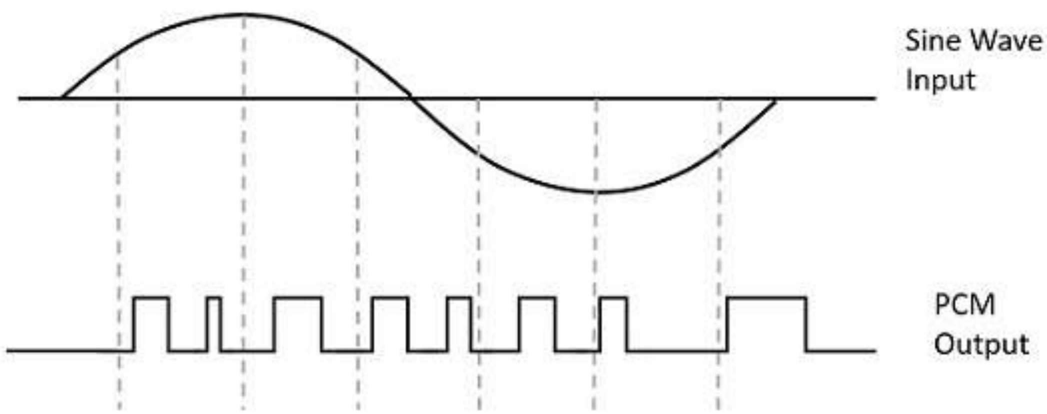
Pulse Digital Modulation: Elements of PCM;

Modulation is the process of varying one or more parameters of a carrier signal in accordance with the instantaneous values of the message signal.

The message signal is the signal which is being transmitted for communication and the carrier signal is a high frequency signal which has no data, but is used for long distance transmission.

There are many modulation techniques, which are classified according to the type of modulation employed. Of them all, the digital modulation technique used is **Pulse Code Modulation (PCM)**.

A signal is pulse code modulated to convert its analog information into a binary sequence, i.e., **1s** and **0s**. The output of a PCM will resemble a binary sequence. The following figure shows an example of PCM output with respect to instantaneous values of a given sine wave.



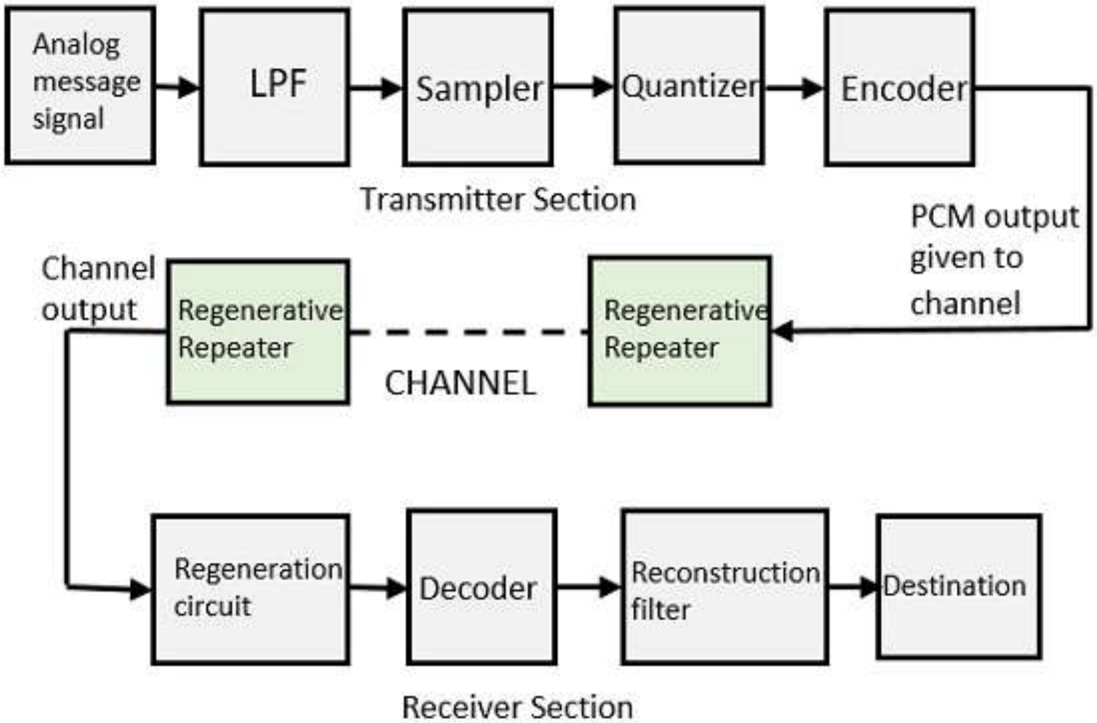
Instead of a pulse train, PCM produces a series of numbers or digits, and hence this process is called as **digital**. Each one of these digits, though in binary code, represent the approximate amplitude of the signal sample at that instant.

In Pulse Code Modulation, the message signal is represented by a sequence of coded pulses. This message signal is achieved by representing the signal in discrete form in both time and amplitude.

Basic Elements of PCM

The transmitter section of a Pulse Code Modulator circuit consists of **Sampling, Quantizing** and **Encoding**, which are performed in the analog-to-digital converter section. The low pass filter prior to sampling prevents aliasing of the message signal.

The basic operations in the receiver section are **regeneration of impaired signals, decoding,** and **reconstruction** of the quantized pulse train. Following is the block diagram of PCM which represents the basic elements of both the transmitter and the receiver sections.



Low Pass Filter

This filter eliminates the high frequency components present in the input analog signal which is greater than the highest frequency of the message signal, to avoid aliasing of the message signal.

Sampler

This is the technique which helps to collect the sample data at instantaneous values of message signal, so as to reconstruct the original signal. The sampling rate must be greater than twice the highest frequency component **W**of the message signal, in accordance with the sampling theorem.

Quantizer

Quantizing is a process of reducing the excessive bits and confining the data. The sampled output when given to Quantizer reduces the redundant bits and compresses the value.

Encoder

The digitization of analog signal is done by the encoder. It designates each quantized level by a binary code. The sampling done here is the sample-and-hold process. These three sections (LPF, Sampler, and Quantizer) will act as an analog to digital converter. Encoding minimizes the bandwidth used.

Regenerative Repeater

This section increases the signal strength. The output of the channel also has one regenerative repeater circuit, to compensate the signal loss and reconstruct the signal, and also to increase its strength.

Decoder

The decoder circuit decodes the pulse coded waveform to reproduce the original signal. This circuit acts as the demodulator.

Reconstruction Filter

After the digital-to-analog conversion is done by the regenerative circuit and the decoder, a low-pass filter is employed, called as the reconstruction filter to get back the original signal.

Hence, the Pulse Code Modulator circuit digitizes the given analog signal, codes it and samples it, and then transmits it in an analog form. This whole process is repeated in a reverse pattern to obtain the original signal.

Bit rate and bandwidth requirements of PCM :The bit rate of a PCM signal can be calculated form the number of bits per sample × the sampling rate. Bit rate $=nb\times fs$ The bandwidth required to transmit this signal depends on the type of line encoding used.

- A digitized signal will always need more bandwidth than the original analog signal. Price we pay for robustness and other features of digital transmission.

Important Relations

- Quantization Noise $(Nq)=\Delta^2/2$
- Signal to Noise ratio
- $(SQNR)=32.22n$ or $SQNR$ in $dB=1.76+6.02n\cong(1.8+6n)dB$
- $Bit\ rate=No.of\ bits\ per\ sample\times sampling\ rate=nfs$
- Bandwidth for PCM signal = n.fm

  Where, n – No. of bits in PCM code

  Fm – signal bandwidth

  fs – sampling rate

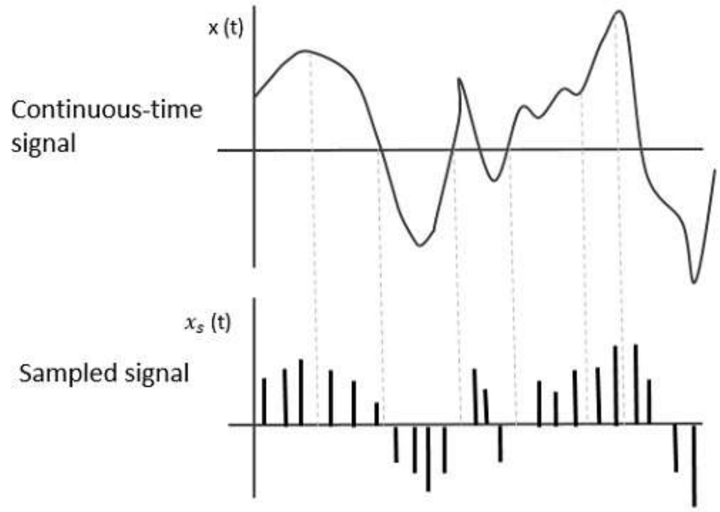
Sampling, quantization and coding

Sampling is defined as, "The process of measuring the instantaneous values of continuous-time signal in a discrete form."

Sample is a piece of data taken from the whole data which is continuous in the time domain.

When a source generates an analog signal and if that has to be digitized, having **1s** and **0s** i.e., High or Low, the signal has to be discretized in time. This discretization of analog signal is called as Sampling.

The following figure indicates a continuous-time signal **x (t)** and a sampled signal $\mathbf{x_s}$ **(t)**. When **x (t)** is multiplied by a periodic impulse train, the sampled signal $\mathbf{x_s}$ **(t)** is obtained.



Sampling Rate

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

SamplingFrequency fs=1/Ts

Where,

- Ts is the sampling time
- fs is the sampling frequency or the sampling rate

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

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

Nyquist Rate

Suppose that a signal is band-limited with no frequency components higher than **W** Hertz. That means, **W** is the highest frequency. For such a signal, for effective reproduction of the original signal, sampling rate should be twice the highest frequency.

This means,

fs=2W

Where,

- fs is the sampling rate
- **W** is the highest frequency

This rate of sampling is called as **Nyquist rate**.

A theorem called, Sampling Theorem, was stated on the theory of this Nyquist rate.

Sampling Theorem

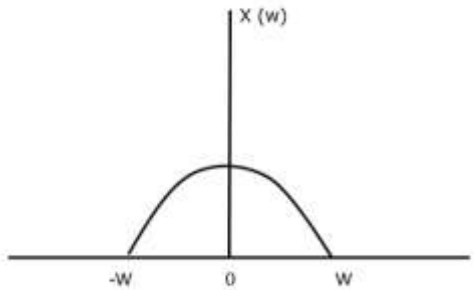
The sampling theorem, which is also called as **Nyquist theorem**, delivers the theory of sufficient sample rate in terms of bandwidth for the class of functions that are band limited.

The sampling theorem states that, "a signal can be exactly reproduced if it is sampled at the rate **f_s** which is greater than twice the maximum frequency **W**."

To understand this sampling theorem, let us consider a band-limited signal, i.e., a signal whose value is **non-zero** between some **–W** and **W** Hertz.

Such a signal is represented as x(f)=0for|f|>W

For the continuous-time signal **x (t)**, the band-limited signal in frequency domain, can be represented as shown in the following figure.

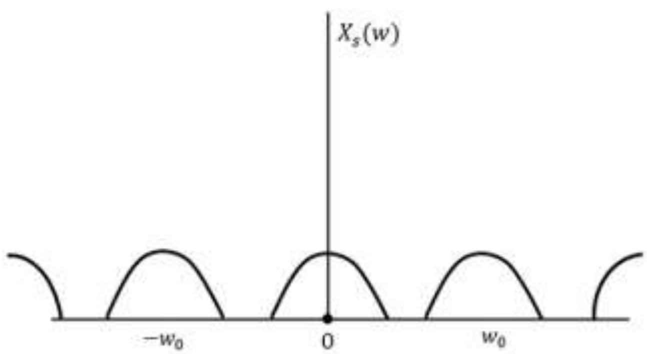


Band limited signal

We need a sampling frequency, a frequency at which there should be no loss of information, even after sampling. For this, we have the Nyquist rate that the sampling frequency should be two times the maximum frequency. It is the critical rate of sampling.

If the signal **x(t)** is sampled above the Nyquist rate, the original signal can be recovered, and if it is sampled below the Nyquist rate, the signal cannot be recovered.

The following figure explains a signal, if sampled at a higher rate than **2w** in the frequency domain.



The above figure shows the Fourier transform of a signal xs(t). Here, the information is reproduced without any loss. There is no mixing up and hence recovery is possible.

The Fourier Transform of the signal xs(t) is

Xs (w)=1Ts∑n=−∞∞X(w−nw0)

Where Ts = **Sampling Period** and w0=2πTs

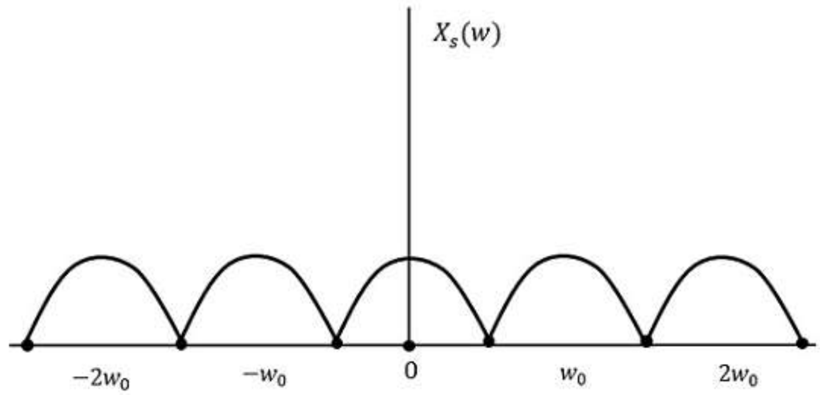
Let us see what happens if the sampling rate is equal to twice the highest frequency (**2W**)

That means,

Fs =2W

Where,

- Fs is the sampling frequency
- **W** is the highest frequency

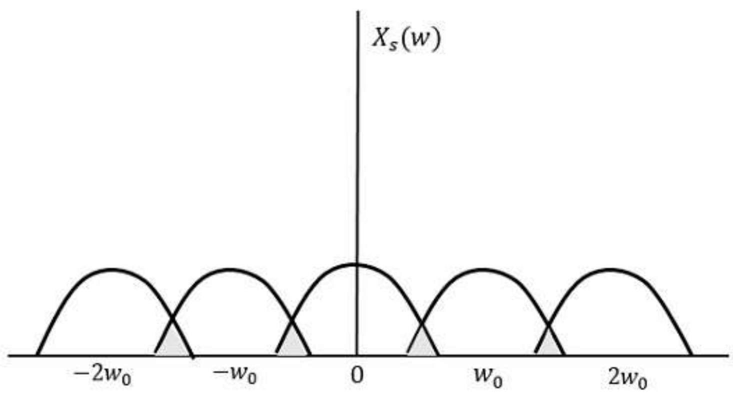


The result will be as shown in the above figure. The information is replaced without any loss. Hence, this is also a good sampling rate.

Now, let us look at the condition,

$$Fs < 2W$$

The resultant pattern will look like the following figure.



We can observe from the above pattern that the over-lapping of information is done, which leads to mixing up and loss of information. This unwanted phenomenon of over-lapping is called as Aliasing.

Aliasing

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

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

- In the transmitter section of PCM, a **low pass anti-aliasing filter** is employed, before the sampler, to eliminate the high frequency components, which are unwanted.
- The signal which is sampled after filtering, is sampled at a rate slightly higher than the Nyquist rate.

This choice of having the sampling rate higher than Nyquist rate, also helps in the easier design of the **reconstruction filter** at the receiver.

Scope of Fourier Transform

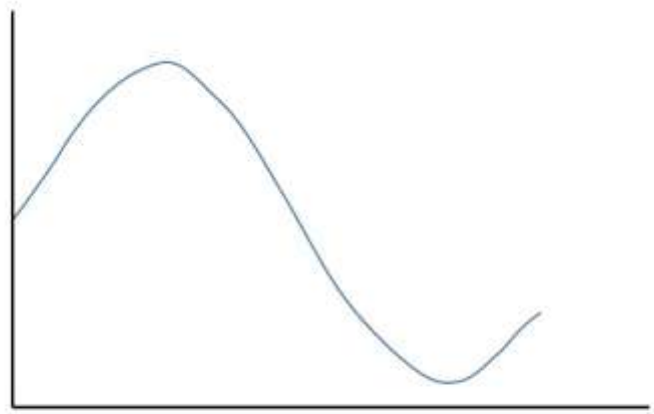
It is generally observed that, we seek the help of Fourier series and Fourier transforms in analyzing the signals and also in proving theorems. It is because −

- The Fourier Transform is the extension of Fourier series for non-periodic signals.
- Fourier transform is a powerful mathematical tool which helps to view the signals in different domains and helps to analyze the signals easily.
- Any signal can be decomposed in terms of sum of sines and cosines using this Fourier transform.

The digitization of analog signals involves the rounding off of the values which are approximately equal to the analog values. The method of sampling chooses a few points on the analog signal and then these points are joined to round off the value to a near stabilized value. Such a process is called as **Quantization**.

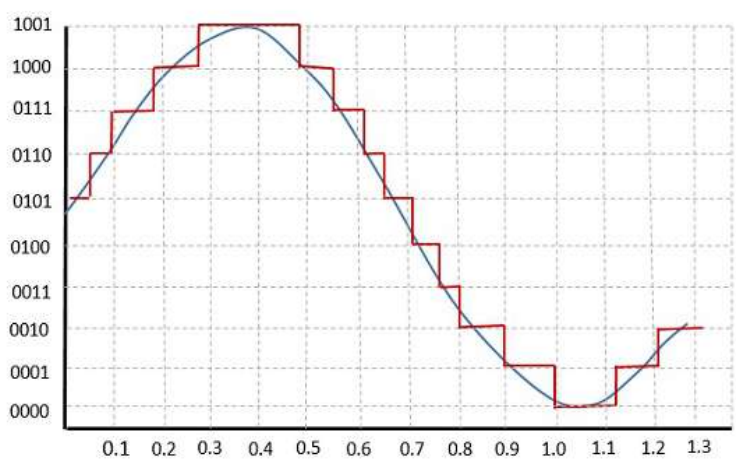
Quantizing an Analog Signal

The analog-to-digital converters perform this type of function to create a series of digital values out of the given analog signal. The following figure represents an analog signal. This signal to get converted into digital has to undergo sampling and quantizing.



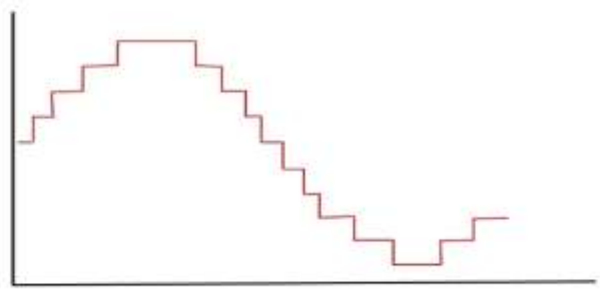
The quantizing of an analog signal is done by discretizing the signal with a number of quantization levels. **Quantization** is representing the sampled values of the amplitude by a finite set of levels, which means converting a continuous-amplitude sample into a discrete-time signal.

The following figure shows how an analog signal gets quantized. The blue line represents analog signal while the brown one represents the quantized signal.



Both sampling and quantization result in the loss of information. The quality of a Quantizer output depends upon the number of quantization levels used. The discrete amplitudes of the quantized output are called as **representation levels** or **reconstruction levels**. The spacing between the two adjacent representation levels is called a **quantum** or **step-size**.

The following figure shows the resultant quantized signal which is the digital form for the given analog signal.



This is also called as **Stair-case** waveform, in accordance with its shape.

Types of Quantization

There are two types of Quantization - Uniform Quantization and Non-uniform Quantization.The type of quantization in which the quantization levels are uniformly spaced is termed as a **Uniform Quantization**. The type of quantization in which the quantization levels are unequal and mostly the relation between them is logarithmic, is termed as a **Non-uniform Quantization**.

There are two types of uniform quantization. They are Mid-Rise type and Mid-Tread type. The following figures represent the two types of uniform quantization.

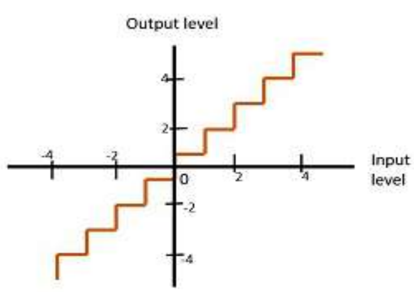


Fig 1 : Mid-Rise type Uniform Quantization

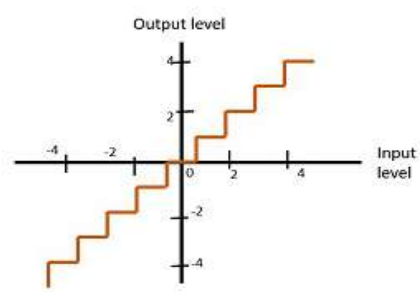


Fig 2 : Mid-Tread type Uniform Quantization

Figure 1 shows the mid-rise type and figure 2 shows the mid-tread type of uniform quantization.

- The **Mid-Rise** type is so called because the origin lies in the middle of a raising part of the stair-case like graph. The quantization levels in this type are even in number.
- The **Mid-tread** type is so called because the origin lies in the middle of a tread of the stair-case like graph. The quantization levels in this type are odd in number.
- Both the mid-rise and mid-tread type of uniform quantizer are symmetric about the origin.

$$\Delta = (max - min)L$$
$$nb = log2L$$

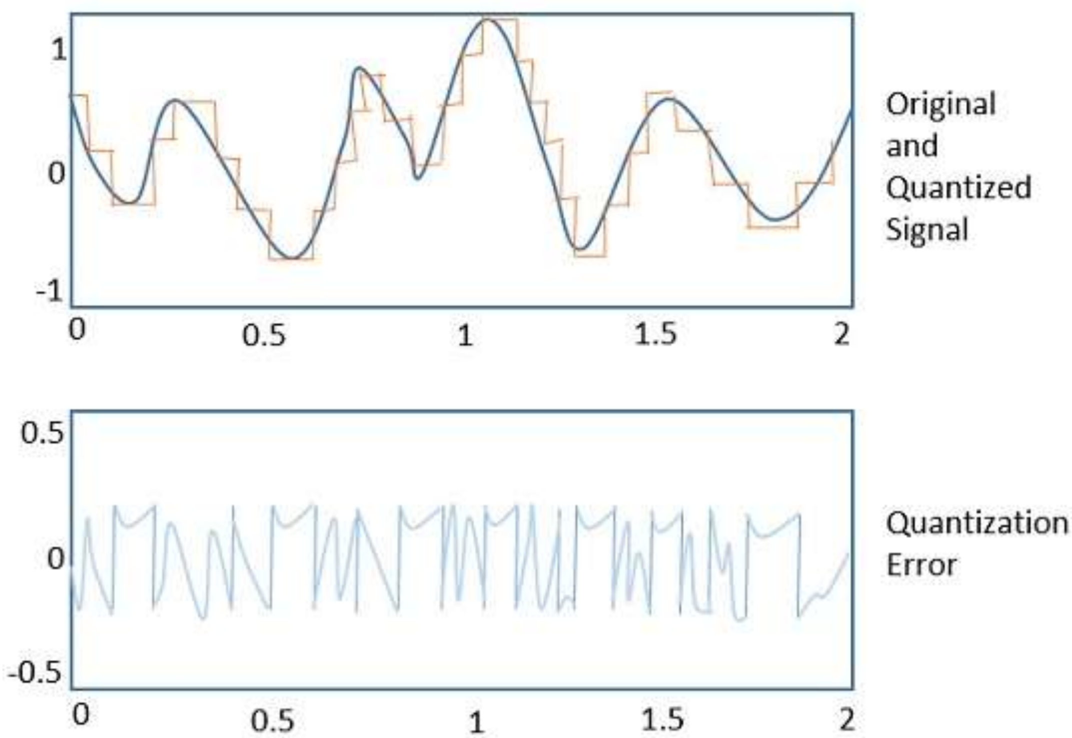
Quantization error, non-uniform quantization and companding

Quantization Error

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

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

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



Quantization Noise

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

Companding in PCM

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

There are two types of Companding techniques. They are −

A-law Companding Technique

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

- A-law Companding is used for PCM telephone systems.

μ-law Companding Technique

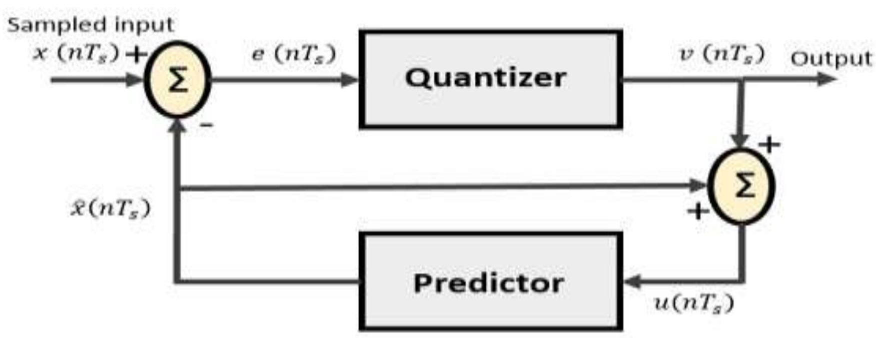
- Uniform quantization is achieved at **μ = 0**, where the characteristic curve is linear and no compression is done.
- μ-law has mid-tread at the origin. Hence, it contains a zero value.
- μ-law companding is used for speech and music signals.

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

Differential PCM (DPCM)

DPCM Transmitter

The DPCM Transmitter consists of Quantizer and Predictor with two summer circuits. Following is the block diagram of DPCM transmitter.



The signals at each point are named as −

- x(nTs) is the sampled input
- x^(nTs) is the predicted sample
- e(nTs) is the difference of sampled input and predicted output, often called as prediction error
- v(nTs) is the quantized output
- u(nTs) is the predictor input which is actually the summer output of the predictor output and the quantizer output

The predictor produces the assumed samples from the previous outputs of the transmitter circuit. The input to this predictor is the quantized versions of the input signal x(nTs).

Quantizer Output is represented as −

$$v(nTs)=Q[e(nTs)]$$
$$=e(nTs)+q(nTs)$$

Where **q (nT_s)** is the quantization error

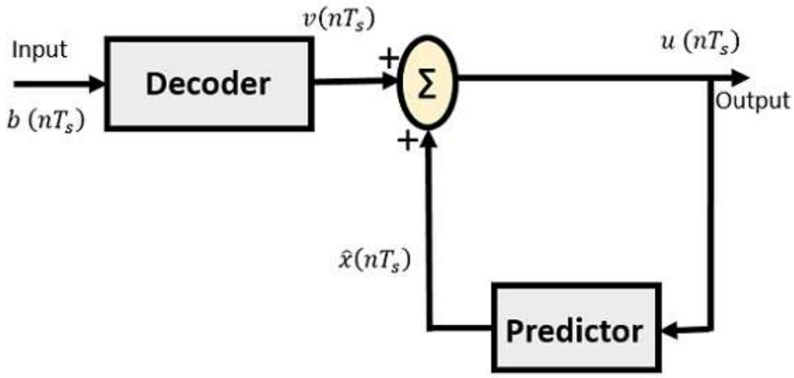
Predictor input is the sum of quantizer output and predictor output,

$$u(nTs)=\hat{x}(nTs)+v(nTs)$$
$$u(nTs)=\hat{x}(nTs)+e(nTs)+q(nTs)$$
$$u(nTs)=x(nTs)+q(nTs)$$

The same predictor circuit is used in the decoder to reconstruct the original input.

DPCM Receiver

The block diagram of DPCM Receiver consists of a decoder, a predictor, and a summer circuit. Following is the diagram of DPCM Receiver.



The notation of the signals is the same as the previous ones. In the absence of noise, the encoded receiver input will be the same as the encoded transmitter output.As mentioned before, the predictor assumes a value, based on the previous outputs. The input given to the decoder is processed and that output is summed up with the output of the predictor, to obtain a better output.

The sampling rate of a signal should be higher than the Nyquist rate, to achieve better sampling. If this sampling interval in Differential PCM is reduced considerably, the sampleto-sample amplitude difference is very small, as if the difference is **1-bit quantization**, then the step-size will be very small i.e., **Δ** (delta).

Advantages of Dpcm:

1) Bandwidth Requirement Of Dpcm Is Less Compared To PCM
2) Quantization Error Is Reduced Because Of Prediction Filter.
3) Numbers Of Bits Used To Represent .One Sample Value Are Also Reduced Compared To Pcm.

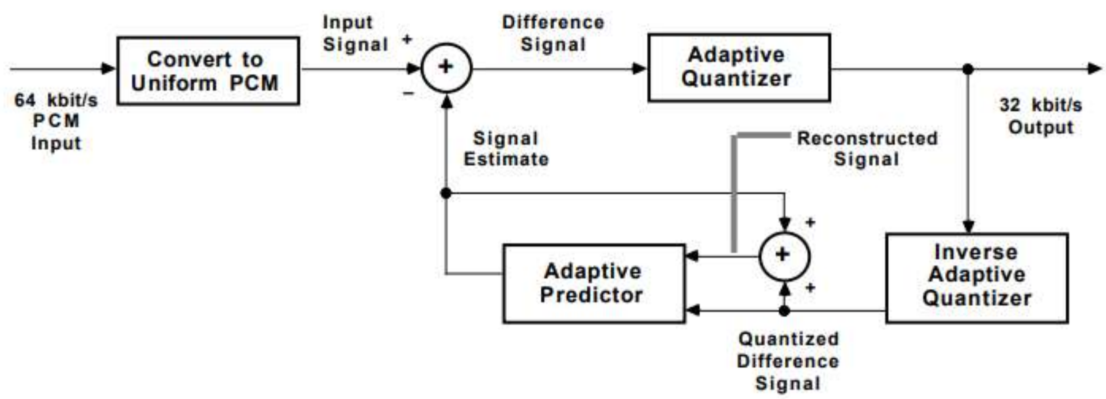
Adaptive DPCM; Delta modulation and its drawbacks;

Adaptive DPCM

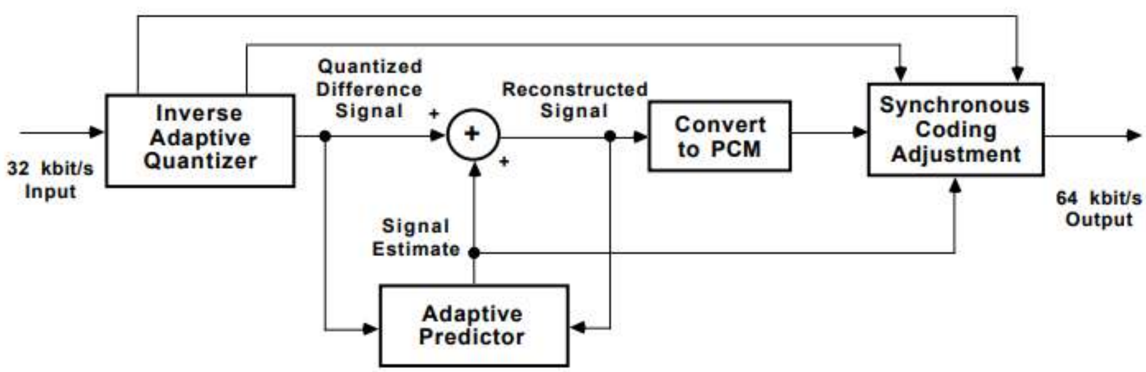
Pulse code modulation (PCM) samples an input signal using a fixed quantizer to produce a digital representation. This technique, although simple to implement, does not take advantage of any of the redundancies in speech signals. The value of the current input sample does not have an effect on the coding of future samples. Adaptive differential PCM (ADPCM), on the other hand, uses an adaptive predictor, one that adjusts according to the value of each input sample, and thereby reduces the number of bits required to represent the data sample from eight (non adaptive PCM) to four. ADPCM does not transmit the value of the speech sample, but rather the difference between a predicted value and the actual sample value. Typically, an ADPCM transcoder is inserted into a PCM system to increase its voice channel capacity. Therefore, the ADPCM encoder accepts PCM values as input, and the ADPCM decoder outputs PCM values.

Adaptive differential pulse code modulation is a very efficient digital coding of waveforms that was developed by Bell Labs in the 1970s for the purpose of voice coding. The concept of ADPCM is to use the past behavior of a signal to forecast it in the future. The resulting signal will represent the error of the prediction, which has no significance. Therefore, the signal must be decoded to rebuild a more meaningful original waveform.

The ADPCM technique is employed to send sound signals through fiber-optic long-distance lines. In the telecommunication field, the ADPCM technique is used mainly in speech compression because the method makes it possible to reduce bit flow without compromising quality. The ADPCM method can be applied to all wave forms, high-quality audio, images and other modern data.



ENCODER



DECODER

The encoder and decoder update their internal variables based on only the generated ADPCM value. This ensures that the encoder and decoder operate in synchronization without the need to send any additional or sideband data. A full decoder is embedded within the encoder to ensure that all variables are updated based on the same data. In the receiving decoder as well as the decoder embedded in the encoder, the transmitted ADPCM value is used to update the inverse adaptive quantizer, which produces a dequantized version of the difference signal. This dequantized value is added to the value generated by the adaptive predictor to produce the reconstructed speech sample. This value is the output of the decoder. The adaptive predictor computes a weighted average of the last six dequantized difference values and the last two predicted values. The coefficients of the filter are updated based on their previous values, the current difference value, and other derived values.

Delta modulation and its drawbacks

The sampling rate of a signal should be higher than the Nyquist rate, to achieve better sampling. If this sampling interval in Differential PCM is reduced considerably, the sampleto-sample amplitude difference is very small, as if the difference is **1-bit quantization**, then the step-size will be very small i.e., **Δ** (delta).

Delta Modulation

The type of modulation, where the sampling rate is much higher and in which the step size after quantization is of a smaller value **Δ**, such a modulation is termed as **delta modulation**.

Features of Delta Modulation

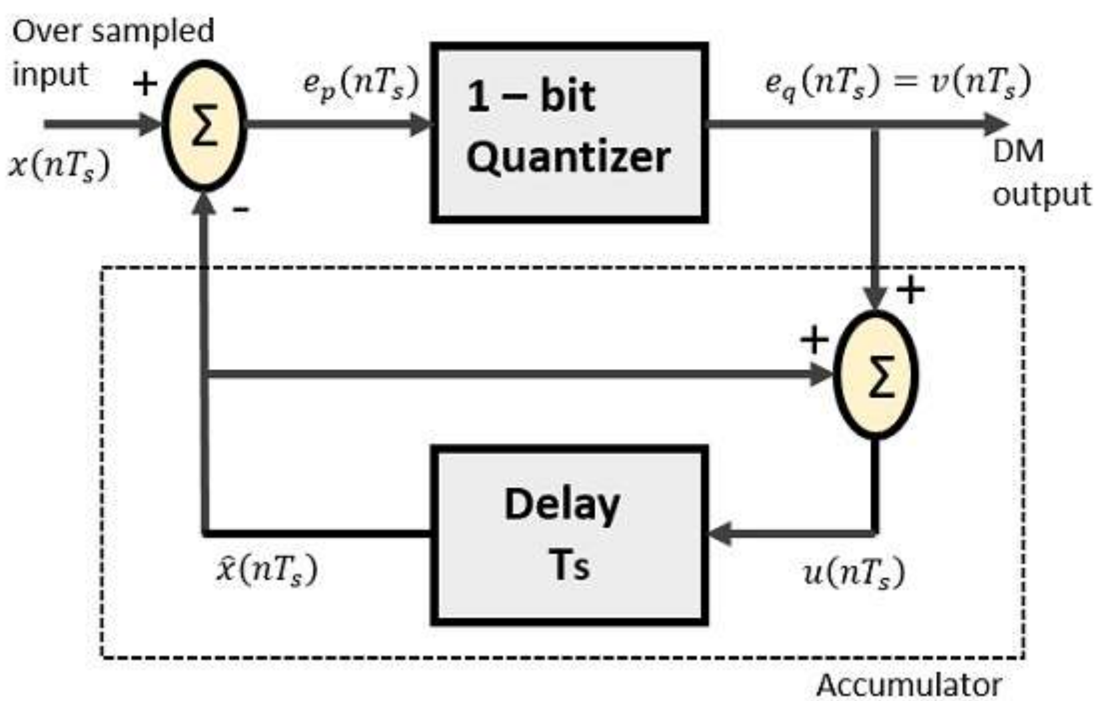
Following are some of the features of delta modulation.

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

Delta Modulation is a simplified form of DPCM technique, also viewed as **1-bit DPCM scheme**. As the sampling interval is reduced, the signal correlation will be higher.

Delta Modulator

The Delta Modulator comprises of a 1-bit quantizer and a delay circuit along with two summer circuits. Following is the block diagram of a delta modulator.



The predictor circuit in DPCM is replaced by a simple delay circuit in DM.

From the above diagram, we have the notations as −

- x(nTs) = over sampled input
- ep(nTs) = summer output and quantizer input
- eq(nTs) = quantizer output = v(nTs)
- x^(nTs) = output of delay circuit
- u(nTs) = input of delay circuit

Using these notations, now we shall try to figure out the process of delta modulation.

ep(nTs)=x(nTs)−xˆ(nTs)---------equation 1

=x(nTs)−u([n−1]Ts)

=x(nTs)−[xˆ[[n−1]Ts]+v[[n−1]Ts]]---------equation 2

Further,

v(nTs)=eq(nTs)=S.sig.[ep(nTs)]---------equation 3

u(nTs)=xˆ(nTs)+eq(nTs)

Where,

- xˆ(nTs) = the previous value of the delay circuit
- eq(nTs) = quantizer output = v(nTs)

Hence,

u(nTs)=u([n−1]Ts)+v(nTs)---------equation 4

Which means?

The present input of the delay unit = (The previous output of the delay unit) + (the present quantizer output)

Assuming zero condition of Accumulation,

u(nTs)=S∑j=1nsig[ep(jTs)]

Accumulated version of DM output = ∑j=1nv(jTs)---------equation 5

Now, note that

xˆ(nTs)=u([n−1]Ts)

=∑j=1n−1v(jTs)---------equation 6

Delay unit output is an Accumulator output lagging by one sample.

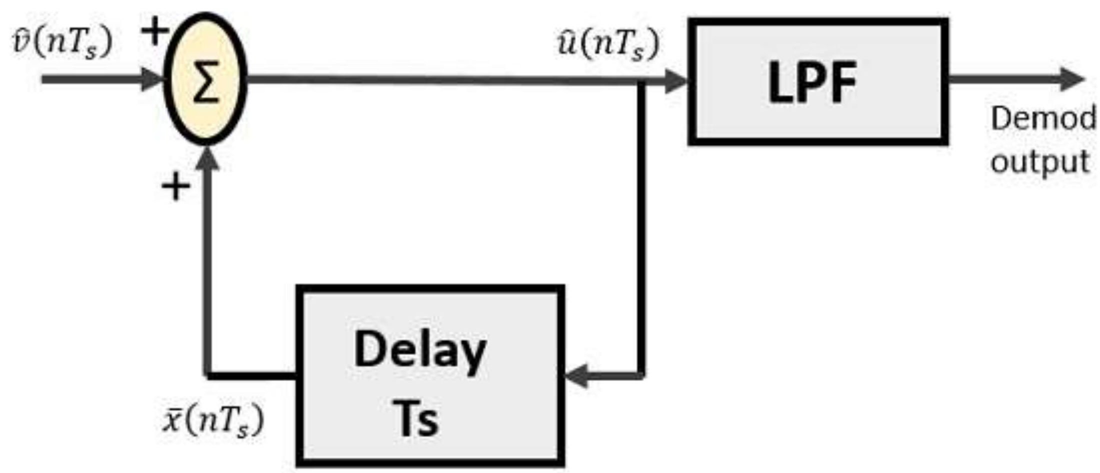
From equations 5 & 6, we get a possible structure for the demodulator.

A Stair-case approximated waveform will be the output of the delta modulator with the step-size as delta (**Δ**). The output quality of the waveform is moderate.

Delta Demodulator

The delta demodulator comprises of a low pass filter, a summer, and a delay circuit. The predictor circuit is eliminated here and hence no assumed input is given to the demodulator.

Following is the diagram for delta demodulator.



From the above diagram, we have the notations as −

- v^(nTs) is the input sample
- u^(nTs) is the summer output
- x¯(nTs) is the delayed output

A binary sequence will be given as an input to the demodulator. The stair-case approximated output is given to the LPF.

Low pass filter is used for many reasons, but the prominent reason is noise elimination for out-of-band signals. The step-size error that may occur at the transmitter is called **granular noise**, which is eliminated here. If there is no noise present, then the modulator output equals the demodulator input.

Advantages of DM Over DPCM

- 1-bit quantizer
- Very easy design of the modulator and the demodulator

However, there exists some noise in DM.

- Slope Over load distortion (when **Δ** is small)
- Granular noise (when **Δ** is large)

Benefits or advantages of Delta Modulation

Following are the benefits or **advantages of Delta Modulation**:

➨In Delta modulation electronic circuit requirement for modulation at transmitter and for demodulation at receiver is substantially simpler compare to PCM.

➨In delta modulation, amplitude of speech signal does not exceed maximum sinusoidal amplitude.

➨PCM has sampling rate higher than nyquist rate. The encode signal contains redundant information. DPCM can efficiently remove this redundancy.

➨DPCM needs less number of quantization levels and hence less number of bits are needed to represent them.

➨Signaling rate and bandwidth of DPCM or delta modulation is less than PCM technique.

Drawbacks or disadvantages of Delta Modulation

Following are the drawbacks or **disadvantages of Delta Modulation**:

➨If changes in signal is less than the step size, then modulator no longer follow signal. Thus produces train of alternating positive and negative pulses.

➨Modulator overloads when slope of signal is too high.

➨High bit rate.

➨It requires predictor circuit and hence it is very complex.

➨Its practical usage is limited.

Delta modulation has two major drawbacks that are:

1. **Slope overload distortion**
   This distortion arises because of large dynamic range of input signal.

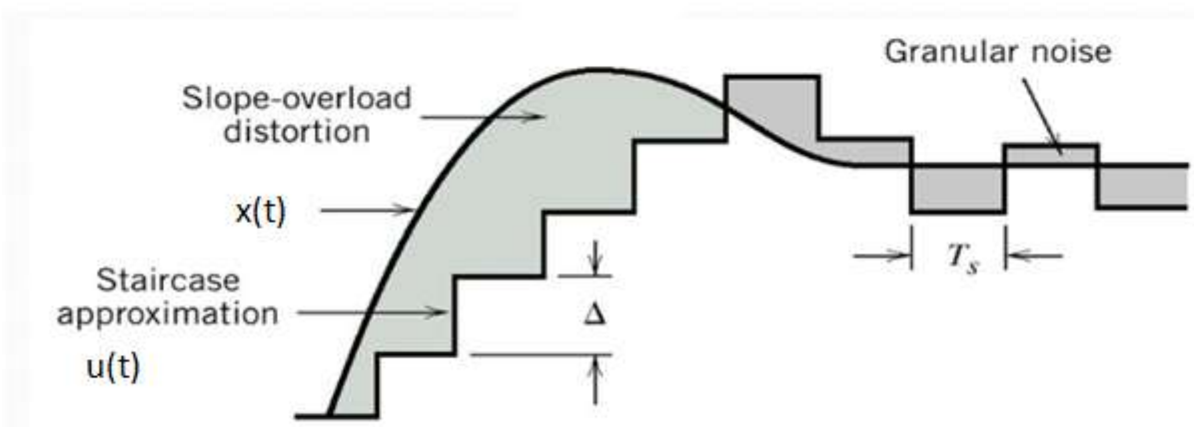


Fig.1: Quantization Errors in Delta Modulation

We can observe from fig.1 , the rate of rise of input signal x(t) is so high that the staircase signal can not approximate it, the step size 'Δ' becomes too small for staircase signal u(t) to follow the step segment of x(t).Hence, there is a large error between the staircase approximated signal and the original input signal x(t).This error or noise is known as **slope overload distortion** .To reduce this error, the step size must be increased when slope of signal x(t) is high. Since the step size of delta modulator remains fixed, its maximum or minimum slopes occur along straight lines. Therefore, this modulator is known as Linear Delta Modulator (LDM).

2. **Granular noise**
   Granular noise occurs when step size is too large compared to small variations in the input signal. This means that for very small variations in the input signal, the staircase signal is changed by large amount because of large step size. The error between the input and approximated signal is called granular noise. The solution to this problem is to make step size small. Adaptive Delta Modulation
   To overcome the quantization error due to slope overload distortion and granular noise, the step size (Δ) is made adaptive to variations in input signal x(t). Particularly in the step segment of the x(t) , the step size is increased. Also, if the input is varying slowly, the step size is reduced. Then

this method is known as Adaptive Delta Modulation (ADM). The adaptive delta modulators can take continuous changes in the step size or discrete changes in the step size

Adaptive delta modulation;

In digital modulation, we have come across certain problem of determining the step-size, which influences the quality of the output wave.A larger step-size is needed in the steep slope of modulating signal and a smaller stepsize is needed where the message has a small slope. The minute details get missed in the process. So, it would be better if we can control the adjustment of step-size, according to our requirement in order to obtain the sampling in a desired fashion. This is the concept of Adaptive Delta Modulation.The performance of a delta modulator can be improved significantly by making the step size of the modulator assume a time-varying form. In particular, during a steep segment of the input signal the step size is increased. Conversely, when the input signal is varying slowly, the step size is reduced. In this way, the size is adapted to the level of the input signal. The resulting method is called adaptive delta modulation (ADM). There are several types of ADM, depending on the type of scheme used for adjusting the step size. In this ADM, a discrete set of values is provided for the step size. Fig.3.17 shows the block diagram of the transmitter and receiver of an ADM System. In practical implementations of the system, the step size

$$\Delta(nT_s) \text{ or } 2\delta(nT_s)$$

is constrained to lie between minimum and maximum values.

The upper limit, δ_{max}, controls the amount of slope-overload distortion. The lower limit, δ_{min}, controls the amount of idle channel noise. Inside these limits, the adaptation rule for $\delta(nT_s)$ is expressed in the general form

$$\delta(nTs) = g(nTs).\ \delta(nTs - Ts) \qquad \text{------ (3.55)}$$

where the time-varying multiplier $g(nT_s)$ depends on the present binary output $b(nT_s)$ of the delta modulator and the M previous values $b(nT_s - T_s)$,.......$b(nT_s - MT_s)$.

This adaptation algorithm is called a constant factor ADM with one-bit memory, where the term "one bit memory" refers to the explicit utilization of the single pervious bit $b(nT_s - T_s)$ because equation (3.55) can be written as,

$$\begin{aligned} g(nTs) &= K \quad \text{if } b(nTs) = b(nTs - Ts) \\ g(nTs) &= K^{-1} \quad \text{if } b(nTs) = b(nTs - Ts) \qquad \text{------ (3.56)} \end{aligned}$$

This algorithm of equation (3.56), with K=1.5 has been found to be well matched to typically speech and image inputs alike, for a wide range of bit rates.

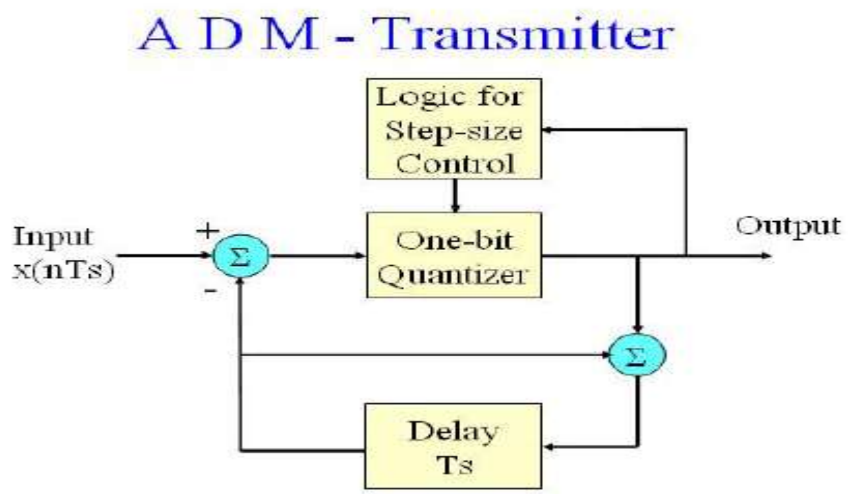


Figure: 3.17a) Block Diagram of ADM Transmitter.

