Telecoms Systems (Week 3)

Dr Vincent (Yuanwei) Liu

yuanwei.liu@qmul.ac.uk

&

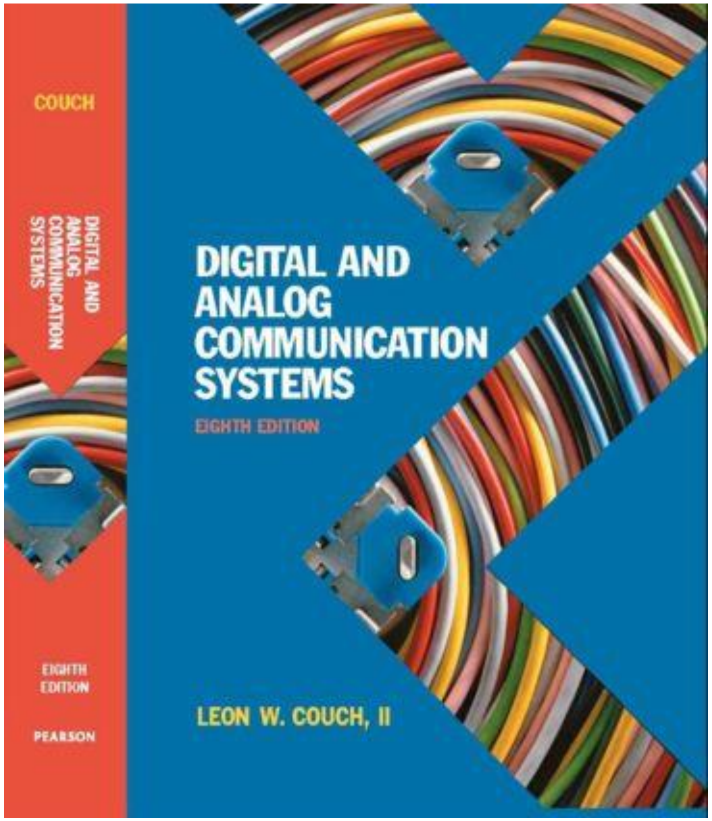
Dr Md Hasanuzzaman Sagor

m.h.sagor@qmul.ac.uk

School of Electronic Engineering & Computer Science,

Queen Mary
University of London

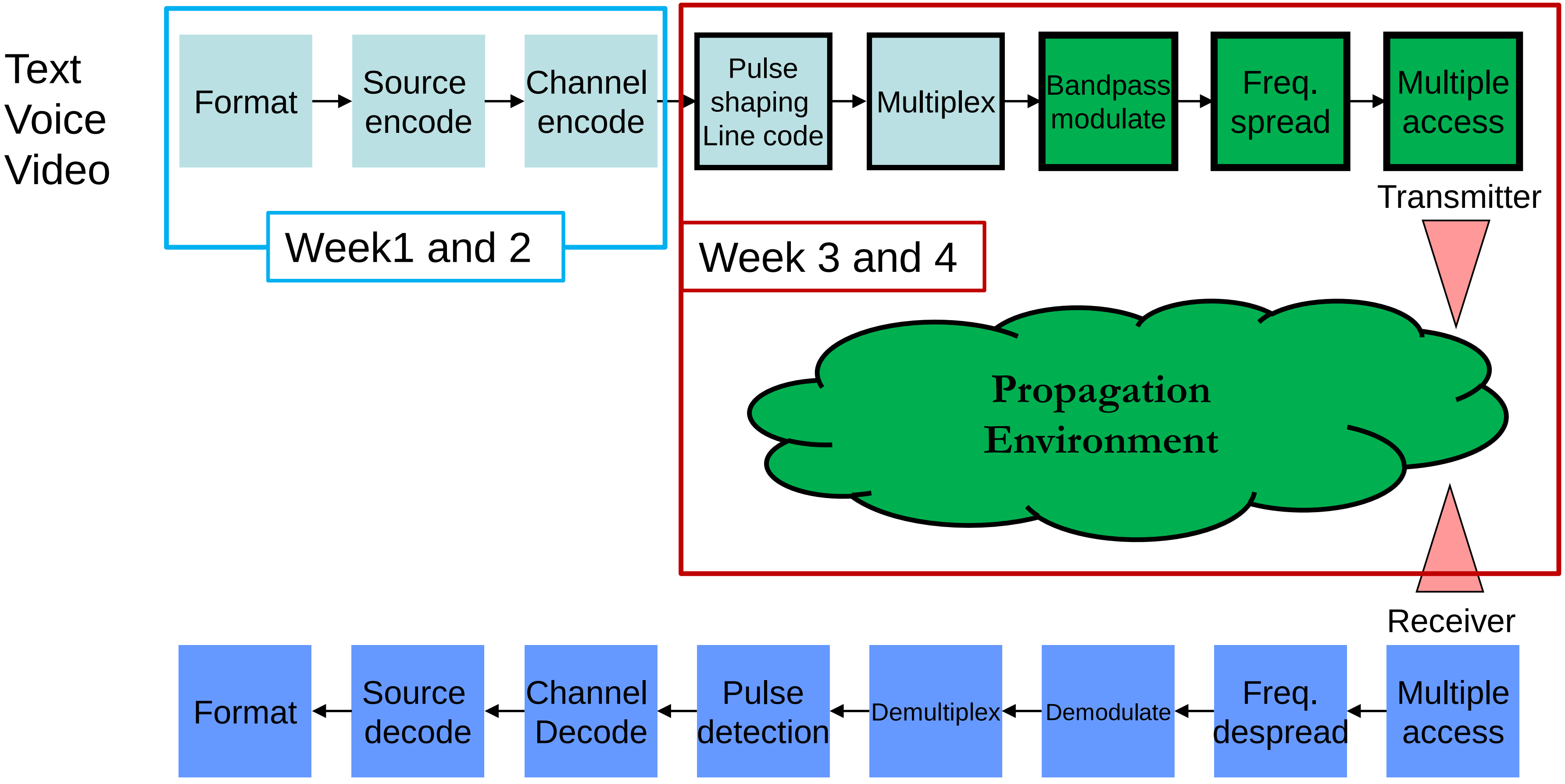
Reference book



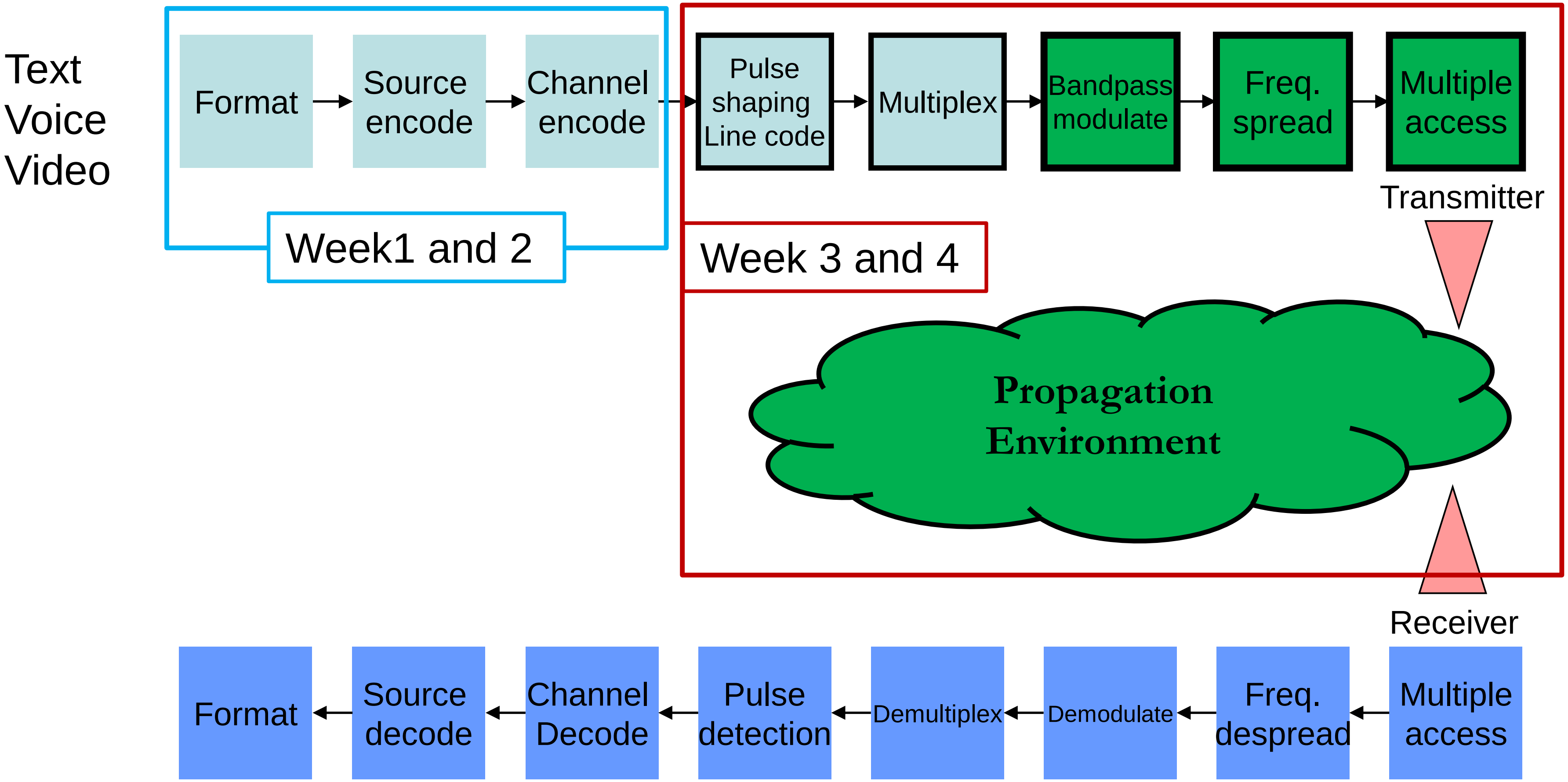
Digital and Analog Communication Systems
8th Edition, 2013 Pearson

LeonW. Couch

Overview of Wireless Communication System



Question: Why do we need those techniques?



Main goals of this week

- Based on how analogue waveforms can be converted to digital waveforms – pulse code modulation (PCM):
  - To learn how to compute the spectrum for digital signals;
  - To examine how the filtering of pulse signals affects our ability to recover the digital information at the receiver, e.g. inter-symbol interference (ISI);
  - To study how we can multiplex (or combine) data from several digital bit streams into one high-speed digital stream for transmissions over a digital system, e.g. time-division multiplexing.

These concepts will be explained with mathematically representations with physical meanings.

Outline of Teaching Plan

- PCM and Delta modulation (addressed)
- Line Codes and Spectra
- Inter Symbol Interference
- Time Division Multiplexing
- Weekly Test in Week 4

Queen Mary
University of London

Line Codes and Spectra

Queen Mary
University of London
web: http://www.eecs.qmul.ac.uk/~yuanwei/

Digital Data to Signal Conversion

- Need to map the signals to the medium
- Four stages:
  - Line coding
  - Block coding
  - Scrambling
  - Modulating
- Block coding and scrambling are optional in data to signal conversion.

Binary Transmission

1 0 1 1 1 1 1 1 1 1 0

- Need transitions for timing recovery
- Need zero (or constant) DC level as communications paths do not usually pass DC

Line coding

- Text, numbers, images, audio and video are converted into sequence of bits on the ADC (Analogue to Digital Converter).

- Line coding is the process of converting digital data to digital signals at the TX (Transmitter)

- At the RX (Receiver), the digital data are recreated by decoding the digital signal.

- To make the regeneration of the original signal more reliable.

Direct-current Component (DC)

- Receivers usually use the mean value of the signal as a reference/baseline to distinguish between 1's and 0's.

- A long string of 0's or 1's can cause a drift in the baseline/reference.

- Direct-current Component (DC) – mean (displacement) of the signal

  - affects the reliability in reproducing the original signal.
  - Frequency just about zero and can not pass through LPF (Low Pass Filter),
  - Telephone line can not pass frequencies below 200Hz.

Timing and synchronisation

- Timing of the incoming signal controls the reading rate.
- It keeps the receiver synchronised with the transmitter – extracted from the pulse pattern.
- A long sequences without a pulse edge makes timing very hard to maintain.

Line Coding Characteristics

- Baseband channels do not include any frequency translation, have increasing attenuation with frequency and often block DC.
- Line coding usually involves pulse amplitude modulation (PAM) and generates a signal that:
  - has a high timing content (transitions) irrespective of data to be transmitted.
  - has a zero (or constant) DC level.
  - must be uniquely decodable.
  - must be transparent to input signal.
  - must minimise the bandwidth required for transmission.

How do we mathematically represent the waveform for a digital signal, such as the PCM signal? - 1

The voltage (or current) waveforms for digital signals can be expressed as an orthogonal series with a finite number of terms N:

$$w(t) = \sum_{k=1}^{N} w_k \varphi_k(t), \qquad 0 < t < T_0 \tag{3-1}$$

w_k represents the digital data

$\varphi_k(t), k = 1,2,3, \ldots, N$ are N orthogonal functions that give the waveform its waveshape.

If the waveform of (3-1) is transmitted over a channel and appears at the input to the receiver, how can a receiver be built to detect the data? – evaluate the orthogonal series coefficient w_k

$$w_k = \frac{1}{K_k} \int_0^{T_0} w(t) \varphi_k^*(t) dt, \quad k = 1,2,\ldots,N \tag{3-2}$$

$w(t)$ is the waveform at the receiver input

$\varphi_k(t)$ is the known orthogonal function that was used to generate the waveform.