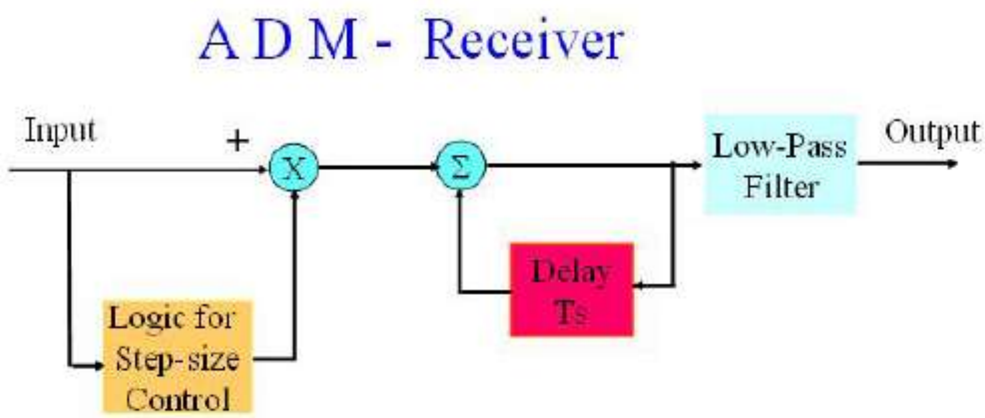


Figure: 3.17 b): Block Diagram of ADM Receiver.

Comparison of PCM and DM systems:

When the analog signal is sampled, it can be quantized and encoded by any one of the following techniques-

i) Pulse code modulation (PCM)
ii) Delta Modulation (DM)
iii) Differential pulse code modulation (DPCM)

PCM: The analog speech waveform is sampled and converted directly into a multibit digital code by an A/D converter. The code is stored and subsequently recalled for playback

DM: Only a single bit is stored for each sample. This bit 1 or 0, represents a greater than or less than condition, respectively as compared to the previous sample. An integrator is then used on the output to convert the stored nit stream to an analog signal.

DPCM: Stores a multibit difference value. A bipolar D/A converter is used for playback to convert the successive difference values to an analog waveform.

ADPCM: Stores a difference value that has been mathematically adjusted according to the slope of the input waveform. Bipolar D/A converter is used to convert the stored digital code to analog for playback.

These techniques convert an analog pulse to its digital equivalent. The digital information is then transmitted over the channel. The major difference among the techniques are given below-

Sr. No.	Parameter	PCM	Delta modulation (DM)	Adaptive Delta Modulation (ADM)	Differential Pulse Code Modulation (DPCM)
1.	Number of bits	It can use 4, 8 or 16 bits per sample.	It uses only one bit for one sample.	Only one bit is used to encode one sample.	Bits can be more than one but are less than PCM.
2.	Levels, step size	The number of levels depend on number of bits. Level size is fixed.	Step size is fixed and cannot be varied.	According to the signal variation, step size varies (Adapted).	Fixed number of levels are used.
3.	Quantization error and distortion	Quantization error depends on number of levels used.	Slope overload distortion and granular noise is present.	Quantization error is present but other errors are absent.	Slope overload distortion and quantization noise is present.
4.	Bandwidth of transmission channel	Highest bandwidth is required since number of bits are high.	Lowest bandwidth is required.	Lowest bandwidth is required.	Bandwidth required is lower than PCM.
5	Feedback.	There is no feedback in transmitter or receiver.	Feedback exists in transmitter.	Feedback exists.	Feedback exists.
6	Complexity of notation	System is complex.	Simple.	Simple.	Simple.
7.	Signal to noise ratio	Good.	Poor.	Better than DM.	Fair.
8.	Area of applications	Audio and video Telephony.	Speech and images.	Speech and images.	Speech and video.

Table 2.4.1 Comparison between PCM, Adaptive Delta Modulation and Differential Pulse Code Modulation

Comparison of PAM,PWM and PPM systems

S.No	Pulse Amplitude Modulation (PAM)	Pulse Duration/Width Modulation (PDM/PWM)	Pulse Position Modulation (PPM)
1	Amplitude of the pulse proportional to amplitude of modulating signal	Width of the pulse is proportional to amplitude of modulating signal	The relative position of the pulse is proportional to amplitude of modulating signal
2	Bandwidth of the transmission channel depends on the pulse width	Bandwidth of the transmission channel depends on the rise time of the pulse	Bandwidth of the transmission channel depends on the rising time of the pulse
3	Instantaneous power of the transmitter varies	Instantaneous power of the transmitter varies	Instantaneous power of the transmitter remains constant
4	Noise interference is high	Noise interference is minimum	Noise interference is minimum
5	System is complex to implement	System is simple to implement	System is simple to implement
6	Similar to amplitude modulation	Similar to frequency modulation	Similar to phase modulation

Noise in PCM and DM systems

Signal to Quantization Noise ratio in PCM:

The signal to quantization noise ratio is given as:

$$\frac{S}{N_q} = \frac{\text{Normalized signal power}}{\text{Normalized noise power}}$$

$$= \frac{\text{Normalized signal power}}{\frac{\Delta^2}{12}}$$

The number of quantization value is equal to:

$$q=2^v$$

Putting this value in eq(6), we get:

$$\Delta = \frac{2X_{max}}{2^V}$$

Substitute this value in eq, we get

$$\frac{S}{N_q} = \frac{\text{Normalized signal power}}{\left[\frac{2X_{max}}{2^v}\right]^2 * \frac{1}{12}}$$

Let the normalized signal power is equal to P then the signal to quantization noise will be given by:

$$\boxed{\frac{S}{N_q} = \frac{3P * 2^{2v}}{X_{max}^2}}$$

Signal to Quantization Noise ratio in DM:

Delta modulation systems are subject to two types of quantization error: (1) slope –overload distortion, and (2) granular noise.
If we consider the maximum slope of the original input waveform x(t), it is clear that in order for the sequence of samples{u(nTs)} to increase as fast as the input sequence of samples {x(nTs)} in a region of maximum slope of x(t), we require that the condition in equation be satisfied.

$$\frac{\delta}{T_s} \geq \max\left|\frac{dx(t)}{dt}\right|$$

Otherwise, we find that the step size = 2δ is too small for the stair caseΔOtherwise, we find that the step size approximation u(t) to follow a steep segment of the input waveform x(t), with the result that u(t) falls behind x(t). This condition is called slope-overload, and the resulting quantization error is called slope-overload distortion (noise). Since the maximum slope of, increases and decreases inΔthe staircase approximation u(t) is fixed by the step size u(t) tend to occur along straight lines. For this reason, a delta modulator using a fixed step size is often referred ton as linear delta modulation (LDM). is too large relative to the localΔThe granular noise occurs when the step size slope characteristics of the input wave form x(t), thereby causing the staircase approximation u(t) to hunt around a relatively flat segment of the input waveform; The granular noise is analogous to quantization noise in a PCM system. The e choice of the optimum step size that minimizes the mean-square value of the quantizing error in a linear delta modulator will be the result of a compromise between slope overload distortion and granular noise.

Output SNR for Sinusoidal Modulation. Consider the sinusoidal signal, x(t) = A cos(2μfot)
The maximum slope of the signal x(t) is given by

$$\max\left|\frac{dx(t)}{dt}\right| = 2\pi\ f_0 A \qquad \text{----- (3.46)}$$

The use of Eq.5.81 constrains the choice of step size $\Delta = 2\delta$, so as to avoid slope-overload. In particular, it imposes the following condition on the value of δ:

$$\frac{\delta}{T_s} \geq \max\left|\frac{dx(t)}{dt}\right| = 2\pi\ f_0 A \qquad \text{----- (3. 47)}$$

Hence for no slope overload error the condition is given by equations 3.48 and 3.49.

$$A \leq \frac{\delta}{2\pi\ f_0 Ts} \qquad \text{------ (3.48)}$$

$$\delta \geq 2\pi\ f_0 A T_s \qquad \text{------ (3.49)}$$

Hence, the maximum permissible value of the output signal power equals

$$P_{\max} = \frac{A^2}{2} = \frac{\delta^2}{8\pi^2 f_0^2 T_s^2} \qquad \text{---- (3.50)}$$

When there is no slope-overload, the maximum quantization error $\pm\delta$. Assuming that the quantizing error is uniformly distributed (which is a reasonable approximation for small δ). Considering the probability density function of the quantization error

$$f_Q(q) = \frac{1}{2\delta}\ \ for\ -\delta \leq\ q \leq +\delta \qquad \text{----- (3.51)}$$
$$0\ \ otherwise$$

The variance of the quantization error is $\sigma^2{}_Q$.

$$\sigma_Q{}^2 = \frac{1}{2\delta}\int_{-\delta}^{+\delta} q^2\ \ dq = \frac{\delta^2}{3} \qquad \text{----- (3.52)}$$

The receiver contains (at its output end) a low-pass filter whose bandwidth is set equal to the message bandwidth (i.e., highest possible frequency component of the message signal), denoted as W such that $f0 \leq W$. Assuming that the average power of the quantization error is uniformly distributed over a frequency interval extending from -1/Ts to 1/Ts, we get the result:

Average output noise power $N_o = \left(\frac{f_c}{f_s}\right)\frac{\delta^2}{3} = WT_s\left(\frac{\delta^2}{3}\right)$ ----- (3.53)

Correspondingly, the maximum value of the output signal-to-noise ratio equals

$$(SNR)_O = \frac{P_{\max}}{N_o} = \frac{3}{8\pi^2 W f_0^2 T_s^3} \qquad \text{----- (3.54)}$$

Problems :

1.The bit stream *{bn}* = 0, 1, 1, 0, 0, 0, 1, 0 is to be sent through a channel (lowpass LTI system with large bandwidth). Assume that rectangular pulses of amplitude *A* are used and the bit rate is 1/*T* bps. In polar mapping, use the rule:

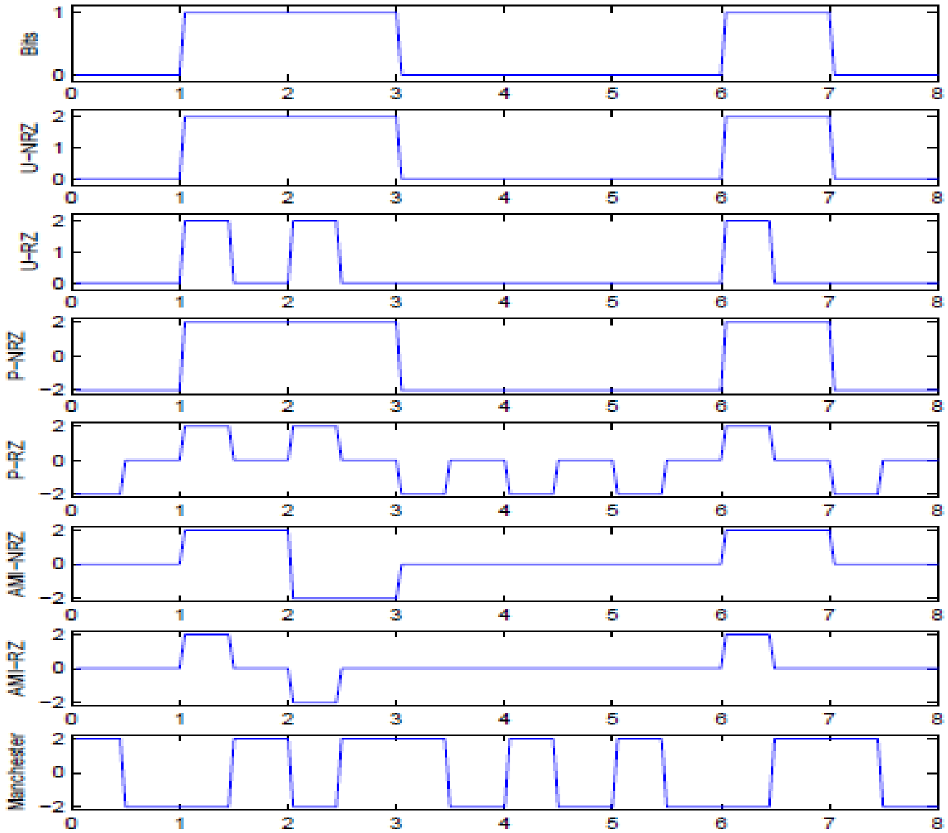
bn	***an***
0	−*A*
1	+*A*

Sketch the transmitted signal for each of the following line coding schemes:

(a) Unipolar NRZ
(b) Unipolar RZ
(c) Polar NRZ
(d) Polar RZ
(e) AMI-NRZ (Assume that −*A* is the inital state).
(f) AMI-RZ (Assume that −*A* is the inital state).
(g) Manchester

Sol.

(Line coding. Bit stream $\{b_n\}$ = 0, 1, 1, 0, 0, 0, 1, 0.) The transmitted signals are shown in the figure below (amplitude A = 2)



2. Compare the seven schemes (a) Unipolar NRZ
(b) Unipolar RZ
(c) Polar NRZ
(d) Polar RZ
(e) AMI-NRZ (Assume that $-A$ is the inital state).
(f) AMI-RZ (Assume that $-A$ is the inital state).
(g) Manchester
in terms of average (DC) power and average (DC) amplitude level.

Sol.

Technique	P_{ave}	$\bar{a_k}$
U-NRZ	$\frac{A^2T}{2}$	$\frac{A}{2}$
U-RZ	$\frac{A^2T}{4}$	$\frac{A}{4}$
P-NRZ	A^2T	0
P-RZ	$\frac{A^2T}{2}$	0
AMI-NRZ	$\frac{A^2T}{2}$	0
AMI-RZ	$\frac{A^2T}{4}$	0
Manhester	A^2T	0

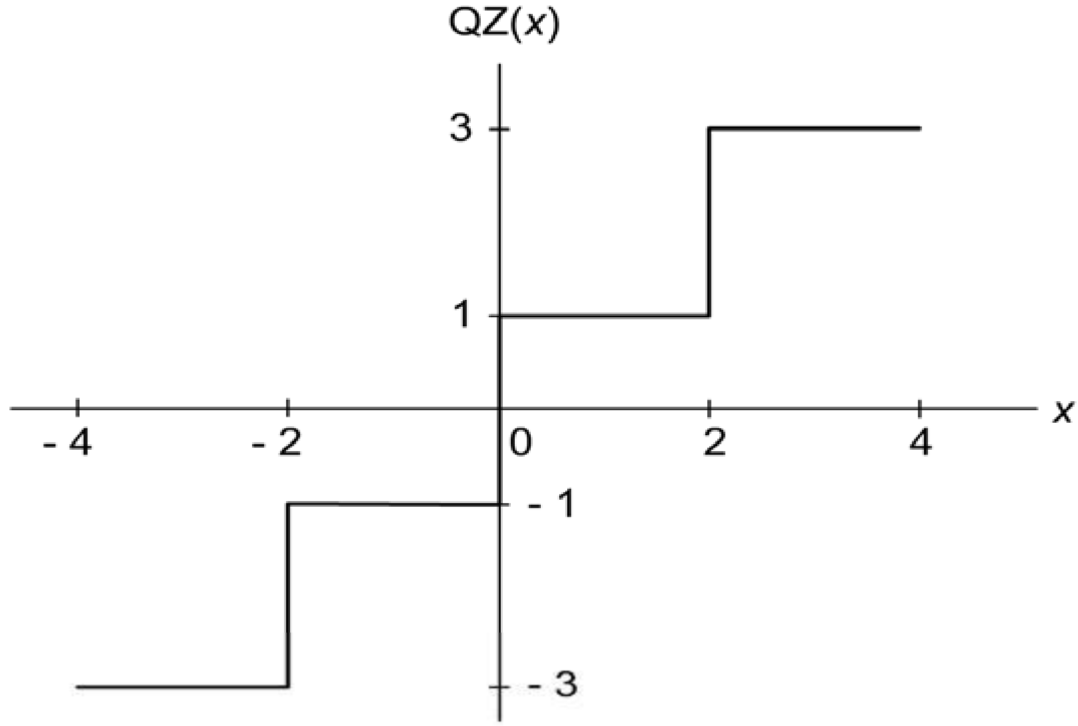
3. Consider the quantizer characteristic shown in Fig. 6.25(a). Let X be the input to the quantizer with $f_X(x)$ as shown at Fig. 6.25(b). Find

a) the value of A

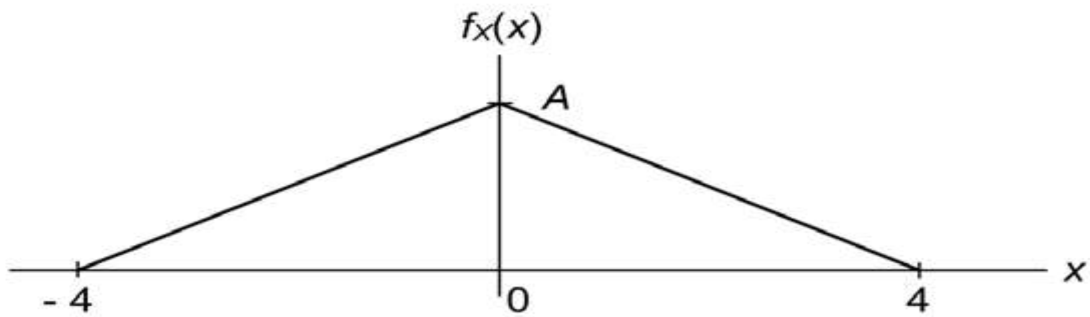
b) the total quantization noise variance,

σ^2_Q

c) Is it is the same as $\Delta^2/12$?



(a)



(b)

Sol.

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

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

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

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

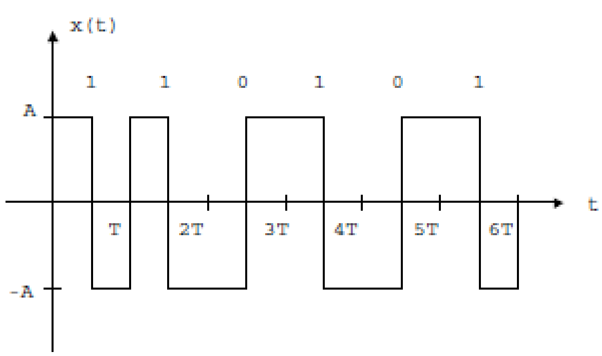
Carrying out the calculations, we have

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

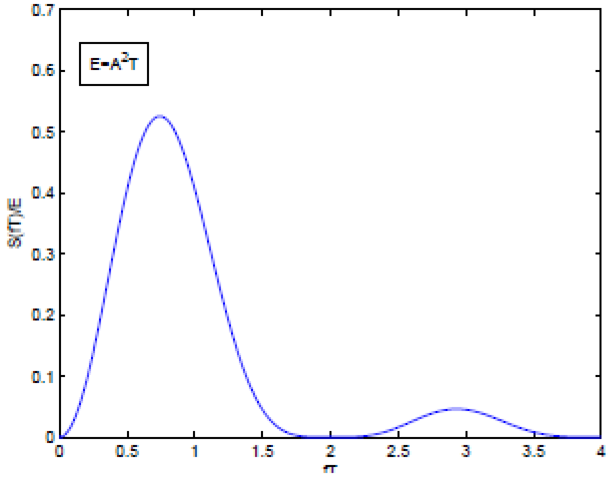
c) As $\frac{\Delta^2}{12} = \frac{4}{12} = \frac{1}{3}$, in this case we have σ_Q^2 the same as $\frac{\Delta^2}{12}$. ◆

4.A line coding scheme uses Manchester encoding with rectangular pulses.
(a) Sketch the signal corresponding to the bit sequence "110101"
(b) Sketch carefully, showing all relevant labels, the power spectral density of this scheme.
(c) The energy per bit is $Eb = 1\times 10^{-9}$ Joules and the bit rate is $Rb = 1\times 10^{6}$ bits/sec. What is the amplitude of the pulses?
(d) The noise at the receiver is AWGN with $\sigma^2 n = 1.25\times10^{-10}$ W/Hz. Determine the probability of a bit error.

Sol.
a)



b)



c)

The bit rate is $R_b = 1/T = 10^6$. The energy per bit is $E_b = A^2 T$ from which it follows that $A = \sqrt{E_b/T} = \sqrt{E_b R_b} = \sqrt{(10^{-9})(10^6)} = \sqrt{10^{-3}} = 31.62$ mV.

d.

$$P_e = Q\left(\sqrt{\frac{2E_b}{N_0}}\right) = Q\left(\sqrt{\frac{2 \cdot 10^{-9}}{2 \cdot 1.25 \times 10^{-10}}}\right) = Q\left(\sqrt{8}\right) = 2.34 \times 10^{-3}.$$

Problems

1.The bit stream *{bn}* = 0, 1, 1, 0, 0, 0, 1, 0 is to be sent through a channel (lowpass LTI system with large bandwidth). Assume that rectangular pulses of amplitude *A* are used and the bit rate is 1/*T* bps. In polar mapping, use the rule:

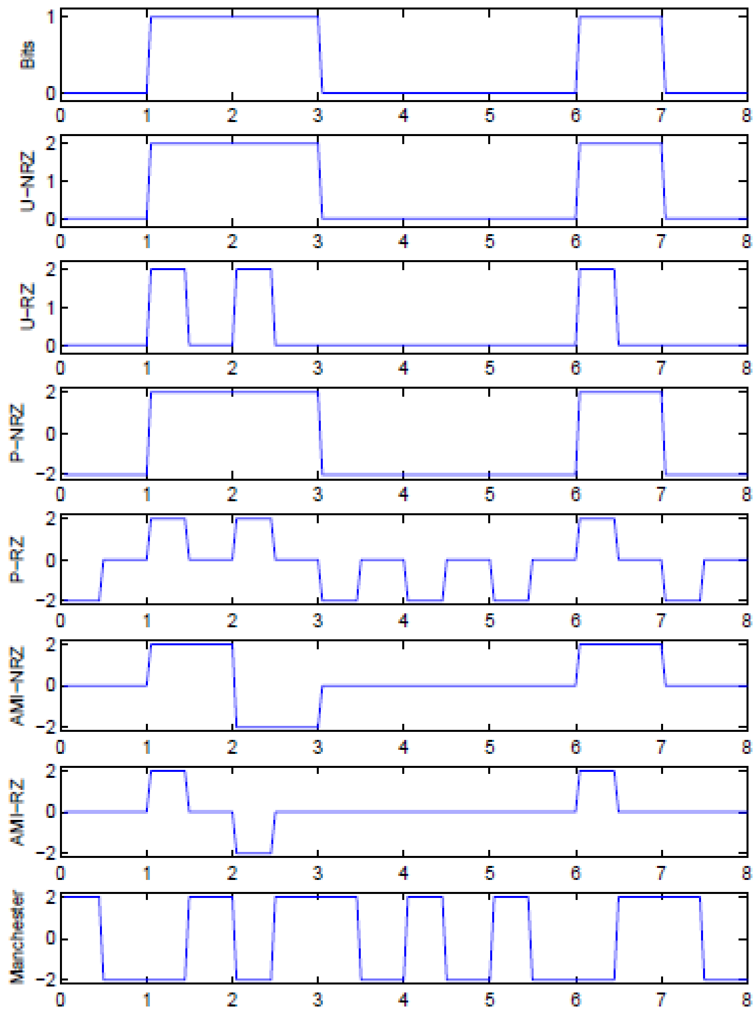
bn	*an*
0	−*A*
1	+*A*

Sketch the transmitted signal for each of the following line coding schemes:

(a) Unipolar NRZ
(b) Unipolar RZ
(c) Polar NRZ
(d) Polar RZ
(e) AMI-NRZ (Assume that −*A* is the inital state).
(f) AMI-RZ (Assume that −*A* is the inital state).
(g) Manchester

Sol.

(Line coding. Bit stream $\{b_n\}$ = 0, 1, 1, 0, 0, 0, 1, 0.) The transmitted signals are shown in the figure below (amplitude A = 2)



2.Compare the seven schemes (a) Unipolar NRZ
(b) Unipolar RZ
(c) Polar NRZ
(d) Polar RZ
(e) AMI-NRZ (Assume that $-A$ is the inital state).
(f) AMI-RZ (Assume that $-A$ is the inital state).
(g) Manchester
in terms of average (DC) power and average (DC) amplitude level.

Sol.

Technique	P_{ave}	$\bar{a_k}$
U-NRZ	$\frac{A^2T}{2}$	$\frac{A}{2}$
U-RZ	$\frac{A^2T}{4}$	$\frac{A}{4}$
P-NRZ	A^2T	0
P-RZ	$\frac{A^2T}{2}$	0
AMI-NRZ	$\frac{A^2T}{2}$	0
AMI-RZ	$\frac{A^2T}{4}$	0
Manhester	A^2T	0

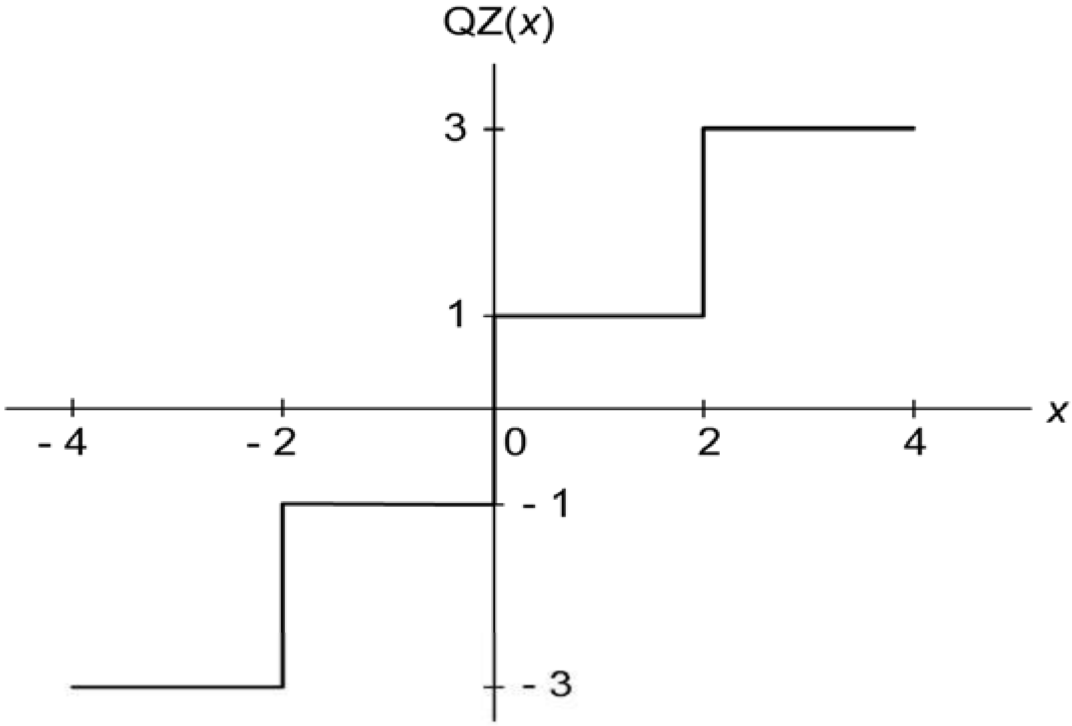
3. Consider the quantizer characteristic shown in Fig. 6.25(a). Let X be the input to the quantizer with $f_X(x)$ as shown at Fig. 6.25(b). Find

a) the value of A

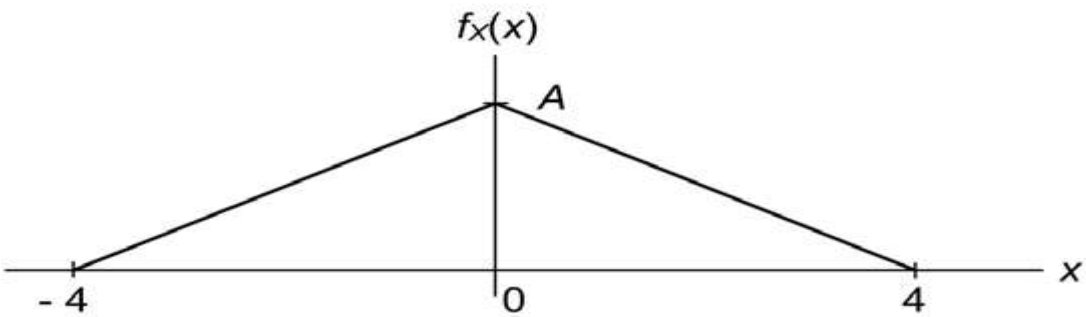
b) the total quantization noise variance,

σ^2_Q

c) Is it is the same as $\Delta^2/12$?



(a)



(b)

Sol.

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

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

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

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

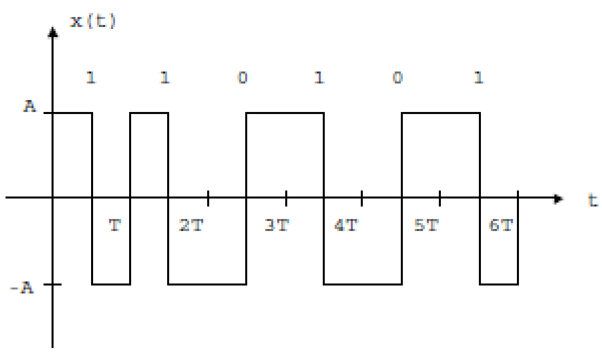
Carrying out the calculations, we have

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

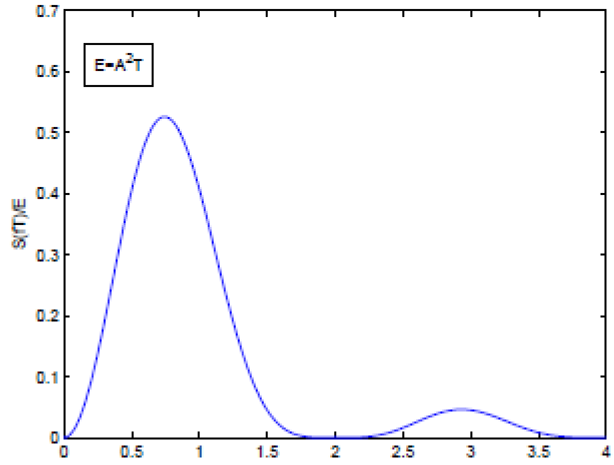
c) As $\frac{\Delta^2}{12} = \frac{4}{12} = \frac{1}{3}$, in this case we have σ_Q^2 the same as $\frac{\Delta^2}{12}$. ♦

4.A line coding scheme uses Manchester encoding with rectangular pulses.
(a) Sketch the signal corresponding to the bit sequence “110101”
(b) Sketch carefully, showing all relevant labels, the power spectral density of this scheme.
(c) The energy per bit is $Eb = 1\times 10^{-9}$ Joules and the bit rate is $Rb = 1\times 10^{6}$ bits/sec. What is the amplitude of the pulses?
(d) The noise at the receiver is AWGN with $\sigma^2 n = 1.25\times10^{-10}$ W/Hz. Determine the probability of a bit error.

Sol.
a)



b)



c)

The bit rate is $R_b = 1/T = 10^6$. The energy per bit is $E_b = A^2 T$ from which it follows that $A = \sqrt{E_b/T} = \sqrt{E_b R_b} = \sqrt{(10^{-9})(10^6)} = \sqrt{10^{-3}} = 31.62$ mV.

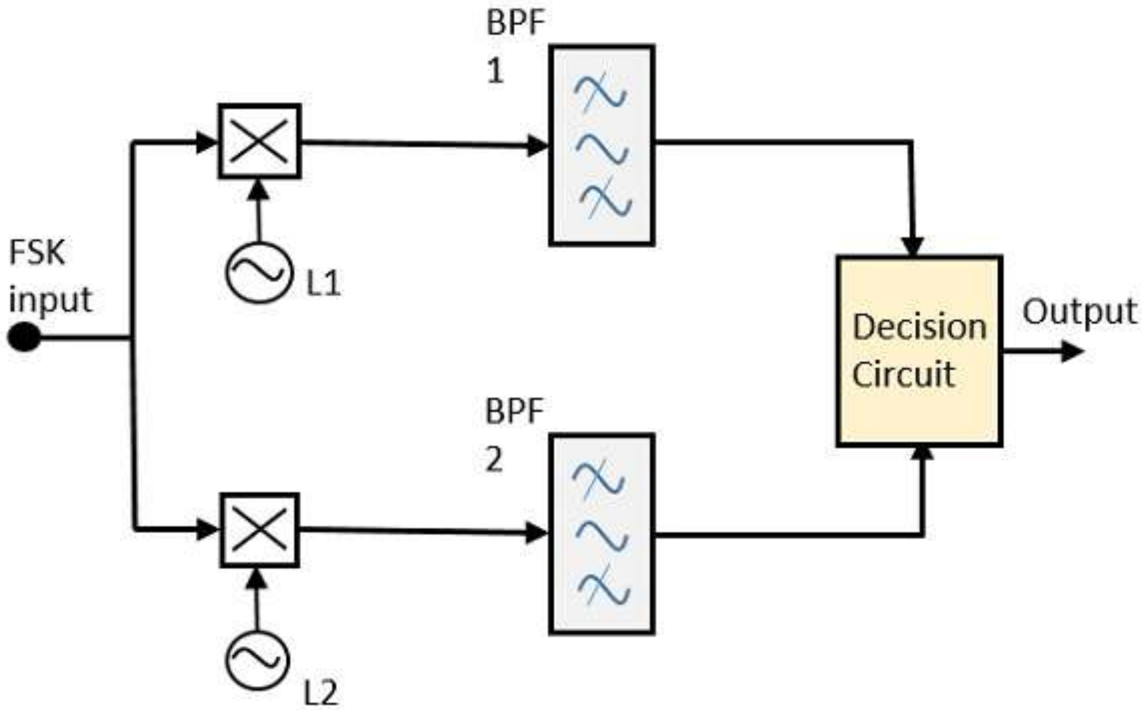
d.

$$P_e = Q\left(\sqrt{\frac{2E_b}{N_0}}\right) = Q\left(\sqrt{\frac{2 \cdot 10^{-9}}{2 \cdot 1.25 \times 10^{-10}}}\right) = Q\left(\sqrt{8}\right) = 2.34 \times 10^{-3}.$$

UNIT – II
DIGITAL MODULATION TECHNIQUES

Coherent FSK detector

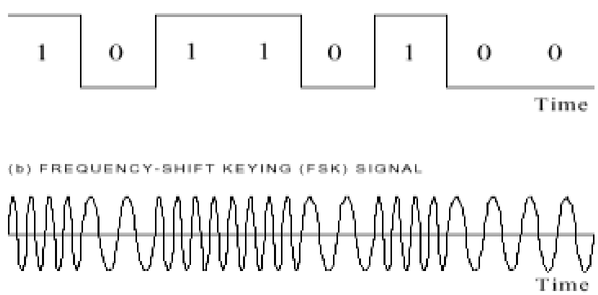
The block diagram of Synchronous(coherent) FSK detector consists of two mixers with local oscillator circuits, two band pass filters and a decision circuit. Following is the diagrammatic representation.



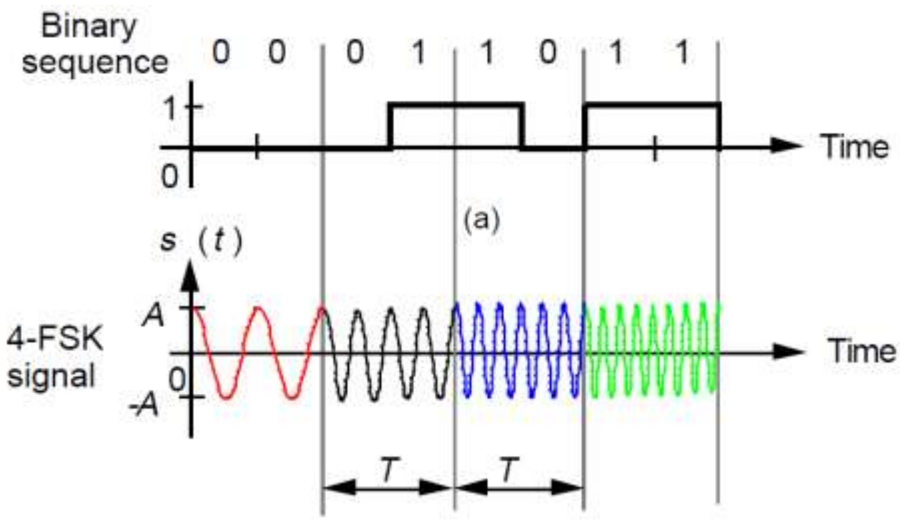
The FSK signal input is given to the two mixers with local oscillator circuits. These two are connected to two band pass filters. These combinations act as demodulators and the decision circuit chooses which output is more likely and selects it from any one of the detectors. The two signals have a minimum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

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

Frequency Shift Keying



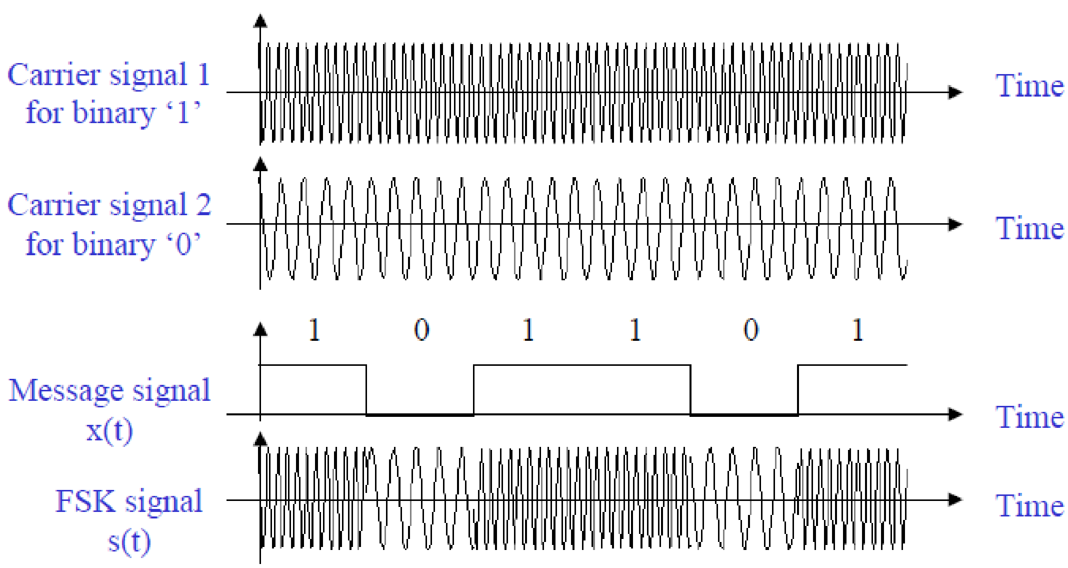
2-FSK



Typical Binary FSK Modulated Signal BFSK waveforms shown in the time domain

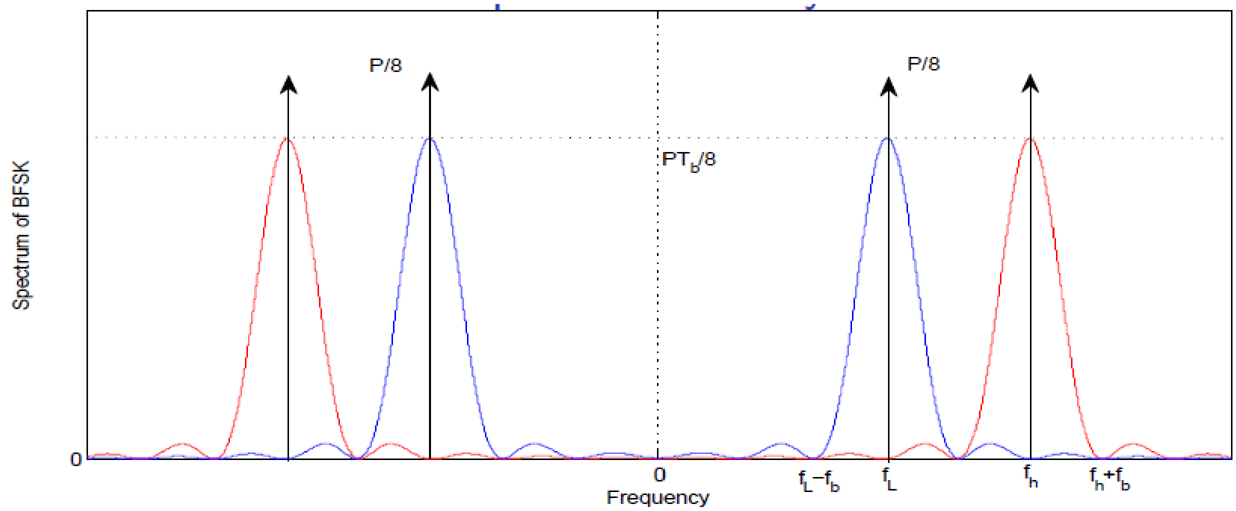
Binary bit ‘1’ and ‘0’ are represented by two different frequencies slightly offset from the carrier frequency.

- 1/0 represented by two different frequencies slightly offset from carrier frequency



where: *fs*= Low ‘space’ carrier frequency = $fo–\Delta f$, hertz .....for a Logic 0.
fm= High ‘mark’ carrier frequency = $fo+ \Delta f$, hertz .....for a Logic 1.

BandpassPower Spectral Density of Binary FSK



- Energy per bit: Eb= PTb, watt-second.
- The Null-to-Null RF transmission bandwidth for Binary FSK is:
- Bnull= (*fh*+ *fb*) –(*fL*–*fb*) = (*fh*–*fL*) + 2*fb*= 2Δf + 2*fb*= 2(Δf + Rb) = *Carson's Rule*.
- BFSK bandwidth with 90% of signal power: B90% = 1.23Rb.
- BFSK bandwidth with 99% of signal power: B99% = 2.12Rb.

Frequency Shift Keying Issues

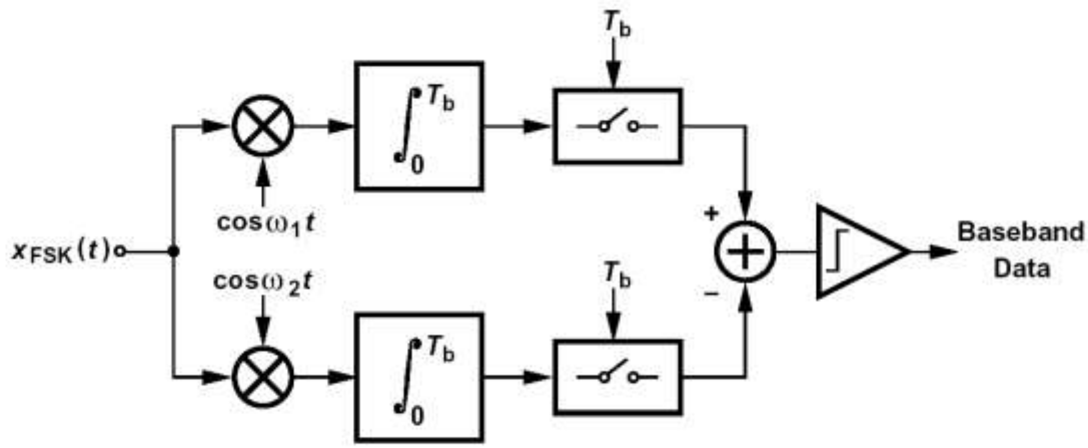
Advantages of FSK:

- FSK is ideally a constant envelope modulation; hence, more power-efficient class-C non-linear Power Amplifiers can be used in the transmitter.
- FSK is more bandwidth efficient than ASK.
- Reasonably simple modulation and demodulation schemes.

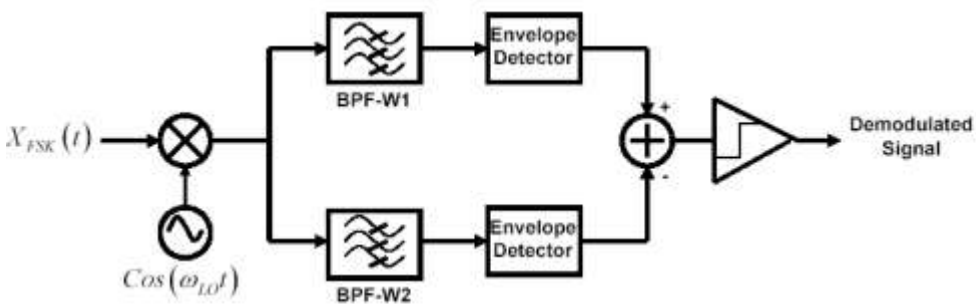
Disadvantages of FSK:

- The difference between coherent FSK detection and non-coherent FSK detection is not significant for higher FSK levels.
- The extra hardware required for coherent FSK detection is hence hard to justify.
- Coherent FSK is not often used in practice due to the difficulty (and cost) in generating two reference frequencies close together at the receiver.

FSK Demodulation (Coherent)



FSK Demodulation (Non Coherent)



Non coherent FSK

Overview

1.The noncoherent FSK requires, at most, only 1 dB more Eb/N0than that for coherent FSK for probability of bit error: Pb< 10^{-4}(See next slide for BER).

2.The noncoherent FSK demodulator is considerably easier to build since coherent reference signals need not be generated.

3.Coherent FSK signals can be no coherent demodulated to avoid the carrier recovery.

4. Non coherently generated FSK can only be non coherently demodulated.

5.Both cases are referred to ‘non coherent FSK’.

6.In both cases, the demodulation problem becomes a problem of detecting signals with unknown phases.

7.For coherent FSK signals to be orthogonal, the FSK frequencies must be integer multiple of 1/(2Ts) and their separation must be a multiple of 1/(2Ts).The null-to-null bandwidth for coherent FSK is: Bnull= (M + 3)Rs/2, Hertz.

8.For noncoherentFSK signals to be orthogonal, the FSK frequencies must be integer multiple of 1/(2Ts) and their separation must be a multiple of 1/Ts. The null-to-null bandwidth for noncoherentFSK is: Bnull= (M + 1) Rs, Hertz.

9.Thus, for the same symbol rate: Rs, more system bandwidth is required for noncoherently detected FSK then for coherently detected FSK.

BPSK, coherent BPSK detection

Binary Phase Shift Keying (BPSK):

The simplest form of PSK is binary phase-shift keying (BPSK), where N = 1 and M = 2. Therefore, with BPSK, two phases (21 = 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°. 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 is a simple but significant carrier modulation scheme. The two time-limited energy signals s1(t) and s2(t) are defined based on a single basis function φ1(t) as:

$$s_1(t) = \sqrt{\frac{2E_b}{T_b}}.\cos 2\pi f_c t \quad \text{and} \quad s_2(t) = \sqrt{\frac{2E_b}{T_b}}.\cos[2\pi f_c t + \pi] = -\sqrt{\frac{2E_b}{T_b}}.\cos 2\pi f_c t$$

The basis function, evidently, is,

$$\varphi_1(t) = \sqrt{\frac{2}{\mathrm{T_b}}}.\cos 2\pi f_c t \quad ; \quad 0 \le \mathrm{t} < \mathrm{T_b}.$$

So, BPSK may be described as a one-dimensional digital carrier modulation scheme. Note that the general form of the basis function is,

$$\varphi_1(t) = \sqrt{\frac{2}{T_b}}.\cos(2\pi f_c t + \phi),$$

where 'Φ' indicates an arbitrary but fixed initial phase offset. For convenience, let us set Φ= 0.
As we know, for narrowband transmission, $f_c >> 1/T_b$. That is, there will be multiple cycles of the carrier sinusoid within one bit duration (Tb). For convenience in description, let us set, $f_c = n \times 1/T_b$ (though this is not a condition to be satisfied theoretically).
Now, we see,

$$s_1(t) = \sqrt{E_b}.\varphi_1(t) \quad \text{and} \quad s_2(t) = -\sqrt{E_b}.\varphi_1(t),$$

The two associated scalars are:

$$s_{11}(t) = \int_0^{T_b} s_1(t).\varphi_1(t)dt = +\sqrt{E_b} \quad \text{and} \quad s_{21} = \int_0^{T_b} s_2(t).\varphi_2(t)dt = -\sqrt{E_b}$$

Fig. 5.24.1 (a) presents a sketch of the basis function φ1(t) and Fig. 1 (b) shows the BPSK modulated waveform for a binary sequence. Note the abrupt phase transitions in the modulated waveform when there is change in the modulating sequence. On every occasion the phase has changed by 180^0. Also note that, in the diagram, we have chosen to set

$$\sqrt{\frac{2E_b}{T_b}} = 1, \text{ i.e. } \frac{E_b}{T_b} = \frac{1}{2} = 0.5,$$

which is the power associated with an unmodulated carrier sinusoid of unit peak amplitude.

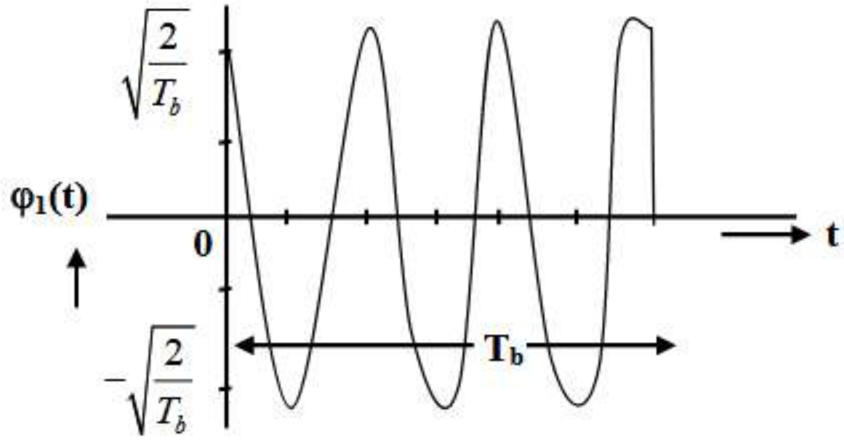


Fig. 1: (a) Sketch of the basis function φ1(t) for BPSK modulation

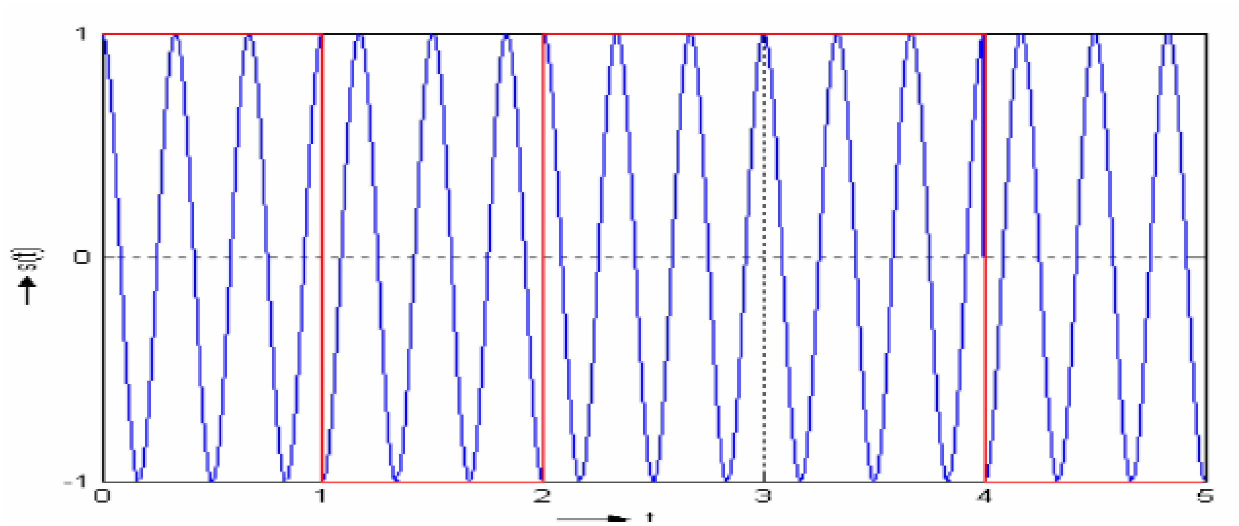


Fig. 1 (b)BPSK modulated waveform for the binary sequence 10110. Note that the amplitude has been normalized to ±1, as is a common practice.

Fig.1: (c)shows the signal constellation for binary PSK modulation. The two pints are equidistant from the origin, signifying that the two signals carry same energy.

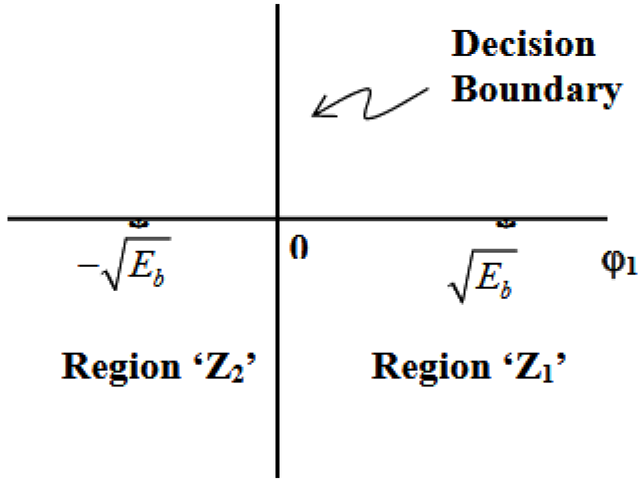


Fig. 1: (c)Signal constellation for binary PSK modulation. The diagram also shows the optimum decision boundary followed by a correlation receiver

Fig. 2shows a simple scheme for generating BPSK modulated signal
without pulse shaping. A commonly available balanced modulator (such as IC 1496) may be used as the product modulator to actually generate the
modulated signal. The basis function φ1(t), shown as the second input to th
e product modulator, can be generated by an oscillator. Note that the oscillator may work independent of the data clock in general.

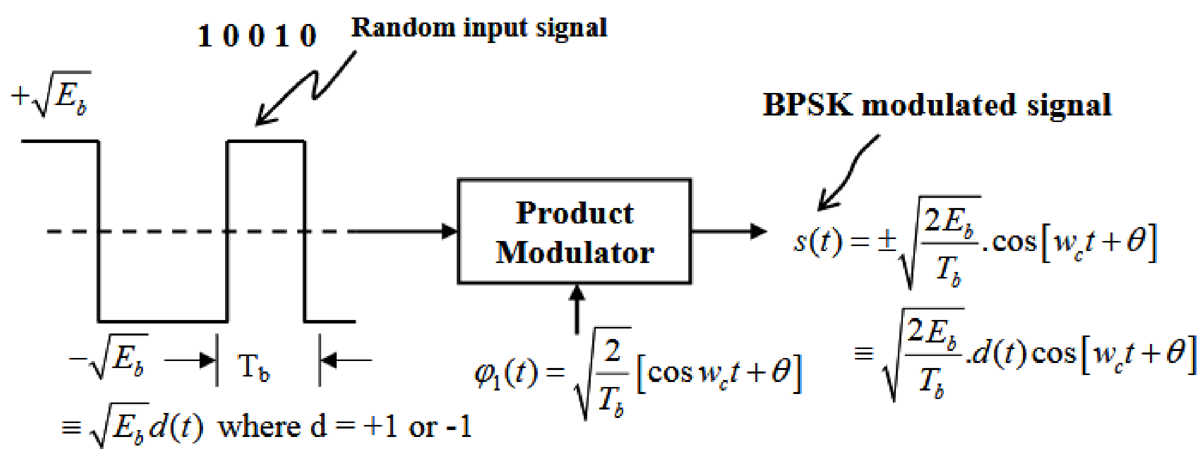


Fig. 2 A simple scheme for generating BPSK modulated signal. No pulse-shaping filter has been used.

coherent BPSK detection

Fig. 3presents a scheme for coherent demodulation of BPSK modulated signal following the concept of optimum correlation receiver. The input signal r(t) to the demodulator is assumed to be centered at an intermediate frequency (IF). This real narrowband signal consists of the desired modulated signal s(t) and narrowband Gaussian noise w(t). As is obvious, the correlation detector consists of the product modulator, shown as an encircled multiplier, and the integrator. The vector receiver is a simple binary decision device,

such as a comparator. For simplicity, we assumed that the basis function phase reference is perfectly known at the demodulator and hence the φ1(t),
shown as an input to the product demodulator, is phase-synchronized to that of the modulator. Now it is straightforward to note that the signal at (A) in
Fig. 3 is

$$r_A(t) = [s(t)+w(t)].\sqrt{\frac{2}{T_b}}.\cos(w_c t+\theta)$$

The signal at (B) is:

$$r_1 = \sqrt{\frac{2}{T_b}}\int_0^{T_b}\left[d(t).\sqrt{\frac{2E_b}{T_b}}.\cos(w_c t+\theta)+w(t)\right]\cos(w_c t+\theta)dt$$

$$= \sqrt{E_b}.d(t)+\sqrt{\frac{2}{T_b}}\int_0^{T_b}w(t).\cos(w_c t+\theta)dt$$

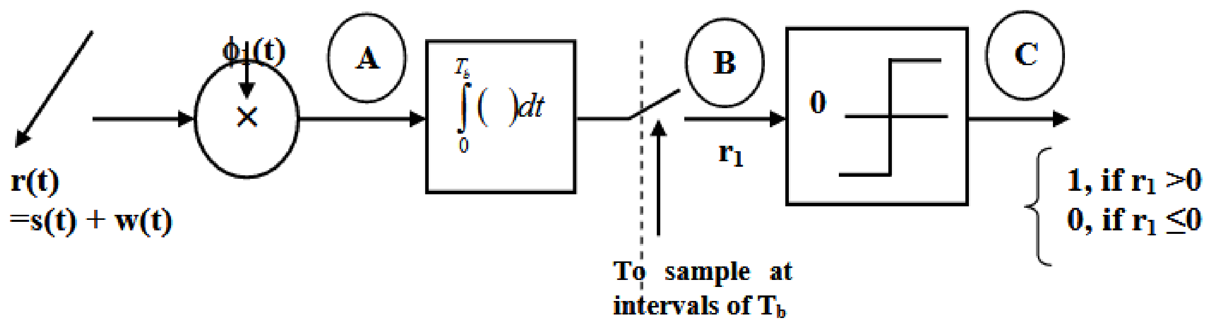


Fig. 3 A scheme for coherent demodulation of BPSK modulated signal following the concept of optimum correlation receiver

Note that the first term in the above expression is the desired term while the
second term represents the effect of additive noise. We have discussed about similar noise component earlier in Module #4 and we know that this term is a Gaussian distributed random variable with zero mean. Its variance is proportional to the noise power spectral density. It should be easy to follow that, if d(t) = +1 and the second term in Eq. 5(i.e. the noise sample voltage) is not less than -1.0, the threshold detector will properly decide the received signal as a logic '1'. Similarly, if d(t) = -1 and the noise sample voltage is not greater than +1.0, the comparator will properly decide the received signal as a logic '0'. These observations are based on 'positive binary logic'.
Power Spectrum for BPSK Modulated Signal
Continuing with our simplifying assumption of zero initial phase of the carrier and with no pulse shaping filtering, we can express a BPSK modulated signal as:

$$s(t)=\sqrt{\frac{E_b.2}{T_b}}.d(t)\cos w_c t\text{, where}\quad d(t)=\pm 1$$

The baseband equivalent of s(t) is,

$$\tilde{u}(t) = u_I(t) = \sqrt{\frac{2E_b}{T_b}}.d(t) = \pm g(t),$$

$$\text{where } g(t) = \sqrt{\frac{2E_b}{T_b}} \text{ and } u_Q(t) = 0.$$

Now, uI(t) is a random sequence of $+\sqrt{\frac{2E_b}{T_b}}$ and $-\sqrt{\frac{2E_b}{T_b}}$ which are equi-probable. So, the power spectrum of the base band signal is:

$$\rightarrow U_B(f) = \frac{2E_b.\sin^2(\pi T_b f)}{(\pi T_b f)^2} \qquad = 2.E_b.\sin c^2(T_b f)$$

Now, the power spectrum S(f) of the modulated signal can beexpressed in terms of UB(f) as:

$$S(f) = \frac{1}{4}\left[U_B(f - f_c) + U_B(f + f_c)\right]$$

Fig.4shows the normalized base band power spectrum of BPSK modulated signal. The spectrum remains the same for arbitrary non-zero initial phase of
carrier oscillator.'

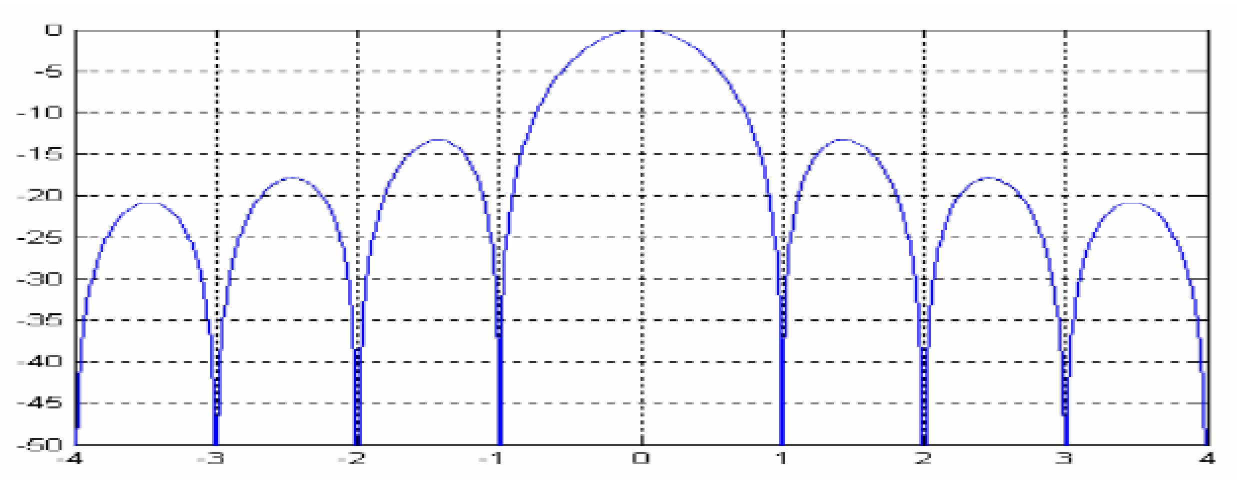


Fig.4:Normalized base band power spectrum of BPSK modulated signal

QPSK; DPSK

Quaternary Phase Shift Keying (QPSK)

This modulation scheme is very important for developing concepts of two-dimensional I-Q modulations as well as for its practical relevance. In a sense, QPSK is an expanded version from binary PSK where in a symbol consists of two bits and two orthonormal basis functions are used. A group of two bits is often called a 'dibit'. So, four dibits are possible. Each symbol carries same energy.
Let, E: Energy per Symbol and T: Symbol Duration = 2. Tb, where Tb: duration of 1 bit. Then, a general expression for QPSK modulated signal, without any pulse shaping, is:

$$s_i(t) = \sqrt{\frac{2E}{T}} \cos\left[2\pi f_c t + (2i-1).\frac{\pi}{4}\right] + 0 \,; \quad 0 \le \mathrm{t} \le \mathrm{T};\ \mathrm{i} = 1,2,3,4$$

where, $f_c = n.\frac{1}{T} = n.\frac{1}{2T_b}$ is the carrier (IF) frequency.

On simple trigonometric expansion, the modulated signal si (t) can also be expressed as:

$$s_i(t) = \sqrt{\frac{2E}{T}}.\cos\left[(2i-1)\frac{\pi}{4}\right].\cos 2\pi f_c t - \sqrt{\frac{2E}{T}}.\sin\left[(2i-1)\frac{\pi}{4}\right].\sin 2\pi f_c t \,;\ 0 \le \mathrm{t} \le \mathrm{T}$$

The two basis functions are:

$$\varphi_1(t) = \sqrt{\frac{2}{T}}.\cos 2\pi f_c t \ ;\ 0 \le \mathrm{t} \le \mathrm{T} \quad \text{and} \quad \varphi_2(t) = \sqrt{\frac{2}{T}}.\sin 2\pi f_c t \ ;\ 0 \le \mathrm{t} \le \mathrm{T}$$

The four signal points, expressed as vectors, are:

$$\overline{s_i} = \left\{\sqrt{E}\cos\left[(2i-1)\pi/4\right] - \sqrt{E}\sin\left[(2i-1)\pi/4\right]\right\} = \begin{bmatrix} s_{i1} \\ s_{i2} \end{bmatrix} \quad ; \quad \mathrm{i} = 1,2,3,4$$

Fig.5.1 shows the signal constellation for QPSK modulation. Note that all the four points are equidistant from the origin and hence lying on a circle. In this plain version of QPSK, a symbol transition can occur only after at least T = 2Tb sec. That is, the symbol rate Rs = 0.5Rb. This is an important observation because one can guess that for a given binary data rate, the transmission bandwidth for QPSK is half of that needed by BPSK modulation scheme. We discuss about it more later.

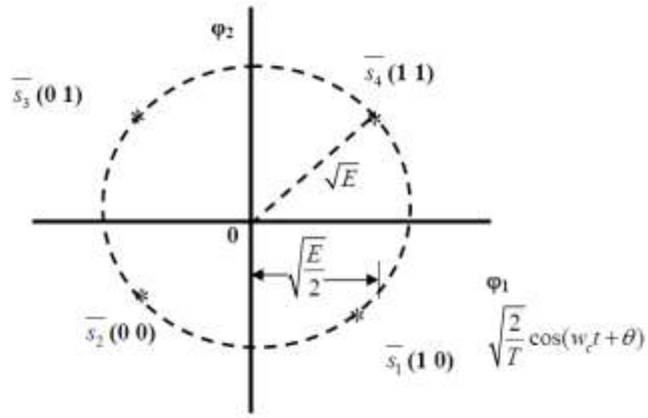


Fig.1 Signal constellation for QPSK. Note that in the above diagram θ has been considered to be zero. Any fixed non-zero initial phase of the basis functions is permissible in general.

Now, let us consider a random binary data sequence: 10111011000110… Let us designate the bits as 'odd' (bo) and 'even' (be) so that one modulation symbol consists of one odd bit and the adjacent even bit. The above sequence can be split into an odd bit sequence (1111001…) and an even bit sequence (0101010…). In practice, it can be achieved by a 1-to-2 DEMUX. Now, the modulating symbol sequence can be constructed by taking one bit each from the odd and even sequences at a time as {(10), (11), (10), (11), (00), (01), (10), …}. We started with the odd sequence. Now we can recognize the binary bit stream as a sequence of signal points which are to be transmitted: {1*s*, 4*s*, 1*s*, 4*s*, 2*s*, 3*s*, 1*s*, …}.

With reference to **Fig.5.25.1**, let us note that when the modulating symbol changes from 1*s* to 4*s*, it ultimately causes a phase shift of $\pi^c/2$ in the pass band modulated signal [from $-\pi^c/4$ to $+\pi^c/4$ in the diagram]. However, when the modulating symbol changes from 4*s* to 2*s*, it causes a phase shift of in the pass band modulated signal [from $+\pi^c/4$ π to $+5\pi^c/4$in the diagram]. So, a phase change of or *c*02*c*π or occurs in the modulated signal every 2Tb sec. It is interesting to note that as no pulse shaping has been used, the phase changes occur almost instantaneously. Sharp phase transitions give rise to significant side lobes in the spectrum of the modulated signal.

Table.1 summarizes the features of QPSK signal constellation.

Input	Dibit		Phase of QPSK	Coordinates of signal points		
	(b_0)	(b_e)		s_{i1}	s_{i2}	i
$\overline{s_1}$	1	0	$\pi/4$	$+\sqrt{E/2}$	$-\sqrt{E/2}$	1
$\overline{s_2}$	0	0	$3\pi/4$	$-\sqrt{E/2}$	$-\sqrt{E/2}$	2
$\overline{s_3}$	0	1	$5\pi/4$	$-\sqrt{E/2}$	$+\sqrt{E/2}$	3
$\overline{s_4}$	1	1	$7\pi/4$	$+\sqrt{E/2}$	$+\sqrt{E/2}$	4

Table .1 *Feature summary of QPSK signal constellation*

Fig.2 shows the QPSK modulated waveform for a data sequence 101110110001. For better illustration, only three carrier cycles have been shown per symbol duration.

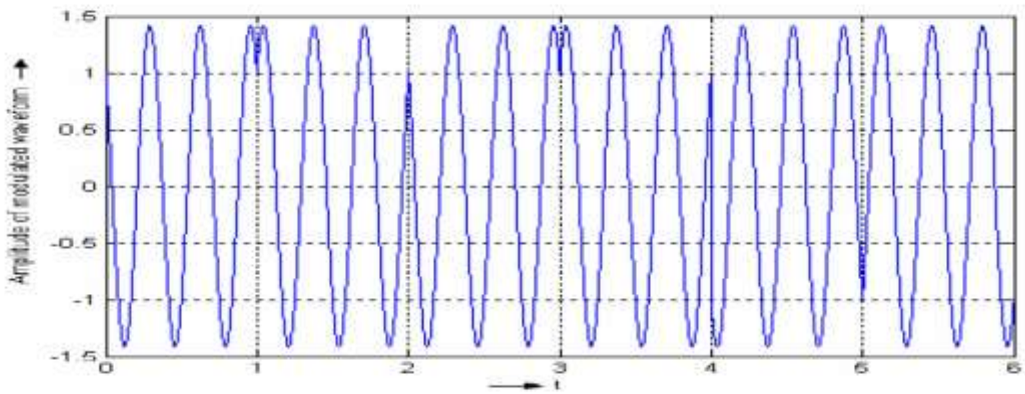


Fig.2 *QPSK modulated waveform*

Generation of QPSK modulated signal

Let us recall that the time-limited energy signals for QPSK modulation can be expressed as,

$$s_i(t) = \sqrt{\frac{2E}{T}}.\cos\left[(2i-1)\pi/4\right].\cos w_c t - \sqrt{\frac{2E}{T}}.\sin\left[(2i-1)\pi/4\right].\sin w_c t$$

$$= \sqrt{E}.\cos\left[(2i-1)\pi/4\right]\sqrt{\frac{2}{T}}.\cos w_c t - \sqrt{E}\sin\left[(2i-1)\pi/4\right].\sqrt{\frac{2}{T}}\sin w_c t$$

$$= s_{i1}\varphi_1(t) + s_{i2}\varphi_2(t) \qquad \text{i = 1,2,3.4}$$

The QPSK modulated wave can be expressed in several ways such as:

$$s(t) = \sqrt{E}.d_{odd}(t).\sqrt{\frac{2}{T}}\cos w_c t + \sqrt{E}.d_{even}(t).\sqrt{\frac{2}{T}}\sin w_c t$$

$$= \sqrt{\frac{2E}{T}}.d_{odd}(t)\cos w_c t + \sqrt{\frac{2E}{T}}.d_{even}(t)\sin w_c t$$

$$= \left\{d_{odd}(t).\sqrt{\frac{2E}{T}}\right\}\cos w_c t + \left\{d_{even}(t).\sqrt{\frac{2E}{T}}\right\}\sin w_c t$$

For narrowband transmission, we can further express s(t) as:

$$s(t) \equiv u_I(t).\cos w_c t - u_Q(t).\sin w_c t$$

where $\tilde{u}(t) = u_I(t) + ju_Q(t)$ is the complex low-pass equivalent representation of s(t).

One can readily observe that, for rectangular bipolar representation of information bits and without any further pulse shaping,

$$u_I(t) = \sqrt{\frac{2E}{T}}.d_{odd}(t) \text{ and } u_Q(t) = \sqrt{\frac{2E}{T}}.d_{even}(t)$$

Note that while expressing Eq. 6, we have absorbed the '-' sign, associated with the quadrature carrier 'sin ωct' in $d_{even}(t)$. We have also assumed that $d_{odd}(t) = +1.0$ for 'bo' ≡1 while $d_{even}(t) = -1.0$ when be ≡ 1. This is not a major issue in concept building as its equivalent effect can be implemented by inverting the quadrature carrier.

Fig. 3(a) shows a schematic diagram of a QPSK modulator following **Eq. 6**. Note that the first block, accepting the binary sequence, does the job of generation of odd and even sequences as well as the job of scaling (representing) each bit appropriately so that its outputs are si1 and si2 (**Eq.5**). **Fig. .3(b)** is largely similar to **Fig.3(a)** but is better suited for simple implementation. Close observation will reveal that both the schemes are equivalent while the second scheme allows adjustment of power of the modulated signal by adjusting the carrier amplitudes. Incidentally, both the in-phase carrier and the quadrature phase carriers are obtained from a single continuous-wave oscillator in practice.

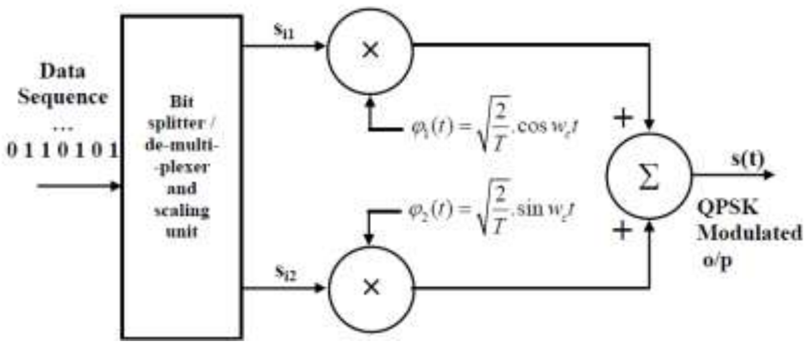


Fig.3 (a) Block schematic diagram of a QPSK modulator

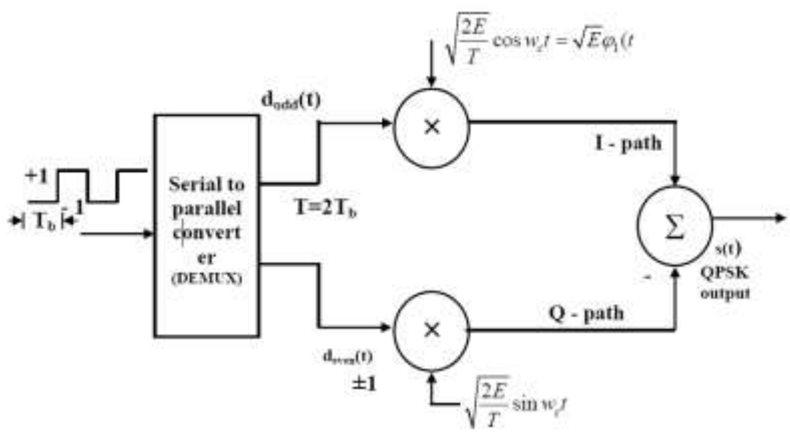


Fig.3 (b) *Another schematic diagram of a QPSK modulator, equivalent to* ***Fig. 3(a)*** *but more suitable in practice*

The QPSK modulators shown in **Fig.3** follow a popular and general structure known as I/Q (In-phase / Quadrature-phase) structure. One may recognize that the output of the multiplier in the I-path is similar to a BPSK modulated signal where the modulating sequence has been derived from the odd sequence. Similarly, the output of the multiplier in the Q-path is a BPSK modulated signal where the modulating sequence is derived from the even sequence and the carrier is a sine wave. If the even and odd bits are independent of each other while occurring randomly at the input to the modulator, the QPSK modulated signal can indeed be viewed as consisting of two independent BPSK modulated signals with orthogonal carriers.

The structure of a QPSK demodulator, following the concept of correlation receiver, is shown in **Fig. 4**. The received signal r(t) is an IF band pass signal, consisting of a desired modulated signal s(t) and in-band thermal noise. One can identify the I- and Q- path correlators, followed by two sampling units. The sampling units work in tandem and sample the outputs of respective integrator output every T = 2Tb second, where 'Tb' is the duration of an information bit in second. From our understanding of correlation receiver, we know that the sampler outputs, i.e. r1 and r2 are independent random variables with Gaussian probability distribution. Their variance is same and decided by the noise variance while their means are $\pm 2E$, following our style of representation. Note that the polarity of the sampler output indicates best estimate of the corresponding information bit. This task is accomplished by the vector receiver, which consists of two identical binary comparators as indicated in **Fig.4.** The output of the comparators are interpreted and multiplexed to generate the demodulated information sequence (in the figure).

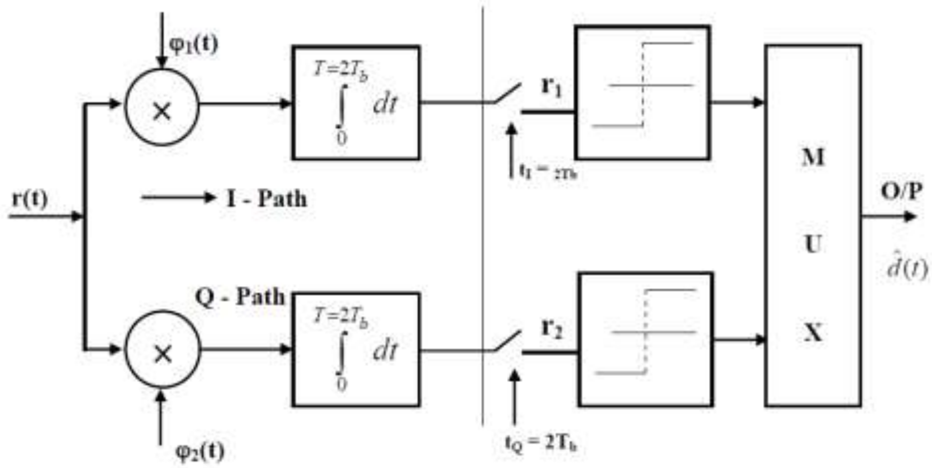


Fig. 5.25.4 *Correlation receiver structure of QPSK demodulator*

We had several ideal assumptions in the above descriptions such as a) ideal regeneration of carrier phase and frequency at the receiver, b) complete knowledge of symbol transition instants, to which the sampling clock should be synchronized, c) linear modulation channel between the modulator output and our demodulator input and so forth. These issues must be addressed satisfactorily while designing an efficient QPSK modem.

Spectrum of QPSK modulated signal

To determine the spectrum of QPSK modulated signal, we follow an approach similar to the one we followed for BPSK modulation in the previous lesson. We assume a long sequence of random independent bits as our information sequence. Without Nyquist filtering, the shaping function in this case can be written as:

$$g(t)=\sqrt{\frac{E}{T}}\;; \qquad 0 \le t \le T = 2\,T_b$$

After some straight forward manipulation, the single-sided spectrum of the equivalent complex baseband signal can be expressed as:

$$U_B(f) = 2E.\sin c^2(Tf)$$

Here 'E' is the energy per symbol and 'T' is the symbol duration. The above expression can also be put in terms of the corresponding parameters associated with one information bit:

$$U_B(f) = 4.E_b.\sin c^2(2T_b f)$$

Fig. 5 shows a sketch of single-sided baseband spectrum of QPSK modulated signal vs. the normalized frequency (fTb). Note that the main lobe has a null at fTb = 0.5.f.T = 0.5 because no Nyquist pulse shaping was adopted. The width of the main lobe is half of that necessary for BPSK modulation. So, for a given data rate, QPSK is more bandwidth efficient. Further, the peak of the first sidelobe is not negligibly small compared to the main lobe peak. The side lobe peak is about 12 dB below the main lobe peak. The peaks of the subsequent lobes monotonically decrease. So, theoretically the spectrum stretches towards infinity. As discussed in Module #4, the spectrum is restricted in a practical system by resorting to pulse shaping. The single-sided equivalent Nyquist bandwidth for QPSK = (1/2) symbol rate (Hz) = *T*21 (Hz) = *bT*41(Hz). So, the normalized single-sided equivalent Nyquist bandwidth = ¼ = 0.25. The Nyquist transmission bandwidth of the real pass band modulated signal s(t) = 2 x single-sided Nyquist bandwidth =1/ 2*Tb*(Hz) = 1/*T* (Hz) ≡ The symbol rate.

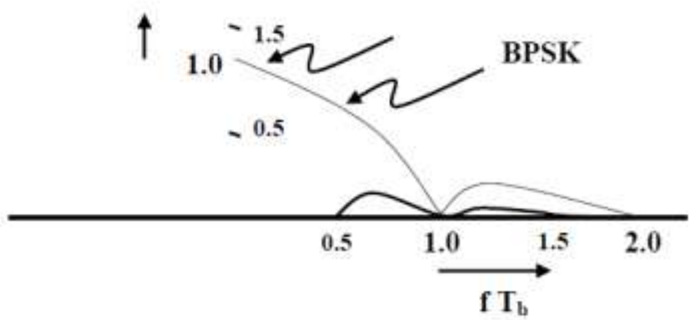


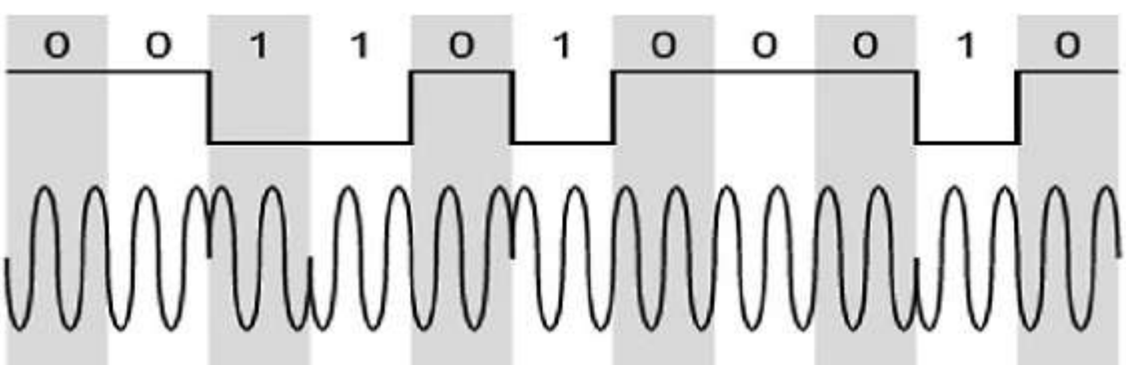
Fig.5 *Normalized base band bandwidth of QPSK and BPSK modulated signals*

The actual transmission bandwidth that is necessary = Nyquist transmission bandwidth) x (1 + α) Hz = (1 + α). *T*1 Hz = (1 + α).Rs Hz, where 'Rs' is the symbol rate in symbols/sec.

DPSK

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

The following figure represents the model waveform of DPSK.

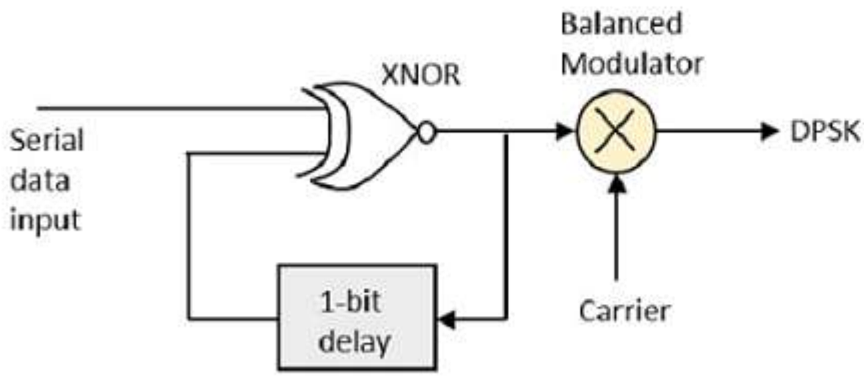


It is seen from the above figure that, if the data bit is Low i.e., 0, then the phase of the signal is not reversed, but continued as it was. If the data is a High i.e., 1, then the phase of the signal is reversed, as with NRZI, invert on 1 (a form of differential encoding).

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

DPSK Modulator

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

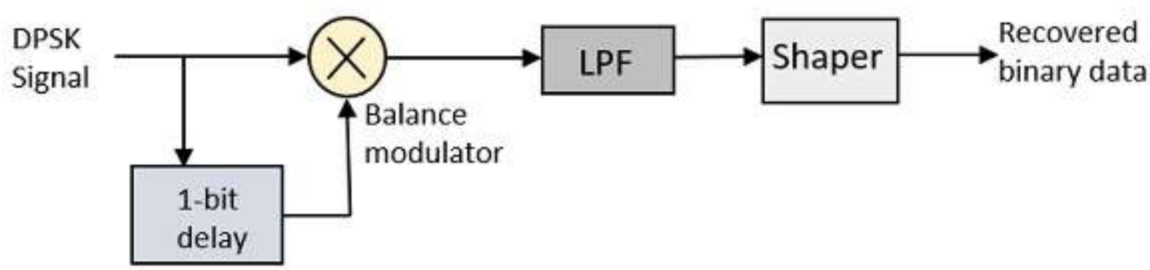


DPSK Modulator

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

DPSK Demodulator

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



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

Differential Encoding of BPSK Modulation (DEBPSK)

Let us assume that for an ordinary BPSK modulation scheme, the carrier phase is 0c when the message bit, 'mk' is logic '1' and it is πc if the message bit mk' is logic '0'.
When we apply differential encoding, the encoded binary '1' will be transmitted by adding 0c to the current phase of the carrier and an encoded binary '0' will be transmitted by adding πc to the current phase of the carrier. Thus, relation of the current message bit to the absolute phase of the received signal is modified by differential encoding. The current carrier phase is dependent on the previous phase and the current message bit. For BPSK modulation format, the differential encoder generates an encoded binary logic sequence {dk} such that, dk = 1 if dk-1 and mk are similar and dk = 0 if dk-1 and mk are not similar.
For completeness, let us assume that the first encoded bit, say, d0 is '1' while the index 'k' takes values 1, 2, ….**Fig. 1(a)** shows a block schematic diagram for differential encoding and BPSK modulation. For clarity, we will refer the modulated signal as 'Differentially Encoded and BPSK modulated (DEBPSK)' signal. The level shifter converts the logic symbols to binary antipodal signals of ±1. Note that the encoding logic is simple to implement:

$$d_k = d_{k-1} m_k + \overline{d_{k-1}}.\overline{m_k}$$

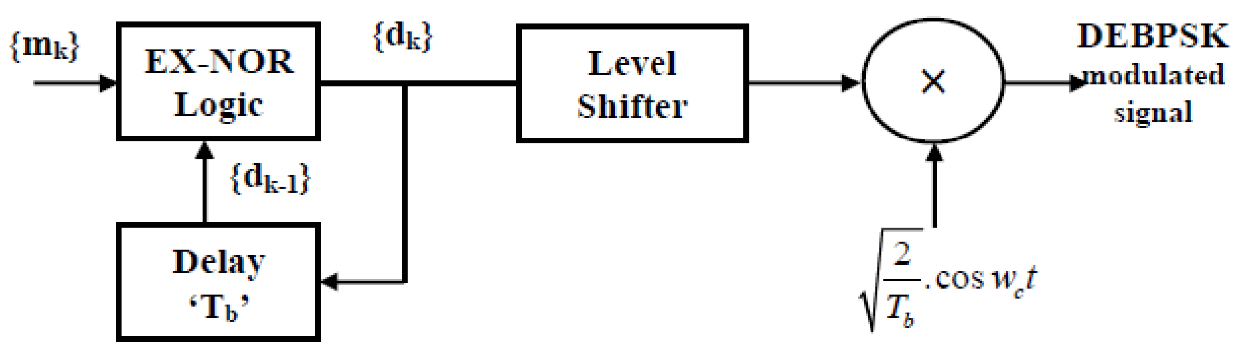
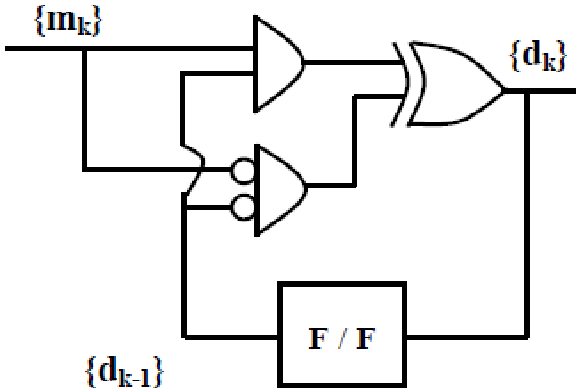


Fig.1(a) Block schematic diagram showing differential encoding for BPSK modulation

Fig. 1(b) shows a possible realization of the differential encoder. It also explains the encoding operation for a sample message sequence {1,0,1,1,0,1,0,0,…} highlighting the phase of the modulated carrier.



$\{m_k\}$:		1	0	1	1	0	1	0	0
$\{d_k\}$:	1	1	0	0	0	1	1	0	1
Tx.Phase :	0	0	π	π	π	0	0	π	0
Demod. Data :		1	0	1	1	0	1	0	0

Fig. 5.26.1(b) A realization of the differential encoder for DEBPSK showing the encoding operation for a sample message sequence

Now, demodulation of a DEBPSK modulated signal can be carried out following the concept of correlation receiver as we have explained earlier in **Lesson #24** (**Fig. 5.24.3**), followed by a differential decoding operation. This ensures optimum (i.e., best achievable) error performance while not requiring a very precise carrier phase recovery scheme. We will refer this combination of correlation receiver with differential encoding-decoding also as the DEBPSK modulation-demodulation scheme.

This is to avoid confusion with another possible scheme of demodulation, which uses a concept of direct differential demodulation. **Fig.5.26.2** explains the differential demodulation scheme for BPSK when differential encoding has been used for BPSK modulation. We refer this demodulator as 'Differential Binary PSK (DBPSK) demodulator'. This is an example of 'non-coherent' demodulation scheme, as it does not require the regenerated carrier for demodulation. So, it is simpler to implement. With reference to the diagram, note that the output x(t) of the multiplier can be expressed without considering the noise component as:

$$x(t) = r(t) \times r(t - T_b) = A_c^{\,2} \left\{ \cos\left[\varpi_c t + \theta + \bar{\mathrm{d}}_k.\pi\right] \times \cos\left[\varpi_c (t - T_b) + \theta + \bar{\mathrm{d}}_{k\text{-}1}.\pi\right] \right\}$$

$$= \frac{A_c^{\,2}}{2} \left\{ \cos\left[\left(\bar{d}_k - \bar{d}_{k-1}\right)\pi\right] + \cos\left[2.\varpi_c t + 2\theta + \left(\bar{d}_k + \bar{d}_{k-1}\right)\pi\right] \right\}$$

$$A_c \cos\left[\varpi_c t + \theta + \bar{d}_k \pi\right]$$

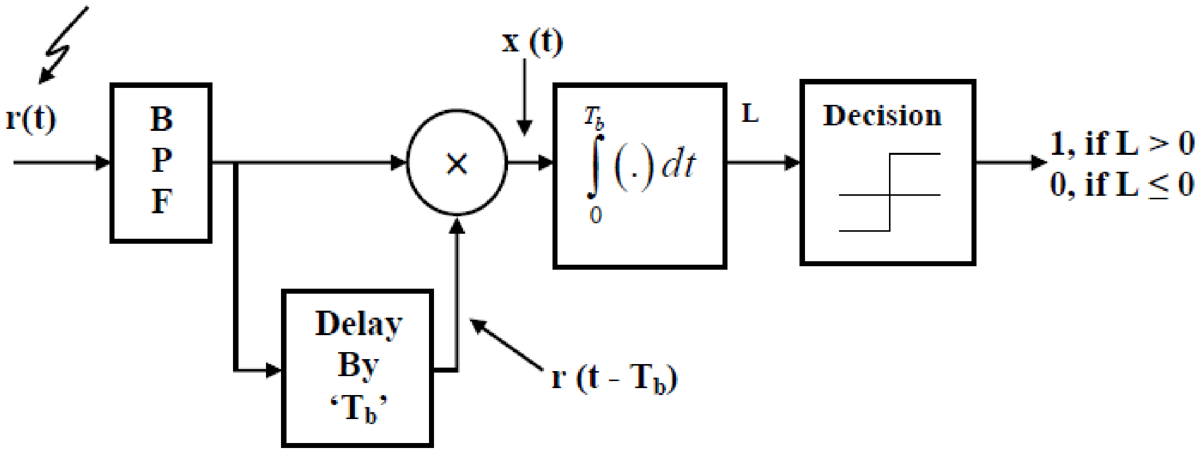


Fig.5.2 Differential demodulation of differentially encoded BPSK modulated signal

Here, the received signal r(t) is:

$$r(t) = A_c \cos\left[\varpi_c t + \theta + \bar{d}_k \pi\right]$$

The integrator, acting as a low pass filter, removes the second term of x(t), which is centered around 2ωc and as a result, the output 'L' of the integrator is ± $A_c^2/2$which is used by the threshold detector to determine estimates of the message bit 'mk' directly. Unlike the DEBPSK demodulation scheme, no separate differential decoding operation is necessary. However, the DBPSK demodulator scheme requires that the IF modulated signal (or equivalently, its time samples) is delayed precisely by 'Tb', the duration of one message bit, and fed to the multiplier. Error performance of the DBPSK demodulation scheme is somewhat inferior to that of ordinary BPSK (or DEBPSK) as one decision error in the demodulator may cause two bits to be in error in quick succession. However, the penalty in error performance is not huge for many applications where lower cost or complexity is preferred more. DBPSK scheme needs about 0.94 dB of additional 0*NEb*to ensure a BER of 10-05, compared to the optimum and coherent BPSK demodulation scheme.

Differential Coding for QPSK

The four possibilities that are to be considered for designing differential encoder and decoder for QPSK are shown in **Table 5.26.1**, assuming that the I-path carrier in the modulator is A cost and the Q-path carrier is -Asint. In any of the four possibilities listed in **Table 5.26.1**, we wish to extract (t) in the I-arm and (t) in the Q-arm using differential encoding and decoding

I-Path Regenerated Carrier	**Q-Path Regenerated carrier**	$\hat{u}_i(t)$	$\hat{u}_q(t)$	**Remarks**
A cos w_c t	-Asin w_c t	u_i(t)	u_q(t)	Correctly derived
-A cos w_c t	Asin w_c t	- u_i(t)	- u_q(t)	Inverted
Asin w_c t	-A cos w_c t	- u_q(t)	- u_i(t)	Swapped and inverted
-Asin w_c t	A cos w_c t	u_q(t)	u_i(t)	Swapped

Table .1 The outputs of the I- and Q- correlators in the demodulator

One can easily verify from the truth table (**Table .2**) that,

$$d_{ik} = \overline{u_{ik}} \,.\, \overline{u_{qk}} \,.\, d_{qk-1} + u_{ik} \,.\, \overline{u_{qk}} \,.\, d_{qk-1} + u_{ik} \,.\, u_{qk} \,.\, d_{ik-1} + u_{qk} \,.\, u_{ik} \,.\, d_{qk-1}$$

$$d_{qk} = \overline{u_{ik}} \,.\, \overline{u_{qk}} \,.\, d_{qk-1} + u_{ik} \,.\, \overline{u_{qk}} \,.\, d_{ik-1} + u_{ik} \,.\, u_{qk} \,.\, \overline{d_{qk-1}} + \overline{u_{ik}} \,.\, u_{qk} \,.\, d_{ik-1}$$

u_{ik}	u_{qk}	d_{ik-1}	d_{qk-1}	d_{ik}	d_{qk}
0	0	0	0	0	0
0	0	0	0	0	1
0	0	1	0	1	0
0	0	1	1	1	1
0	1	0	0	0	1
0	1	0	1	1	1
0	1	1	0	0	0
0	1	1	1	1	0
1	0	0	0	1	0
1	0	0	1	0	0
1	0	1	0	1	1
1	0	1	1	0	1
1	1	0	0	1	1
1	1	0	1	1	0
1	1	1	0	0	1
1	1	1	1	0	0

Table.2 Truth table for differential encoder for QPSK

A feed forward logic circuit is used in a differential decoder in a DEQPSK scheme, which considers the output from the quadrature demodulator to recover and in the correct arms. *ikuqku*

Let us consider a situation, represented by the 11th row of the encoder truth table,

$u_{qk}=0,\ u_{ik}=1,\ d_{ik-1}=1,\ d_{qk-1}=0,\ d_{ik}=0$ and $d_{qk}=1$.

Now, let us consider the four possible phase combination of the quadrature demodulator at the receiver to write the values ofe_i 's and e_q's (**Table 3).**

I-Path carrier	**Q-path carrier**	d_{ik-1} d_{qk-1}	d_{ik} d_{qk}	e_{ik-1} e_{qk-1}	e_{ik} e_{qk}	u_{ik} u_{qk}	**Remarks**
A cos w_c t	-A sin w_c t	1 0	1 1	1 0	1 1	1 0	Phase OK
-A cos w_c t	A sin w_c t	1 0	1 1	0 1	0 0	1 0	Phase inverted
A sin w_c t	-A cos w_c t	1 0	1 1	1 0	0 0	1 0	Data swapped and inverted
-A sin w_c t	A cos w_c t	1 0	1 1	0 1	0 1	1 0	Data swapped

Table.3 *Four phase combinations of the quadrature demodulator at the receiver*

Related to the values of e_i's and e_q's

The last three columns showing 's, 's and the desired outputs partially indicate the necessary logic for designing a differential decoder. Continuing in a similar fashion, one can construct the complete truth table of a differential decoder (**Table.4**).

e_{ik-1} e_{qk-1}	e_{ik}	e_{qk}	$\hat{u}_{ik}$	$\hat{u}_{qk}$
0 0	0	0	0	0
	0	1	0	1
	1	0	1	0
	1	1	1	1
0 1	0	0	1	0
	0	1	0	0
	1	0	1	1
	1	1	0	1
1 0	0	0	0	1
	0	1	1	1
	1	0	0	0
	1	1	1	0
1 1	0	0	1	1
	0	1	1	0
	1	0	0	1
	1	1	0	0

Table .4 *Truth Table of the differential decoder for QPSK*

It is easy to deduce that,

$$\hat{u}_{ik}=\overline{e_{ik}}\cdot\overline{e_{qk}}\cdot e_{qk-1}+e_{ik}\cdot\overline{e_{qk}}\cdot\overline{e_{ik-1}}+e_{ik}\cdot e_{qk}\cdot\overline{e_{qk-1}}+\overline{e_{ik}}\cdot e_{qk}\cdot e_{ik-1}$$

$$\hat{u}_{qk}=\overline{e_{ik}}\cdot\overline{e_{qk}}\cdot e_{ik-1}+e_{ik}\cdot\overline{e_{qk}}\cdot e_{qk-1}+e_{ik}\cdot e_{qk}\cdot\overline{e_{ik-1}}+\overline{e_{ik}}\cdot e_{qk}\cdot\overline{e_{qk-1}}$$

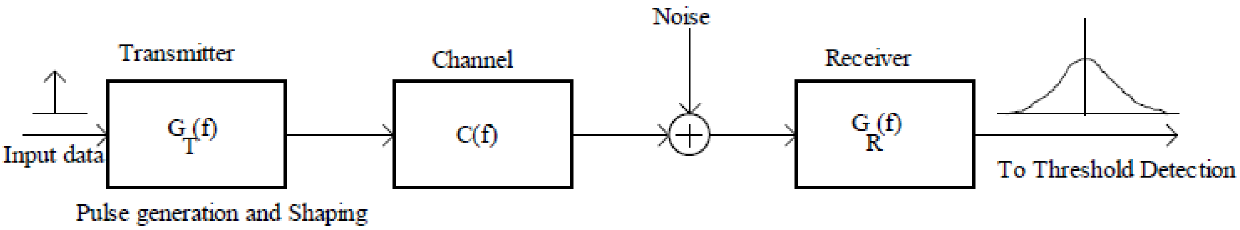
Somewhat analogous to DBPSK, one can design a QPSK modulation-demodulation scheme-using differential encoding in the modulator and employing noncoherent differential demodulation at the receiver. The resultant scheme may be referred as DQPSK. The complexity of such a scheme is less compared to a coherent QPSK scheme because precise recovery of carrier phase is not necessary in the receiver. However, analysis shows that the error performance of DQPSK scheme is considerably poorer compared to the coherent DEQPSK or ordinary coherent QPSK receiver. The differential demodulation approach requires more than 2dB extra E_b/N_0 ensure a BER of 10-05 when compared to ordinary uncoded QPSK with correlation receiver structure.