How do we mathematically represent the waveform for a digital signal, such as the PCM signal? - 2

The orthogonal function space representation of (3-1) corresponds to the orthogonal vector space represented by:

$$\boldsymbol{w} = \sum_{j=1}^{N} w_j \varphi_j \qquad (3\text{-}3)$$

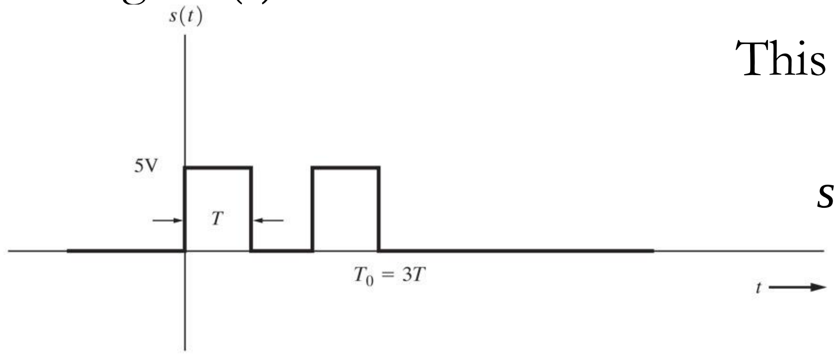
$\boldsymbol{w}$ is N-dimensional vector in Euclidean vector space
w_j is weighting coefficients
φ_j is an orthogonal set of N-directional vectors

A shorthand notation for the vector **w** is given by a row vector

$$\boldsymbol{w} = (w_1, w_2, w_3, \ldots . w_N)$$

Example – Vector representation of a binary signal - 1

- Examine the representation for the waveform of a 3-bit (binary) signal in the Figure (a) below.



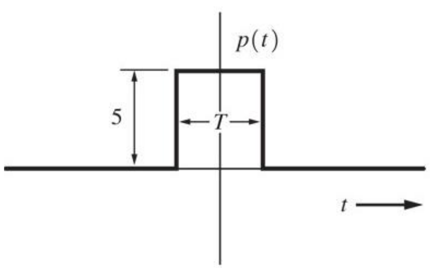
(a) A Three-Bit Signal Waveform

This signal could be directly represented by

$$s(t) = \sum_{j=1}^{N=3} d_j p\left[t - \left(j - \frac{1}{2}\right)T\right] = \sum_{j=1}^{N=3} d_j p_j(t)$$

where p(t) is a set of orthogonal functions that are not normalized as shown in Figure (b), and

$$p_j(t) \triangleq p\left[t - \left(j - \frac{1}{2}\right)T\right]$$



(b) Bit Shape Pulse

The vector $\bar{d} = (d_1, d_2, d_3) = (1, 0, 1)$ is the binary message word with 1 representing a binary 1 and 0 representing a binary 0.

Example – Vector representation of a binary signal - 2

- The set $\{p_j(t)\}$ is a set of orthogonal functions.
- The function p(t) is the pulse shape for each bit as shown in Figure (c).
- Using the orthogonal function approach, we can represent the waveform s(t) as exact linear combinations of $\{\varphi_j(t)\}$ by

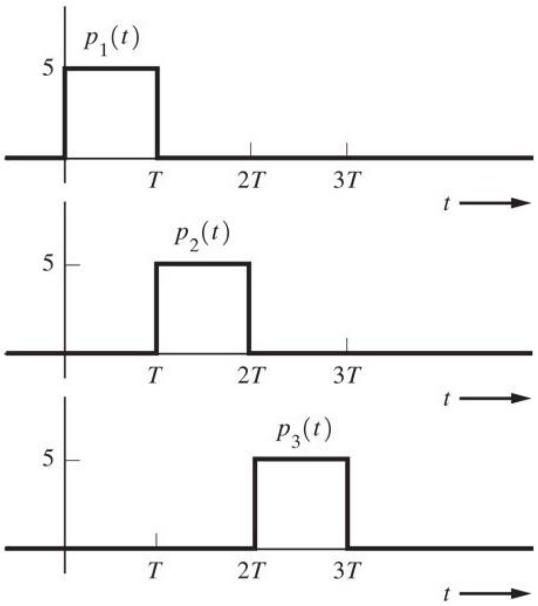
$$s(t) = \sum_{j=1}^{N=3} s_j \varphi_j$$

where $\{\varphi_j(t)\}$be the corresponding set of *orthogonal basis* (normalized).

$$\varphi_j(t) = \frac{p_j(t)}{\sqrt{K_j}} = \frac{p_j(t)}{\sqrt{\int_{-\infty}^{\infty} p_j^2\,(t)\,dt}} = \frac{p_j(t)}{\sqrt{\int_0^{T_0} p_j^2\,(t)\,dt}} = \frac{p_j(t)}{5\sqrt{T}}$$

- We can find the series as

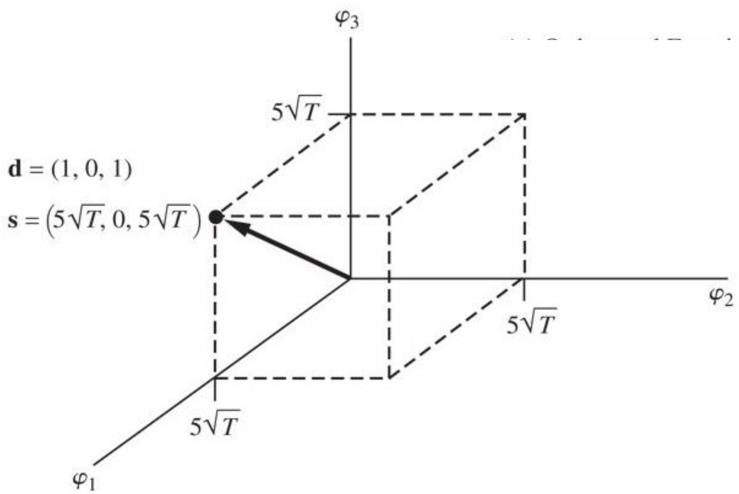
$$\bar{s} = (s_1, s_2, s_3) = (5\sqrt{T}, 0, 5\sqrt{T})$$



(c) Orthogonal Function Set

Example – Vector representation of a binary signal - 3

- The vector representation for s(t) is shown in Figure (d).
- Note that for this N=3 – dimensional case with binary signalling, only $2^3=8$ different waveforms could be represented (M=8).
- Each waveform corresponds to a vector that terminates on a vertex of a cube.



(d) Vector Representation of the 3-Bit Signal

Matlab code (vector_represnetation. m) for the waveform generated by the orthonormal series as Figure (a).