Optimal reception of digital signal: Baseband signal receiver;

Baseband Pulse Transmission

Baseband digital signals - signals whose spectrum extend down to or near zero frequency.

Model of the transmission link



Consider a *baseband binary PAM system*

i/p
$$a(t) = \sum_k a_k \delta(t - kT)$$

o/p (i.e. from the transmit filter into the channel)

$$s(t) = \sum_k a_k g(t - kT)$$

The receiver filter o/p

$$y(t) = \sum_k a_k h(t - kT) + n(t)$$

n(t) is filtered white noise.

$$h(t) = g_T(t) * c(t) * g_R(t)$$

in the frequency domain

$$H(f) = G_T(f).\ C(f).\ G_R(f)$$

The receive filter output y(t) is sampled at time t= iT (i integer)

$$y(iT) = \sum_{k=-\infty}^{\infty} a_k h[(i-k)T] + n(iT)$$

$$= a_i h(0) + \sum_{k=-\infty, k\neq i}^{\infty} a_k\, h[(i-k)T] + n(iT)$$

The first term aih(0) represents the contribution of the i th transmitted pulse. The second term represents the residual effect of all other transmitted pulses on the decoding of the i th bit.

The residual effect due to the occurrence of pulses before and after sampling instant iT is called ***intersymbol interference (ISI)***.

In the absence of noise and ISI

$$y(iT) = a_i h(0) = a_i \text{ if } h(0) = 1 \text{ (normalizing without loss of generality)}$$

i.e., under ideal conditions i th transmitted bit is decoded correctly.

The presence of noise and ISI degrades the receiver performance and the objective is to minimize these effects and hence deliver the data to its destination with the smalleset error rate possible.

When the signal to noise ratio is high (telephone system PSTN) the operation of the system is largely limited by ISI rather than noise.

Thus first we neglect the effect of noise and focus attention on ISI.

Nyquist's Criterion for Distortionless Baseband Binary Transmission

* Typically, the transfer function of the channel and the transmitted pulse shape are specified.

Objective: To determine the transfer functions of the transmit and receive filters to reconstruct the original data sequence.

* The ISI free condition is satisfied if

$$h(iT - kT) = \begin{cases} 1, & i=k \\ 0, & i\neq k \end{cases}$$

i.e., if $(i-k) = n$ then

$$h(nT) = \begin{cases} 1, & n=0 \\ 0, & n\neq 0 \end{cases}$$

where h(0) =1 by normalization. Thus ignoring noise term,

$$y(iT) = a_i$$

Therefore condition given in (**) ensures "*perfect reception in the absence of noise*".

Let's try to put this in the frequency domain perspective.

If h(t) is sampled at a rate of 1/T, i.e., h(t) multiplied by a periodic impulse train with period T,

$$i(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT)$$

gives us the signal

$$h_s(t) = h(t)\, i(t) = h(t) \sum_{n=-\infty}^{\infty} \delta(t - nT) = \sum_{n=-\infty}^{\infty} h(nT)\delta(t - nT)$$

$$H_s(f) = \sum_{n=-\infty}^{\infty} h(nT) e^{-j2\pi nT} \quad = 1 \text{ since } h(0)=1,\ h\ (n\neq 0\ T) = 0$$

$$h_s(t) = h(0)\ \delta(t) = \delta(t)$$

Hence the Fourier transform

$$H_s(f) = F\{\delta(t)\} = 1$$

Using frequency domain too we can obtain Hs(f).
The Fourier transform of a periodic impulse train is a periodic impulse train

$$I(f) = \frac{1}{T} \sum_{n=-\infty}^{\infty} \delta(f - nf_s)$$

where fs = 1/T is the sampling rate (frequency)
The Fourier transform of hs(t) is

$$H_s(f) = H(f) * I(f) = H(f) * \frac{1}{T} \sum_{n=-\infty}^{\infty} \delta(f - nf_s)$$

$$= \frac{1}{T} \sum_{n=-\infty}^{\infty} H(f - nf_s)$$

Combining these results we have

$$\frac{1}{T}\sum_{n=-\infty}^{\infty} H(f - nf_s) = 1 \quad \text{or} \quad (f_s = 1/T)$$

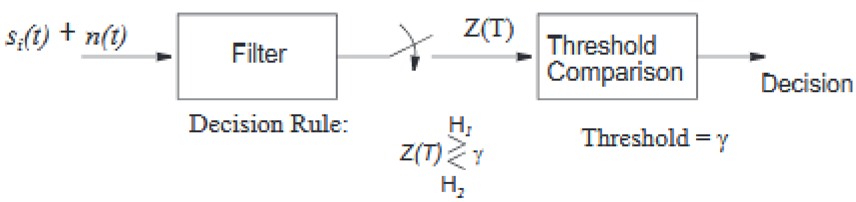
$$\boxed{\sum_{n=-\infty}^{\infty} H(f - n/T) = T}$$

This is the frequency domain requirement for zero ISI.

---> *Nyquist's Criterion for distortionless baseband transmission* in the absence of noise.

* Important to remember that H(f) refers to the overall system, incorporating the transmit filter, the channel, and the receive filter.

Digital receiver and optimum detection



The receiver block consists of a processorwhose structure that maximizes the output signal to noise ratio at the end of each symboltime (T) can be determined as follows. The input to the receiver block is the signals1(contaminated by an additive white Gaussian noise (AWGN),n(t), having a two sided power spectral density $N_0/2$. The output of the block at t=Tis given By

$$v_0(T) = \int_{-\infty}^{T} s(\tau)h(T-\tau) + n(\tau)h(T-\tau)d\tau$$
$$= v_T + n_T$$

It is mathematically more convenient to write (2) in frequency domain as

$$V_0(f) = \int_{-\infty}^{\infty} S(f)H(f) + \sqrt{\frac{N_0}{2}}H(f)df$$

The signal power is given by $|\int_{-\infty}^{\infty} S(f)H(f)df|^2$ and the output noise power $|\int_{-\infty}^{\infty} \frac{N_o}{2}df|H(f)|^2$. To maximize the signal to noise ratio we use Schwartz's inequality,i.e.

$$\frac{|\int_{-\infty}^{\infty} S(f)H(f)df|^2}{\int_{-\infty}^{\infty} \frac{N_o}{2}|H(f)df|^2} \leq \frac{|\int_{-\infty}^{\infty} S(f)df|^2|\int_{-\infty}^{\infty} H(f)df|^2}{|\int_{-\infty}^{\infty} \frac{N_o}{2}|H(f)df|^2}$$

The equality, representing maximum signal to noise ratio, holds when H(f)∗ 4 is equal to kS(f), which means that the H∗ (f) is aligned with a scaled version of S(f). The amplitude scaling represents the gain of the filter, which without a loss of generality can be taken to be unity. When this condition is used in (4) and by taking its inverse Fourier transform a time domain relation between the signaling waveform and the receiver impulse response is obtained. The resulting relationship leads to a concept, which is known as matched filter receiver. It turns out that h(t) = ks(T − t)∗ , which means that when the receiver impulse response equals the complex conjugate of time reflected signalling waveform, the output signal to noise ratio is maximized. Using the condition in (3) the maximum output signal to noise ratio equals Eb No . It is important to note that to achieve maximum signal to noise ratio at the decision device input does not require preservation of the transmitted signal waveform since the maximum signal to noise ratio is equal to the signal energy to noise power spectral density ratio. This result distinguishes digital communication from analog communication where the latter requires that the waveform of the transmitted signal must be reproduced exactly at the receiver output. Another version of the matched filter receiver is obtained when the optimum condition is substituted in (2) to obtain:

$$v_0(T) = \int_{-\infty}^{T} s(\tau)^* s(t-\tau) + n(\tau) s(t-\tau) d\tau$$

Equation (5) give an alternate receiver structure, which also delivers optimum decisions. The correlation receiver as it is called is shown in Figure 3. The two optimum structures defined above can be used for any signaling format. An important question that remains relates to measure of digital communication performance. The probability of making an incorrect decision (probability of error), Pe, is a universally accepted performance measure of digital communication. Consider a long string of binary data symbols consisting of 1s and 0s contaminated by AWGN is received by the receiver. The input to the decision device is then a random process with a mean equal to s1 = vs, s2 = −vs depending whether the transmitted signal was 1 or 0 and its distribution is Gaussian because of the presence of AWGN. The conditional error probability is then determined by finding the area under the probability density curve from −∞ to γ when a 1 is transmitted and from γ to ∞ when a 0 is transmitted as shown in Figure 4. The error probability is then given by

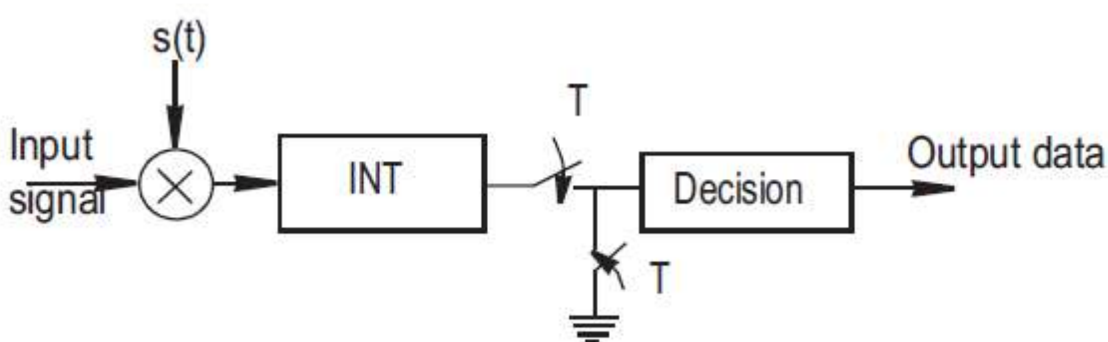
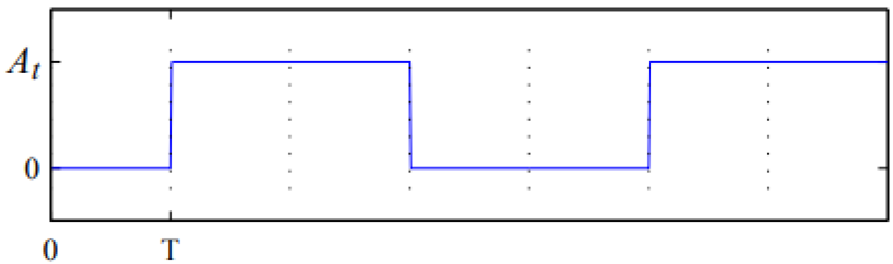


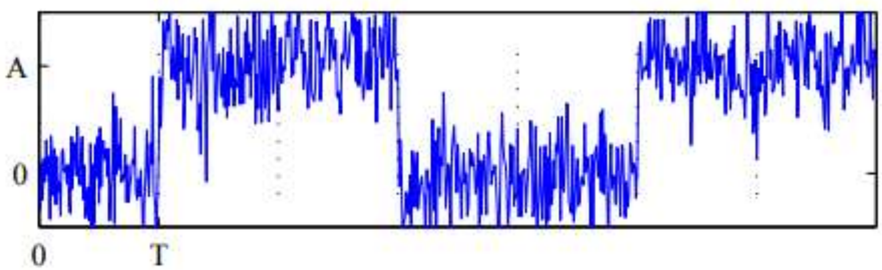
Fig. 3. The Digital Correlation Receiver

Probability of error in transmission

In a binary PCM system, binary digits may be represented by two pulse levels. If these levels are chosen to be 0 and A, the signal is termed an on-off binary signal. If the level switches between −A/2 and A/2 it is called a polar binary signal. Suppose we are transmitting digital information, and decide to do this using two-level pulses each with period T :



The binary digit 0 is represented by a signal of level 0 for the duration T of the transmission, and the digit 1 is represented by the signal level At . In what follows we do not consider modulating the signal — it is transmitted at baseband. In the event of a noisy Gaussian channel (with high bandwidth) the signal at the receiver may look as follows:



Here the binary levels at the receiver are nominally 0 (signal absent) and A (signal present) upon receipt of a 0 or 1 digit respectively. The function of a receiver is to distinguish the digit 0 from the digit 1. The most important performance characteristic of the receiver is the probability

that an error will be made in such a determination. Consider the received signal waveform for the bit transmitted between time 0 and time T . Due to the presence of noise the actual waveform y(t) at the receiver is

y(t) = f (t) + n(t),

where f (t) is the ideal noise-free signal.

In the case described the signal f (t) is

$$f(t) = \begin{cases} 0 & \text{symbol 0 transmitted (signal absent)} \\ 1 & \text{symbol 1 transmitted (signal present).} \end{cases}$$

In what follows, it is assumed that the transmitter and the receiver are synchronised, so the receiver has perfect knowledge of the arrival times of sequences of pulses. The means of achieving this synchronisation is not considered here. This means that without loss of generality we can always assume that the bit to be received lies in the interval (0, T).

Simple detection

A very simple detector could be obtained by sampling the received signal at some time instant Ts in the range (0, T), and using the value to make a decision. The value obtained would be one of the following:

$$y(T_s) = n(T_s) \qquad \text{signal absent}$$
$$y(T_s) = A + n(T_s) \qquad \text{signal present.}$$

Since the value n(T) is random, we cannot decide with certainty whether a signal was present or not at the time of the sample. However, a reasonable rule for the decision of whether a 0 or a 1 was received is the following:

$$y(T) \leq \mu \qquad \text{signal absent — 0 received}$$
$$y(T) > \mu \qquad \text{signal present — 1 received.}$$

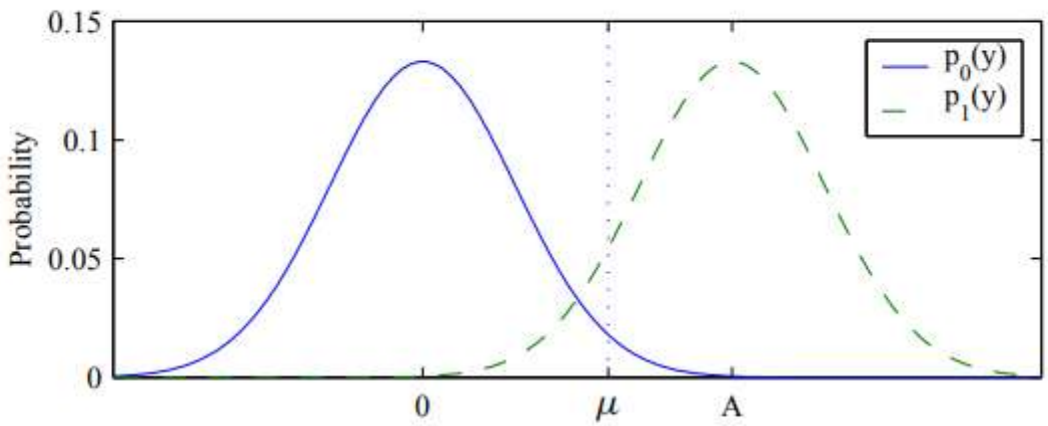
The quantity μ is a threshold which we would usually choose somewhere between 0 and A. For convenience we denote y(Ts) by y. Suppose now that n(Ts) has a Gaussian distribution with a mean of zero and a variance of σ 2 . Under the assumption that a zero was received the probability density of y is

$$p_0(y) = \frac{1}{\sqrt{2\pi}\sigma} e^{-y^2/(2\sigma^2)}.$$

Similarly, when a signal is present, the density of y is

$$p_1(y) = \frac{1}{\sqrt{2\pi}\sigma} e^{-(y-A)^2/(2\sigma^2)}.$$

These are shown below:



Using the decision rule described, it is evident that we sometimes decide that a signal is present even when it is in fact absent. The probability of such a false alarm occurring (mistaking a zero for a one) is

$$P_{\epsilon_0} = \int_{\mu}^{\infty} \frac{1}{\sqrt{2\pi}\sigma} e^{-y^2/(2\sigma^2)} dy.$$

Similarly, the probability of a missed detection (mistaking a one for a zero) is

$$P_{\epsilon_1} = \int_{-\infty}^{\mu} \frac{1}{\sqrt{2\pi}\sigma} e^{-(y-A)^2/(2\sigma^2)} dy.$$

Letting P0 and P1 be the source digit probabilities of zeros and ones 3 respectively, we can define the overall probability of error to be

$$P_\epsilon = P_0 P_{\epsilon_0} + P_1 P_{\epsilon_1}.$$

In the equiprobable case this becomes

$$P_\epsilon = \frac{1}{2}(P_{\epsilon_0} + P_{\epsilon_1}).$$

The sum of these two errors will be minimised for μ = A/2. This sets the decision threshold for a minimum probability of error for P0 = P1 = 1 2 . In that case the probabilities of each type of error are equal, so the overall probability of error is

$$P_\epsilon = \int_{A/2}^{\infty} \frac{1}{\sqrt{2\pi}\sigma} e^{-y^2/(2\sigma^2)} dy.$$

Making the change of variables z = y/σ this integral becomes

$$P_\epsilon = \int_{A/(2\sigma)}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-z^2/2} dz = \mathrm{erfc}\left(\frac{A}{2\sigma}\right).$$

A graph of Pas a function of A/(2σ) can be found in Stremler. This may be written in a more useful form by noting that the average signal power is $S = A^2/2$, and the noise power is $N = \sigma^2$. The probability of error for on-off binary is therefore

$$P_\epsilon = \mathrm{erfc}\sqrt{\frac{S}{2N}}.$$

The selection of voltages 0 and A may be difficult for baseband transmission, since an overall DC current flow is implied. If instead a zero is represented by the voltage −A/2 and a one by A/2, then the entire calculation can be repeated — the only difference is that now $S = (A/2)^2$, so for polar binary

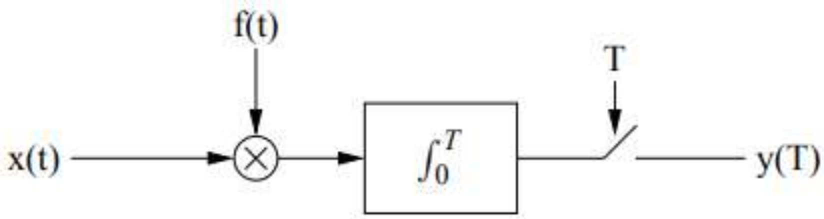
$$P_\epsilon = \mathrm{erfc}\sqrt{\frac{S}{N}}.$$

The on-off binary signal therefore requires twice the signal power of the polar binary signal to achieve the same error rate.

Matched filter

The simple detector just described can be improved upon. Suppose now that the ideal received signal over the interval (0, T) is 0 when the digit 0 is transmitted, and f (t) when 1 is transmitted. Once again there is white Gaussian noise added to the received waveform: x(t) = n(t) signal absent x(t) = f (t) + n(t) signal present.

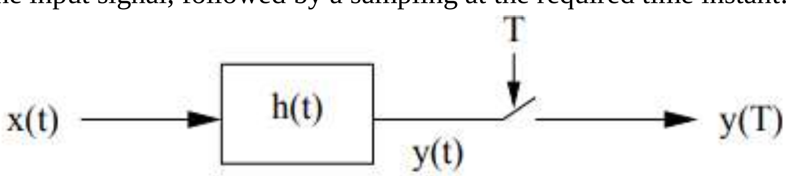
The best detector in this case is the matched filter receiver:



This receiver attempts to recognise the known signal f (t) in the input x(t) by calculating the integral

$$y(T) = \int_0^T x(t) f(t) dt,$$

over the complete signal duration. Recall that the matched filter is the most powerful of all detectors for a known signal in Gaussian noise. The matched filter can more usefully be viewed as a linear convolution of the input signal, followed by a sampling at the required time instant:



The impulse response of this filter should be chosen as

$$h(t) = \begin{cases} f(T-t) & 0 \le t \le T \\ 0 & \text{otherwise} \end{cases}$$

to yield the optimal formulation. The result of the convolution is then

$$y(t) = \int_{-\infty}^{\infty} x(p) h(t-p) dp = \int_{-\infty}^{\infty} x(p) f(p-(t-T)) dp$$
$$= \int_{t-T}^{t} x(p) f(p-(t-T)) dp,$$

since f (t) = 0 for t < 0 and t > T . The output at t = T is then

$$y(T) = \int_0^T x(p) f(p) dp,$$

which is the required match filter value. Note also that subsequent samples also give the required integral for later bit intervals: for t = 2T ,

$$y(2T) = \int_T^{2T} x(p) f(p-T) dp.$$

When the input signal x(t) is identically the signal f (t), the output at t = T is

$$y(T) = \int_0^T |f(t)|^2 dt = E,$$

the energy of the signal. If the input signal is Gaussian white noise with PSD η/2, then the PSD at the filter output is

$$\frac{\eta}{2}|H(\omega)|^2.$$

The total output power over all frequencies is therefore

$$\sigma_{n_0}^2 = \overline{n_0^2(t)} = \frac{1}{2\pi}\int_{-\infty}^{\infty} \frac{\eta}{2}|H(\omega)|^2 d\omega = \frac{\eta}{2}E$$

The output of the matched filter will be

$$y(T) = n_0(T) \qquad \text{signal absent}$$
$$y(T) = E + n_0(T) \qquad \text{signal present.}$$

Here E is the energy in f (t), or

$$E = \int_0^T f^2(t)dt,$$

and n0(T) is the component due to noise. This quantity is a zero-mean random variable, and has a mean-square power (variance) equal to

$$\sigma^2 = \overline{n_0^2(t)} = E\eta/2,$$

where η is the noise power spectral density. Again defining y = y(T), we see that the analysis for the case of simple (sampled) detection continues to apply, except that • The value A must now be replaced by E, and • The value σ 2 must be replaced by n 2 0 (t) = Eη/2. The probability of error for matched filter detection is therefore

$$P_\epsilon = \text{erfc}\sqrt{\frac{E}{2\eta}}.$$

The polar binary case proceeds in the same way, except that the output of the matched filter is
y(T) = −E + n0(t) signal absent
y(T) = E + n0(T) signal present.
Once gain we can substitute 2E for A and √ Eη/2 for σ into the results for the simple case, so

$$P_\epsilon = \text{erfc}\sqrt{\frac{2E}{\eta}}.$$

.

Probability of error for various line encoding formats

The output of a multiplexer is coded into electrical pulses or waveforms for the purpose of transmission over the channel. • This process is called a line coding or transmission coding • Many possible ways of assigning waveforms (pulses) to the digital data.

Properties of Line Coding

Following are the properties of line coding −

- As the coding is done to make more bits transmit on a single signal, the bandwidth used is much reduced.
- For a given bandwidth, the power is efficiently used.
- The probability of error is much reduced.
- Error detection is done and the bipolar too has a correction capability.
- Power density is much favorable.
- The timing content is adequate.
- Long strings of **1s** and **0s** is avoided to maintain transparency.

Types of Line Coding

There are 3 types of Line Coding

- Unipolar
- Polar
- Bi-polar

Unipolar Signaling

Unipolar signaling is also called as **On-Off Keying** or simply **OOK**.

The presence of pulse represents a **1** and the absence of pulse represents a **0**.

There are two variations in Unipolar signaling −

- Non Return to Zero (NRZ)
- Return to Zero (RZ)

Polar Signaling

There are two methods of Polar Signaling. They are −

- Polar NRZ
- Polar RZ

Applications of Line Coding

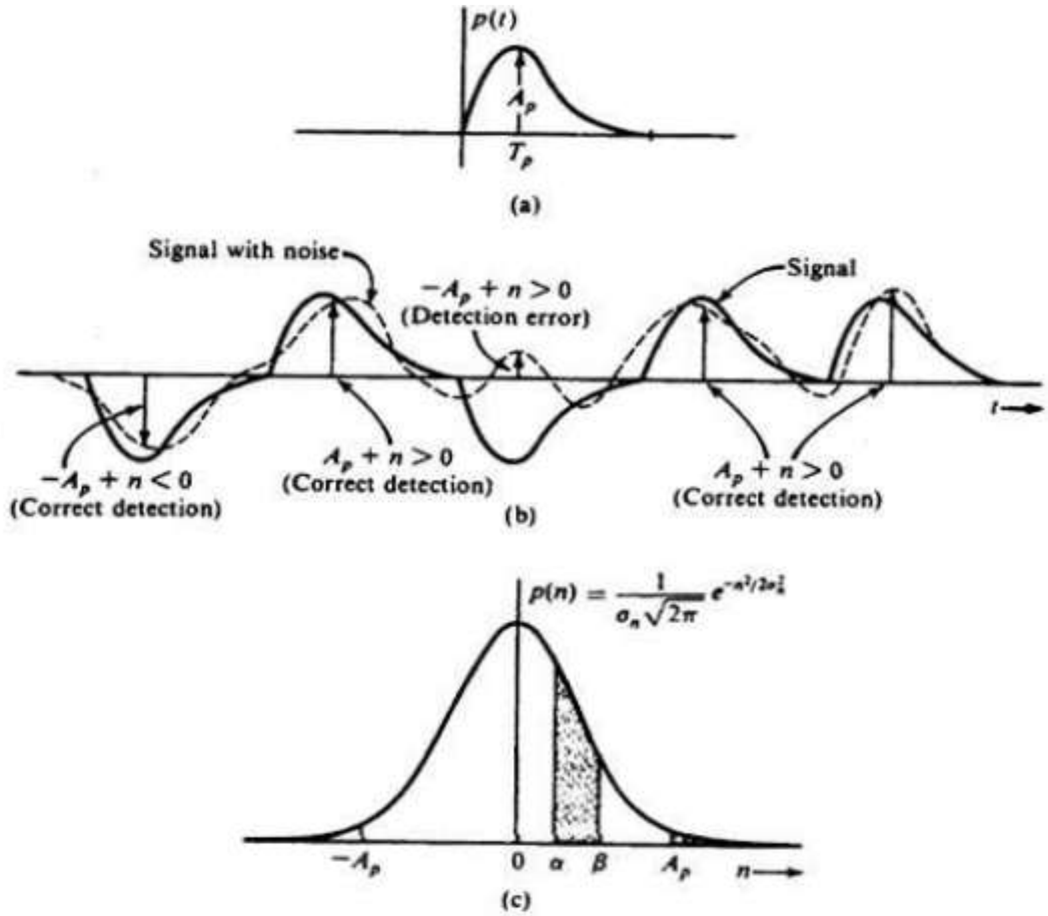
- NRZ encoding: RS232 based protocols
- Manchester encoding: Ethernet networks
- Differential Manchester encoding: token-ring networks
- NRZ-Inverted encoding: Fiber Distributed Data Interface (FDDI)

Comparison of Line Codes:

Sr. No.	Parameters	Polar RZ	Polar NRZ	AMI	Manchester
1	Transmission of DC component	YES	YES	NO	NO
2	Signaling Rate	1/Tb	1/Tb	1/Tb	1/Tb
3	Noise Immunity	LOW	LOW	HIGH	HIGH
4	Synchronizing Capability	Poor	Poor	Very Good	Very Good
5	Bandwidth Required	1/Tb	1/2Tb	1/2Tb	1/Tb
6	Crosstalk	HIGH	HIGH	LOW	LOW

29

Error Probability for Polar Signal



With respect to Figure
P (e/0)= probability that n > Ap
P (e/1)= probability that n < -Ap

Correlation receiver

A Correlation Receiver, consisting of a Correlation Detector and a Vector Receiver implements the M – L decision rule by, (a) first finding r G with a correlation detector and then (b) computing the metric in and taking decision in a vector receiver. Fig.1 shows the structure of a Correlation Detector for determining the received vector r from the received signal r(t). Fig. 1 highlights the operation of a Vector Receiver.

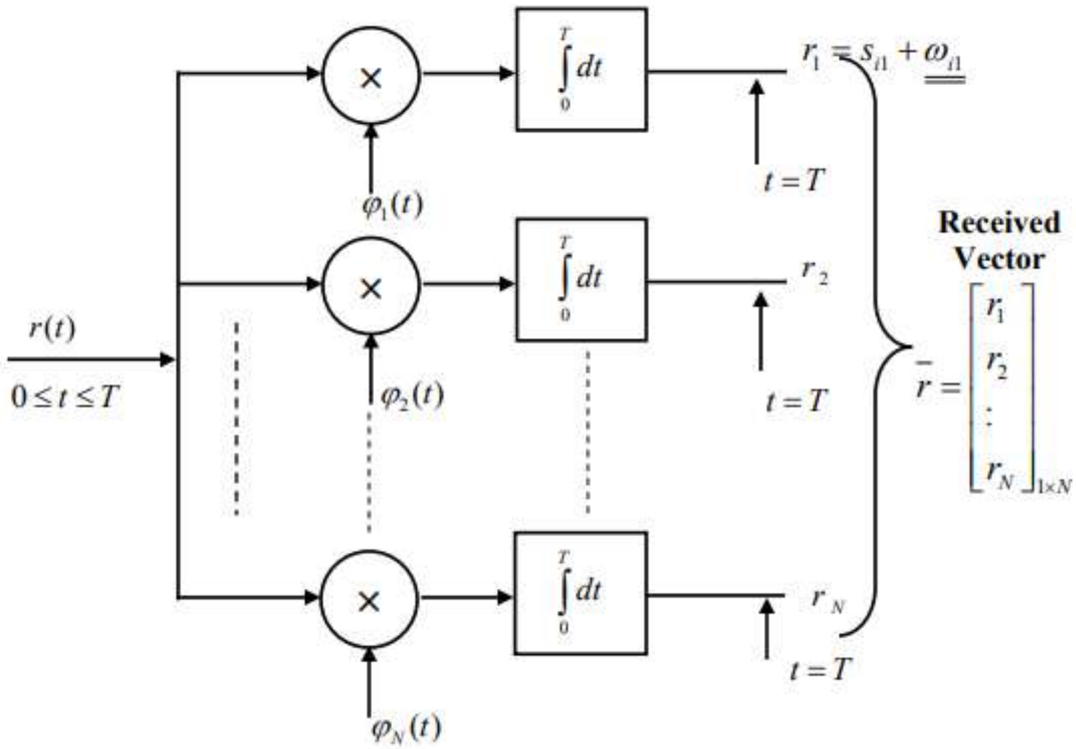
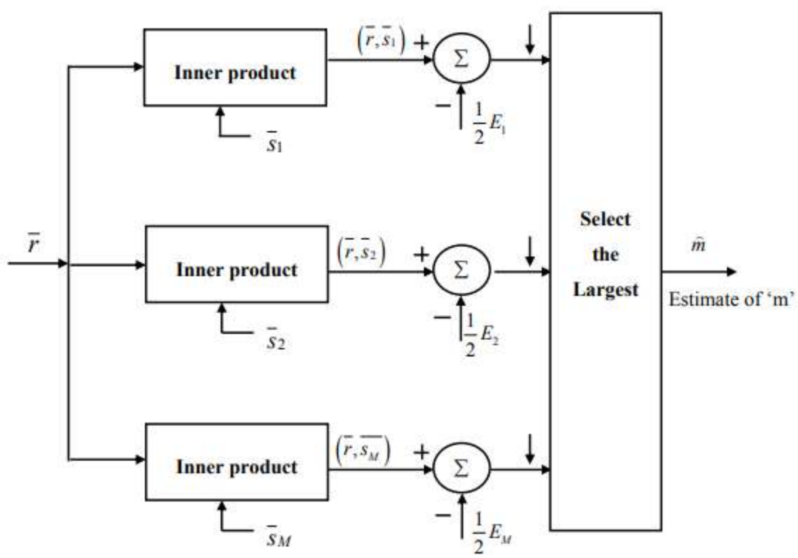


Fig. 1 The structure of a Correlation Detector for determining the received vector r from the received signal r(t)



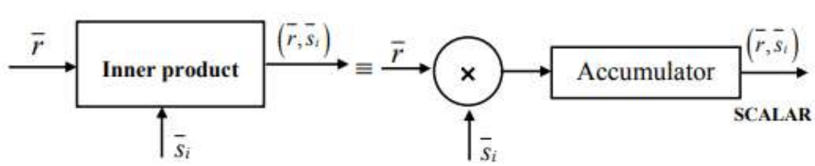


Fig. 2 Block schematic diagram for the Vector Receiver

Features of the received vector r We will now discuss briefly about the statistical features of the received vector r as obtained at the output of the correlation detector . The j-th element of r , which is obtained at the output of the j-th correlator once in T second, can be expressed as:

$$r_j = \int_0^T r(t)\Phi_j(t)dt = \int_0^T [s_i(t) + w(t)]\Phi_j(t)dt$$
$$= s_{ij} + w_j \; ; \quad j=1,2,\ldots, N$$

Here wj is a Gaussian distributed random variable with zero mean and sij is a scalar signal component of si . Now, the mean of the correlator out put is,

$$E[r_j] = E[s_{ij} + w_j] = E[s_{ij}] = s_{ij} = m_{rj},$$

say. We note that the mean of the correlator out put is independent of the noise process. However, the variances of the correlator outputs are dependent on the strength of accompanying noise:

$$Var[r_j] = \sigma_{r_j}^2 = E\left[(r_j - s_{ij})^2\right] = E[w_j^2]$$
$$= E\left[\int_0^T w(t)\Phi_j(t)dt \int_0^T w(u)\Phi_j(u)du\right]$$
$$= E\left[\int_0^T\int_0^T \Phi_j(t)\Phi_j(u).w(t)w(u)dtdu\right]$$

Taking the expectation operation inside, we can write

$$\sigma_{rj}^2 = \int_0^T\int_0^T \Phi_j(t)\Phi_j(u)E[w(t).w(u)]dtdu$$
$$= \int_0^T\int_0^T \Phi_j(t)\Phi_j(u)R_w(t,u)dtdu$$

Here, Rw (t-u) is the auto correlation of the noise process. As we have learnt earlier, additive white Gaussian noise process is a WSS random process and hence the autocorrelation function may be expressed as, Rw (t,u ,) = R (t− u) and further

$$R_w(t-u) = \frac{N_0}{2}\delta(t-u),$$

where 'No' is the single-sided noise power spectral density in Watt/Hz. So, the variance of the correlator output now reduces to:

$$\sigma_{rj}^2 = \frac{N_0}{2}\int_0^T\int_0^T \Phi_j(t)\Phi_j(u)\delta(t-u)dtdu$$
$$= \frac{N_0}{2}\int_0^T \Phi_j^2(t)dt = \frac{N_0}{2}$$

It is interesting to note that the variance of the random signals at the out puts of all N correlators are a) same, b) independent of information-bearing signal waveform and c) dependent only on the noise psd. Now, the likelihood function for si(t), as introduced earlier in Eq.4.19.3 and the ML decision rule [4.19.5] ,

can be expressed in terms of the output of the correlation detector. The likelihood function for 'mi'
$\Pr(\bar{r}|m_i) = f_{\bar{r}}(\bar{r}|m_i) = f_{\bar{r}}(\bar{r}|s_i(t))$,

where, fr(r|mi) is the conditional pdf of 'r' given 'mi'.

In our case, $f_{\bar{r}}(\bar{r}|m_i) = \prod^{N} f_{r_j}(r_j|m_i)$, i = 1,2,…,M

where, $f_{r_j}(r_j|m_i)$ is the pdf of a Gaussian random variable with mean s_{ij} & var. = $\sigma^2_{r_j}$ = $\frac{N_0}{2}$, i.e., $f_{r_j}(r_j|m_i) = \frac{1}{\sqrt{2\pi\sigma^2_{r_j}}}.e^{-\frac{(r_j - s_{ij})^2}{2\sigma_{ij}^2}}$

Combining Eq. 4.19.12 and 4.19.13, we finally obtain,

$$f_{\bar{r}}(\bar{r}|m_i) = (\pi N_0)^{-\frac{N}{2}}.\exp\left[-\frac{1}{N_0}\sum_{j=1}^{N}(r_j - s_{ij})^2\right], \quad i=1,2,\ldots\ldots,M$$

Calculation of probability of error for ASK

5.13.1 Error Probability of ASK

In Amplitude Shift Keying (ASK), some number of carrier cycles are transmitted to send '1' and no signal is transmitted for binary '0'. Thus,

Binary '1' $\Rightarrow$ $x_1(t) = \sqrt{2P_s}\cos(2\pi f_0 t)$ and

Binary '0' $\Rightarrow$ $x_2(t) = 0$ (i.e. no signal) ... (5.13.1)

Here P_s is the normalized power of the signal in 1Ω load. i.e. power $P_s = \frac{A^2}{2}$. Hence $A = \sqrt{2P_s}$. Therefore in above equation for $x_1(t)$ amplitude 'A' is replaced by $\sqrt{2P_s}$.

We know that the probability of error of the optimum filter is given as,

$$P_e = \frac{1}{2}\,erfc\left\{\frac{x_{01}(T) - x_{02}(T)}{2\sqrt{2}\,\sigma}\right\} \quad \text{... (5.13.2)}$$

Here $\left[\frac{x_{01}(T) - x_{02}(T)}{\sigma}\right]^2_{max} = \int_{-\infty}^{\infty} \frac{|X(f)|^2}{S_{ni}(f)}\,df$

The above equations can be applied to matched filter when we consider white Gaussian noise. The power spectral density of white Gaussian noise is given as,

$$S_{ni}(f) = \frac{N_0}{2}$$

Putting this value of $S_{ni}(f)$ in above equations we get,

$$\left[\frac{x_{01}(T) - x_{02}(T)}{\sigma}\right]^2_{max} = \int_{-\infty}^{\infty} \frac{|X(f)|^2}{\frac{N_0}{2}}\,df$$

$$= \frac{2}{N_0}\int_{-\infty}^{\infty} |X(f)|^2\,df \quad \text{... (5.13.3)}$$

Parseval's power theorem states that,

$$\int_{-\infty}^{\infty} |X(f)|^2\,df = \int_{-\infty}^{\infty} x^2(t)\,dt$$

Hence equation 5.13.3 becomes,

$$\left[\frac{x_{01}(T) - x_{02}(T)}{\sigma}\right]^2_{max} = \frac{2}{N_0}\int_{-\infty}^{\infty} x^2(t)\,dt$$

We know that $x(t)$ is present from 0 to T. Hence limits in above equation can be changed as follows :

$$\left[\frac{x_{01}(T)-x_{02}(T)}{\sigma}\right]^2_{max} = \frac{2}{N_0}\int_0^T x^2(t)\,dt \qquad \text{... (5.13.4)}$$

We know that $x(t) = x_1(t) - x_2(t)$. For ASK $x_2(t)$ is zero, hence $x(t) = x_1(t)$. Hence above equation becomes,

$$\left[\frac{x_{01}(T)-x_{02}(T)}{\sigma}\right]^2_{max} = \frac{2}{N_0}\int_0^T x_1^2(t)\,dt$$

Putting equation of $x_1(t)$ from equation 5.13.1 in above equation we get,

$$\left[\frac{x_{01}(T)-x_{02}(T)}{\sigma}\right]^2_{max} = \frac{2}{N_0}\int_0^T \left[\sqrt{2P_s}\cos(2\pi f_0 t)\right]^2 dt$$

$$= \frac{4P_s}{N_0}\int_0^T \cos^2(2\pi f_0 t)\,dt$$

We know that $\cos^2\theta = \frac{1+\cos 2\theta}{2}$. Here applying this formula to $\cos^2(2\pi f_0 t)$ we get,

$$\left[\frac{x_{01}(T)-x_{02}(T)}{\sigma}\right]^2_{max} = \frac{4P_s}{N_0}\int_0^T \frac{1+\cos 4\pi f_0 t}{2}\,dt$$

$$= \frac{4P_s}{N_0}\cdot\frac{1}{2}\left\{\int_0^T dt + \int_0^T \cos 4\pi f_0 t\,dt\right\}$$

$$= \frac{2P_s}{N_0}\left\{[t]_0^T + \left[\frac{\sin 4\pi f_0 t}{4\pi f_0}\right]_0^T\right\}$$

$$= \frac{2P_s}{N_0}\left\{T + \frac{\sin 4\pi f_0 T}{4\pi f_0}\right\} \qquad \text{... (5.13.5)}$$

We know that T is the bit period and in this one bit period, the carrier has integer number of cycles. Thus the product $f_0 T$ is an integer. This is illustrated in Fig. 5.13.1

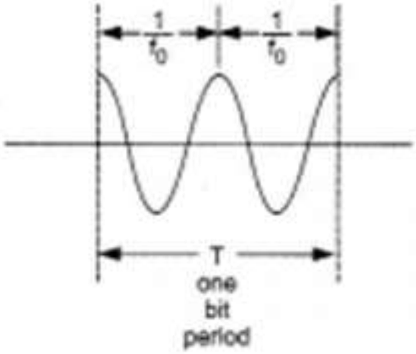


Fig. 5.13.1 In one bit period T, the carrier completes its two cycles. The carrier has frequency f_0. From figure we can write,

$$T = \frac{1}{f_0} + \frac{1}{f_0}$$

i.e. $$T = \frac{2}{f_0}$$

$\therefore \quad f_0 T = 2$ **(integer no. of cycles)**

As shown in above figure, the carrier completes two cycles in one bit duration. Hence

$$f_0 T = 2$$

Therefore, in general if carrier completes 'k' number of cycles, then,

$$f_0 T = k \qquad \text{(Here } k \text{ is an integer)}$$

Therefore the sine term in equation 5.13.5 becomes, $\sin 4\pi k$ and k is integer.

For all integer values of k, $\sin 4\pi k = 0$. Hence equation 5.13.5 becomes,

$$\left[\frac{x_{01}(T) - x_{02}(T)}{\sigma}\right]^2_{max} = \frac{2P_s T}{N_0} \qquad \text{... (5.13.6)}$$

$$\therefore \quad \left[\frac{x_{01}(T) - x_{02}(T)}{\sigma}\right]_{max} = \sqrt{\frac{2P_s T}{N_0}} \qquad \text{... (5.13.7)}$$

Putting this value in equation 5.13.2 we get error probability of ASK using matched filter detection as,

$$P_e = \frac{1}{2} erfc\left\{\frac{1}{2\sqrt{2}} \cdot \sqrt{\frac{2P_s T}{N_0}}\right\} = \frac{1}{2} erfc \sqrt{\frac{P_s T}{4N_0}}$$

Here $P_s T = E$, i.e. energy of one bit hence above equation becomes,

$$\textbf{Error probability of ASK : } P_e = \frac{1}{2} erfc \sqrt{\frac{E}{4N_0}} \qquad \text{... (5.13.8)}$$

This is the expression for error probability of ASK using matched filter detection.

Calculation of probability of error for FSK

BFSK is a two-dimensional modulation scheme with two time-limited signals as reproduced below:

$$s_i(t) = \begin{cases} \sqrt{\dfrac{2E_b}{T_b}} \cos 2\pi f_i t, & 0 \le t \le T_b, i = 1,2 \\ 0, & \text{elsewhere.} \end{cases}$$

We assume appropriately chosen 'mark' and 'space' frequencies such that the two basis functions are orthonormal:

$$\varphi_i(t) = \sqrt{\frac{2}{T_b}} \cos 2\pi f_i t \quad ; \quad 0 \le t \le T_b \text{ and } j = 1,2$$

We will consider the coherent demodulator structure [Fig. 5.28.1(b)] obtain the optimum error performance for binary FSK. For ease of reference, the signal constellation for BFSK is reproduced in Fig. 5.28.1(a). As we can see, the two signal vectors are $\bar{s}_1 = \begin{bmatrix} \sqrt{E_b} \\ 0 \end{bmatrix}$ and $\bar{s}_2 = \begin{bmatrix} 0 \\ \sqrt{E_b} \end{bmatrix}$ while the two associated scalars are s11 = s22 = $\sqrt{E_b}$. The decision zones are shown by the discontinuous line.

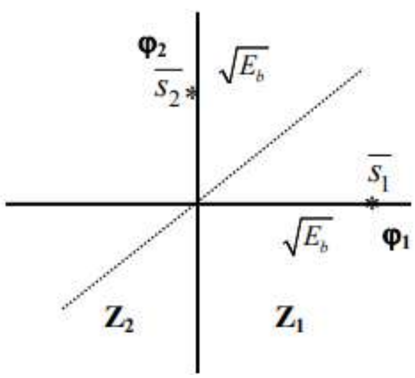


Fig. 5.28.1(a) Signal constellation for BFSK showing the decision zones Z1 and Z2

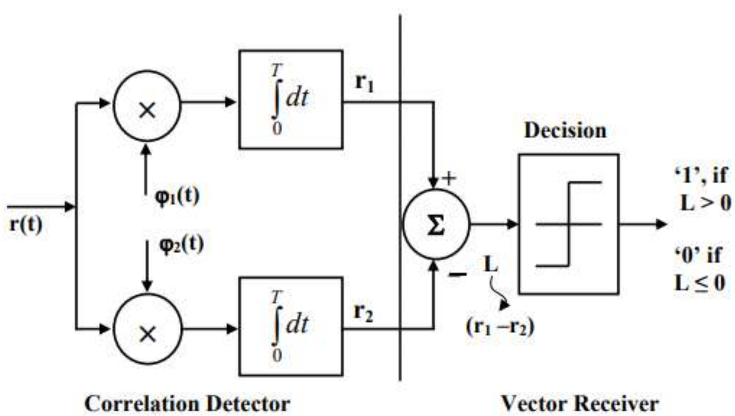


Fig. 5.28.1(b) Coherent demodulator structure for BFSK highlighting the decision process
Now, suppose 1s represents logic '1' and 2 s represents logic '0'. If s1(t) is transmitted and if noise is absent, the output of the upper correlator in Fig. 5.28.1(b), i.e. r1 is $\sqrt{E_b}$ while the output of the lower correlator, i.e. r2 is zero. So, we see that the intermediate parameter L = (r1 – r2) > 0. Similarly, it is easy to see that if s2(t) is transmitted, L < 0. Now, from Fig. 5.28.1(a) we see that the decision boundary is a straight line with unit slope. This implies that, if the received vector r at the output of the correlator bank is in decision zone Z1, then L > 0 and otherwise it is in zone Z2.

When we consider additive noise, 'L' represents a random variable whose mean is + $\sqrt{E_b}$ if message '1' is transmitted. For message '0', the mean of 'L' is - $\sqrt{E_b}$. Further, r1 and r2 are independent and identically distributed random variables with the same variance 2 No . So, variance of 'L' = variance of 'r1' + variance of 'r2' = 2 No + 2 No = 5.28.3 No Now, assuming that a '0' has been transmitted, the likelihood function is:

$$f_L(l\,|\,0) = \frac{1}{\sqrt{2\pi N_0}}.\exp\left[-\frac{\left\{l-\left(-\sqrt{E_b}\right)\right\}^2}{2N_0}\right]$$

$$= \frac{1}{\sqrt{2\pi N_0}}.\exp\left[-\frac{\left(l+\sqrt{E_b}\right)^2}{2N_0}\right]$$

In the above expressions, 'l' represents a sample value of the random variable 'L'. From the above expression, we can determine the average probability of error when '0'-s are transmitted as: Pe ( ) 0 =

Average probability of error when '0'– s are transmitted $= \int_0^{\infty} f_L(l\,|\,0)\,dl$

$$= \frac{1}{\sqrt{2\pi N_0}} \cdot \int_0^{\infty} \exp\left[-\frac{\left(l + \sqrt{E_b}\right)^2}{2N_0} \right] dl$$

Putting $\frac{l + \sqrt{E_b}}{\sqrt{2N_0}} = Z$ in the above expression, we readily get,

$$\mathrm{P}e(0) = \frac{1}{\sqrt{\pi}} \int_{\sqrt{\frac{E_b}{2N_0}}}^{\infty} \exp\left(-Z^2\right) dz$$

$$= \frac{1}{2} erfc\left(\sqrt{\frac{E_b}{2N_0}} \right)$$

Following similar approach, we get,

$P_e(1)$ = Average probability of error when '1'-s are transmitted

$$= \frac{1}{2} erfc\left(\sqrt{\frac{E_b}{2N_0}} \right)$$

Now, using the justification that '1' and '0' are equally likely to occur at the input of the modulator, the

overall BER = Pe . 1/2 Pe (0) + . 1/2 Pe (1)= $\frac{1}{2} erfc\left(\sqrt{\frac{E_b}{2N_0}} \right)$

Calculation of probability of error for BPSK

We consider ordinary BPSK modulation with optimum demodulation by a correlation receiver Fig. 5.27.1 shows the familiar signal constellation for the scheme including an arbitrarily chosen .

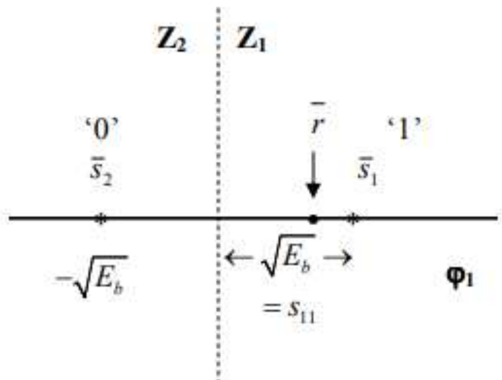


Fig.5.27.1 *Signal constellation for BPSK showing an arbitrary received vector* $\bar{r}$

Now, with reference to **Fig. 5.27.1**, we observe that, an error occurs if

a) $s_1(t)$ is transmitted while $\bar{r}$ is in Z_2 or
b) $s_2(t)$ is transmitted while $\bar{r}$ is in Z_1.

Further, if 'r' denotes the output of the correlator of the BPSK demodulator, we know the decision zone in which $\bar{r}$ lies from the following criteria:

a) $\bar{r}$ lies in Z_1 if $r = \int_0^{T_b} r(t)\varphi_1(t)dt > 0$

b) $\bar{r}$ lies in Z_2 if $r \le 0$

Now, from **Eq. 5.27.1**, we can construct an expression for a Likelihood Function:
$f_r\left(\bar{r} \middle| s_2(t)\right) = f_r(r(t)$/message '0' was transmitted)

From our previous discussion,

$$f_r\left(\bar{r}\middle|s_2(t)\right) = f_r\left(\bar{r}\middle|'0'\right)$$

$$= \frac{1}{\sqrt{\pi N_0}}.\exp\left[-\frac{1}{N_0}\left(r - s_{21}\right)^2\right]$$

$$= \frac{1}{\sqrt{\pi N_0}}.\exp\left\{\frac{-\left[r-\left(-\sqrt{E_b}\right)\right]^2}{N_0}\right\}$$

$$= \frac{1}{\sqrt{\pi N_0}}.\exp\left[-\frac{1}{N_0}\left(r+\sqrt{E_b}\right)^2\right]$$

$\therefore$ The conditional Probability that the receiver decides in favour of '1' while '0' was transmitted $= \int_0^\infty f_r\left(r|0\right)dr = P_e(0)$, say.

$$\therefore P_e(0) = \frac{1}{\sqrt{\pi N_0}}\int_0^\infty \exp\left[-\frac{1}{N_0}\left(r+\sqrt{E_b}\right)^2\right]dr \qquad 5.27.3$$

Now, putting $\frac{1}{\sqrt{N_0}}\left(r+\sqrt{E_b}\right) = Z$, we get,

$$P_e(0) = \frac{1}{\sqrt{\pi}}\int_{\sqrt{E_b/N_0}}^{\infty} \exp\left(-Z^2\right)dz$$

$$= \frac{1}{2}.\frac{2}{\sqrt{\pi}}\int_{\sqrt{\frac{E_b}{N_0}}}^{\infty} e^{-z^2}dz$$

$$= \frac{1}{2}.erfc\left(\sqrt{\frac{E_b}{N_o}}\right) = Q\left(\sqrt{\frac{2E_b}{N_o}}\right), \qquad \left[\because erf_c(u) = 2Q\left(\sqrt{2}\,u\right)\right] \qquad 5.27.4$$

Fig. 5.27.2 shows the profiles for the error function erf(x) and the complementary error function erfc(x)

Following a similar approach as above, we can determine the probability of error when '1' is transmitted from the modulator, i.e. (1) Pe as,

$$P_e(1) = \frac{1}{2}.erfc\left(\sqrt{\frac{E_b}{N_o}}\right)$$

Now, as we have assumed earlier, the '0'-s and '1'-s are equally likely to occur at the input of the modulator and hence, the average probability of a received bit being decided erroneously (Pe) is,

$$P_e = \frac{1}{2}.P_e(0) + \frac{1}{2}P_e(1) = \frac{1}{2}.erfc\left(\sqrt{\frac{E_b}{N_o}}\right)$$

We can easily recognize that Pe is the BER, or equivalently the SER(Symbol error rate) for the optimum BPSK modulator. This is the best possible error performance any BPSK modulator-demodulator can achieve in presence of AWGN. Fig. 5.27.3 depicts the above relationship. This figure demands some attention as it is often used as a benchmark for comparing error performance of other carrier modulation schemes. Careful observation reveals that about 9.6 dB of o b N E is necessary to achieve a BER of 10-05 while an o b N E of 8.4 dB implies an achievable BER of 10^{-04}.

UNIT – III

BASE BAND TRANSMISSSION AND PULSE SHAPING

Base Band Transmission: Requirements of a line encoding format

There are many reasons for using line coding. Each of the line codes you will be examining offers one or more of the following advantages: spectrum shaping and relocation without modulation or filtering. This is important in telephone line applications, for example, where the transfer characteristic has heavy attenuation below 300 Hz. bit clock recovery can be simplified. DC component can be eliminated; this allows AC (capacitor or transformer) coupling between stages (as in telephone lines). Can control baseline wander (baseline wander shifts the position of the signal waveform relative to the detector threshold and leads to severe erosion of noise margin). error detection capabilities. bandwidth usage; the possibility of transmitting at a higher rate than other schemes over the same bandwidth. At the very least the LINE-CODE ENCODER serves as an interface between the TTL level signals of the transmitter and those of the analog channel. Likewise, the LINE-CODE DECODER serves as an interface between the analog signals of the channel and the TTL level signals required by the digital receiver.

The two new modules to be introduced are the LINE-CODE ENCODER and the LINE-CODE DECODER. You will not be concerned with how the coding and decoding is performed. You should examine the waveforms, using the original TTL sequence as a reference. In a digital transmission system line encoding is the final digital processing performed on the signal before it is connected to the analog channel, although there may be simultaneous bandlimiting and wave shaping. Thus in TIMS the LINE-CODE ENCODER accepts a TTL input, and the output is suitable for transmission via an analog channel. At the channel output is a signal at the TIMS ANALOG REFERENCE LEVEL, or less. It could be corrupted by noise. Here it is re-generated by a detector. The TIMS detector is the DECISION MAKER module (already examined in the experiment entitled Detection with the DECISION MAKER in this Volume). Finally the TIMS LINE-CODE DECODER module accepts the output from the DECISION MAKER and decodes it back to the binary TTL format. Preceding the line code encoder may be a source encoder with a matching decoder at the receiver. These are included in the block diagram of Figure 1, which is of a typical baseband digital transmission system. It shows the disposition of the LINECODE ENCODER and LINE-CODE DECODER. All bandlimiting is shown concentrated in the channel itself, but could be distributed between the transmitter, channel, and receiver.

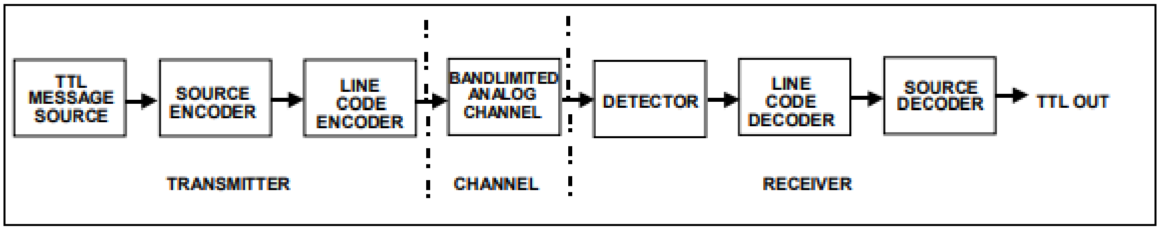


Figure 1: baseband transmission system

The LINE-CODE ENCODER serves as a source of the system bit clock. It is driven by a master clock at 8.333 kHz (from the TIMS MASTER SIGNALS module). It divides this by a factor of four, in order to derive some necessary internal timing signals at a rate of 2.083 kHz. This then becomes a convenient source of a 2.083 kHz TTL signal for use as the system bit clock. Because the LINE-CODE DECODER has some processing to do, it introduces a time delay. To allow for this, it provides a re-timed clock if required by any further digital processing circuits (eg, for decoding, or error counting modules).

NRZ-L Non return to zero - level (bipolar): this is a simple scale and level shift of the input TTL waveform.

NRZ-M Non return to zero - mark (bipolar): there is a transition at the beginning of each '1', and no change for a '0'. The 'M' refers to 'inversion on mark'. This is a differential code. The decoder will give the correct output independently of the polarity of the input.

UNI-RZ Uni-polar - return to zero (uni-polar): there is a half-width output pulse if the input is a '1'; no output if the input is a '0'. This waveform has a significant DC component.

BIP-RZ Bipolar return to zero (3-level): there is a half-width +ve output pulse if the input is a '1'; or a half-width -ve output pulse if the input is a '0'. There is a return-to-zero for the second half of each bit period.

RZ-AMI Return to zero - alternate mark inversion (3-level): there is a half-width output pulse if the input is a '1'; no output if the input is a '0'. This would be the same as UNIRZ. But, in addition, there is a polarity inversion of every alternate output pulse.

Biφ-L Biphase - level (Manchester): bipolar ±V volts. For each input '1' there is a transition from +V to -V in the middle of the bit-period. For each input '0' there is a transition from -V to +V in the middle of the bit period.

DICODE-NRZ Di-code non-return to zero (3-level): for each transition of the input there is an output pulse, of opposite polarity from the preceding pulse. For no transition between input pulses there is no output. The codes offered by the line-code encoder are illustrated in Figure 2 below.

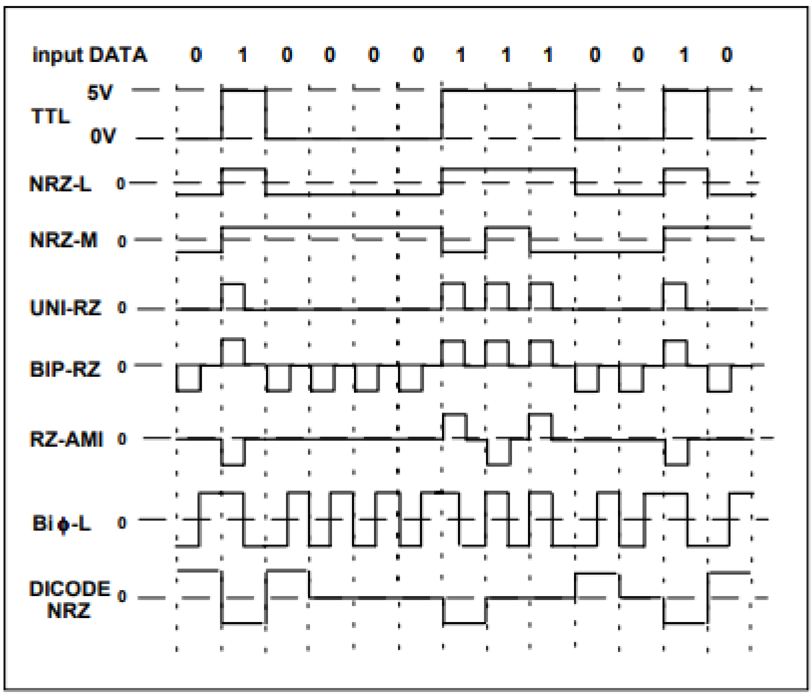


Figure 2: TIMS line codes

The output waveforms, apart from being encoded, have all had their amplitudes adjusted to suit a TIMS analog channel (not explicitly shown in Figure 2). When connected to the input of the LINE-CODE DECODER these waveforms are de-coded back to the original TTL sequence.

Various line encoding formats: Unipolar, Polar, Bipolar

A **line code** is the code used for data transmission of a digital signal over a transmission line. This process of coding is chosen so as to avoid overlap and distortion of signal such as inter-symbol interference.

Properties of Line Coding

Following are the properties of line coding −

- As the coding is done to make more bits transmit on a single signal, the bandwidth used is much reduced.
- For a given bandwidth, the power is efficiently used.
- The probability of error is much reduced.
- Error detection is done and the bipolar too has a correction capability.
- Power density is much favorable.
- The timing content is adequate.
- Long strings of **1s** and **0s** is avoided to maintain transparency.

Types of Line Coding

There are 3 types of Line Coding

- Unipolar
- Polar
- Bi-polar

Unipolar Signaling

Unipolar signaling is also called as **On-Off Keying** or simply **OOK**.

The presence of pulse represents a **1** and the absence of pulse represents a **0**.

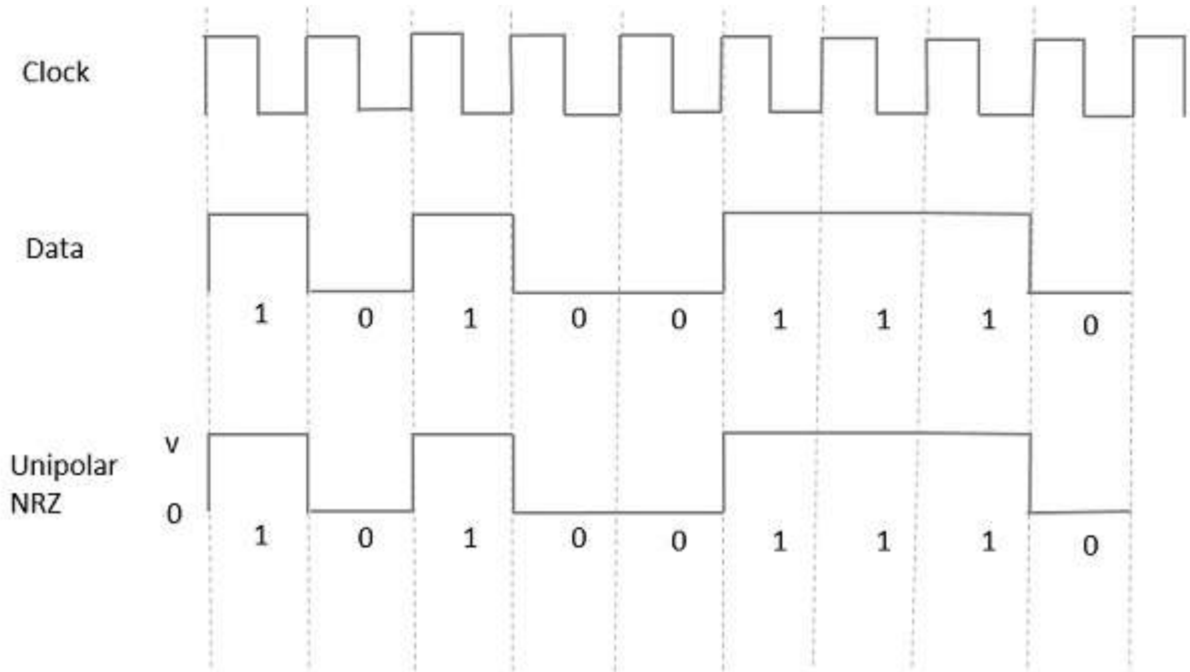
There are two variations in Unipolar signaling −

- Non Return to Zero (NRZ)
- Return to Zero (RZ)

Unipolar Non-Return to Zero (NRZ)

In this type of unipolar signaling, a High in data is represented by a positive pulse called as **Mark**, which has a duration T_0 equal to the symbol bit duration. A Low in data input has no pulse.

The following figure clearly depicts this.



Advantages

The advantages of Unipolar NRZ are −

- It is simple.
- A lesser bandwidth is required.

Disadvantages

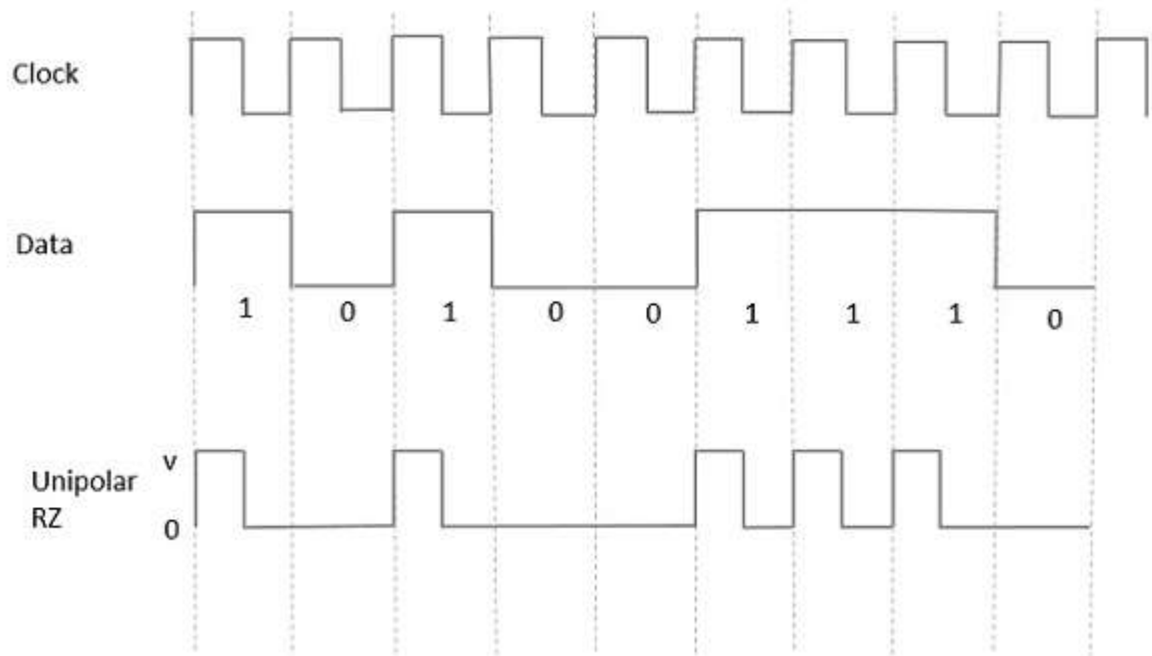
The disadvantages of Unipolar NRZ are −

- No error correction done.
- Presence of low frequency components may cause the signal droop.
- No clock is present.
- Loss of synchronization is likely to occur (especially for long strings of **1s** and **0s**).

Unipolar Return to Zero (RZ)

In this type of unipolar signaling, a High in data, though represented by a **Mark pulse**, its duration $\mathbf{T_0}$ is less than the symbol bit duration. Half of the bit duration remains high but it immediately returns to zero and shows the absence of pulse during the remaining half of the bit duration.

It is clearly understood with the help of the following figure.



Advantages

The advantages of Unipolar RZ are −

- It is simple.
- The spectral line present at the symbol rate can be used as a clock.

Disadvantages

The disadvantages of Unipolar RZ are −

- No error correction.
- Occupies twice the bandwidth as unipolar NRZ.
- The signal droop is caused at the places where signal is non-zero at 0 Hz.

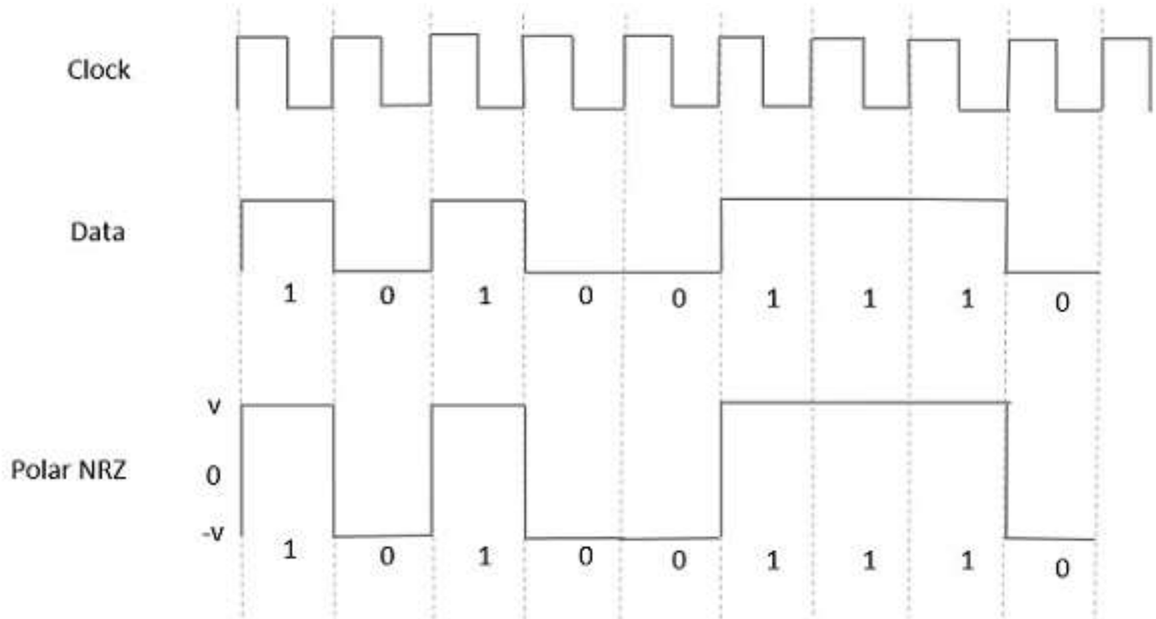
Polar Signaling

There are two methods of Polar Signaling. They are −

- Polar NRZ
- Polar RZ

Polar NRZ

In this type of Polar signaling, a High in data is represented by a positive pulse, while a Low in data is represented by a negative pulse. The following figure depicts this well.



Advantages

The advantages of Polar NRZ are −

- It is simple.
- No low-frequency components are present.

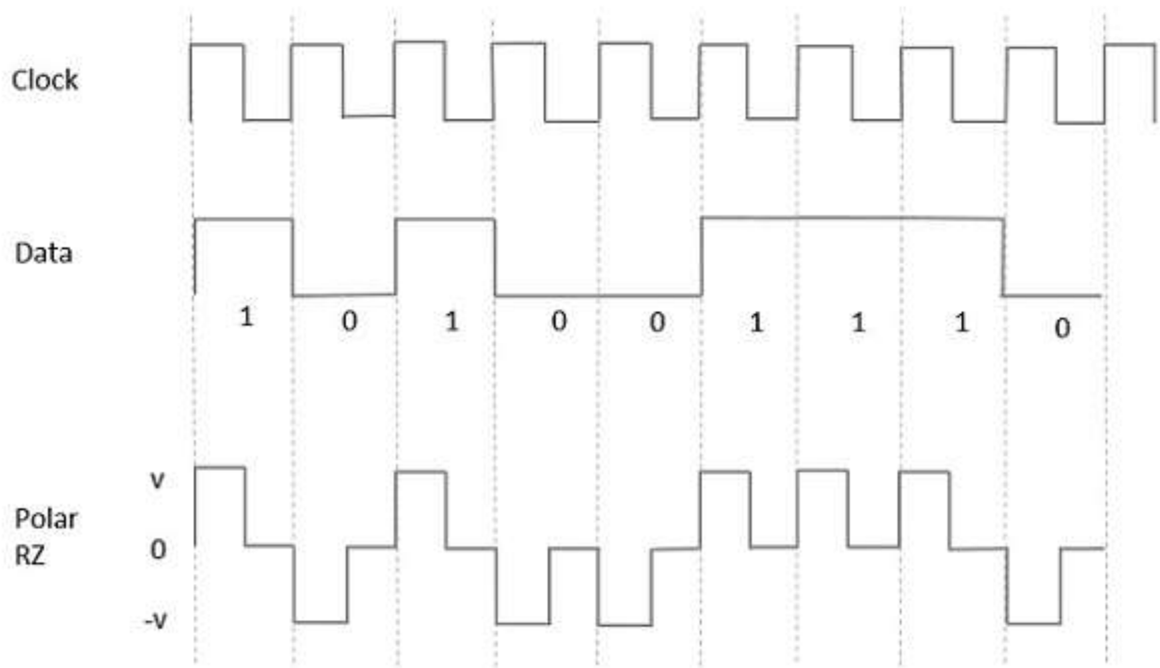
Disadvantages

The disadvantages of Polar NRZ are −

- No error correction.
- No clock is present.
- The signal droop is caused at the places where the signal is non-zero at **0 Hz**.

Polar RZ

In this type of Polar signaling, a High in data, though represented by a **Mark pulse**, its duration $\mathbf{T_0}$ is less than the symbol bit duration. Half of the bit duration remains high but it immediately returns to zero and shows the absence of pulse during the remaining half of the bit duration.However, for a Low input, a negative pulse represents the data, and the zero level remains same for the other half of the bit duration. The following figure depicts this clearly.



Advantages

The advantages of Polar RZ are −

- It is simple.
- No low-frequency components are present.

Disadvantages

The disadvantages of Polar RZ are −

- No error correction.
- No clock is present.
- Occupies twice the bandwidth of Polar NRZ.
- The signal droop is caused at places where the signal is non-zero at **0 Hz**.

Bipolar Signaling

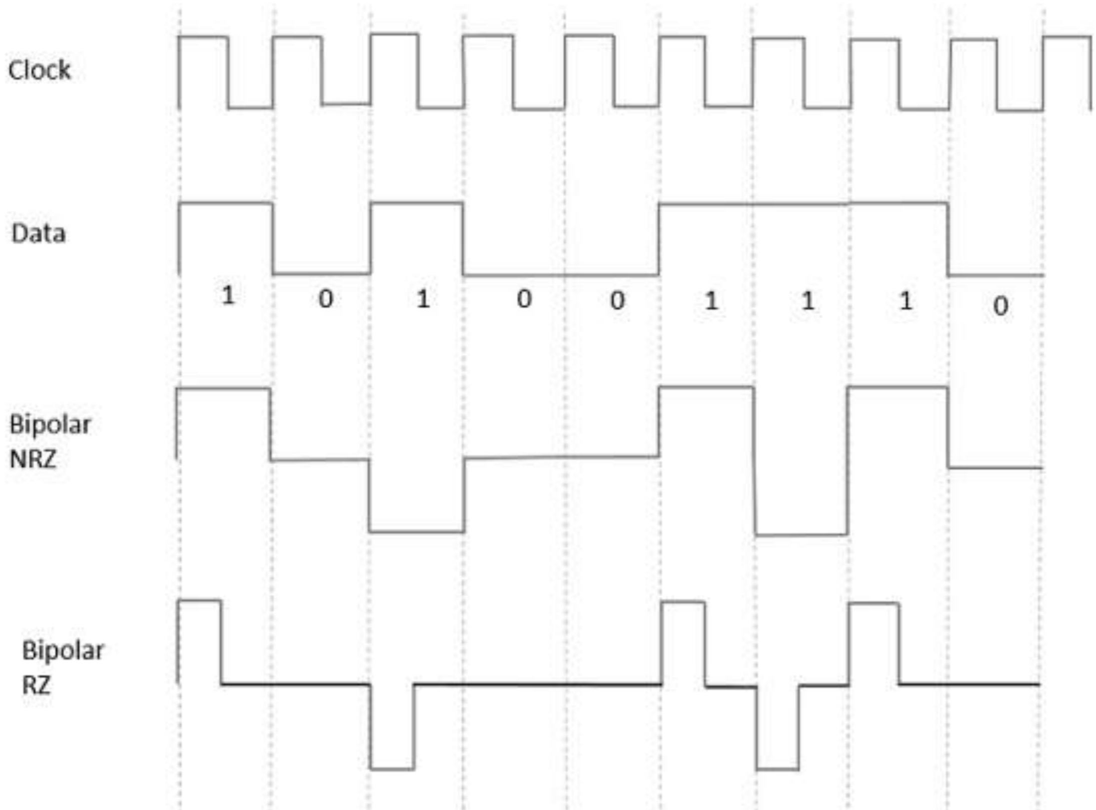
This is an encoding technique which has three voltage levels namely **+, -** and **0**. Such a signal is called as **duo-binary signal**.

An example of this type is **Alternate Mark Inversion (AMI)**. For a **1**, the voltage level gets a transition from + to – or from – to +, having alternate **1s**to be of equal polarity. A **0** will have a zero voltage level.

Even in this method, we have two types.

- Bipolar NRZ
- Bipolar RZ

From the models so far discussed, we have learnt the difference between NRZ and RZ. It just goes in the same way here too. The following figure clearly depicts this.



The above figure has both the Bipolar NRZ and RZ waveforms. The pulse duration and symbol bit duration are equal in NRZ type, while the pulse duration is half of the symbol bit duration in RZ type.

Advantages

Following are the advantages −

- It is simple.
- No low-frequency components are present.
- Occupies low bandwidth than unipolar and polar NRZ schemes.
- This technique is suitable for transmission over AC coupled lines, as signal drooping doesn't occur here.
- A single error detection capability is present in this.

Disadvantages

Following are the disadvantages −

- No clock is present.
- Long strings of data causes loss of synchronization.

Scrambling technique BZ8S, HDB3

Scrambling Schemes: *Extension of Bipolar AMI. Used in case of long distance applications. Goals:*

- No dc component
- No long sequences of 0-level line signal
- No increase in bandwidth
- Error detection capability
- Examples: B8ZS, HBD3

Bipolar with 8-zero substitution (B8ZS): *The limitation of bipolar AMI is overcome in B8ZS, which is used in North America. A sequence of eight zero's is replaced by the following encoding*

A sequence of eight 0's is replaced by 000+ - 0 + -, if the previous pulse was positive.
A sequence of eight 0's is replaced by 000 - + 0 + -, if the previous pulse was negative

High Density Bipolar-3 Zeros: *Another alternative, which is used in Europe and Japan is HDB3. It replaces a sequence of 4 zeros by a code as per the rule given in the following table. The encoded signals are shown in Fig. 2.4.15.*

HDB3 substitution rule		
Polarity of the Preceding pulse	Number of bipolar pulses (ones) since last substitution	
	odd	even
—	000 —	+ 00 +
+	000 +	— 00 —

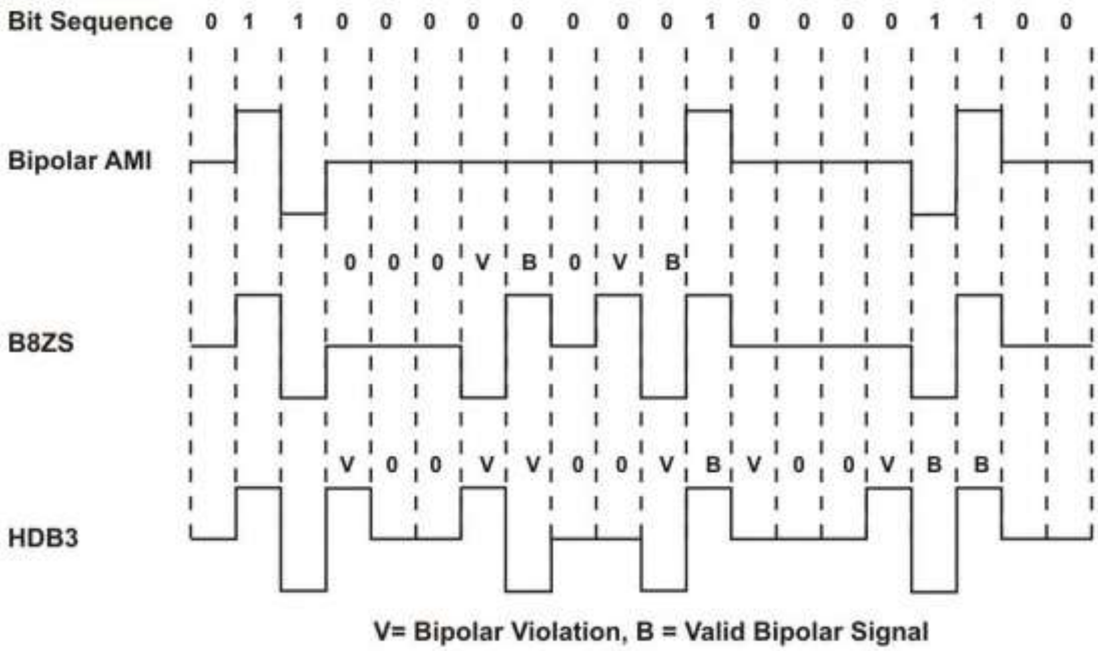


Figure 2.4.15 *B8ZS and HDB3 encoding techniques*

Computation of power spectral densities of various line encoding formats

Encoding is the process of converting the data or a given sequence of characters, symbols, alphabets etc., into a specified format, for the secured transmission of data. **Decoding** is the reverse process of encoding which is to extract the information from the converted format.

Data Encoding

Encoding is the process of using various patterns of voltage or current levels to represent **1s** and **0s** of the digital signals on the transmission link.

The common types of line encoding are Unipolar, Polar, Bipolar, and Manchester.

Encoding Techniques

The data encoding technique is divided into the following types, depending upon the type of data conversion.

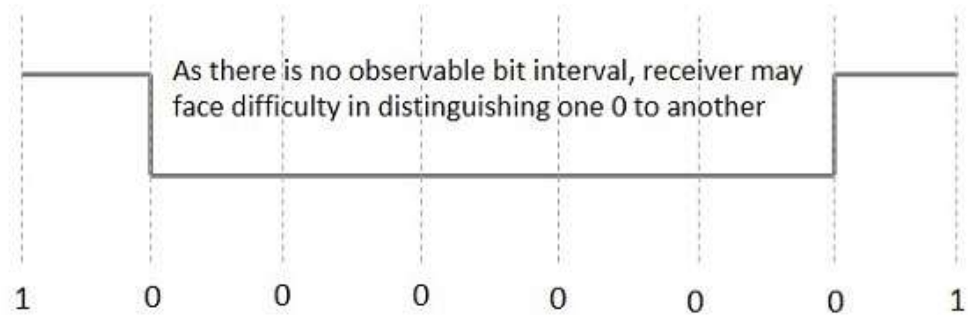
- **Analog data to Analog signals** − The modulation techniques such as Amplitude Modulation, Frequency Modulation and Phase Modulation of analog signals, fall under this category.
- **Analog data to Digital signals** − This process can be termed as digitization, which is done by Pulse Code Modulation (PCM). Hence, it is nothing but digital modulation. As we have already discussed, sampling and quantization are the important factors in this. Delta Modulation gives a better output than PCM.
- **Digital data to Analog signals** − The modulation techniques such as Amplitude Shift Keying (ASK), Frequency Shift Keying (FSK), Phase Shift Keying (PSK), etc., fall under this category. These will be discussed in subsequent chapters.

- **Digital data to Digital signals** − These are in this section. There are several ways to map digital data to digital signals. Some of them are −

Non Return to Zero (NRZ)

NRZ Codes has **1** for High voltage level and **0** for Low voltage level. The main behavior of NRZ codes is that the voltage level remains constant during bit interval. The end or start of a bit will not be indicated and it will maintain the same voltage state, if the value of the previous bit and the value of the present bit are same.

The following figure explains the concept of NRZ coding.



NRZ Coding

If the above example is considered, as there is a long sequence of constant voltage level and the clock synchronization may be lost due to the absence of bit interval, it becomes difficult for the receiver to differentiate between 0 and 1.

There are two variations in NRZ namely −

NRZ - L (NRZ – LEVEL)

There is a change in the polarity of the signal, only when the incoming signal changes from 1 to 0 or from 0 to 1. It is the same as NRZ, however, the first bit of the input signal should have a change of polarity.

NRZ - I (NRZ – INVERTED)

If a **1** occurs at the incoming signal, then there occurs a transition at the beginning of the bit interval. For a **0** at the incoming signal, there is no transition at the beginning of the bit interval.

NRZ codes has a **disadvantage** that the synchronization of the transmitter clock with the receiver clock gets completely disturbed, when there is a string of **1s** and **0s**. Hence, a separate clock line needs to be provided.

Bi-phase Encoding

The signal level is checked twice for every bit time, both initially and in the middle. Hence, the clock rate is double the data transfer rate and thus the modulation rate is also doubled. The clock is taken from the signal itself. The bandwidth required for this coding is greater.

There are two types of Bi-phase Encoding.

- Bi-phase Manchester
- Differential Manchester

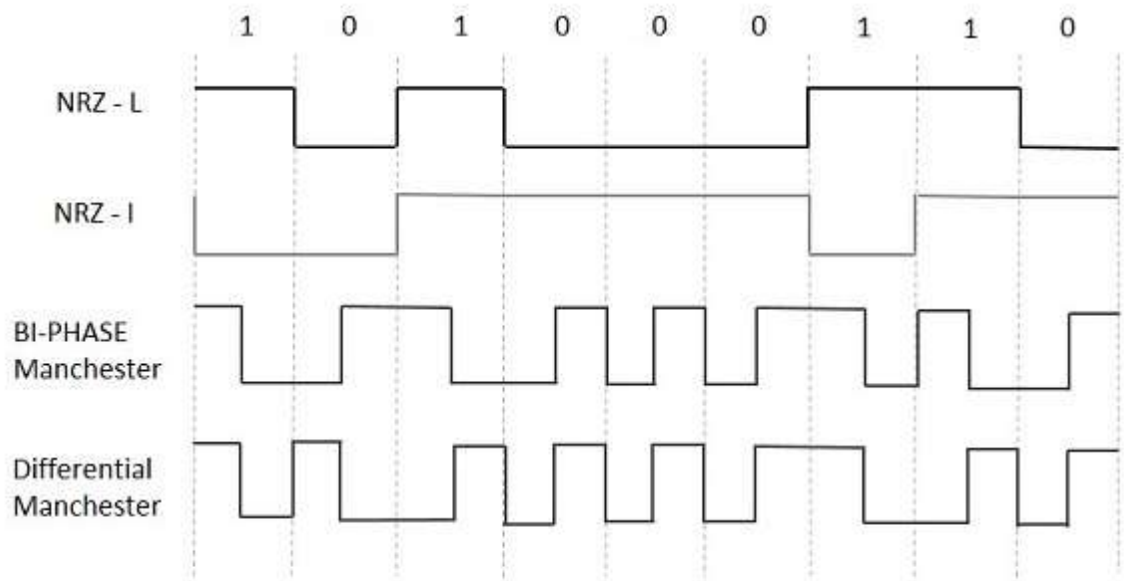
Bi-phase Manchester

In this type of coding, the transition is done at the middle of the bit-interval. The transition for the resultant pulse is from High to Low in the middle of the interval, for the input bit 1. While the transition is from Low to High for the input bit **0**.

Differential Manchester

In this type of coding, there always occurs a transition in the middle of the bit interval. If there occurs a transition at the beginning of the bit interval, then the input bit is **0**. If no transition occurs at the beginning of the bit interval, then the input bit is **1**.

The following figure illustrates the waveforms of NRZ-L, NRZ-I, Bi-phase Manchester and Differential Manchester coding for different digital inputs.



Block Coding

Among the types of block coding, the famous ones are 4B/5B encoding and 8B/6T encoding. The number of bits is processed in different manners, in both of these processes.

4B/5B Encoding

In Manchester encoding, to send the data, the clocks with double speed is required rather than NRZ coding. Here, as the name implies, 4 bits of code is mapped with 5 bits, with a minimum number of **1** bits in the group.

The clock synchronization problem in NRZ-I encoding is avoided by assigning an equivalent word of 5 bits in the place of each block of 4 consecutive bits. These 5-bit words are predetermined in a dictionary.

The basic idea of selecting a 5-bit code is that, it should have **one leading 0**and it should have **no more than two trailing 0s**. Hence, these words are chosen such that two transactions take place per block of bits.

8B/6T Encoding

We have used two voltage levels to send a single bit over a single signal. But if we use more than 3 voltage levels, we can send more bits per signal.

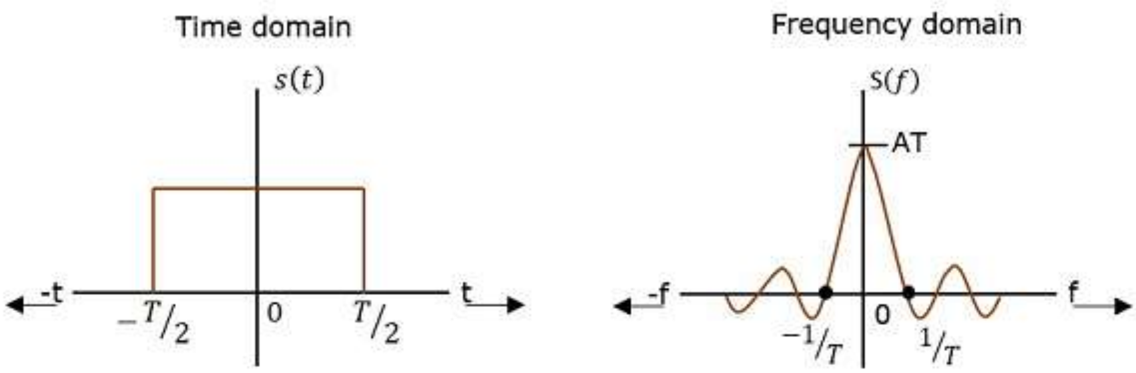
For example, if 6 voltage levels are used to represent 8 bits on a single signal, then such encoding is termed as 8B/6T encoding. Hence in this method, we have as many as 729 (3^6) combinations for signal and 256 (2^8) combinations for bits.

These are the techniques mostly used for converting digital data into digital signals by compressing or coding them for reliable transmission of data.

Power Spectral Density

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

PSD is the Fourier Transform of Auto-Correlation (Similarity between observations). It is in the form of a rectangular pulse.



$$s(t) = A \quad |t| < T/2$$
$$= 0 \quad |t| > T/2$$

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

PSD Derivation

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

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

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

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

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

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

Getting to know that $R_n = R_{-n}$ for real signals, we have

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

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

$$s_y(w) = | F(w) |^2 S_x(w)$$

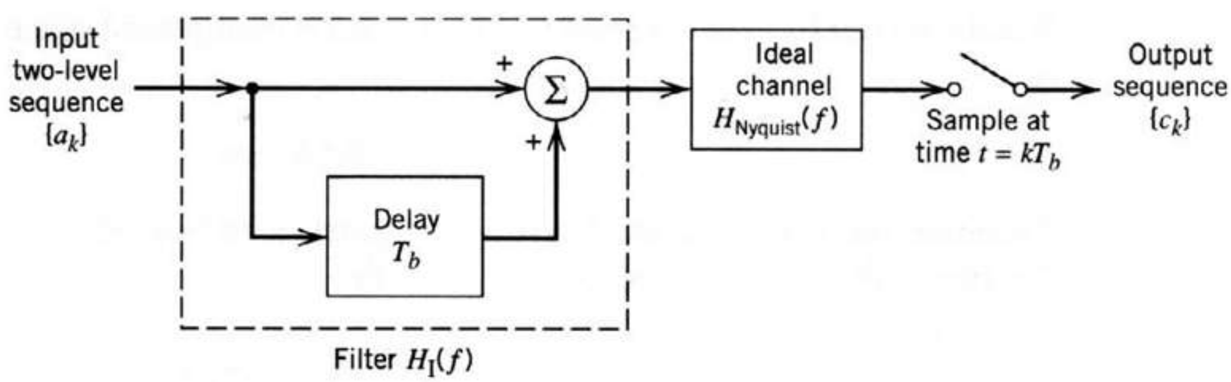
$$= \frac{|F(w)|^2}{T} (\sum_{n=-\infty}^{\infty} R_n e^{-jnwT_b})$$

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

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

Duobinary signaling

Duobinary



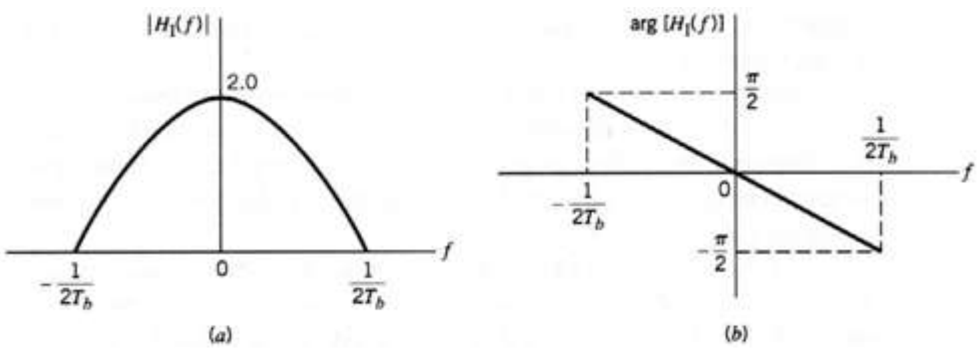
$$
\begin{aligned}
H_I(f) &= H_{\text{Nyquist}}(f)[1 + \exp(-j2\pi f T_b)] \\
&= H_{\text{Nyquist}}(f)[\exp(j\pi f T_b) + \exp(-j\pi f T_b)] \exp(-j\pi f T_b) \\
&= 2H_{\text{Nyquist}}(f) \cos(\pi f T_b) \exp(-j\pi f T_b)
\end{aligned}
$$

$$
H_{\text{Nyquist}}(f) = \begin{cases} 1, & |f| \le 1/2T_b \\ 0, & \text{otherwise} \end{cases}
$$

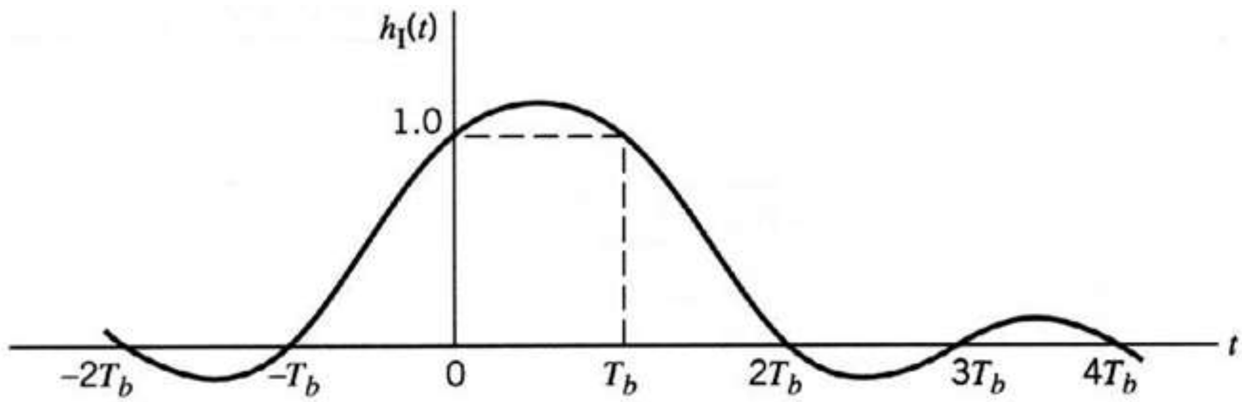
Duobinary signal and Nyguist Criteria:

- Nyguist second criteria: but twice the bandwidth

$$
H_I(f) = \begin{cases} 2\cos(\pi f T_b)\exp(-j\pi f T_b), & |f| \le 1/2T_b \\ 0, & \text{otherwise} \end{cases}
$$



$$
\begin{aligned}
h_I(t) &= \frac{\sin(\pi t/T_b)}{\pi t/T_b} + \frac{\sin[\pi(t - T_b)/T_b]}{\pi(t - T_b)/T_b} \\
&= \frac{\sin(\pi t/T_b)}{\pi t/T_b} - \frac{\sin(\pi t/T_b)}{\pi(t - T_b)/T_b} \\
&= \frac{T_b^2 \sin(\pi t/T_b)}{\pi t(T_b - t)}
\end{aligned}
$$



Differential Coding

- The response of a pulse is spread over more than one signaling interval.
- The response is partial in any signaling interval.
- Detection
- Major drawback: error propagation.
- To avoid error propagation, need deferential coding (precoding).