Representation for a 3-bit binary digital signal

Bandwidth estimation

- The symbol rate (baud) is defined as:

$$D = {}^{N}/_{T_0} \text{ symbols/second}$$

where N is the number of dimensions used in T_0

- The bit rate is defined as:

$$R = {}^{n}/_{T_0} \text{ bits/second}$$

where n is the number of bits sent in T_0, for binary, n=N

- The lower bound for the bandwidth of the waveform representing the digital signal can be obtained:

$$B \geq \frac{N}{2T_0} = \frac{1}{2} D\ Hz \qquad (3\text{-}4)$$

- If $\varphi_k(t)$ are of the sin(x)/x type, the lower bound absolute bandwidth of $\frac{N}{2T_0} = \frac{D}{2}$ will be achieved.

- For other pulse shapes, the bandwidth will be larger than this lower bound.

Class exercise - 1

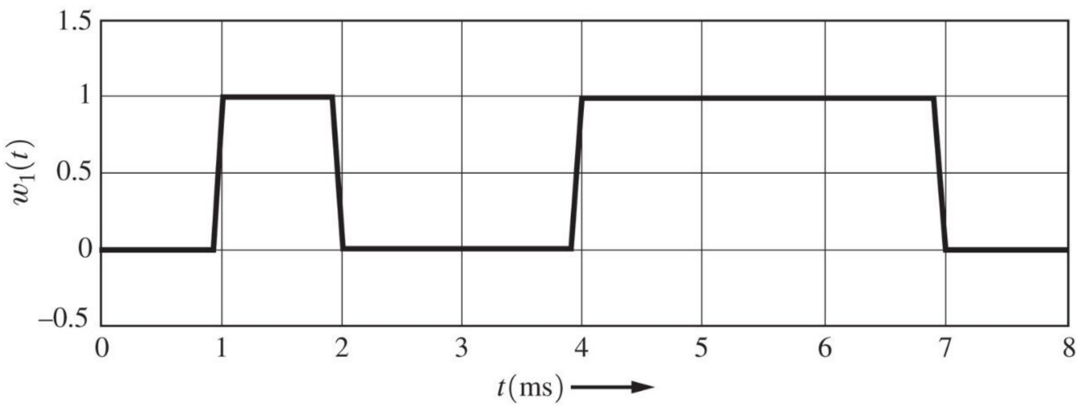
- Let us examine some properties of binary signalling from a digital source that can produce M=256 distinct message. Each message could be represented by n=8-bit binary words. Assume that it takes T_0=8ms to transmit one message and that a particular message corresponding to the code word 01001110 is to be transmitted. Then, $w_1 = 0, w_2 = 1, w_3 = 0, w_4 = 0, w_5 = 1, w_6 = 1, w_7 = 1, w_8 = 0$

- **Case 1: Rectangular pulse orthogonal functions**
- **Case 2: Sin(x)/x pulse orthogonal functions**

Class exercise - 1

Case 1: Rectangular pulse orthogonal functions

Assume that the orthogonal functions $\varphi_k(t)$ are given by unity-amplitude rectangular pulses that are $T_b = \frac{T_0}{n} = \frac{8}{8} = 1ms$ wide, where T_b is the time that it takes to send 1 bit of data. When rectangular pulse shape is used as shown in Figure (a), the digital source information is transmitted via a binary digital waveform. That is, the digital signal is a digital waveform.



(a) Rectangular Pulse Shape, $T_b = 1$ ms

Class exercise - 1

- The bit rate and the baud (symbol rate) of the binary signal are $R = \frac{n}{T_0} = \frac{8}{8} = 1\ kbit/s$, and $D = \frac{N}{T_0} = 1\ kbaud$, because N=n=8 and $T_0 = 8\ ms$. That is, the bit rate and the symbol rate are equal for binary signalling.
- **What is the bandwidth for the waveform in Figure (a)?**
- Using equation (3-4), we find that the lower bound for the bandwidth is $B = \frac{1}{2}D = 500\ Hz$. In the later this week, it will be shown that the actual null bandwidth of this binary signal with rectangular pulse shape is $B = \frac{1}{T_s} = D = 1000Hz$. This is larger than the lower bound for the bandwidth, so what is the waveshape that gives the lower bound bandwidth of 500 Hz?

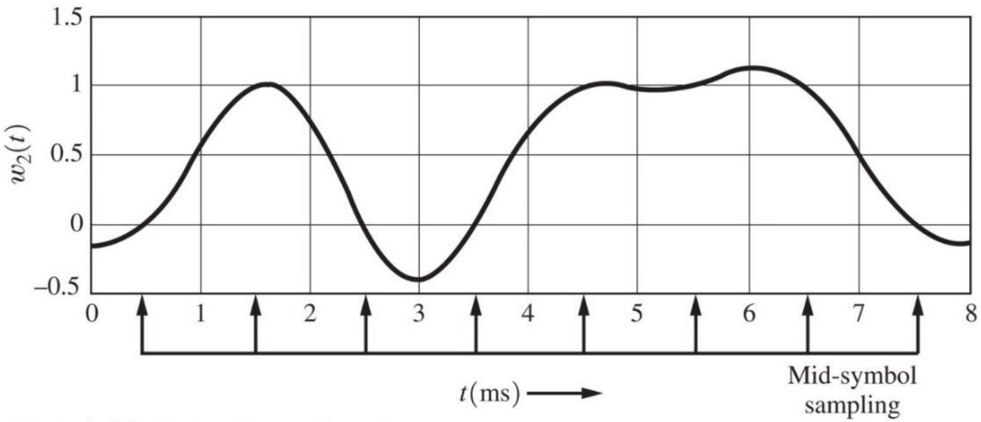
The answer is one with sin(x)/x pulses.

Class exercise - 1

Case 2: Sin(x)/x pulse orthogonal functions

From an intuitive viewpoint, we know that the sharp corners of the rectangular pulse need to be rounded to reduce the bandwidth of the wave. The resulting waveform that is transmitted is shown in Figure (b).

The data can be detected at the receiver by evaluating the orthogonal series coefficients as sin(x)/x orthogonal functions are used. This can be achieved by sampling the received waveform at the midpoint of each symbol interval as shown in Figure (b).



(b) $\sin(x)/x$ Pulse Shape, T_b = 1 ms

Matlab code (Class_exercise_1.m) available for Figure (b).

Class exercise - 1

Case 2: Sin(x)/x pulse orthogonal functions

The bit rate and the symbol rate are still $R = 1\ kbit/s$ and D= 1 kbaud. The absolute bandwidth can be evaluated by $B = \frac{1}{2T_s} = 500\ Hz$. Thus, the lower-bound bandwidth is achieved.