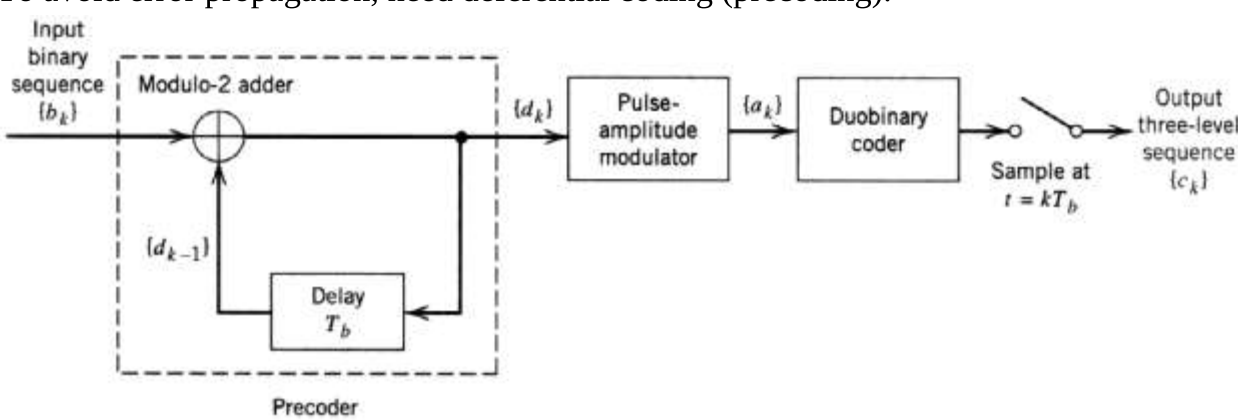


FIGURE 4.14 A precoded duobinary scheme; details of the duobinary coder are given in Figure 4.11.

Modified duo binary signaling

- Modified duobinary signaling
- In duobinary signaling, H(f) is nonzero at the origin.
- We can correct this deficiency by using the class IV partial response.

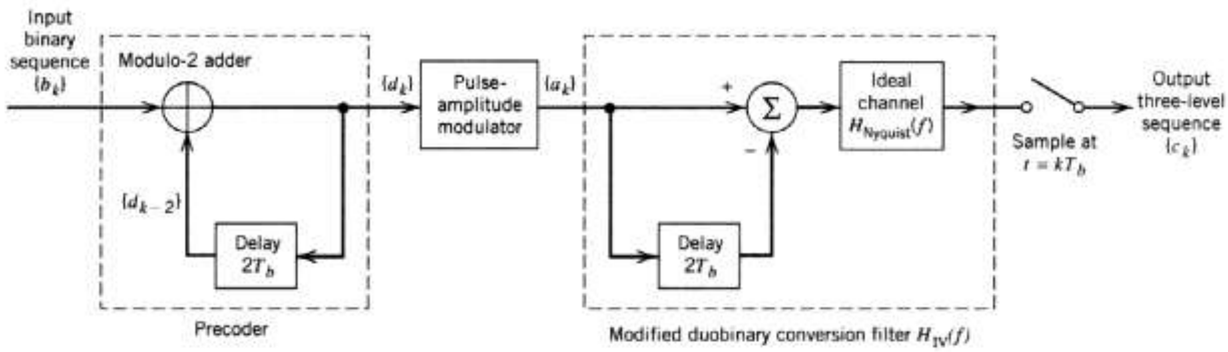


FIGURE 4.16 Modified duobinary signaling scheme.

Spectrum

$$H_{IV}(f) = H_{Nyquist}(f)[1 - \exp(-j4\pi f T_b)]$$
$$= 2jH_{Nyquist}(f)\sin(2\pi f T_b)\exp(-j2\pi f T_b)$$

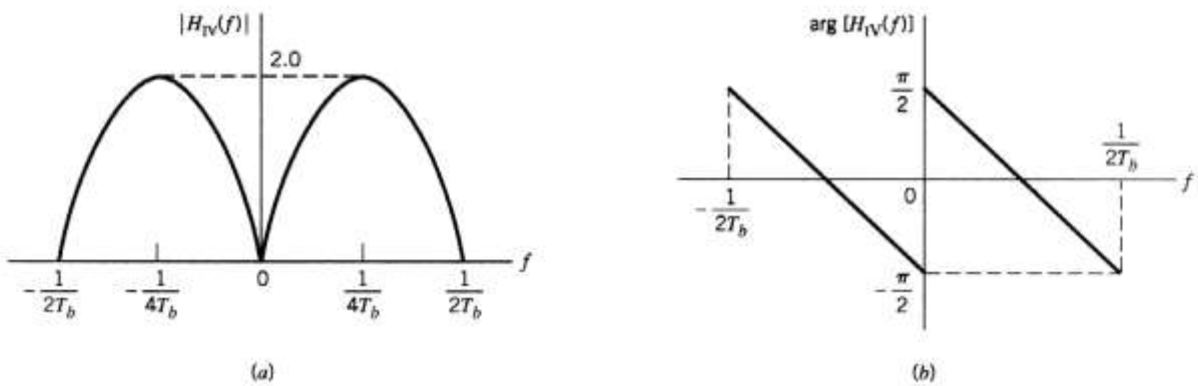


FIGURE 4.17 Frequency response of the modified duobinary conversion filter. (*a*) Magnitude response. (*b*) Phase response.

- Time Sequence: interpretation of receiving 2, 0, and -2?

$$h_{IV}(t) = \frac{\sin(\pi t/T_b)}{\pi t/T_b} - \frac{\sin[\pi(t - 2T_b)/T_b)]}{\pi(t - 2T_b)/T_b}$$

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

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

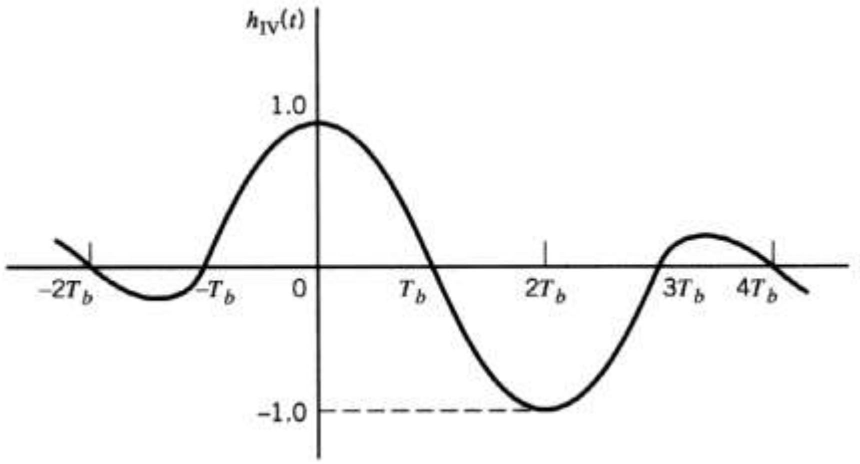
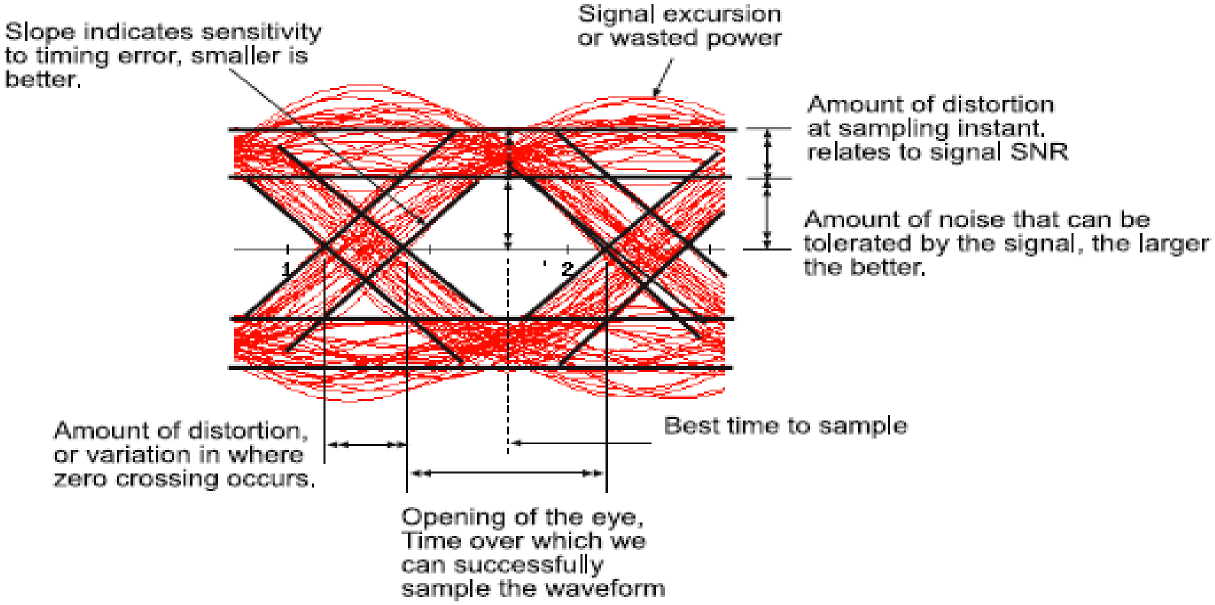


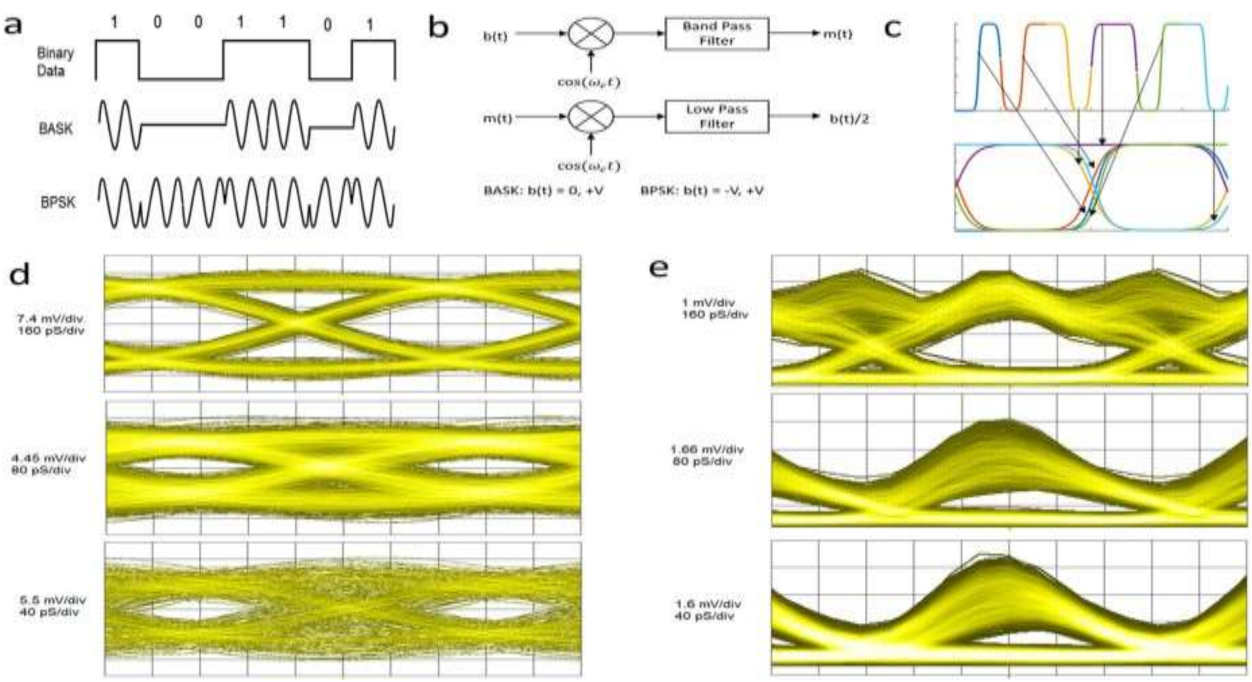
FIGURE 4.18 Impulse response of the modified duobinary conversion filter.

Eye Diagram

- Eye diagram is a means of evaluating the quality of a received "digital waveform"
- By quality is meant the ability to correctly recover symbols and timing
- The received signal could be examined at the input to a digital receiver or at some stage within the receiver before the decision stage
- Eye diagrams reveal the impact of ISI and noise
- Two major issues are 1) sample value variation, and 2) jitter and sensitivity of sampling instant
- Eye diagram reveals issues of both
- Eye diagram can also give an estimate of achievable BER
- Check eye diagrams at the end of class for participation

Interpretation of Eye Diagram





Crosstalk

- It is an unwanted coupling between signal paths. It can occur by electrical coupling between nearby twisted pairs.
- Typically, crosstalk is of the same order of magnitude as, or less than, thermal noise.

\

UNIT – IV
INFORMATION THEORY AND SOURCE CODING

Information theory:

- It is a study of Communication Engineering plus Maths.
- A Communication Engineer has to Fight with
- Limited Power
- Inevitable Background Noise
- Limited Bandwidth

Information Theory deals with

- The Measure of Source Information
- The Information Capacity of the channel
- Coding

If The rate of Information from a source does not exceed the capacity of the Channel, then there exist a Coding Scheme such that Information can be transmitted over the Communication Channel with arbitrary small amount of errors despite the presence of Noise

- This is utilized to determine the information rate of discrete Sources

Consider two Messages

A Dog Bites a Man → High probability → Less information

A Man Bites a Dog → Less probability → High Information

So we can say that

Information α (1/Probability of Occurrence)

Also we can state the three law from Intution

Rule 1: Information $I(m_k)$ approaches to 0 as P_k approaches infinity.

Mathematically $I(m_k) = 0$ as $P_k \rightarrow 1$

e.g. Sun Rises in East

Rule 2: The Information Content $I(m_k)$ must be Non Negative contity.

It may be zero

Mathematically $I(m_k) >= 0$ as $0 <= P_k <=1$

e.g. Sun Rises in West.

Rule 3: The Information Content of message having Higher probability is less than the Information Content of Message having Lower probability

Mathematically $I(m_k) > I(m_j)$

Also we can state for the Sum of two messages that the information content in the two combined messages is same as the sum of information content of each message Provided the occurrence is mutually independent.

e.g. There will be Sunny weather Today.

There will be Cloudy weather Tomorrow

Mathematically

$$I\,(m_k \text{ and } m_j) = I(m_k\, m_j)$$

$$= I(m_k)+I(m_j)$$

- So Question is which function that we can use that measure the Information?

Information = F(1/Probability)

Requirement that function must satisfy

1. Its output must be non negative Quantity.
2. Minimum Value is 0.
3. It Should make Product into summation.

Information $I(m_k) = \text{Log}_b (1/P_k)$

Here b may be 2, e or 10

If b = 2 then unit is bits

b = e then unit is nats

b = 10 then unit is decit

- It is necessary to define the information content of the particular symbol as communication channel deals with symbol.
- Here we make following assumption…..

1. The Source is stationery, so Probability remains constant with time.
2. The Successive symbols are statistically independent and come out at avg rate of r symbols per second
3. Suppose a source emits M Possible symbols s1, s2, …..SM having Probability of occurrence

p1,p2,…….pm

For a long message having symbols N (>>M)

s1 will occur P1N times, like also

s2 will occur P2N times so on…….

- Since s1 occurs p1N times so information Contribution by s1 is p1Nlog(1/p1).
- Similarly information Contribution by s2 is p2Nlog(1/p2). And So on…….
- Hence the Total Information Content is
- And Average Information is obtained by
- It means that In long message we can expect H bit of information per symbol. Another name of H is entropy.
- Information Rate = Total Information/ time taken
- Here Time Taken
- n bits are transmitted with r symbols per second. Total Information is nH.
- Information rate

H satisfies following Equation

Maximum H Will occur when all the message having equal Probability.

Hence H also shows the uncertainty that which of the symbol will occur.

As H approaches to its maximum Value we can't determine which message will occur.

Consider a system Transmit only 2 Messages having equal probability of occurrence 0.5. at that Time H=1

And at every instant we cant say which one of the two message will occur.

So what would happen for more then two symbol source?

- Let's Consider a Binary Source,

means M=2

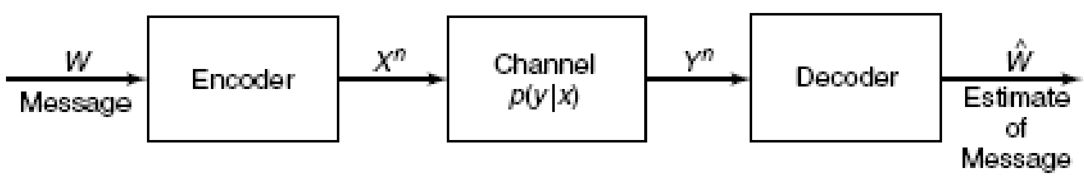
Let the two symbols occur at the probability p and 1-p Respectively.

Where o < p < 1.

- Joint probability of X and Y, $p(X, Y)$, is probability that X and Y simultaneously assume particular values
  - If X, Y independent, $p(X, Y) = p(X)p(Y)$
- Roll die, toss coin
  - $p(X = 3, Y = \text{heads}) = p(X = 3)p(Y = \text{heads}) = 1/6 \times 1/2 = 1/12$
- Conditional probability of X given Y, $p(X|Y)$, is probability that X takes on a particular value given Y has a particular value
- Continuing example …
  - $p(Y=7|X=1) = 1/6$
  - $p(Y=7|X=3) = 1/6$

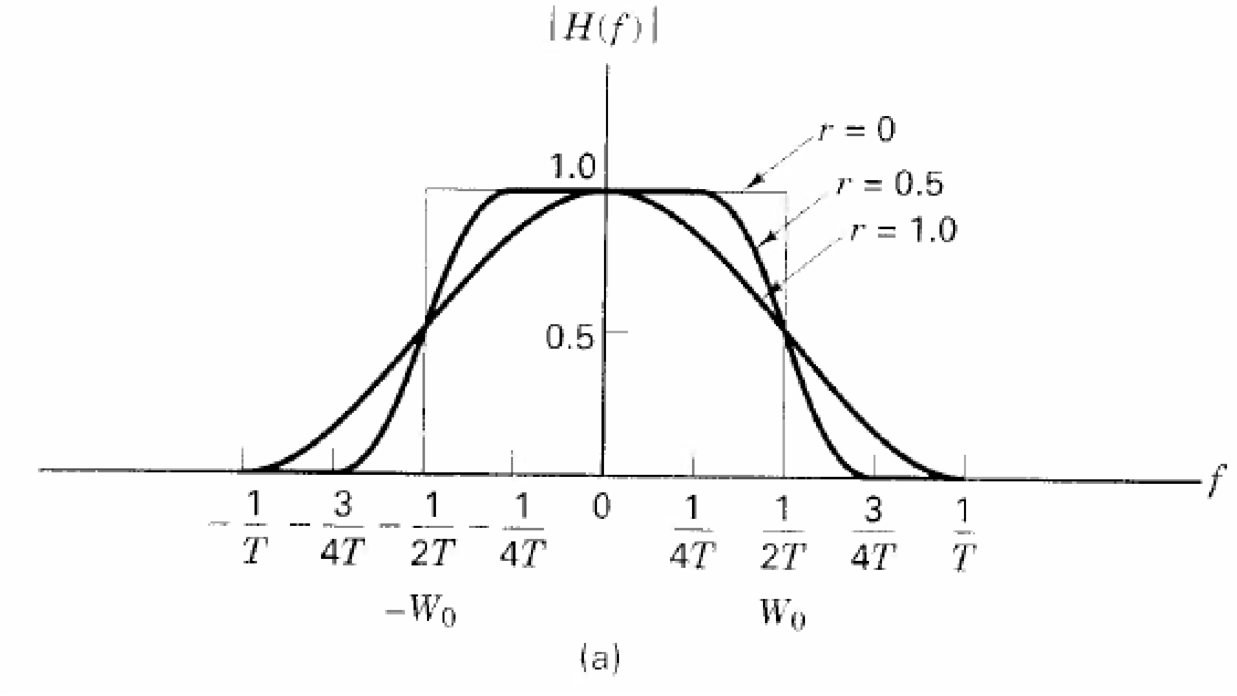
Channel capacity

The communication between A and B is a consequence of a physical act performed by A to induce the desired state in B. This transfer of information is subject to noise. The communication is successful if the transmitter A and the receiver B agree on what was sent. In this part we define the channel capacity as the logarithm of the number of distinguishable signals that can be sent through the channel.Source symbols from some finite alphabet are mapped into some sequence of channel symbols, which then produces the output sequence of the channel. The output sequence is random but has a distribution that depends on the input sequence. Each of the possible input sequences induces a probability distribution on the output sequences. We show that we can choose a “no confusable” subset of input sequences so that with high probability there is only one highly likely input that could have caused the particular output. We can transmit a message with very low probability of error and reconstruct the source message at the output. The maximum rate at which this can be done is called the capacity of the channel.

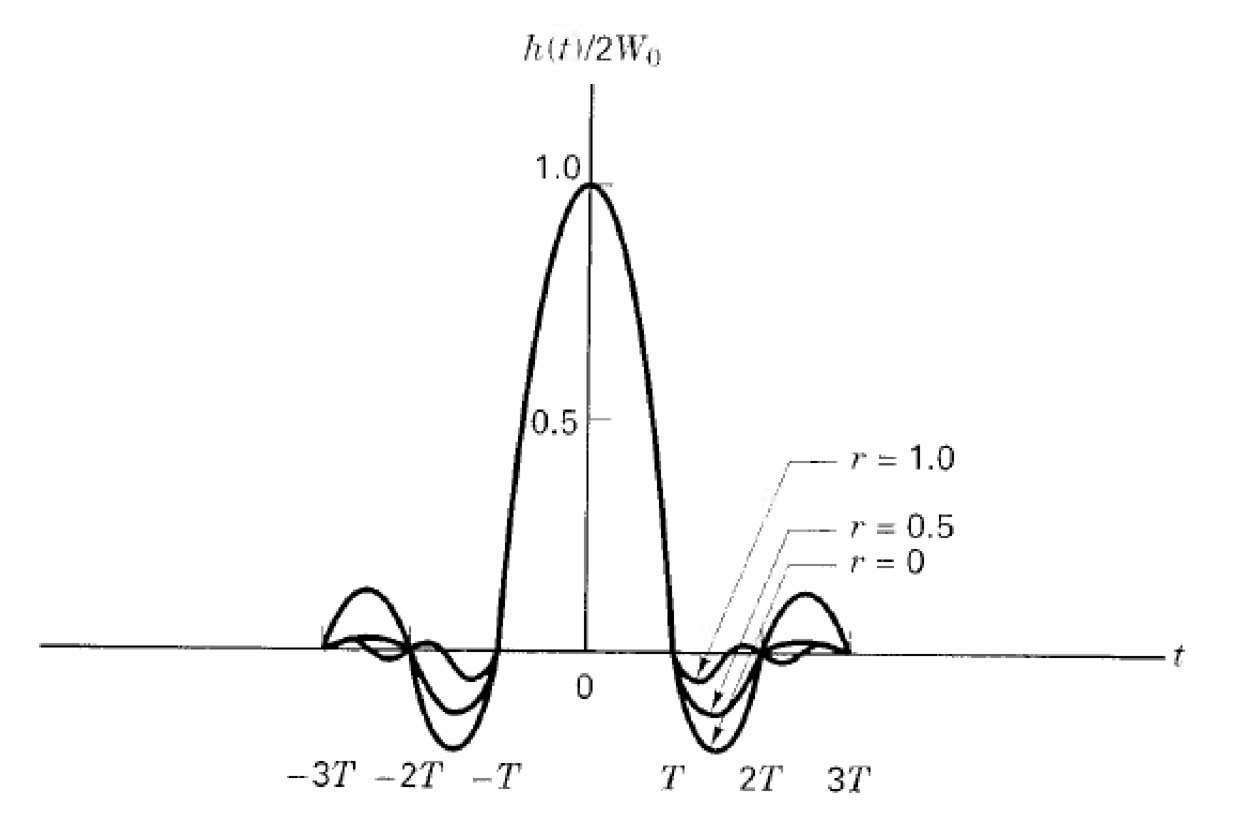


Definition We define a discrete channel to be a system consisting of an input alphabet X and output alphabet Y and a probability transition matrix p(y|x) that expresses the probability of observing the output symbol y given that we send the symbol x.The channel is said to be memoryless if the probability distribution of the output depends only on the input at that time and is conditionally independent of previous channel inputs or outputs. Definition We define the “information” channel capacity of a discrete memory less channel as where the maximum is taken over all possible input distributions p(x).This means: the capacity is the maximum entropy of Y, reduced by the contribution of information given by Y. We shall soon give an operational definition of channel capacity as the highest rate in bits per channel use at which information can be sent with arbitrarily low probability of error.Shannon’s second theorem establishes that the information channel capacity is equal to the operational channel capacity.There is a duality between the problems of data compression and data transmission. During compression, we remove all the redundancy in the data to form the most compressed version possible. During data transmission, we add redundancy in a controlled fashion to combat errors in the channel.

Raised Cosine Filter Transfer Functionin the *f* domain:



Raised Cosine Filter Impulse Response (time domain):



Channel capacity:

The analog of the binary symmetric channel in which some bits are lost (rather than corrupted) is the binary erasure channel. In this channel, a fraction α of the bits are erased. The receiver knows which bits have been erased. The binary erasure channel has two inputs and three outputs

We calculate the capacity of the binary erasure channel as follows:

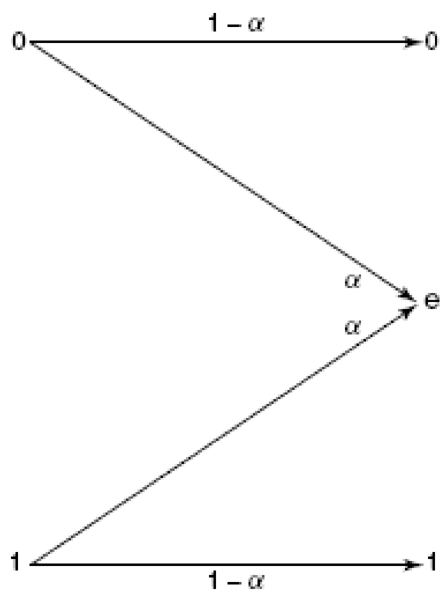
$$
\begin{aligned}
C &= \max I(X;Y) \\
&= \max(H(Y) - H(Y \mid X)) \\
&= \max H(Y) - H(\alpha)
\end{aligned}
$$

The first guess for the maximum of H(Y) would be log 3, but we cannot achieve this by any choice of input distribution p(x). Letting E be the event {Y = e}, using the expansion:

Expansion: H(Y)=H(Y,E)=H(E)+H(Y|E)
And letting Pr(X=1)=□ we have:

$$
\begin{aligned}
H(Y) &= H((1-\pi)(1-\alpha), \alpha, \pi(1-\alpha)) \\
&= H(\alpha) + (1-\alpha)H(\pi)
\end{aligned}
$$

$$
\begin{aligned}
C &= \max H(Y) - H(\alpha) \\
&= \max(1-\alpha)H(\pi) + H(\alpha) - H(\alpha) \\
&= \max(1-\alpha)H(\pi) \\
&= 1-\alpha
\end{aligned}
$$



Since a proportion α of the bits are lost in the channel, we can recover (at most) a proportion 1 − α of the bits. Hence the capacity is at most 1 − α. It is not immediately obvious that it is possible to achieve this rate. This will follow from Shannon's second theorem.In many practical channels, the sender receives some feedback from the receiver. If feedback is available for the binary erasure channel, it is very clear what to do: If a bit is lost, retransmit it until it gets through.Since the bits get through with probability 1 −

α, the effective rate of transmission is 1 − α. In this way we are easily able to achieve a capacity of 1 − α with feedback.The capacity of the binary symmetric channel is C = 1 − H(p) bits per transmission, and the capacity of the binary erasure channel is C = 1 − α bits per transmission. Now consider the channel with transition matrix:

$$p(y \mid x) = \begin{bmatrix} 0.3 & 0.2 & 0.5 \\ 0.5 & 0.3 & 0.2 \\ 0.2 & 0.5 & 0.3 \end{bmatrix}$$

Here the entry in the xth row and the yth column denotes the conditional probability p(y|x) that y is received when x is sent.

In this channel, all the rows of the probability transition matrix are permutations of each other and so are the columns. Such a channel is said to be symmetric.

1. C ≥ 0 since I (X; Y) ≥ 0.
2. C ≤ log |X| since C = max I (X; Y) ≤ maxH(X) = log |X|.
3. C ≤ log |Y| for the same reason.
4. I (X; Y) is a continuous function of p(x).
5. I (X; Y) is a concave function of p(x)

Hartley Shannon law;

channel capacity , the tightest upper bound on information rate (excluding error correcting codes) of arbitrarily low bit error rate data that can be sent with a given average signal power S through an additive white Gaussian noise channel of power N, is:

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

C is the channel capacity in bits per second
B is the bandwidth of the channel in hertz
S is the total received signal power over bandwidth, in watts
N is the total noise or interference power over bandwidth, in watts
S/N is the signal-to-noise ratio (SNR) expressed as a linear power ratio (not as logarithmic decibels).

For SNR of 0, 10, 20, 30 dB, one can achieve C/B of 1, 3.46, 6.66, 9.97 bps/Hz, respectively

Example:
Consider the operation of a modem on an ordinary telephone line. The SNR is usually about 1000. The bandwidth is 3.4 KHz. Therefore:

C = 3400 X log2(1 + 1000)
= (3400)(9.97)
≈34 kbps

The channel capacity is

$$C = \omega\ log\left(1 + \frac{S}{N}\right) = \omega\ log\left(1 + \frac{S}{\eta\omega}\right)$$

Now if $S/N = \infty$ the capacity C should be ∞. But it is not. Again, on the other hand C is not ∞ even if $\omega = \infty$. Thus for a fixed signal power in presence of Gaussian noise the channel capacity approaches upper limit called as Shannons limit. This can be analysed as fallows:

$$C = \omega\ log\left(1 + \frac{S}{\eta\omega}\right)$$

$$C = \frac{S}{\eta}\frac{\eta\omega}{S}\ log\left(1 + \frac{S}{\eta\omega}\right)$$

$$C = \frac{S}{\eta}\ log\left(1 + \frac{S}{\eta\omega}\right)^{\eta\omega/S}$$

Source coding: Fixed length:

We can view compression as an encoding method which transforms a sequence of bits/bytes into an artificial format such that the output (in general) takes less space than the input.

Compression is made possible by exploiting redundancy in source data, For example, redundancy found in:

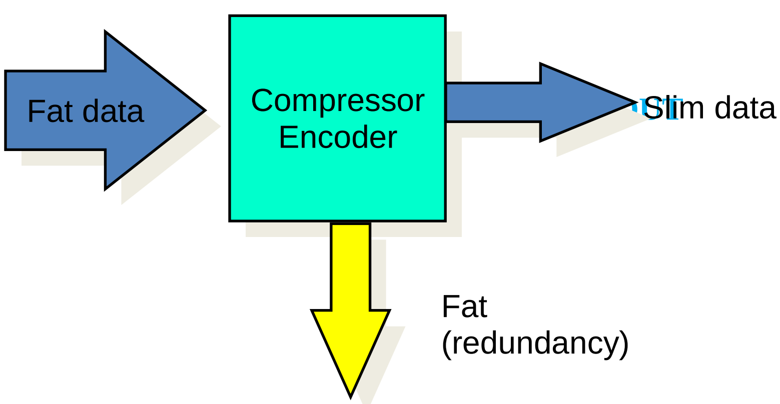
- Character distribution
- Character repetition
- High-usage pattern
- Positional redundancy

We assume a piece of source data that is subject to compression (such as text, image, etc.) is represented by a sequence of symbols. Each symbol is encoded in a computer by a code (or codeword, value), which is a bit string.
Example:
English text: abc (symbols), ASCII (coding)
Chinese text: (symbols), BIG5 (coding)
Image: color (symbols), RGB (coding)

- Some symbols are used more frequently than others.
- In English text, 'e' and space occur most often.
- Fixed-length encoding: use the same number of bits to represent each symbol.
- With fixed-length encoding, to represent n symbols, we need log2 n bits for each code.

Fixed-length encoding may not be the most space-efficient method to digitally represent an object.
Example: "abacabacabacabacabacabacabac"
With fixed-length encoding, we need 2 bits for each of the 3 symbols.
Alternatively, we could use '1' to represent 'a' and '00' and '01' to represent 'b' and 'c', respectively.
How much saving did we achieve?

Technique: whenever symbols occur with different frequencies, we use variable-length encoding.
For your interest:
In 1838, Samuel Morse and Alfred Vail designed the "Morse Code" for telegraph.
Each letter of the alphabet is represented by dots (electric current of short duration) and dashes (3 dots).
Morse & Vail determined the code by counting the letters in each bin of a printer's type box.



Source coding:

Entropy Encoding	Huffman Code, Shannon-Fano Code	
	Run-Length Encoding	
	LZW	
Source Coding	Prediction	DPCM
		DM
	Transformation	DCT
	Vector Quantization	
Hybrid Coding	JPEG	
	MPEG	
	H.261	

For a code C with associated probabilities p(c) the average length is defined as

$$l_a(C) = \sum_{c \in C} p(c) l(c)$$

We say that a prefix code C is optimal if for all prefix codes Theorem (lower bound): For any probability distribution p(S) with associated uniquely decodable code C,

Theorem (upper bound): For any probability distribution p(S) with associated optimal prefix code C,

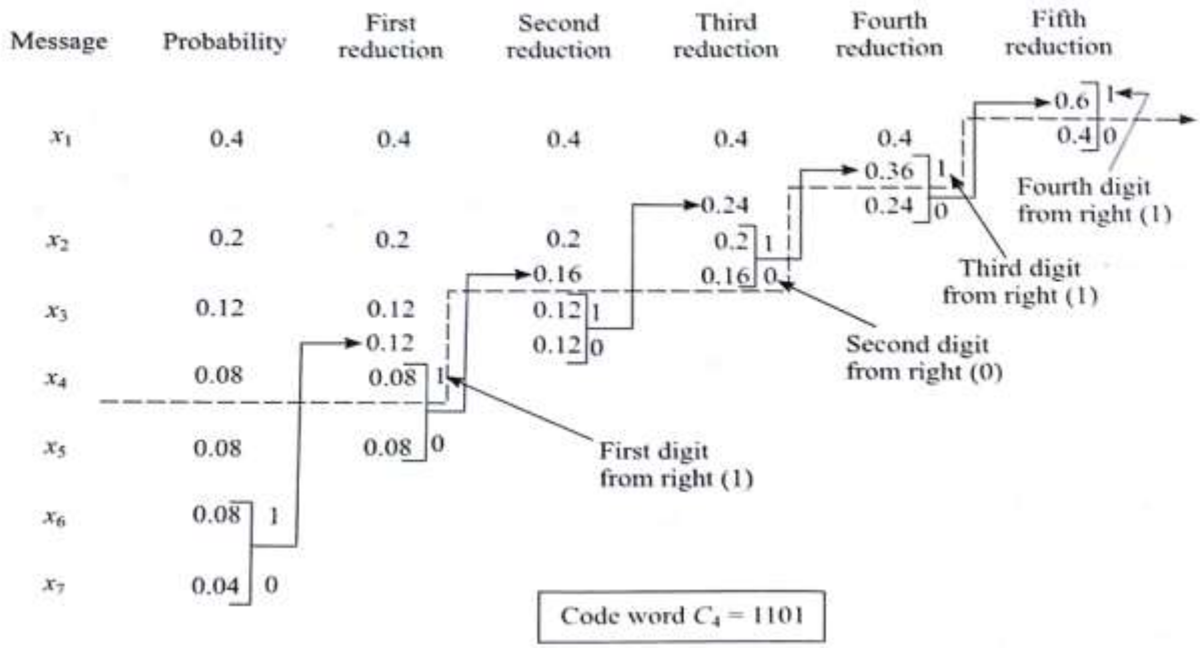
$$H(S) \le l_a(C)$$

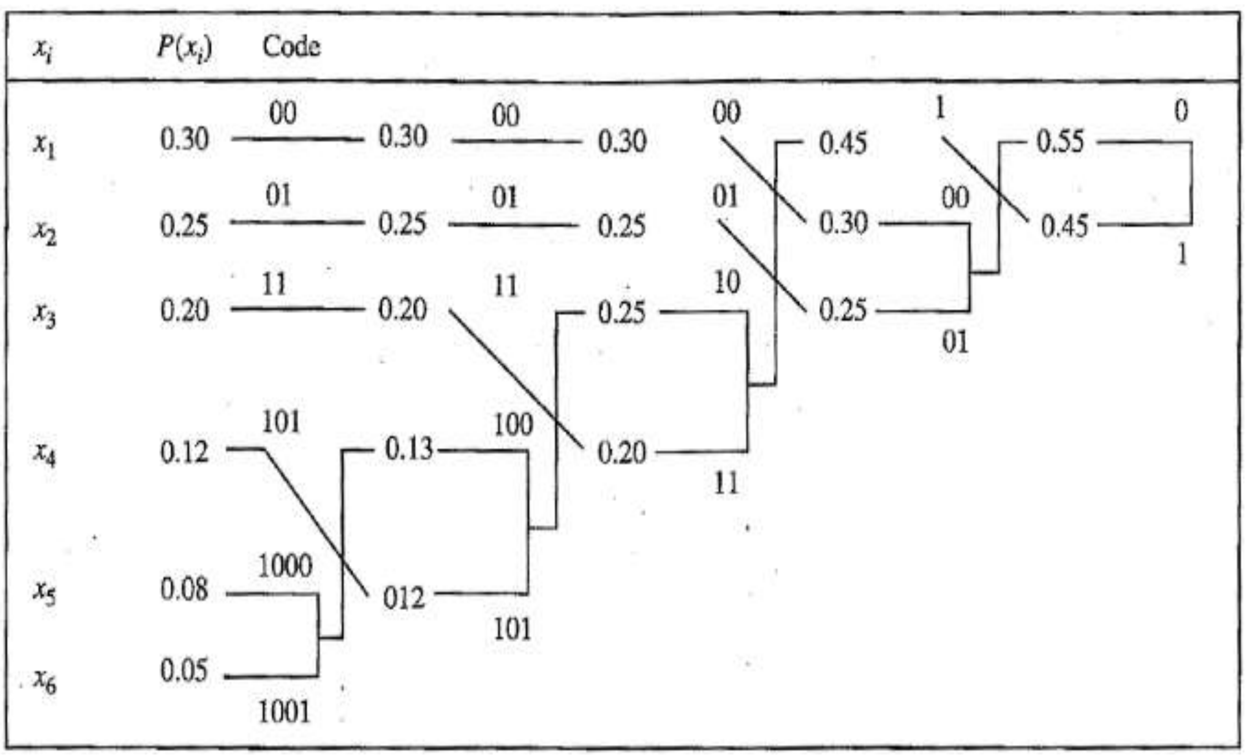
$$l_a(C) \le H(S) + 1$$

$$l_a(C) \le H(S) + 1$$

Huffman coding

Example 11.3.1 Solve Ex. 11.2.2 by the Huffman method.





Computation of power spectral densities:

Input is fed into a channel encoder Produces analog signal with narrow bandwidth Signal is further modulated using sequence of digits Spreading code or spreading sequence Generated by pseudo noise, or pseudo-random number generator Effect of modulation is to increase bandwidth of signal to be transmitted. On receiving end, digit sequence is used to demodulate the spread spectrum signal Signal is fed into a channel decoder to recover data.

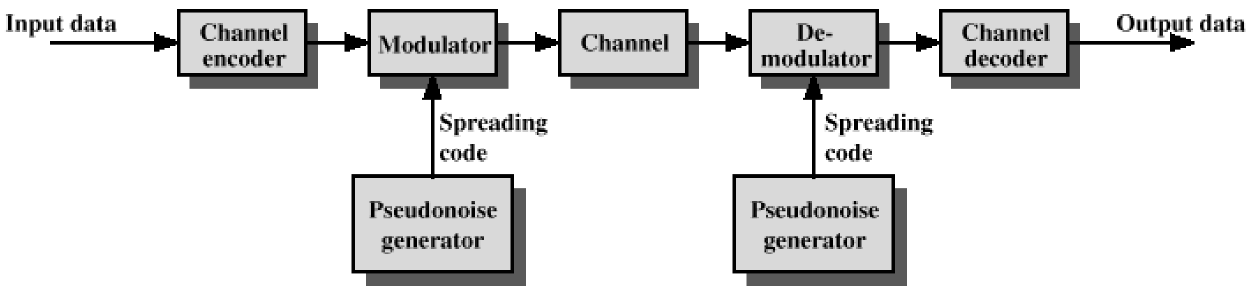


Figure 7.1 General Model of Spread Spectrum Digital Communication System

- What can be gained from apparent waste of spectrum?
- Immunity from various kinds of noise and multipath distortion
- Can be used for hiding and encrypting signals
- Several users can independently use the same higher bandwidth with very little interference

For Arbitrary Pulse Shapes:

- Each bit in original signal is represented by multiple bits in the transmitted signal.
- Spreading code spreads signal across a wider frequency band.
- Spread is in direct proportion to number of bits used.
- One technique combines digital information stream with the spreading code bit stream using exclusive-OR.

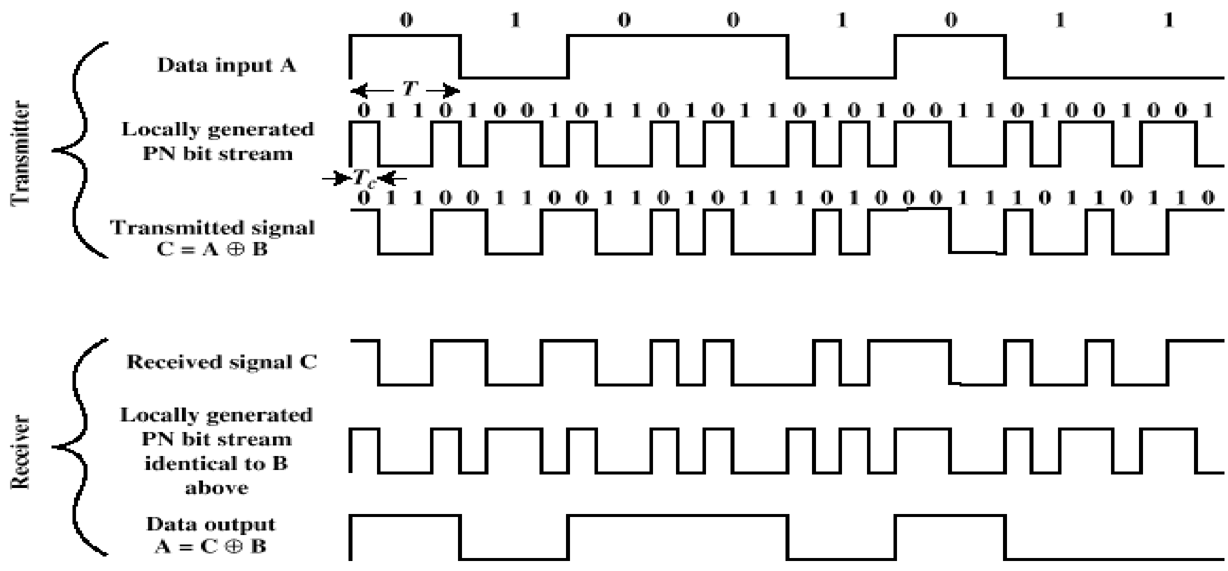


Figure 7.6 Example of Direct Sequence Spread Spectrum

Multiply BPSK signal,

sd(t) = A d(t) cos(2□ fct)

by c(t) [takes values +1, -1] to get

s(t) = A d(t)c(t) cos(2pie fct)

A = amplitude of signal

fc = carrier frequency

d(t) = discrete function [+1, -1]

At receiver, incoming signal multiplied by c(t) ,

Since, c(t) x c(t) = 1, incoming signal is recovered

Code division multiple access using DSSS

- Basic Principles of CDMA
- D = rate of data signal Break
- each bit into k chips
- Chips are a user-specific fixed pattern
- Chip data rate of new channel = kD
- If k=6 and code is a sequence of 1s and -1s
- For a ‘1’ bit, A sends code as chip pattern
- <c1, c2, c3, c4, c5, c6>
- For a ‘0’ bit, A sends complement of code
- <-c1, -c2, -c3, -c4, -c5, -c6>
- Receiver knows sender’s code and performs electronic decode function
- <d1, d2, d3, d4, d5, d6> = received chip pattern
- <c1, c2, c3, c4, c5, c6> = sender’s code
- User A code = <1, –1, –1, 1, –1, 1>

• To send a 1 bit = <1, –1, –1, 1, –1, 1>
• To send a 0 bit = <–1, 1, 1, –1, 1, –1>
• User B code = <1, 1, –1, – 1, 1, 1>
• To send a 1 bit = <1, 1, –1, –1, 1, 1>
• Receiver receiving with A's code
• (A's code) x (received chip pattern)
• User A '1' bit: 6 -> 1
• User A '0' bit: -6 -> 0
• User B '1' bit: 0 -> unwanted signal ignored

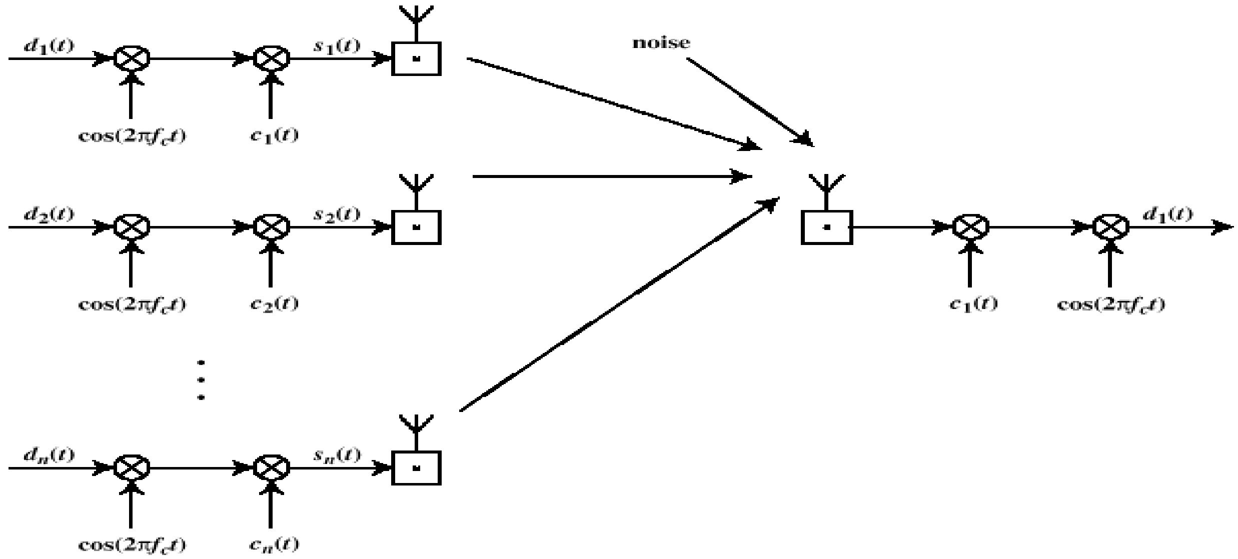


Figure 7.11 CDMA in a DSSS Environment

- Spreading Sequence Categories
  - PN sequences
  - Orthogonal codes
- For FHSS systems
  - PN sequences most common
- For DSSS systems not employing CDMA
  - PN sequences most common
- For DSSS CDMA systems
  - PN sequences
  - Orthogonal codes

Frequency hopping spread spectrum:

- Signal is broadcast over seemingly random series of radio frequencies
  - A number of channels allocated for the FH signal
  - Width of each channel corresponds to bandwidth of input signal

- Signal hops from frequency to frequency at fixed intervals
  - Transmitter operates in one channel at a time
  - Bits are transmitted using some encoding scheme
  - At each successive interval, a new carrier frequency is selected
- Channel sequence dictated by spreading code
- Receiver, hopping between frequencies in synchronization with transmitter, picks up message
- Advantages
  - Eavesdroppers hear only unintelligible blips
  - Attempts to jam signal on one frequency succeed only at knocking out a few bits

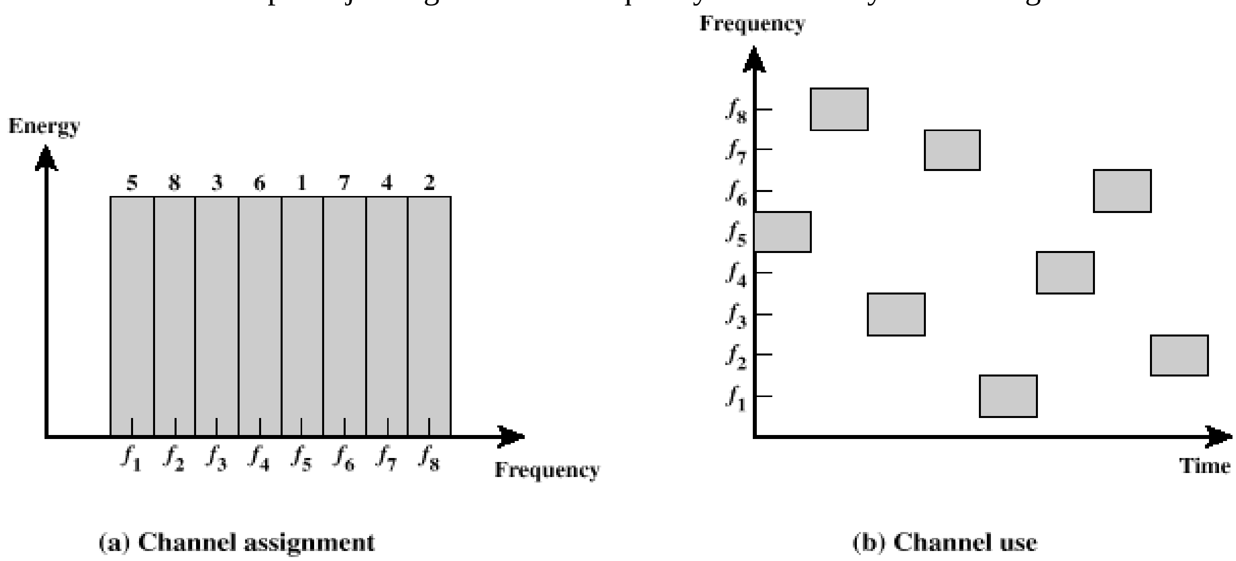


Figure 7.2 Frequency Hopping Example

- MFSK signal is translated to a new frequency every T_c seconds by modulating the MFSK signal with the FHSS carrier signal
- For data rate of R:
  - duration of a bit: $T = 1/R$ seconds
  - duration of signal element: $T_s = LT$ seconds
- $T_c \geq T_s$ - slow-frequency-hop spread spectrum
- $T_c < T_s$ - fast-frequency-hop spread spectrum
- Large number of frequencies used
- Results in a system that is quite resistant to jamming
  - Jammer must jam all frequencies
  - With fixed power, this reduces the jamming power in any one frequency band

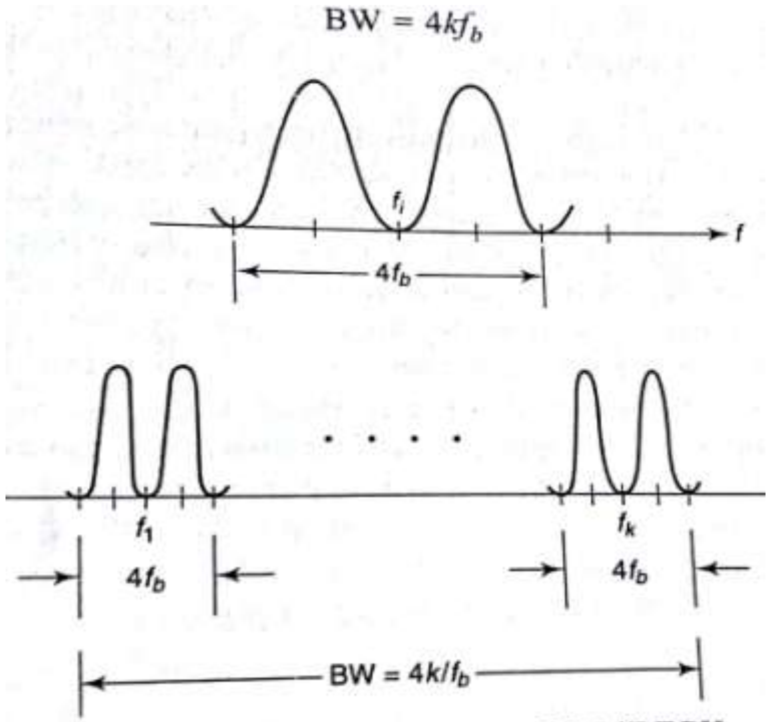


Fig. 15.4 Spectrum of FH/BFSK.