Note:

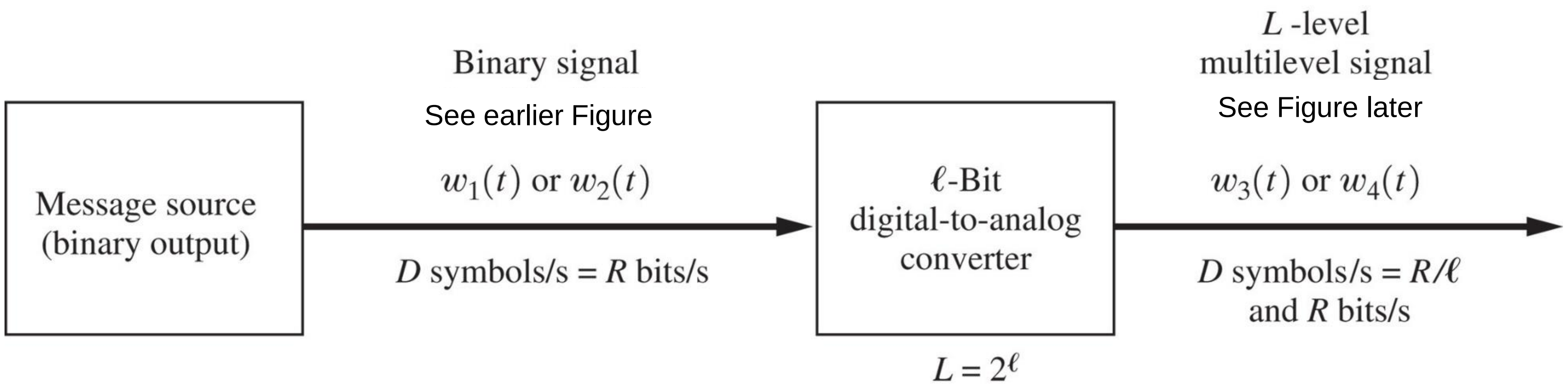
- When rectangular pulse shape is used as shown in Figure (a), the digital source information is transmitted via a binary digital waveform. That is, the digital signal is a digital waveform.
- When the sin(x)/x pulse shape is used as shown in Figure (b), the digital source information is transmitted via an analog waveform (i.e. an infinite number of voltage values ranging between -0.4 and 1.2 V are used.)

Multilevel signalling

- In the case of binary signalling discussed in the class exercise 1, the lower-bound bandwidth of $B = \frac{N}{2T_0}$, was achieved. That is, for Case 2, N=8 pulses were required and give a bandwidth of 500 Hz, for a message duration of $T_0 = 8\ ms$.
- However, this bandwidth could be made smaller if N could be reduced.
- N, (and consequently, the bandwidth) can be reduced by letter w_k's of equation (3-1) take on $L > 2$ possible values (instead of just the two possible values that were used for binary signalling).
- When the w_k's have $L > 2$ possible values, the resulting waveform obtained from equation (3-1) is called a **multilevel signal**.

Class exercise - 2

- Here, the M=256-msessage source of Class exercise 1 will be encoded into an L=4 multilevel signal, and the message will be sent in $T_0 = 8\ ms$. Multilevel data can be obtained by encoding the $l - bit$ binary data of the message source into L-level data by using a digital-to-analog converter (DAC) as shown in Figure below.



Binary-to-multilevel signal conversion.

Class exercise - 2

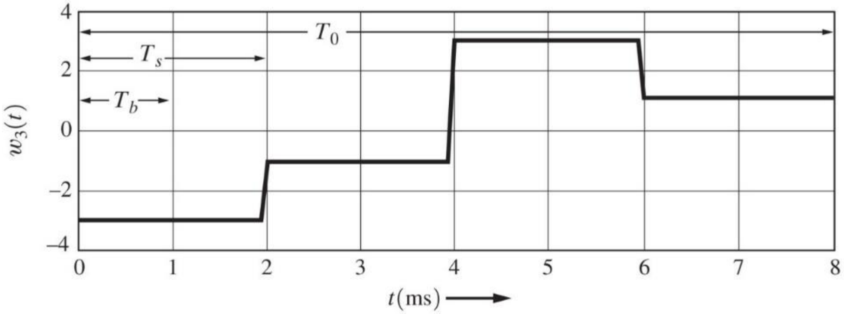
- For example, one possible encoding scheme for an $l = 2 - bit$ DAC is shown in Table below l =2, bits are read in at a time to produce an output that is one of L=4 possible levels, which $L=2^{l}$.

Binary Input ($\ell = 2$ bits)	Output Level (V)
11	+3
10	+1
00	−1
01	−3

Table 3-1 A 2-BIT DIGITAL-TO-ANALOG (DAC) CONVERTER

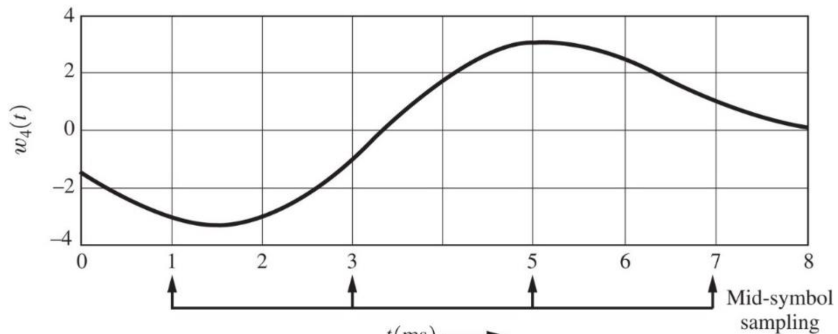
Class exercise - 2

- For the binary code word 01001110, the sequence of four-level outputs world be -3, -1, +3 and +1.
- The w_k's of equation (3-1) would be $w_1 = -3, w_2 = -1, w_3 = +3,$ $and\ w_4 = +1$, where N=4 dimensions are used.
- The corresponding L=4-level waveform is shown in Figure below.
- Figure (a) gives the multilevel waveform when rectangular pulses are used.
- Figure (b) gives the multilevel wave from when sin(x)/x waveform are used.



(a) Rectangular Pulse Shape $T_b = 1$ ms

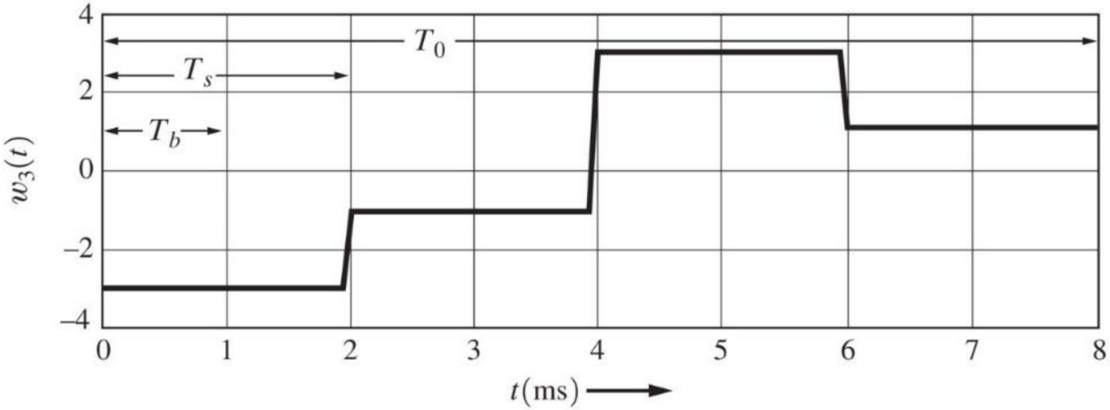
L = 4-level signaling



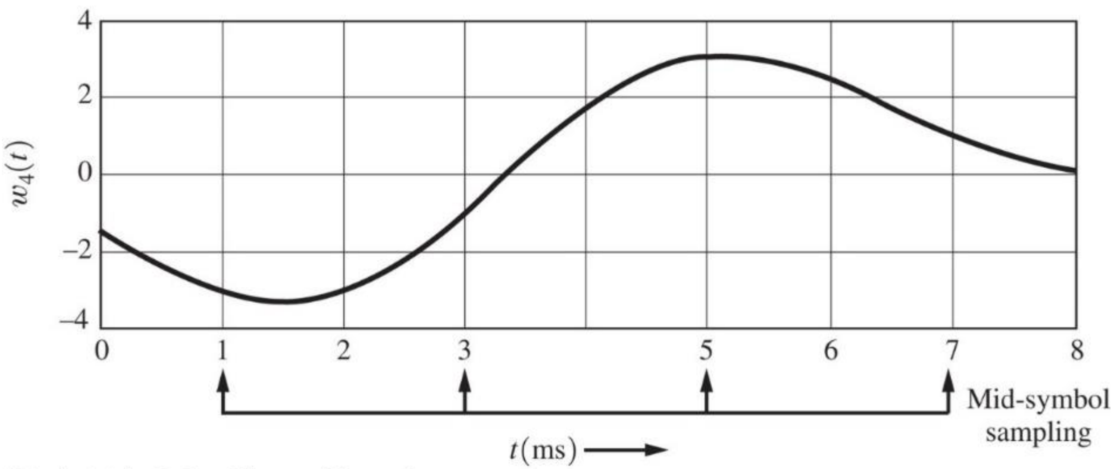
(b) sin(x)/x Pulse Shape, $T_b = 1$ ms

Matlab code (Class_exercise_2.m)

Class exercise - 2



(a) Rectangular Pulse Shape $T_b = 1$ ms

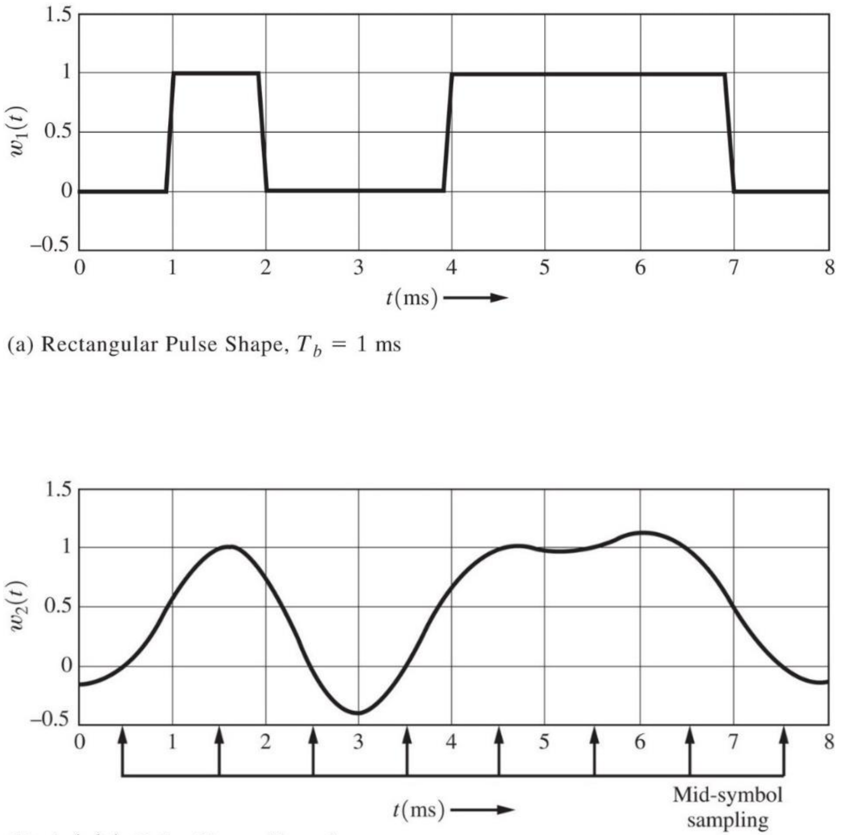


(b) $\sin(x)/x$ Pulse Shape, $T_b = 1$ ms

Class exercise - 2

- For these L=4-level signals, the equivalent bit interval is $T_b = 1$ ms because each symbol carries $l = 2\ bits$ of data, e.g. one of L=4-levels as shown in Table 3-1.
- The bit rate is $R = \frac{n}{T_0} = \frac{l}{T_s} = 1\ kbit/s$, which is same as class exercise 1.
- The baud (symbol) rate is $D = \frac{N}{T_0} = \frac{1}{T_s} = 0.5\ kbaud$, which is different from that of class exercise 1. The bite rate and the symbol rate are related by where $l = log_2(L)$ is the number of bits read in by the DAC on each clock cycle.
- The null bandwidth of the rectangular-pulse multilevel wave in Figure (a) is $B = \frac{1}{T_s} = D = 500\ Hz$
- The absolute bandwidth of the sin(x)/x-pulse multilevel waveform in Figure (b) is $B = \frac{N}{2T_0} = \frac{1}{2T_s} = \frac{D}{2} = 250\ Hz$.