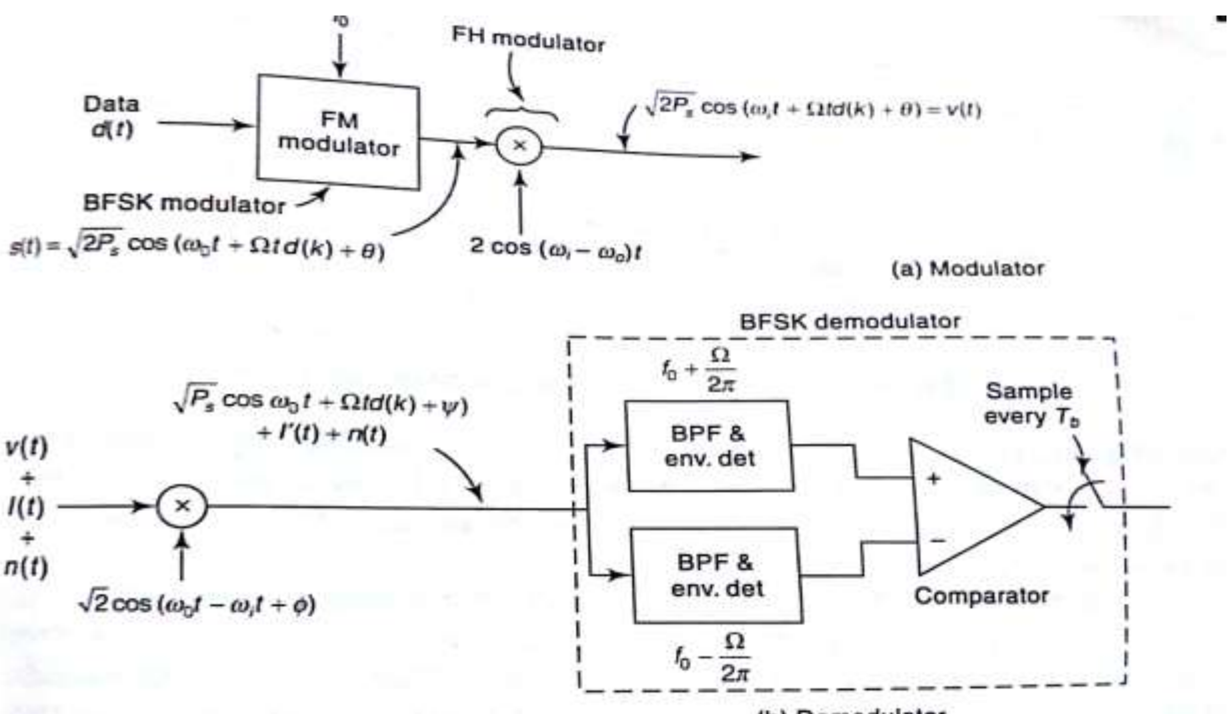
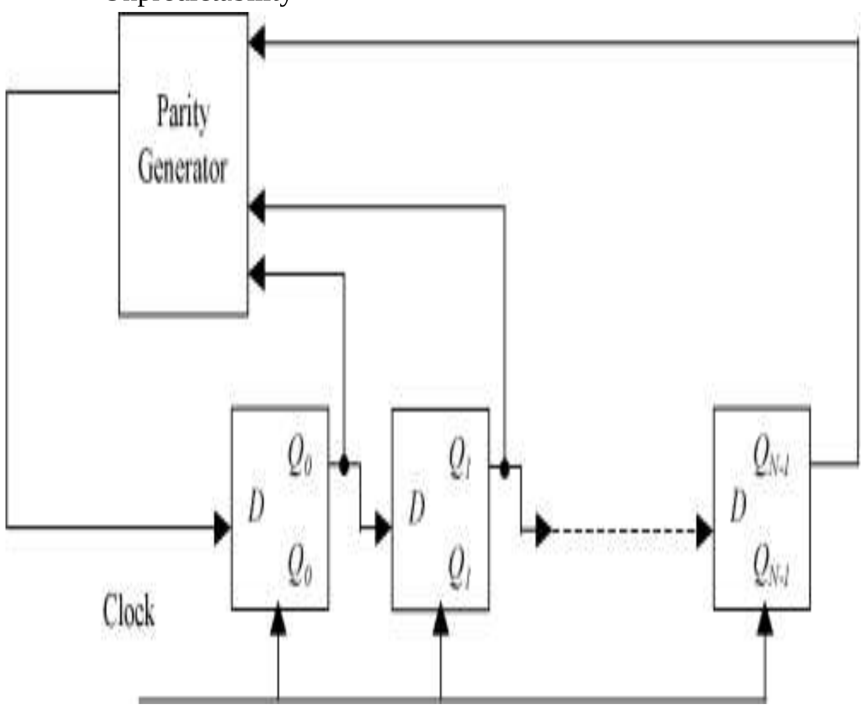


Fig. 15.5 FH/BFSK modulation/demodulation system.

PN- sequence generation and characteristics

- PN generator produces periodic sequence that appears to be random
- PN Sequences
  - Generated by an algorithm using initial seed
  - Sequence isn't statistically random but will pass many test of randomness
  - Sequences referred to as pseudorandom numbers or pseudonoise sequences
  - Unless algorithm and seed are known, the sequence is impractical to predict
- Randomness
  - Uniform distribution
    - Balance property
    - Run property
  - Independence
  - Correlation property
- Unpredictability



Synchronization in spread spectrum systems:

- Non coherent communication such as FSK requires bit synchronizer to allow signal recovery at the receiver. Coherent systems require carrier and phase synchronization to permit modulated received signal to be mixed down to base band.
- IN SS system a third synchronizer is required to allow the regeneration at the receiver of a duplicate of the chipping waveform used at the transmitter. This synchronization is made by two steps.

1. Acquisition of an FH signal (Coarse synchronisation)
2. Tracking of an FH signal(fine synchronization)

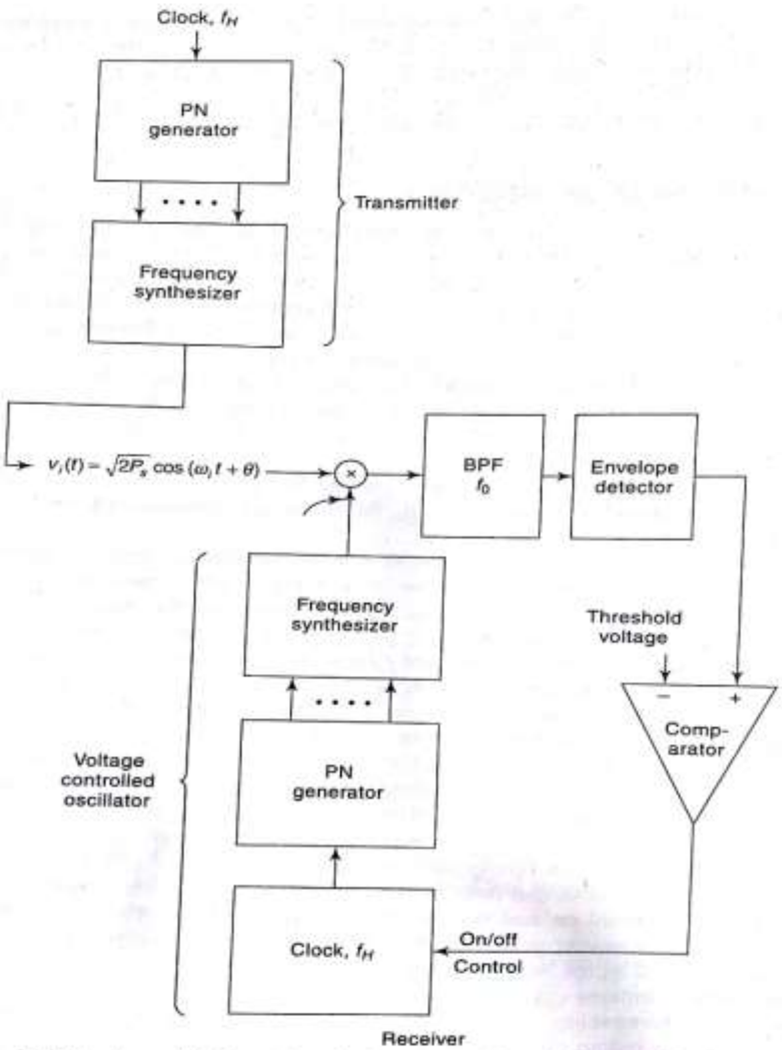


Fig. 15.11 Acquisition circuit (camp-and-wait) for a FH signal.

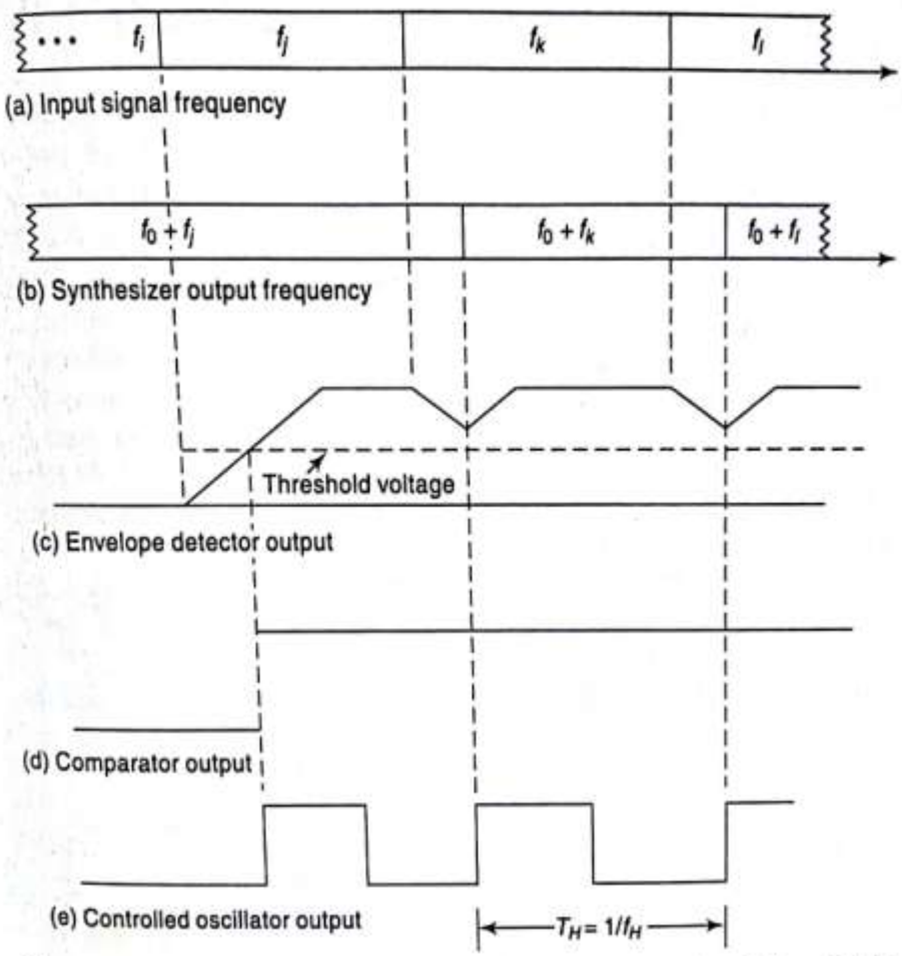


Fig. 15.12 Waveforms for the acquisition circuit of Fig. 15.11.

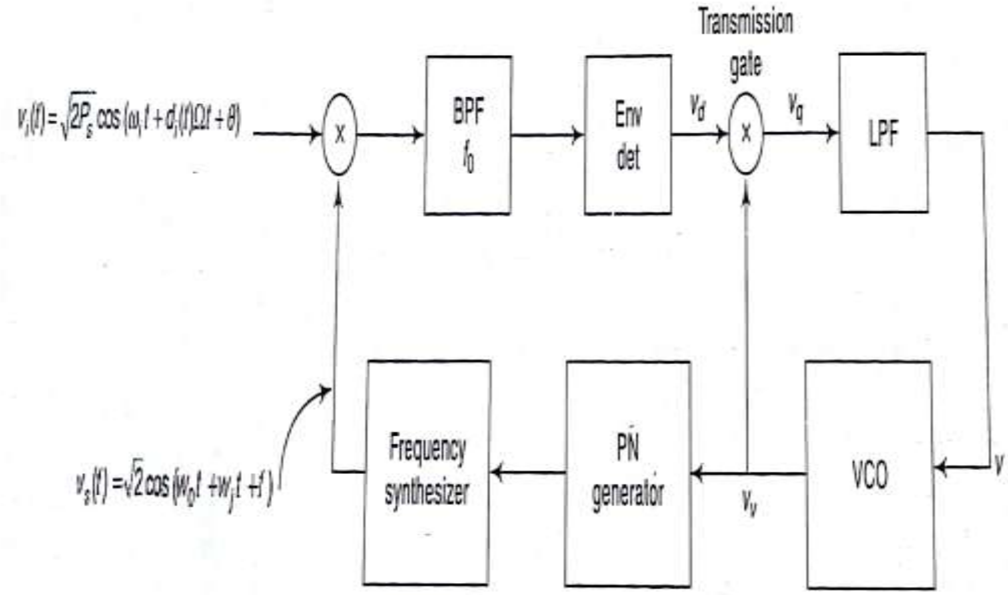
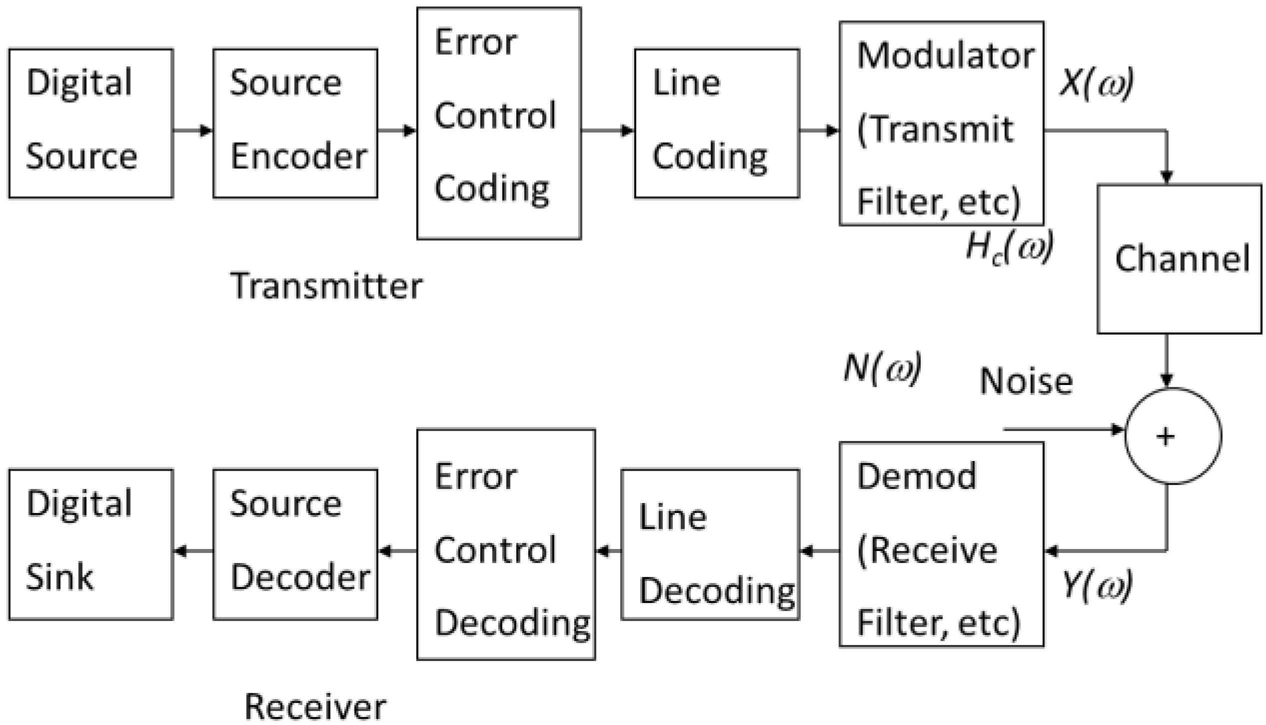


Fig. 15.13 Early-late gate tracking circuit.

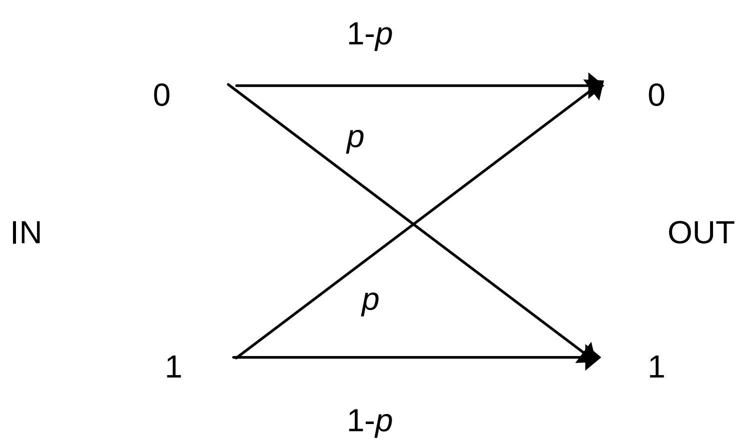
UNIT – V
LINEAR BLOCK CODES AND CONVOLUTION CODES

Linear block codes:

- Error Control Coding (ECC)
- Extra bits are added to the data at the transmitter (redundancy) to permit error detection or correction at the receiver
- Done to prevent the output of erroneous bits despite noise and other imperfections in the channel
- The positions of the error control coding and decoding are shown in the transmission model



- Binary Symmetric Memory less Channel
- Assumes transmitted symbols are binary
- Errors affect '0's and '1's with equal probability (i.e., symmetric)
- Errors occur randomly and are independent from bit to bit (memory less)



p is the probability of bit error or the Bit Error Rate (BER) of the channel

Matrix description of linear block codes:

If there are k data bits, all that is required is to hold k linearly independent codeword, i.e., a set of k codeword's none of which can be produced by linear combinations of 2 or more codeword in the set. The easiest way to find k linearly independent codeword is to choose those which have '1' in just one of the first k positions and '0' in the other k-1 of the first k positions.
For example for a (7,4) code, only four codeword's are required, e.g.,

1	0	0	0	1	1	0
0	1	0	0	1	0	1
0	0	1	0	0	1	1
0	0	0	1	1	1	1

So, to obtain the codeword for dataword 1011, the first, third and fourth codewords in the list are added together, giving 1011010
This process will now be described in more detail,

- An (n,k) block code has code vectors
  $d=(d_1\ d_2 \ldots d_k)$ and
  $c=(c_1\ c_2 \ldots\ldots c_n)$
- The block coding process can be written as $c=dG$
  Where G is the Generator Matrix

$$G = \begin{bmatrix} a_{11} & a_{12} & \ldots & a_{1n} \\ a_{21} & a_{22} & \ldots & a_{2n} \\ . & . & \ldots & . \\ a_{k1} & a_{k2} & \ldots & a_{kn} \end{bmatrix} = \begin{bmatrix} \mathrm{a}_1 \\ \mathrm{a}_2 \\ . \\ \mathrm{a}_k \end{bmatrix}$$

- Thus,

$$c = \sum_{i=1}^{k} d_i \mathrm{a}_i$$

ai must be linearly independent, i.e.
Since codeword's are given by summations of the ai vectors, then to avoid 2 data words having the same codeword the ai vectors must be linearly independent.

- Sum (mod 2) of any 2 codeword's is also a codeword, i.e.,
  Since for data words d_1 and d_2 we have; $d_3 = d_1 + d_2$

$$c_3 = \sum_{i=1}^{k} d_{3i}\mathrm{a}_i = \sum_{i=1}^{k} (d_{1i} + d_{2i})\mathrm{a}_i = \sum_{i=1}^{k} d_{1i}\mathrm{a}_i + \sum_{i=1}^{k} d_{2i}\mathrm{a}_i$$

$$c_3 = c_1 + c_2$$

0 is always a codeword, i.e., since all zeros is a dataword then,

$$c = \sum_{i=1}^{k} 0\, a_i = 0$$

Error detection and correction capabilities of linear block codes:

Example: Find linear block code encoder G if code generator polynomial $g(x)=1+x+x^3$ for a (7, 4) code; n = total number of bits = 7, k = number of information bits = 4, r = number of parity bits = n - k = 3

I is the identity matrix

P is the parity matrix

- The Generator Polynomial can be used to determine the Generator Matrix G that allows determination of parity bits for a given data bits of m by multiplying as follows:
- Other combinations of m can be used to determine all other possible code words

$$p_1 = \text{Re}\left[\frac{x^3}{1+x+x^3}\right] = 1+x \rightarrow [110]$$

$$p_2 = \text{Re}\left[\frac{x^4}{1+x+x^3}\right] = x+x^2 \rightarrow [011]$$

$$p_3 = \text{Re}\left[\frac{x^5}{1+x+x^3}\right] = 1+x+x^2 \rightarrow [111]$$

$$p_4 = \text{Re}\left[\frac{x^6}{1+x+x^3}\right] = 1+x^2 \rightarrow [101]$$

$$G = \begin{bmatrix} 1000|110 \\ 0100|011 \\ 0010|111 \\ 0001|101 \end{bmatrix} = [I \mid P]$$

$$\begin{cases} c_1 = m_1 \\ c_2 = m_2 \\ \dots \\ c_k = m_k \\ c_{k+1} = m_1 p_{1(k+1)} \oplus m_2 p_{2(k+1)} \oplus \dots \oplus m_k p_{k(k+1)} \\ \dots \\ c_n = m_1 p_{1n} \oplus m_2 p_{2n} \oplus \dots \oplus m_k p_{kn} \end{cases}$$

Hamming code:

- Syndrome has R bits
- Number of parity bits R = $n - k$ and n = $2^R - 1$
- We will consider a special class of SEC codes (i.e., Hamming distance = 3) where,
- 0 value implies zero errors
- $2^R - 1$ other syndrome values, i.e., one for each bit that might need to be corrected
- This is achieved if each column of H is a different binary word – remember s = eH^T

Systematic form of (7,4) Hamming code is,

$$G = [I \mid P] = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix} \qquad H = [-P^T \mid I] = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

The original form is non-systematic,

$$G = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix} \qquad H = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}$$

Compared with the systematic code, the column orders of both G and H are swapped so that the columns of H are a binary count

The column order is now 7, 6, 1, 5, 2, 3, 4, i.e., col. 1 in the non-systematic H is col. 7 in the systematic H.

- For a non-systematic (7,4) code

d = 1011
c = 1110000
+ 0101010
+ 1101001
= 0110011
e = 0010000
c_r= 0100011
s = $c_r H^T = eH^T$ = 011

- Note the error syndrome is the binary address of the bit to be corrected

Double errors will always result in wrong bit being corrected, since A double error is the sum of 2 single errors .The resulting syndrome will be the sum of the corresponding 2 single error syndromes this syndrome will correspond with a third single bit error .Consequently the ‘corrected’ codeword will now contain 3 bit errors, i.e., the original double bit error plus the incorrectly corrected bit!

Cyclic codes:

The code with the generator matrix has codewords

c1 = 1011100 c2 = 0101110 c3 =0010111

c1 + c2 = 1110010 c1 + c3 = 1001011 c2 + c3 = 0111001

c1 + c2 + c3 = 1100101

and it is cyclic because the right shifts have the following impacts

c1 ® c2, c2 ® c3, c3 ® c1 + c3

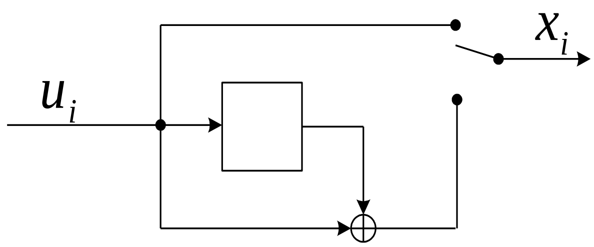
c1 + c2 ® c2 + c3, c1 + c3 ® c1 + c2 + c3, c2 + c3 ® c1

c1 + c2 + c3 ® c1 + c2

Convolution codes: Introduction

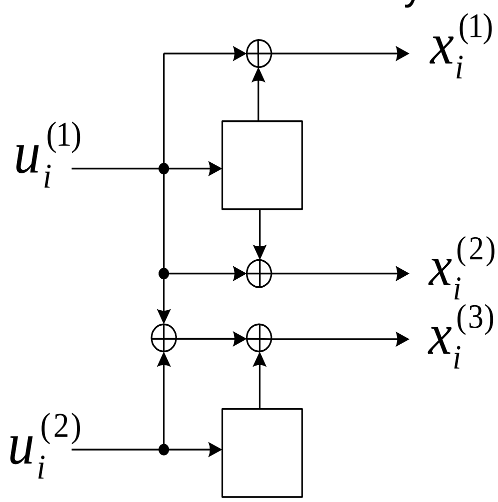
Convolutional codes map information to code bits sequentially by convolving a sequence of information bits with “generator” sequences. A Convolutional encoder encodes K information bits to N>K code bits at one time step ,Convolutional codes can be regarded as block codes for which the encoder has a certain structure such that we can express the encoding operation as convolution. The Convolutional code is linear The encoding mapping is injective Code bits generated at time step i are affected by information bits up to M time steps i – 1, i – 2, …, i – M back in time. M is the maximal delay of information bits in the encoder Code memory is the (minimal) number of registers to construct an encoding circuit for the code. Constraint length is the overall number of information bits affecting code bits generated at time step i: =code memory + K=MK + K=(M + 1)K

A convolutional code is systematic if the N code bits generated at time step i contain the K information bits.



has delay *M*=1, memory 1, constraint length 2, and it is systematic

Example: the rate 2/3 code defined by the circuit



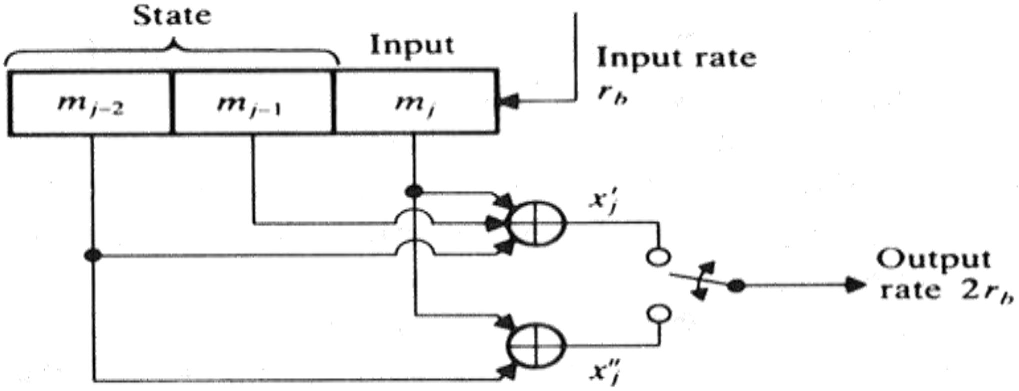
has delay *M*=1, memory 2, constraint length 4, and not systematic

- Most widely used channel code
- Encoding of information stream rather than information block
- Decoding is mostly performed by the Viterbi Algorithm (not covered here)
- The output constraint length K for a convolution code is defined as $K = M + 1$ where M is the maximum number of stages in any shift register
- The code rate r is defined as $r = k/n$ where k is the number of parallel information bits and n is the number of parallel output encoded bits at one time interval
- A convolution code encoder with n=2 and k=1 or code rate r = 1/2 is shown next

Nyquist's First Criterion for Zero lSI

Convolutional codes are applied in applications that require good performance with low implementation complexity. They operate on code streams (not in blocks) ,Convolution codes have memory that utilizes previous bits to encode or decode following bits (block codes are memory less),Convolutional codes are denoted by (n,k,L), where L is code (or encoder) Memory depth (number of register stages) Constraint length C=n(L+1) is defined as the number of encoded bits a message bit can influence to Convolutional codes achieve good performance by expanding their memory depth.

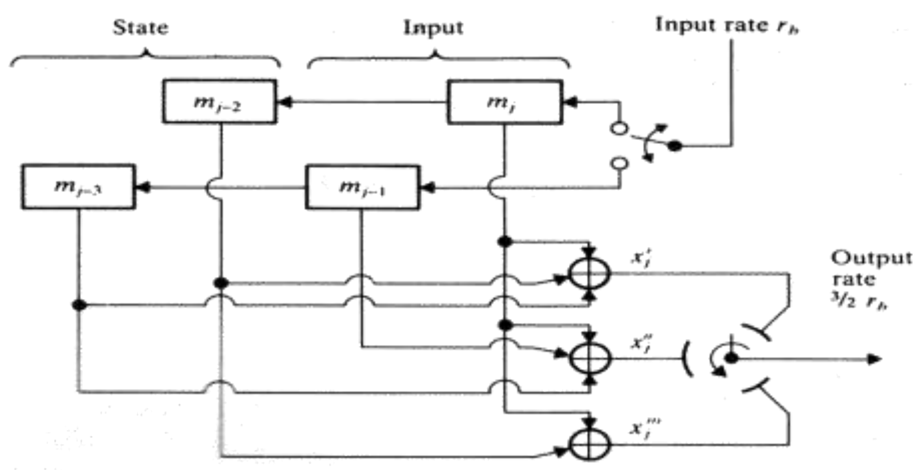
Convolutional encoder is a finite state machine (FSM) processing information bits in a serial manner
Thus the generated code is a function of input and the state of the FSM
In this (n,k,L) = (2,1,2) encoder each message bit influences a span of C= n(L+1)=6 successive output bits = constraint length C
Thus, for generation of n-bit output, we require in this example n shift registers in k = 1 convolutional encoder.



$$\begin{cases} x'_j = m_{j-2} \oplus m_{j-1} \oplus m_j \\ x''_j = m_{j-2} \oplus m_j \end{cases}$$

$$x_{out} = x'_1\, x''_1\, x'_2\, x''_2\, x'_3\, x''_3 \ldots$$

(n,k,L) = (2,1,2) encoder



After each new block of k input bits follows a transition into new state. Hence, from each input state transition, 2k different output states may follow,
Each message bit influences a span of C = n(L+1) = 3(1+1) = 6
successive output bits

$$x'_j = m_{j-3} \oplus m_{j-2} \oplus m_j$$

$$x''_j = m_{j-3} \oplus m_{j-1} \oplus m_j$$

$$x'''_j = m_{j-2} \oplus m_j$$

Encoding of Convolutional Codes; Transform Domain Approach

In any linear system, we know that the time domain operation involving the convolution integral can be replaced by the more convenient transform domain operation, involving polynomial multiplication. Since a Convolutional encoder can be viewed as a 'linear time invariant finite state machine, we may simplify computation of the adder outputs by applying appropriate transformation. As is done in cyclic codes, each 'sequence in the encoding equations can' be replaced by a corresponding polynomial and the convolution operation replaced by polynomial multiplication. For example, for a (2, 1, m) code, the encoding equations become:

$$v^{(1)}(X) = u(X)\, g^{(1)}(X) \qquad \ldots\ldots\ldots\ldots\ldots\ldots \qquad (8.12a)$$

$$v^{(2)}(X) = u(X)\, g^{(2)}(X) \qquad \ldots\ldots\ldots\ldots\ldots\ldots \qquad (8.12b)$$

Where $u(X) = u_1 + u_2X + u_3X^2 + \ldots$ is the information polynomial,

$v^{(1)}(X) = v_1^{(1)} + v_2^{(1)}X + v_3^{(1)}X^2 + \ldots..$, and

$v^{(2)}(X) = v_1^{(2)} + v_2^{(2)}X + v_3^{(2)}X^2 + \ldots..$

are the encoded polynomials.

$g^{(1)}(X) = g_1^{(1)} + g_2^{(1)}X + g_3^{(1)}X^2 + \ldots..$, and

$g^{(2)}(X) = g_1^{(2)} + g_2^{(2)}X + g_3^{(2)}X^2 + \ldots..$

"Generator polynomials" of' the code; and all operations are modulo-2. After multiplexing, the code word becomes:

$$v(X) = v^{(1)}(X^2) + X\, v^{(2)}(X^2) \qquad \ldots\ldots\ldots\ldots\ldots\ldots \qquad (8.13)$$

The indeterminate 'X' can be regarded as a "unit-delay operator", the power of X defining the number of time units by which the associated bit is delayed with respect to the initial bit in the sequence.

Example 8.5:

For the (*2, 1, 3*) encoder of Fig 8.3, the impulse responses were: $g^{(1)} = (1,0,1,1)$, and $g^{(2)} = (1,1,1,1)$

The generator polynomials are: $g^{(1)}(X) = 1 + X^2 + X^3$, and $g^{(2)}(X) = 1 + X + X^2 + X^3$

For the information sequence $u = (1, 0, 1, 1, 1)$; the information polynomial is: $u(X) = 1+X^2+X^3+X^4$

The two code polynomials are then:

$$v^{(1)}(X) = u(X)\, g^{(1)}(X) = (1 + X^2 + X^3 + X^4)\,(1 + X^2 + X^3) = 1 + X^7$$

$$v^{(2)}(X) = u(X)\, g^{(2)}(X) = (1 + X^2 + X^3 + X^4)\,(1 + X + X^2 + X^3) = 1 + X + X^3 + X^4 + X^5 + X^7$$

From the polynomials so obtained we can immediately write:

$$v^{(1)} = (1\ 0\ 0\ 0\ 0\ 0\ 0\ 1), \text{ and } v^{(2)} = (1\ 1\ 0\ 1\ 1\ 1\ 0\ 1)$$

Pairing the components we then get the code word $v = (11, 01, 00, 01, 01, 01, 00, 11)$.

$$K_l = \max_{1 \le j \le n} \left\{\deg\ g_l^{(j)}(X)\right\},\ 1 \le l \le k \qquad \ldots\ldots\ldots\ldots\ldots \qquad (8.15)$$

Where $g_l^{(j)}(X)$ is the generator polynomial relating the *l*-th input to the *j*-th output and the encoder memory order *m* is:

$$m = \max_{1 \le l \le k} K_l = \max_{\substack{1 \le j \le \\ 1 \le l \le k}} \left\{\deg\ g_l^{(j)}(X)\right\} \qquad \ldots\ldots\ldots\ldots \qquad (8.16)$$

Since the encoder is a linear system and $u^{(l)}(X)$ represents the *l*-th input sequence and $v^{(j)}(X)$ represents the *j*-th output sequence the generator polynomial $g_l^{(j)}(X)$ can be regarded as the 'encoder transfer function' relating the input - *l* to the output – *j*. For the *k*-input, *n*- output linear system there are a total of $k \times n$ transfer functions which can be represented as a $(k \times n)$ "***transfer function matrix***".

$$G(X) = \begin{bmatrix} g_1^{(1)}(X), & g_1^{(2)}(X), & \cdots & g_1^{(n)}(X) \\ g_2^{(1)}(X), & g_2^{(2)}(X), & \cdots & g_2^{(n)}(X) \\ \vdots & \vdots & \vdots & \vdots \\ g_k^{(1)}(X), & g_k^{(2)}(X), & \cdots & g_k^{(n)}(X) \end{bmatrix} \qquad \ldots\ldots\ldots\ldots\ldots\ldots \qquad (8.17)$$

Using the transfer function matrix, the encoding equations for an (n, k, m) code can be expressed as

$$V(X) = U(X)\, G(X) \qquad \ldots\ldots\ldots\ldots\ldots \qquad (8.18)$$

$U(X) = [u^{(1)}(X), u^{(2)}(X) \ldots u^{(k)}(X)]$ is the k-vector, representing the information polynomials, and. $V(X) = [v^{(1)}(X), v^{(2)}(X) \ldots v^{(n)}(X)]$ is the n-vector representing the encoded sequences. After multiplexing, the code word becomes:

$$v(X) = v^{(1)}(X^n) + X\,v^{(2)}(X^n) + X^2\,v^{(3)}(X^n) + \ldots + X^{n-1}\,v^{(n)}(X^n) \qquad \ldots\ldots\ldots\ldots\ldots \qquad (8.19)$$

Example 8.6:

For the encoder of Fig 8.4, we have:

$$g_1^{(1)}(X) = 1 + X, \quad g_2^{(1)}(X) = X$$

$$g_1^{(2)}(X) = 1, \quad g_2^{(2)}(X) = 1 + X$$

$$g_1^{(3)}(X) = 1, \quad g_2^{(3)}(X) = 0$$

$$\therefore G(X) = \begin{bmatrix} 1+X & 1 & 1 \\ X & 1+X & 0 \end{bmatrix}$$

For the information sequence $u^{(1)} = (1\ 0\ 1)$, $u^{(2)} = (1\ 1\ 0)$, the information polynomials are:

$$u^{(1)}(X) = 1 + X^2, \; u^{(2)}(X) = 1 + X$$

Then $\quad V(X) = [v^{(1)}(X), v^{(2)}(X), v^{(3)}(X)]$

$$= [1 + X^2, 1 + X] \begin{bmatrix} 1+X & 1 & 1 \\ X & 1+X & 0 \end{bmatrix} = [1 + X^3, 0, 1 + X^2]$$

Hence the code word is:

$$v(X) = v^{(1)}(X^3) + Xv^{(2)}(X^3) + X^2v^{(3)}(X^3)$$

$$= (1 + X^9) + X\,(0) + X^2(1 + X^6)$$

$$= 1 + X^2 + X^8 + X^9$$

$$\therefore v = (1\ 0\ 1, 0\ 0\ 0, 0\ 0\ 1, 1\ 0\ 0).$$

General approach, State, Tree And Trellis Diagram:

Unlike block codes, convolutional codes are not of fixed length. The encoder instead processes using a sliding window the information bit sequence to produce a channel bit sequence. The window operates on a number of information bits at a time to produce a number of channel bits. For example, the encoder shown below examines three consecutive information bits and produces two channel bits. The encoder then shifts in a new information bit and produces another set of two channel bits based on the new information bit and the previous two information bits. In general the encoder stores M information bits. Based on these bits and the current set of k input bits produces n channel bits. The memory of the encoder is M. The constraint length is the largest number of consecutive input bits that any particular output depends. In the above example the outputs depend on a maximum of 3 consecutive input bits. The rate is kn. The operation to produce the channel bits is a linear combination of the information bits in the encoder. Because of this linearity each output of the encoder is a convolution of the input information stream with some impulse response of the encoder and hence the name convolutional code.

Example: K=3,M=2, rate 1/2 code

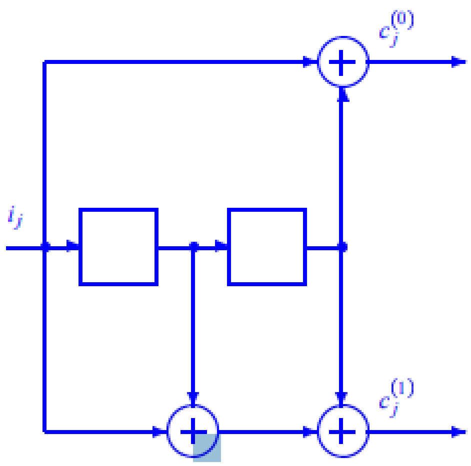


Figure 95: Convolutional Encoder

The operation of the encoder can be determined completely by way of a state transition diagram. The state transition diagram is a directed graph with nodes for each possible encoder content and transition between nodes corresponding to the results of different input bits to the encoder. The transitions are labeled with the output bits of the code. This is shown for the previous example

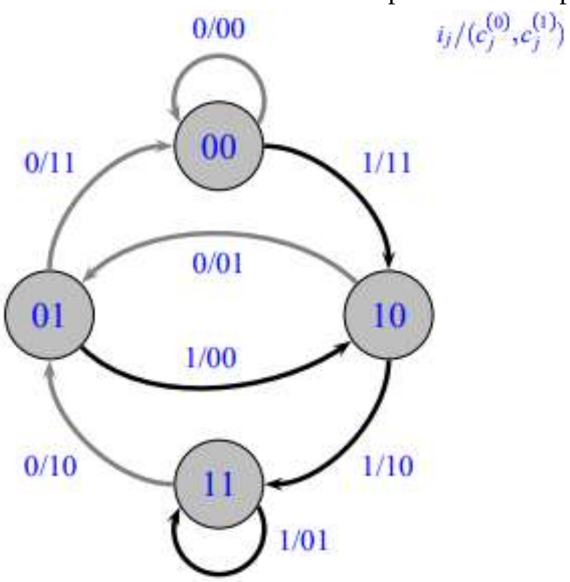
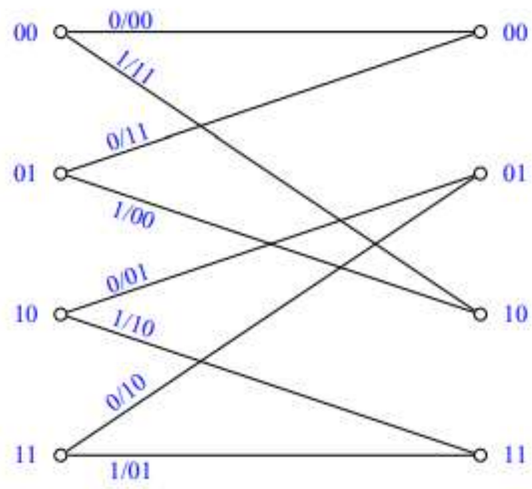
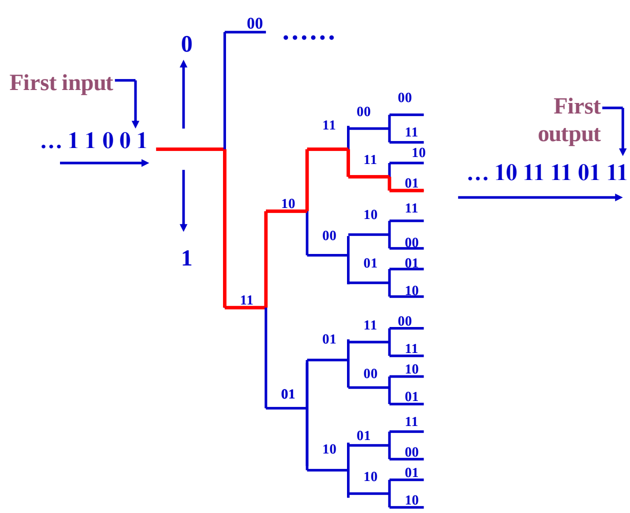


Figure : State Transition Diagram of Encoder

Trellis Section



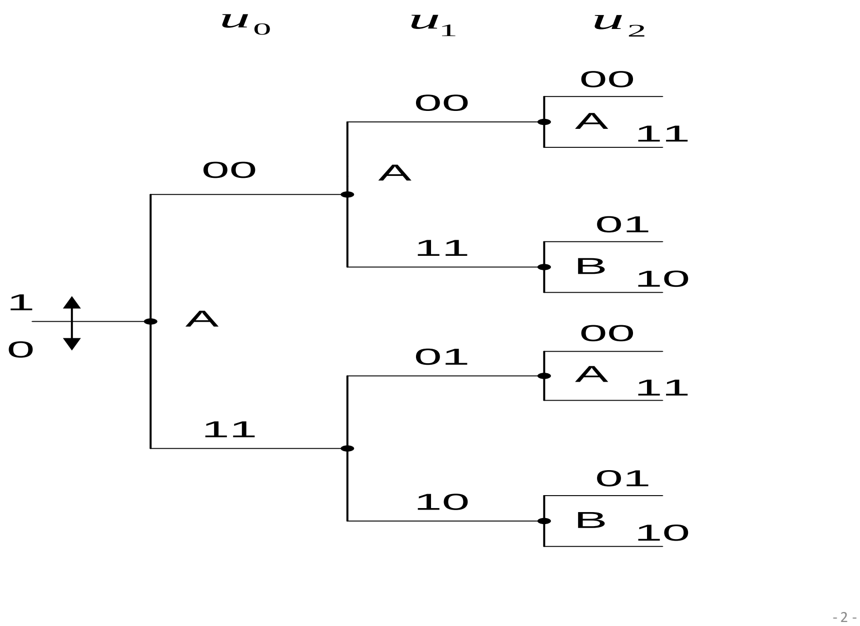
Tree Diagram



3

General approach; State, Tree And Trellis Diagram

Tree



General approach; State, Tree And Trellis Diagram

Convolutional codes

- k = number of bits shifted into the encoder at one time
- k=1 is usually used!!
- n = number of encoder output bits corresponding to
- the k information bits
- Rc = k/n = code rate
- K = constraint length, encoder memory.

Each encoded bit is a function of the present input bits and their past ones.

- **Example:**
  Consider the binary convolutional encoder with constraint length K=3, k=1, and n=3.
- The generators are: g1=[100], g2=[101], and g3=[111].
- The generators are more conveniently given in octal form as (4,5,7).

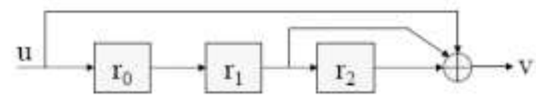
There are three alternative methods that are often used to describe a convolutional code:

- Tree diagram
- Trellis diagram
- State diagram

Assume a (2,1) convolutional coder with constraint length 6.Draw the tree diagram, state diagram and trellis diagram for the assumed coder [AUC APR/MAY 2011] Design block code for a message block of size eight that can correct for single errors Briefly discuss on various error control codes and explain in detail with one example for convolution code.