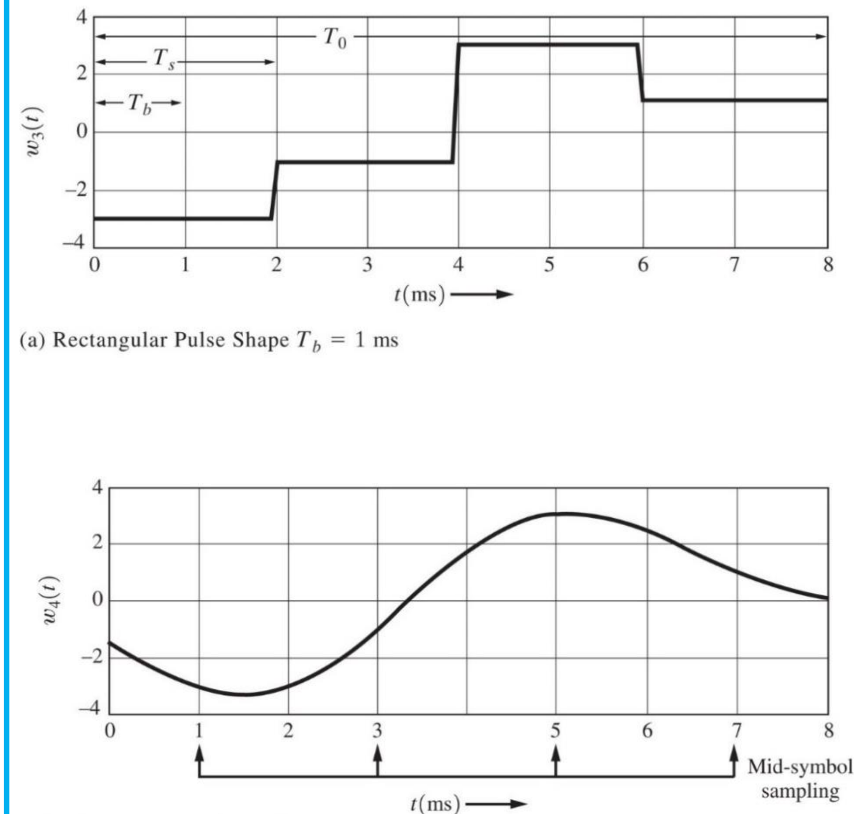
Comparison between class exercise 1 and 2



(a) Rectangular Pulse Shape, $T_b = 1$ ms

(b) $\sin(x)/x$ Pulse Shape, $T_b = 1$ ms

class exercise 1



(a) Rectangular Pulse Shape $T_b = 1$ ms

(b) $\sin(x)/x$ Pulse Shape, $T_b = 1$ ms

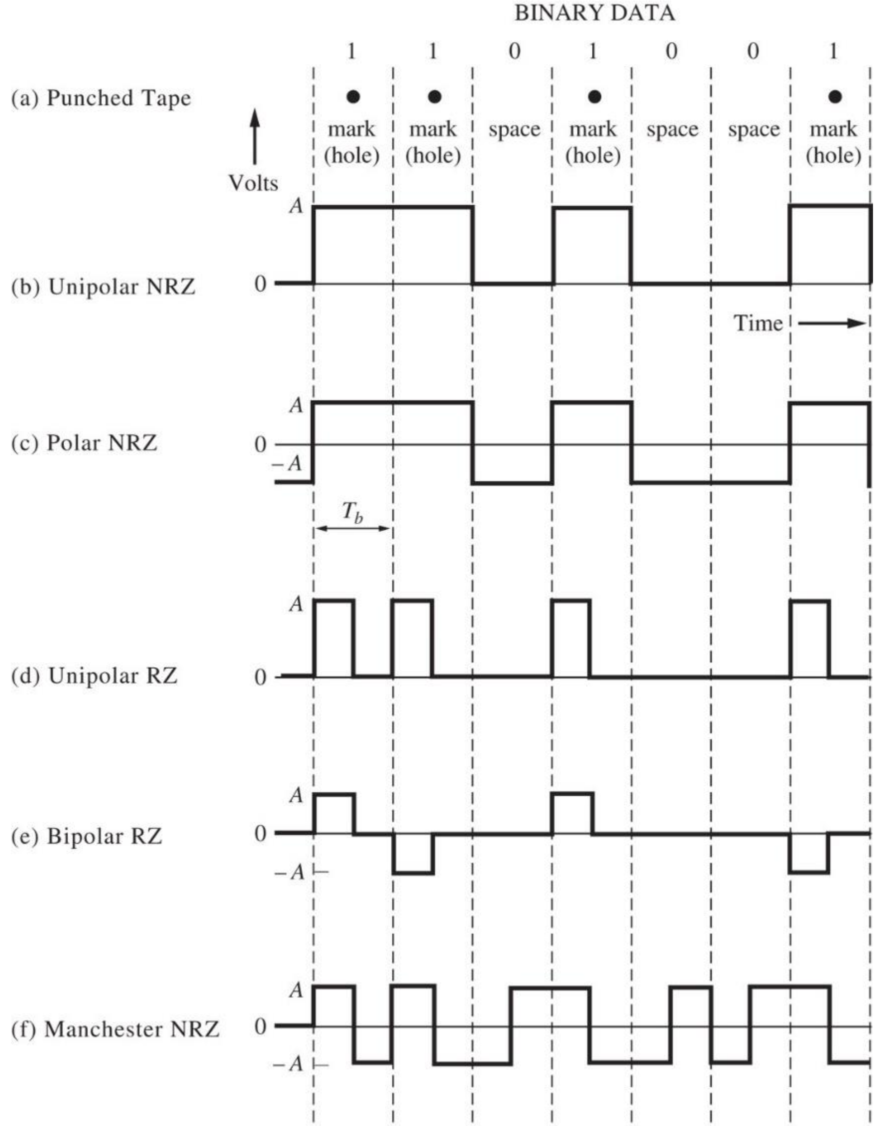
class exercise 2

Line coding schemes

- Unipolar
  - All signal levels are organised on one side of the time axis.
- Polar
  - The signal voltage levels are organised on the both sides of the time axis.
- Bipolar
  - Uses three levels: positive, zero and negative. The voltage level for one data element is at zero while other element alternates between positive and negative.

Return-to-zero (RZ), and no return-to-zero (NRZ)

Binary signalling formats



Advantage and Disadvantage of line codes

- Each of the line codes has advantages and disadvantages:
  - The **unipolar NRZ** line code has the advantage of using circuits that required only one power supply, e.g. a signal 5V power supply for the circuit, but it has the disadvantage of requiring channels that are DC coupled, e.g. with frequency response down to f=0, because the waveform has a nonzero DC value.
  - The **polar NRZ** line code does not required a DC-control channel, provided that the data toggles between binary 1's and 0's often and that equal numbers of binary 1's and 0's are sent. However, the circuitry that produces the polar NRZ signal requires a negative voltage power supply as well as a positive voltage power supply.
  - The **Manchester NRZ** line code has the advantage of always having a 0-DC value, regardless of the data sequence, but it has twice the bandwidth of the unipolar NRZ or polar NRZ code because the pulses are half the width.

Summary of desirable properties of a line code

- Self-synchronization.
- Low probability of bit error.
- A spectrum that is suitable for the channel
- Transmission bandwidth
- Error detection capability.
- Transparency

Power spectra for binary line codes

- The power spectrum density (PSD) can be evaluated by using either a deterministic or a stochastic technique.
- A digital signal (or line code) can be represented by

$$s(t) = \sum_{n=-\infty}^{n=+\infty} a_n f(t - nT_s) \qquad (3\text{-}5)$$

Where f(t) is the symbol pulse shape
Ts is the duration of one symbol.