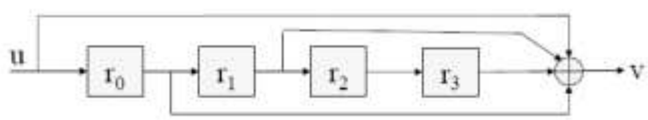
N=2, K=1 AND K=6(CONSTRAIN LENGHT) M=K/n=6/2=3, snce constrain length k=n*M 3 storage element in shift register N=2 two output bits One set k=1 of shift register having 3 storage element the Convolutional code structure is easy to draw from its parameters. First draw m boxes representing the m memory register. Then draw n modulo-2 adders to represent the n output bits. Now connect the memory registers to the adders using the generator polynomial Convolutional codes k = number of bits shifted into the encoder at one time • k=1 is usually used!! • n = number of encoder output bits corresponding to the k0020information bits • r = k/n = code rate • K = constraint length, encoder memory Each encoded bit is a function of the present input bits and their past ones.

Generator Sequence



$g_0^{(1)} = 1,\ g_1^{(1)} = 0,\ g_2^{(1)} = 1,$ and $g_3^{(1)} = 1.$

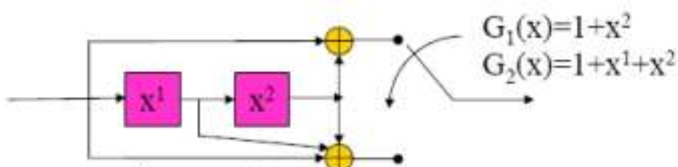
Generator Sequence: $g^{(1)}$=(1 0 1 1)



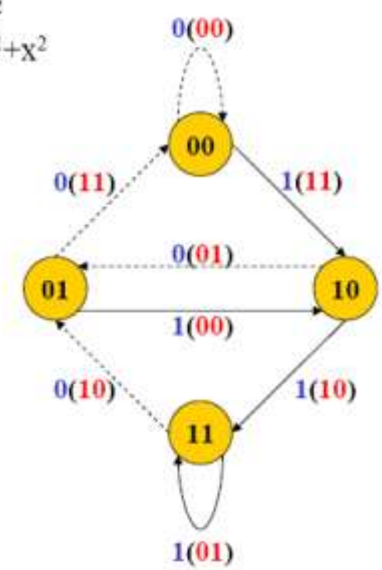
$g_0^{(2)} = 1,\ g_1^{(2)} = 1,\ g_2^{(2)} = 1,\ g_3^{(2)} = 0,$ and $g_4^{(2)} = 1$

Generator Sequence: $g^{(2)}$=(1 1 1 0 1)

Convolutional Codes An Example – (rate=1/2 with K=2)

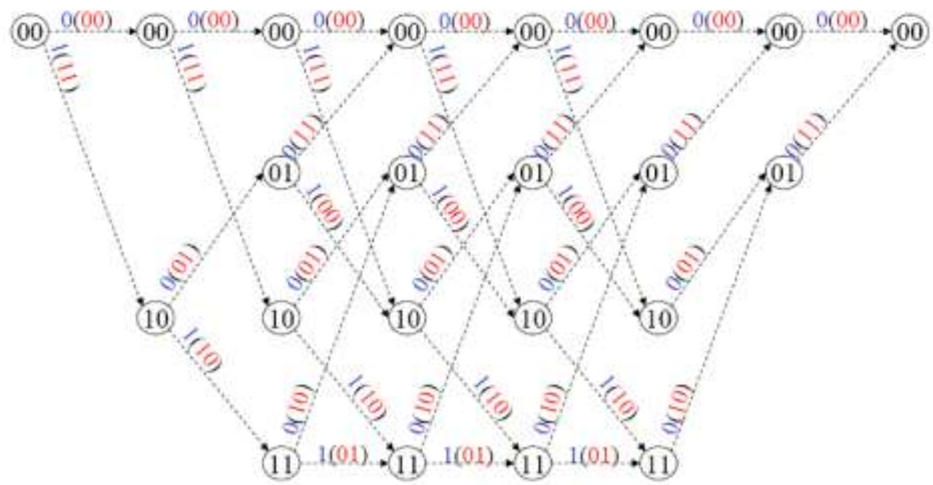


	Present	Next	Output
0	00	00	00
1	00	10	11
0	01	00	11
1	01	10	00
0	10	01	01
1	10	11	10
0	11	01	10
1	11	11	01

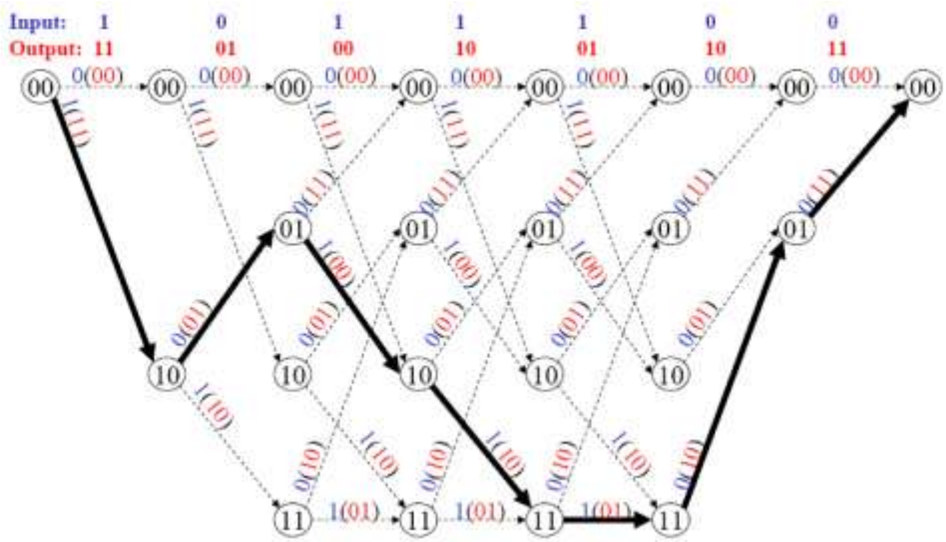


State Diagram

Trellis Diagram Representation



Encoding Process



Decoding using Viterbi Algorithm;

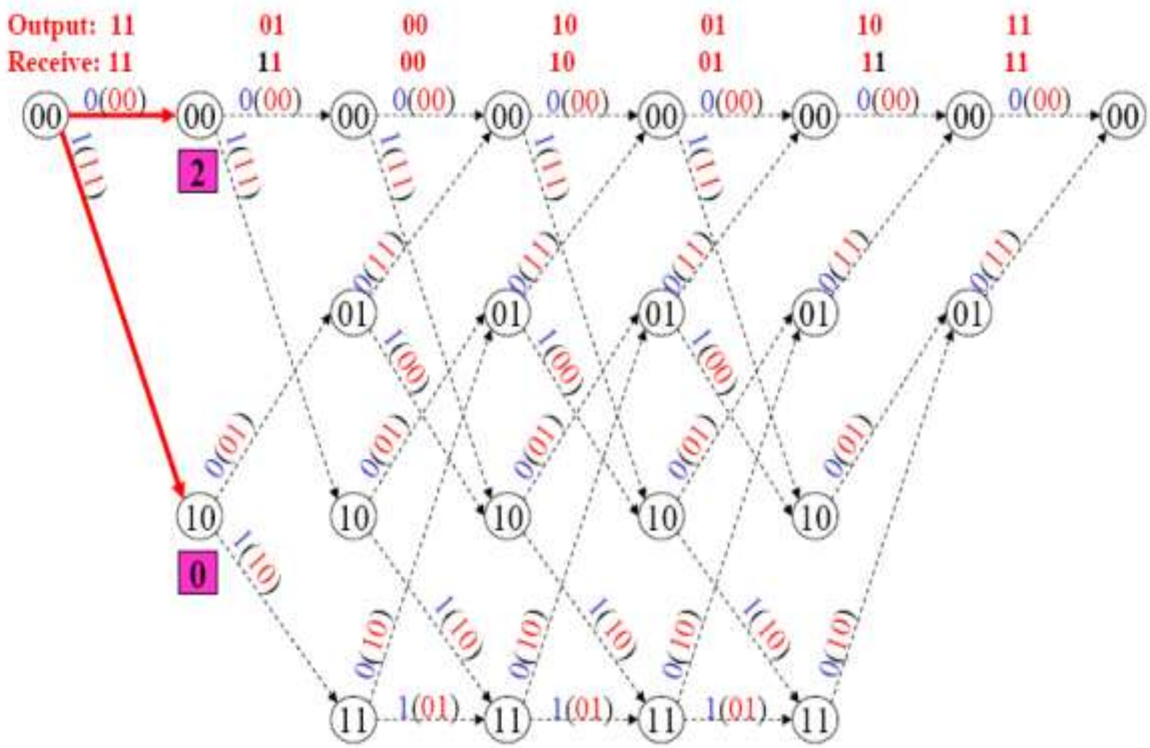
The equivalence between maximum likelihood decoding and minimum distance decoding implies that we may decode a Convolutional code by choosing a path in the code tree whose coded sequence differs from the received sequence in the fewest number of places. Since a code tree is equivalent to a trellis, we may limit our choice to the possible Paths in the trellis representation of the code. Viterbi algorithm operates by computing a metric or discrepancy for every possible Path in the trellis The metric for a particular path is defined as the Hamming distance between the Coded sequence represented by that path and the received sequence. For each state in the trellis, the algorithm compares the two paths entering the node And the path with the lower metric is retained. The retained paths are called survivors. The sequence along the path with the smallest metric is the maximum likelihood Choice and represents the transmitted sequence

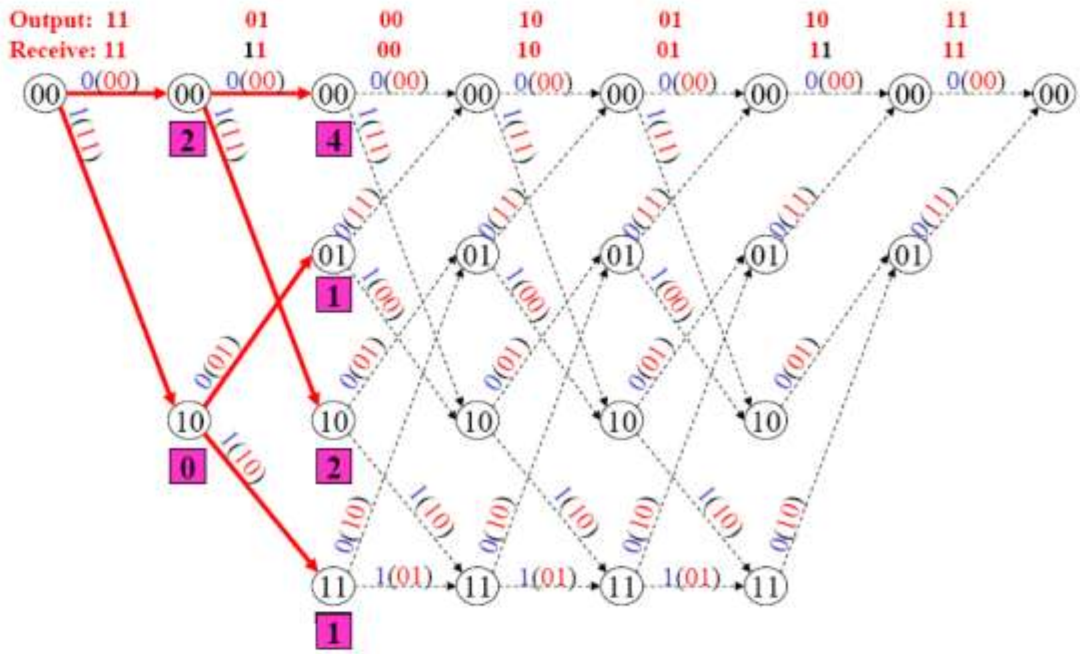
In the circuit below, suppose that the encoder generates an all-zero sequence and that the received sequence is (0100010000…) in which the are two errors due to channel noise: one in second bit and the other in the sixth bit. We can show that this double-error pattern is correctable using the Viterbi algorithm.

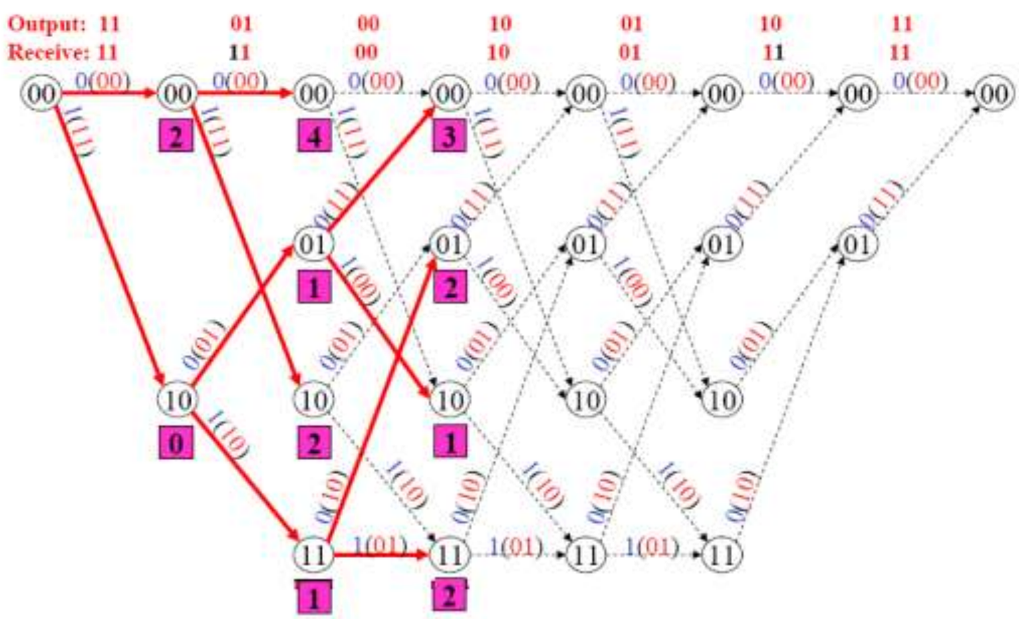
Decoding using Viterbi Algorithm;

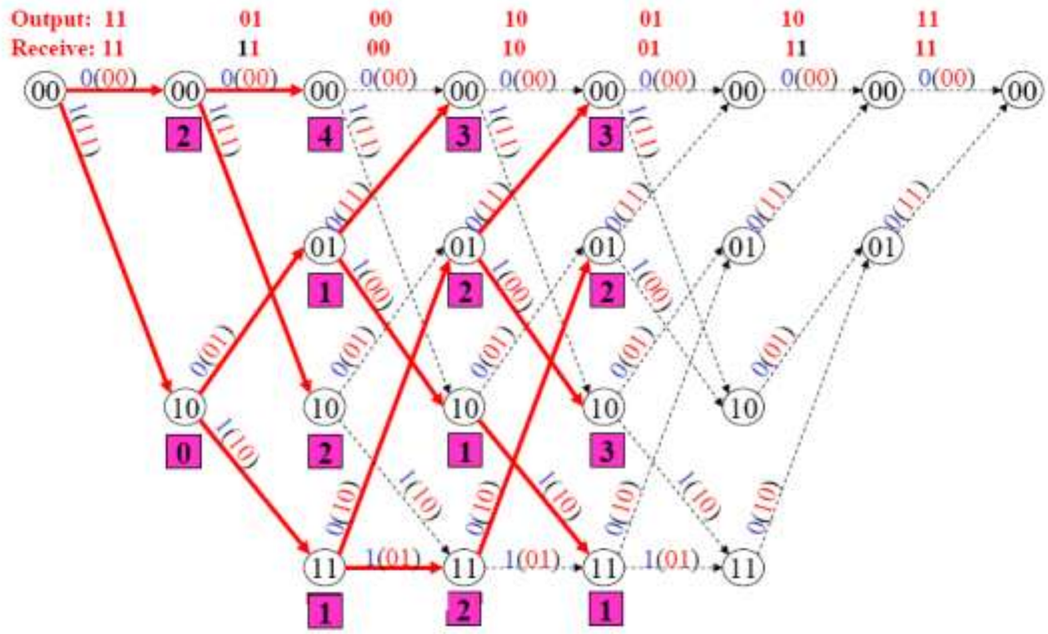
- An efficient search algorithm Performing ML decoding rule. Reducing the computational complexity. Basic concept
- Generate the code trellis at the decoder The decoder penetrates through the code trellis level by level in search for the transmitted code sequence
- At each level of the trellis, the decoder computes and compares the metrics of all the partial paths entering a node
- The decoder stores the partial path with the larger metric and eliminates all the other partial paths. The stored partial path is called the survivor.

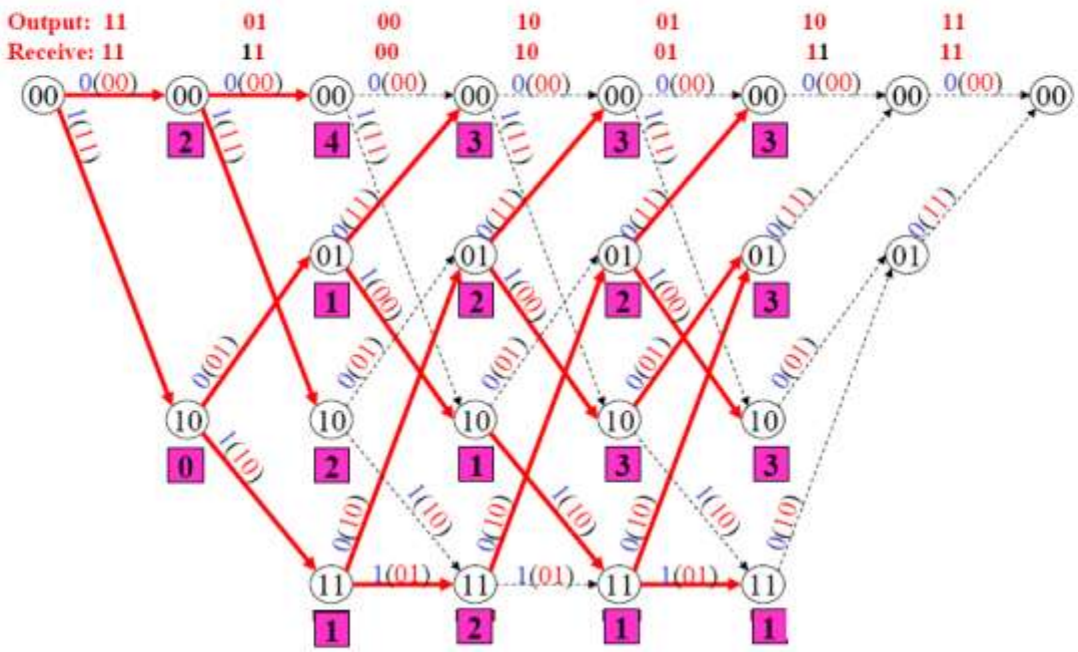
Viterbi Decoding Process

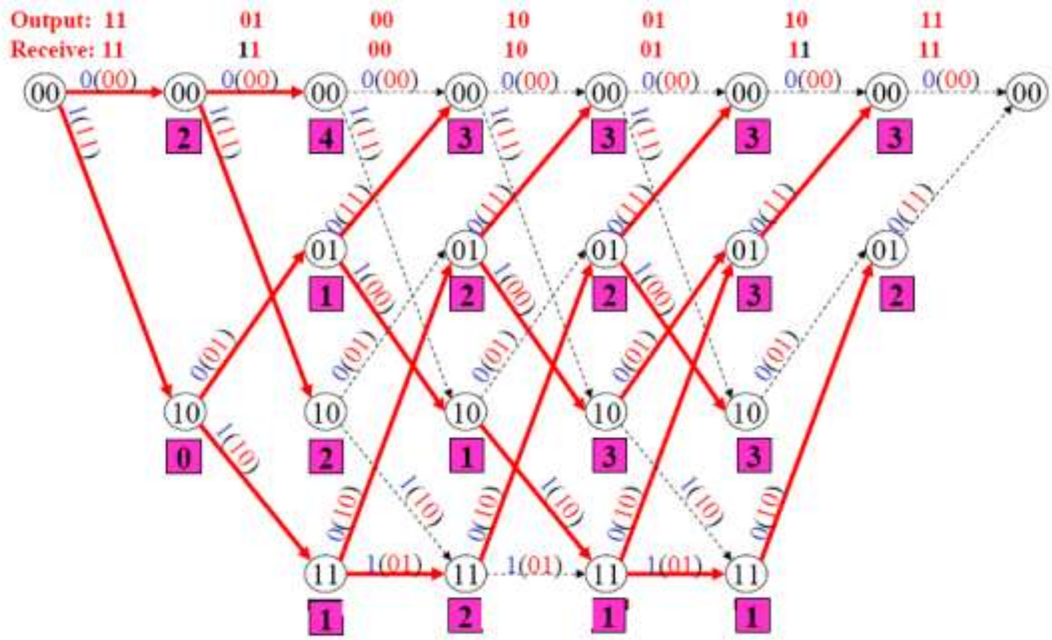


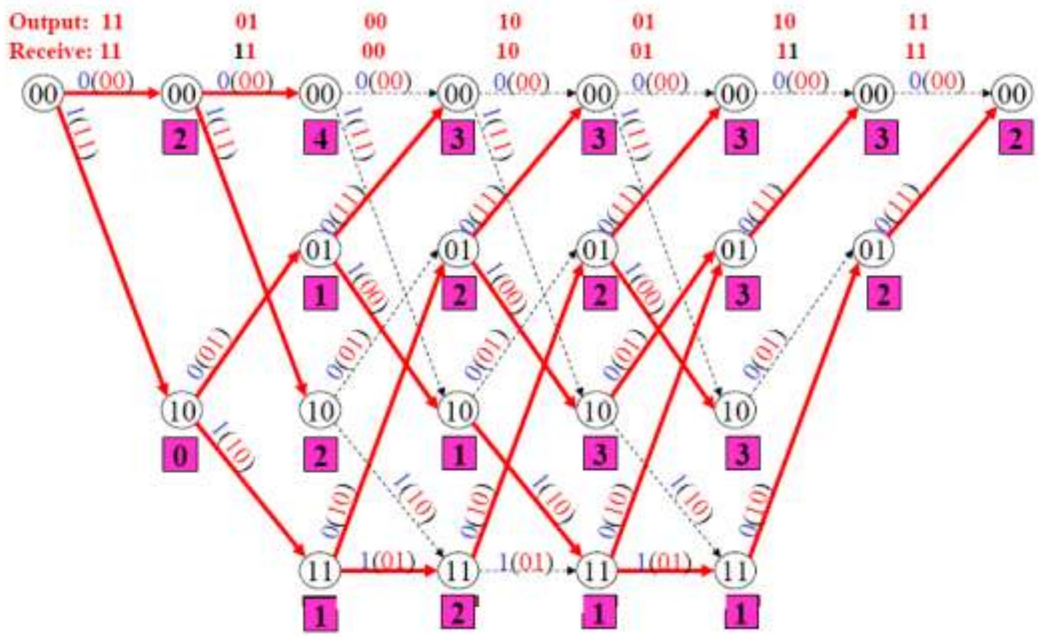

Output: 11 01 00 10 01 10 11
Receive: 11 11 00 10 01 11 11
0(00)
1(11)
0(11)
1(00)
0(01)
1(10)
0(10)
1(01)
2
4
1
0
2
1

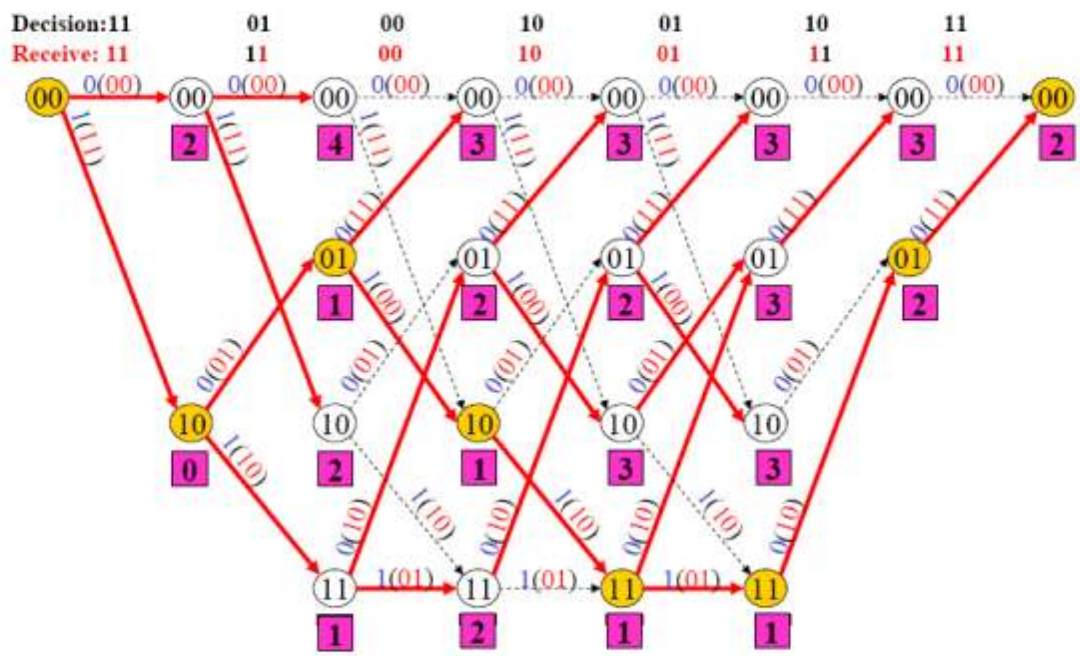

Output: 11 01 00 10 01 10 11
Receive: 11 11 00 10 01 11 11
0(00)
1(11)
0(11)
1(00)
0(01)
1(10)
0(10)
1(01)
2
4
3
1
2
0
2
1
1
2


Output: 11 01 00 10 01 10 11
Receive: 11 11 00 10 01 11 11


Output: 11 01 00 10 01 10 11
Receive: 11 11 00 10 01 11 11


Output: 11 01 00 10 01 10 11
Receive: 11 11 00 10 01 11 11


Output: 11 01 00 10 01 10 11
Receive: 11 11 00 10 01 11 11



Burst Error Correction

In **coding theory**, **burst error-correcting codes** employ methods of **correcting burst** errors, which are errors that occur in many consecutive bits rather than occurring in bits independently of each other. ... The methods used to **correct** random errors are inefficient to **correct burst** errors.

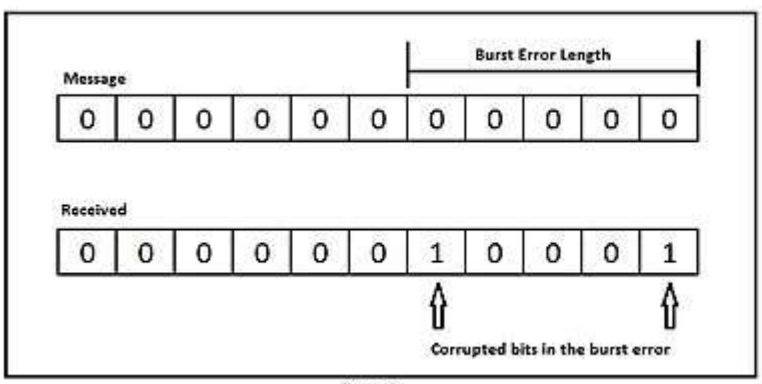


Figure 1

Many codes have been designed to correct random errors. Sometimes, however, channels may introduce errors which are localized in a short interval. Such errors occur in a burst (called burst errors) because they occur in many consecutive bits. Examples of burst errors can be found extensively in storage mediums. These errors may be due to physical damage such as scratch on a disc or a stroke of lightning in case of wireless channels. They are not independent; they tend to be spatially concentrated. If one bit has an error, it is likely that the adjacent bits could also be corrupted. The methods used to correct random errors are inefficient to correct burst errors.

A burst of length

Say a codeword is transmitted, and it is received as Then, the error vector is called a burst of length if the nonzero components of are confined to consecutive components. For example, is a burst of length
Although this definition is sufficient to describe what a burst error is, the majority of the tools developed for burst error correction rely on cyclic codes. This motivates our next definition.

A cyclic burst of length

An error vector is called a cyclic burst error of length if its nonzero components are confined to cyclically consecutive components. For example, the previously considered error vector , is a cyclic burst of length , since we consider the error starting at position and ending at position . Notice the indices are -based, that is, the first element is at position .

A burst : Consider a binary representation of length **l** such that **l** > 1. Now, if non-zero bits of the representation are cyclically confined to **l** consecutive positions with nonzero first and last positions, we say that this is burst of length **l**.

l-burst-error-correcting code : A code is said to be *l-burst-error-correcting code* if it has ability to correct burst errors up to length **l**.
Example: 00110010000 is a burst of length 5, while 010000000000001000 is a burst of length 6.

Wraparound burst of length l : A burst of length l that is obtained by any cyclic shift of a burst of length l is called Wraparound burst of length l.
Following are typical parameters that a burst can have

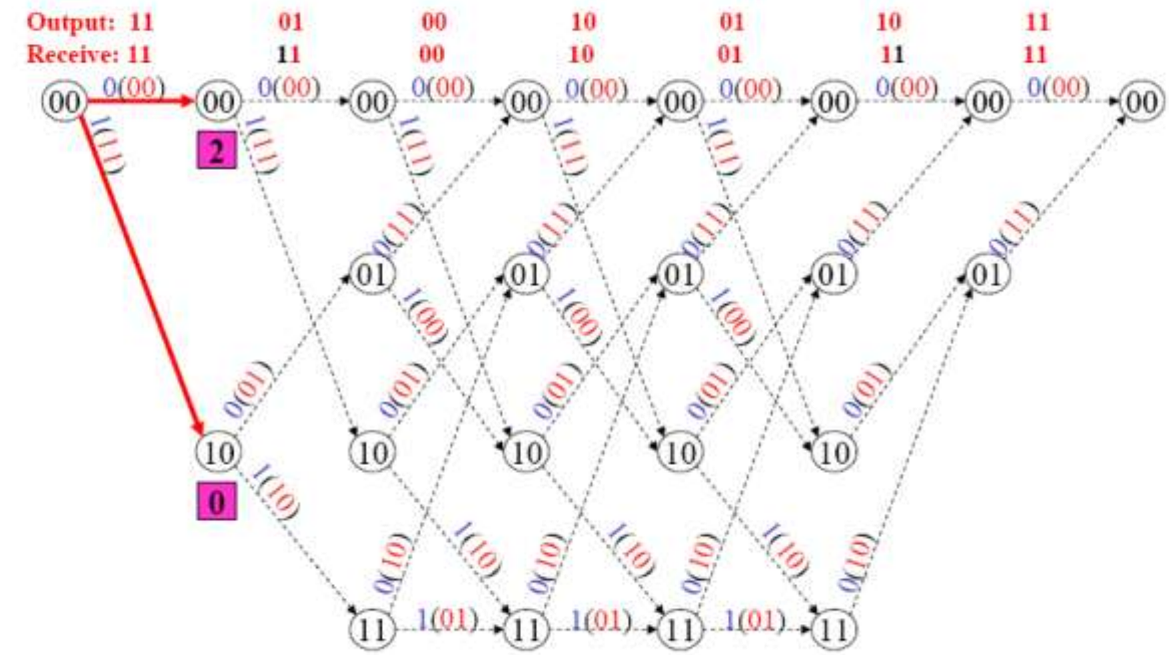
1. Location of burst - Least significant digit of burst is called as location of that burst.
2. Pattern of burst - A burst pattern of a burst of length l is defined as the polynomial b(x) of degree l − 1.

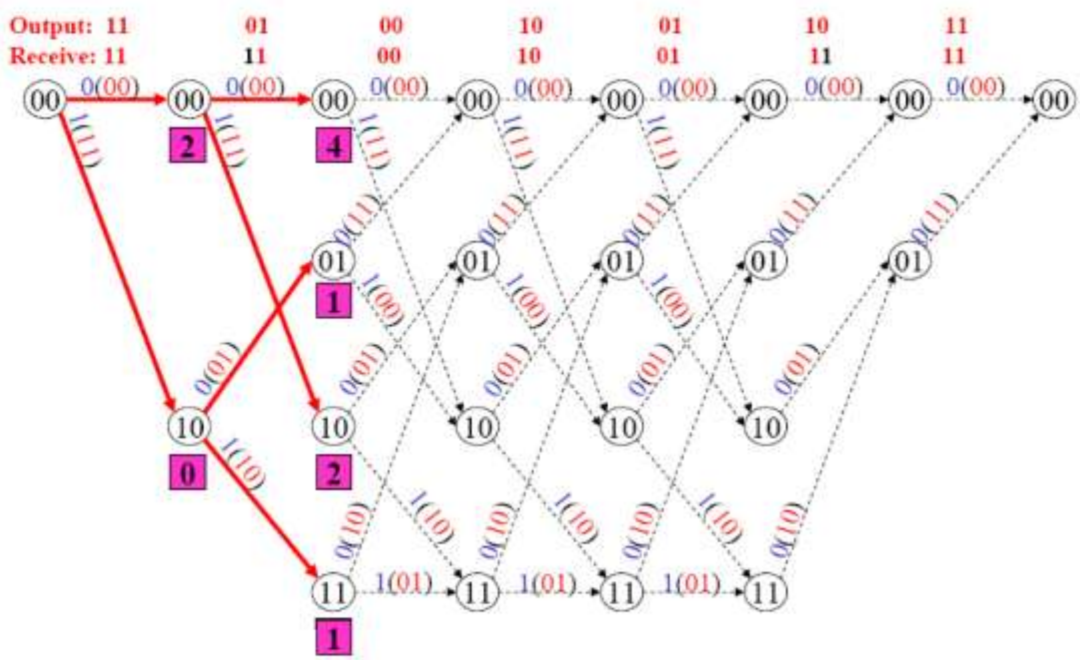
The codes we have considered so far have been designed to correct random errors. In general, a t-error correcting code corrects all error patterns of weight t or less in a codeword of block length n. It may be, however, that certain channels introduce errors localized in short intervals rather than at random. For example, in storage mediums, errors resulting from physical irregularities or structural alteration, perhaps flaws in the original medium or damage due to wear and tear, are not independent, but rather tend to be spatially concentrated. Similarly, interference over short time intervals in serially transmitted radio signals causes errors to occur in bursts. There exist codes for correcting such burst errors. Many of these codes are cyclic.
A cyclic burst error of length t is a vector whose non-zero components are contained within a cyclic run of length t, the first and last components in the run being non-zero. Examples: (i) e 1 = (01010110000) is a burst of length 6 in V11 (2). (ii) e 2 = (00000010001) is a burst of length 5 in V11 (2). (iii) e 3 = (01000000100) is a burst of length 5 in V11 (2).

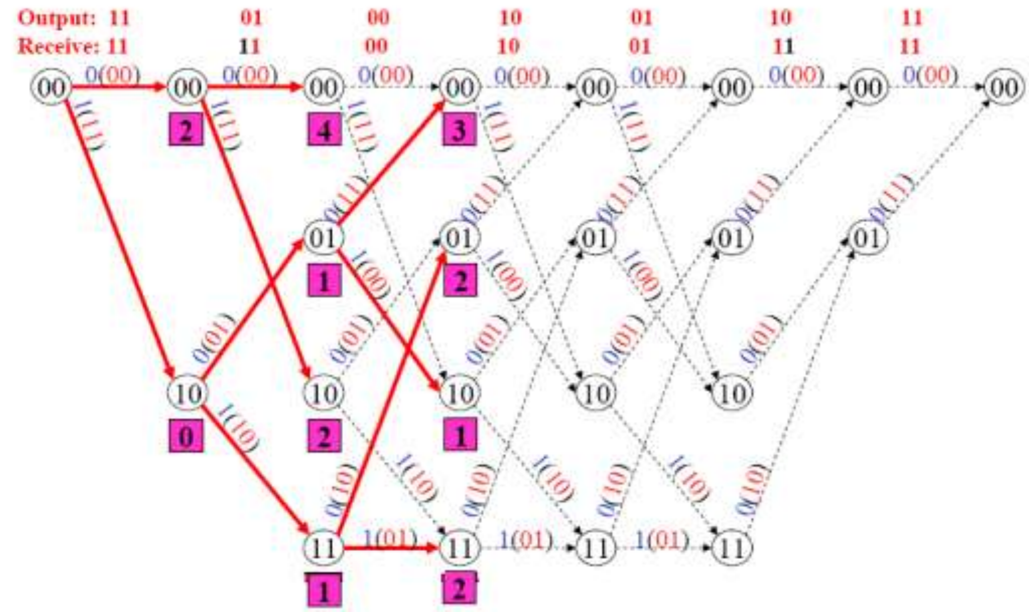
Block Interleaving and convolution interleaving

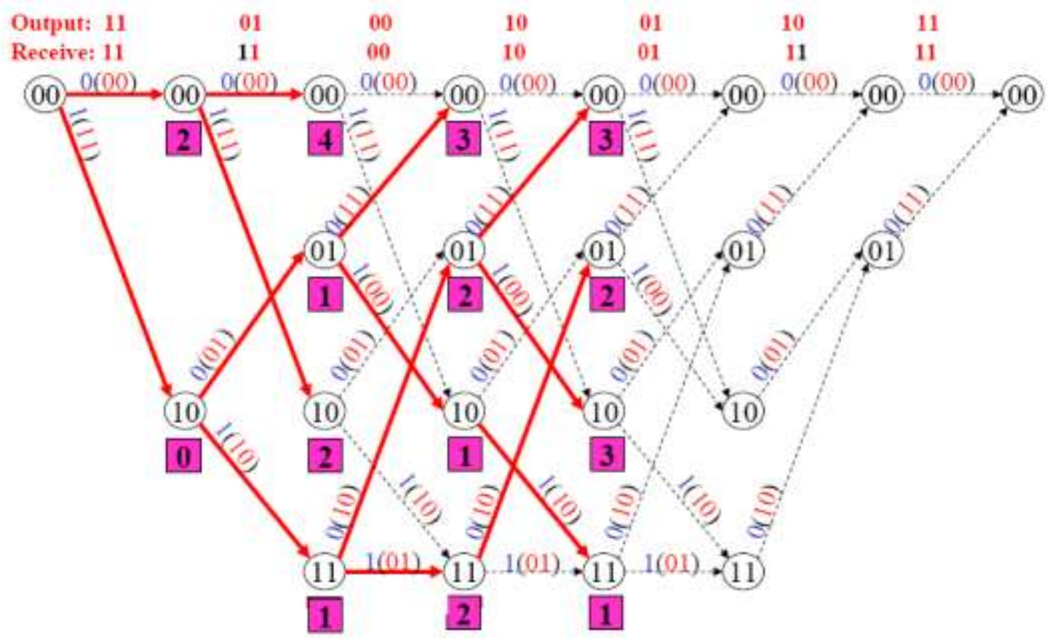
Decoding using Viterbi Algorithm:

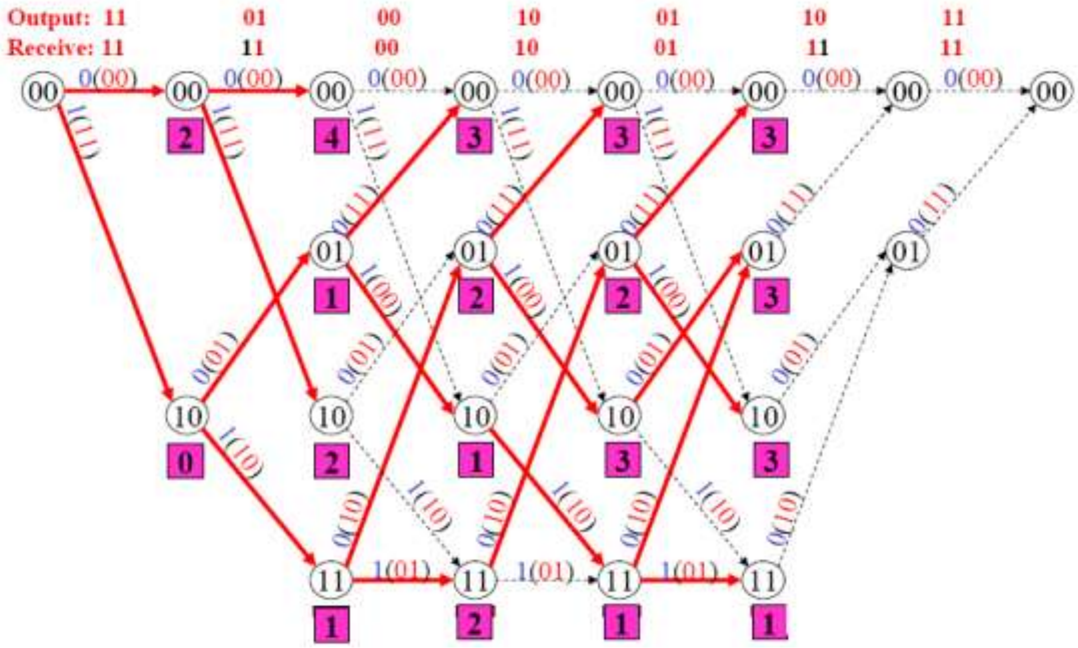
An efficient search algorithm Performing ML decoding rule. Reducing the computational complexity.
Basic concept .Generate the code trellis at the decoder The decoder penetrates through the code trellis level by level in search for the transmitted code sequence .At each level of the trellis, the decoder computes and compares the metrics of all the partial paths entering a node .The decoder stores the partial path with the larger metric and eliminates all the other partial paths. The stored partial path is called the survivor.
Viterbi Decoding Process

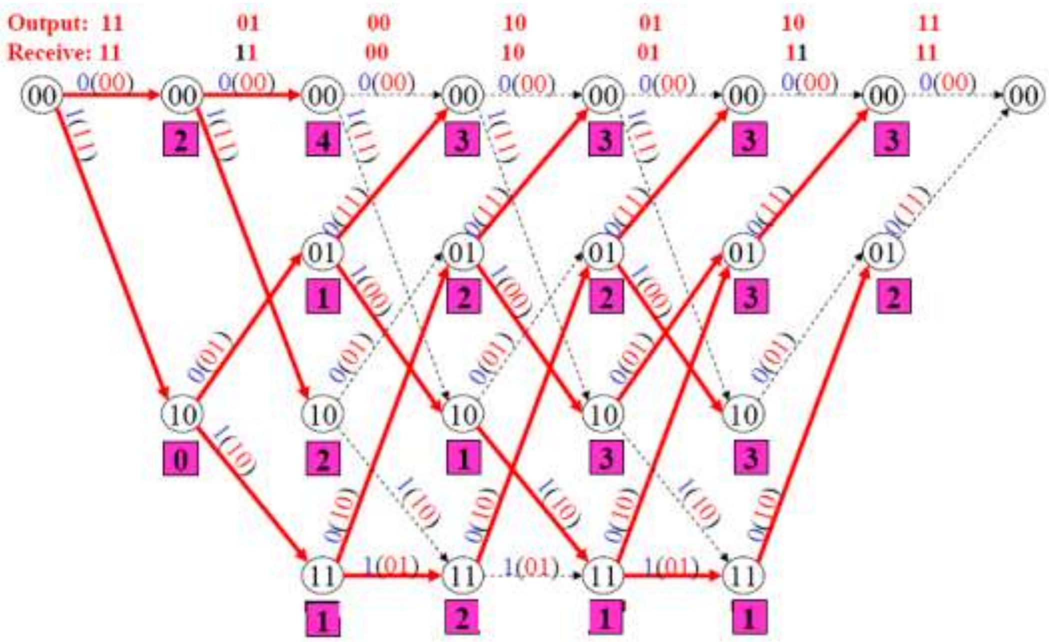

Output: 11 01 00 10 01 10 11
Receive: 11 11 00 10 01 11 11
00
01
10
11
0(00)
1(11)
0(11)
1(00)
0(01)
1(10)
0(10)
1(01)
2
0

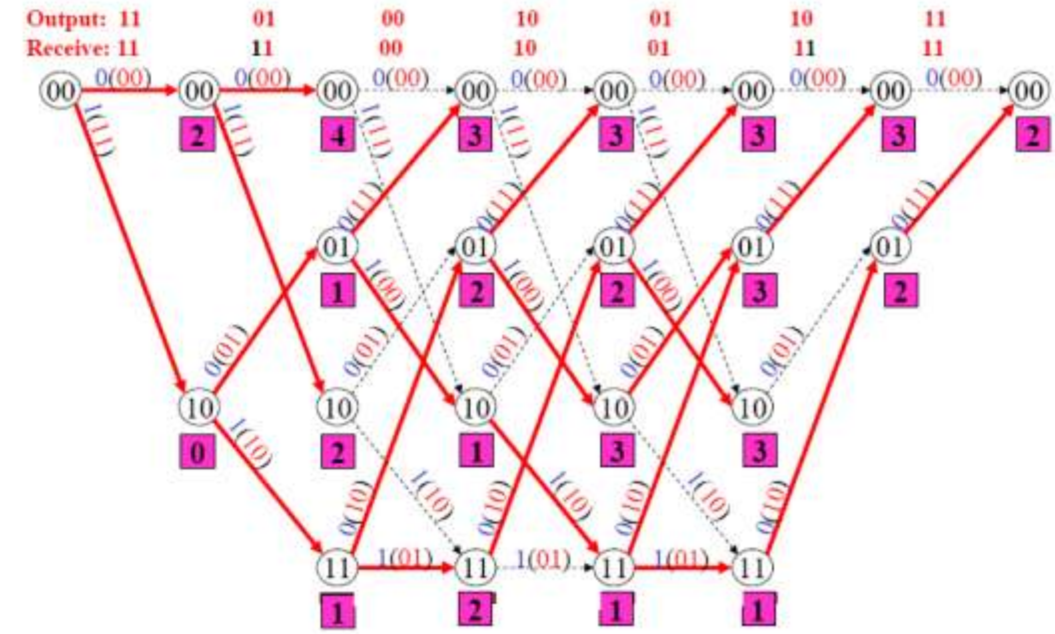

Output: 11 01 00 10 01 10 11
Receive: 11 11 00 10 01 11 11
00
01
10
11
0(00)
1(11)
0(11)
1(00)
0(01)
1(10)
0(10)
1(01)
2
0
4
1
2
1


Output: 11 01 00 10 01 10 11
Receive: 11 11 00 10 01 11 11


Output: 11 01 00 10 01 10 11
Receive: 11 11 00 10 01 11 11


Output: 11 01 00 10 01 10 11
Receive: 11 11 00 10 01 11 11


Output: 11 01 00 10 01 10 11
Receive: 11 11 00 10 01 11 11


Output: 11 01 00 10 01 10 11
Receive: 11 11 00 10 01 11 11

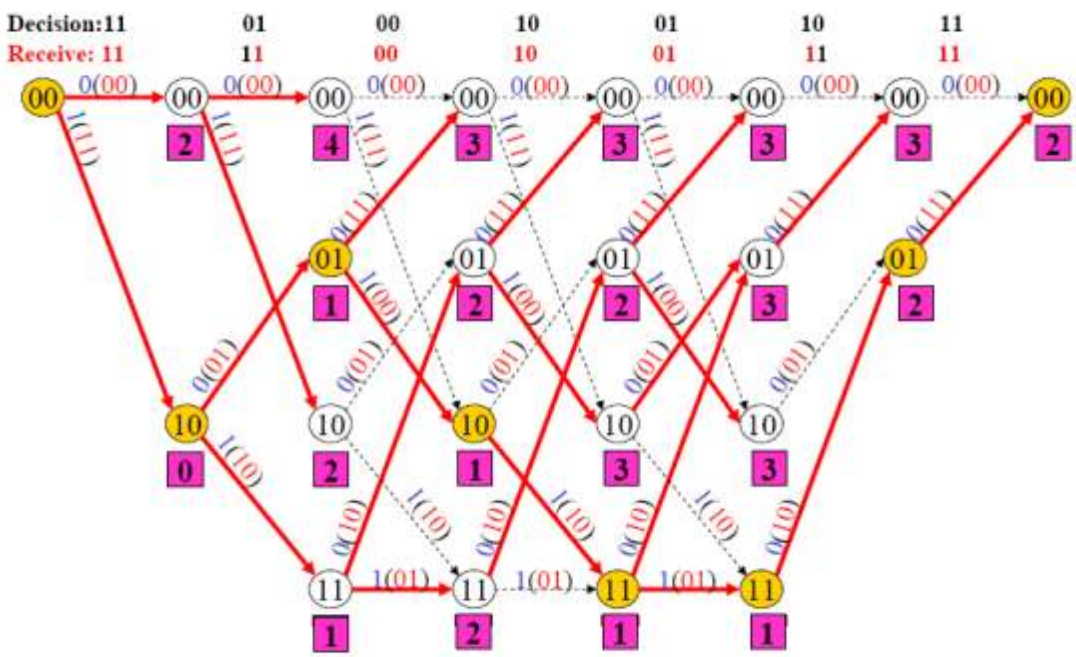

Decision:11 01 00 10 01 10 11
Receive: 11 11 00 10 01 11 11

Figure : Interleaving