- For binary signalling, $T_s = T_b$, where T_b is the time that it takes to send 1 bit.
- For multilevel signalling, $T_s = lT_b$
- The set $\{a_n\}$ is the set of random data. For example, for the unipolar NRZ line code, $f(t) = Rect\left(\frac{t}{T_b}\right)$ and $a_n = +A$ V when a binary 1 is sent, and $a_n = 0$ V when a binary 0 is sent.

General expression for the PSD of a digital signal

- The general expression for the PSD of a digital signal can be expressed by:

$$P(f) = \frac{|F(f)|^2}{T_s} \sum_{k=-\infty}^{\infty} R(k) e^{j2\pi k f T_s} \qquad \text{(3-6a)}$$

where F(f) is the Fourier transform of the pulse shape f(t).
R(k) is the autocorrelation of the data. The autocorrelation is given by

$$\mathrm{R}(k) = \sum_{i=1}^{l} (a_n a_{n+k})_i \, P_i \qquad \text{(3-6b)}$$

where a_n and a_{n+k} are the (voltage) levels of the data pulses at the nth and (n+k)th symbol positions, respectively. Pi is the probability of having the ith $a_n a_{n+k}$ product.

- Note that equation (3-6a) shows that the spectrum of the digital signal depends on two things:
  - The pulse shape used
  - Statistical properties of the data

Unipolar NRZ signalling - 1

- For unipolar signalling, the possible levels for the a's are +A and 0 V.
- Assume that these values are equally likely to occur and that data are independent.
- Now, evaluate R(k) as defined by equation (3-6b):
- For k=0, the possible products of $a_n a_n$ are $A \times A = A^2$ and $0 \times 0 = 0$, and consequently, $I = 2$. For random data, the probability of having A^2 is 50%, and the probability of having 0 is 50%, so that

$$\mathrm{R}(0) = \sum_{i=1}^{2} (a_n a_n)_i \, P_i = A^2 \cdot \frac{1}{2} + 0 \cdot \frac{1}{2} = \frac{1}{2} A^2$$

- For $k \neq 0$, there are I=4 possibilities for the product values: $A \times A, A \times 0, and\ 0 \times A, 0 \times 0$. They all occur with a probability of ¼. Thus,

$$\mathrm{R}(k) = \sum_{i=1}^{4} (a_n a_{n+k})_i \, P_i = A^2 \cdot \frac{1}{4} + 0 \cdot \frac{1}{4} + 0 \cdot \frac{1}{4} + 0 \cdot \frac{1}{4} = \frac{1}{4} A^2$$

- Overall:

$$\mathrm{R}_{unipolar}(k) = \begin{cases} \frac{1}{2} A^2 & k = 0 \\ \frac{1}{4} A^2 & k \neq 0 \end{cases} \qquad (3\text{-}7)$$

Unipolar NRZ signalling - 2

- For rectangular NRZ pulse shape, the Fourier transform pair is

$$f(t) = Rect\left(\frac{t}{T_b}\right) \leftrightarrow F(f) = T_b \frac{\sin \pi f(T_b)}{\pi f T_b} \tag{3-8}$$

- Using equation (3-6a) with $T_s = T_b$, we find that the PSD for the unipolar NRZ line code is

$$P_{unipolar\ NRZ}(f) = \frac{A^2 T_b}{4}\left(\frac{sin\pi f T_b}{\pi f T_b}\right)^2 \left[1 + \sum_{k=-\infty}^{\infty} e^{j2\pi k f T_b}\right]$$

Note: with a weight of ½.

where

$$\sum_{k=-\infty}^{\infty} e^{j2\pi k f T_b} = \frac{1}{T_b}\sum_{n=-\infty}^{\infty} \delta\left(f - \frac{n}{T_b}\right)$$

- Thus,

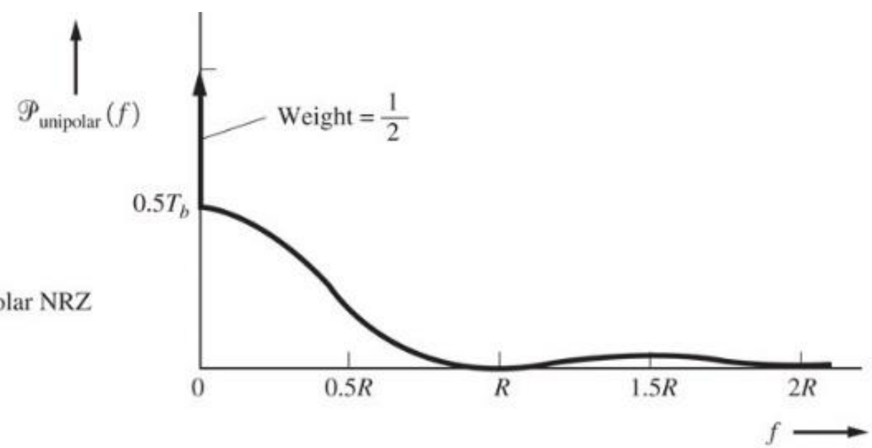
$$P_{unipolar\ NRZ}(f) = \frac{A^2 T_b}{4}\left(\frac{sin\pi f T_b}{\pi f T_b}\right)^2 \left[1 + \frac{1}{T_b}\sum_{n=-\infty}^{\infty} \delta\left(f - \frac{n}{T_b}\right)\right] \tag{3-9a}$$

Unipolar NRZ signalling - 3

- Because of $[\sin(\pi f T_b)/\pi f T_b] = 0$ at $f = n/T_b$ for $n \neq 0$, this reduces equation (3-9a) to

$$P_{unipolar\ NRZ}(f) = \frac{A^2 T_b}{4}\left(\frac{\sin \pi f T_b}{\pi f T_b}\right)^2 \left[1 + \frac{1}{T_b}\delta(f)\right] \qquad (3\text{-}9b)$$

- If A is selected so that the normalised average power of the unipolar NRZ signal is unity, then $A = \sqrt{2}$.
- The PSD is plotted in Figure (a), where $\frac{1}{T_b} = R$, the bit rate of the line code.



(a) Unipolar NRZ

Polar NRZ signalling - 1

- For polar NRZ signalling, the possible levels for a's are +A and –A V. For equally likely occurrences of +A and –A, and assuming that the data are independent from bit to bit, we get

$$\mathrm{R}(0) = \sum_{i=1}^{2} (a_n a_n)_i P_i = A^2 \cdot \frac{1}{2} + (-A)^2 \cdot \frac{1}{2} = A^2$$

- For $k \neq 0$,

$$\mathrm{R}(k) = \sum_{i=1}^{4} (a_n a_{n+k})_i P_i = A^2 \cdot \frac{1}{4} + (-A)(+A) \cdot \frac{1}{4} + (+A)(-A) \cdot \frac{1}{4} + (-A)^2 \cdot \frac{1}{4} = 0$$

- Thus,

$$\mathrm{R}_{polar}(k) = \begin{Bmatrix} A^2 & k = 0 \\ 0 & k \neq 0 \end{Bmatrix} \qquad (3\text{-}10)$$

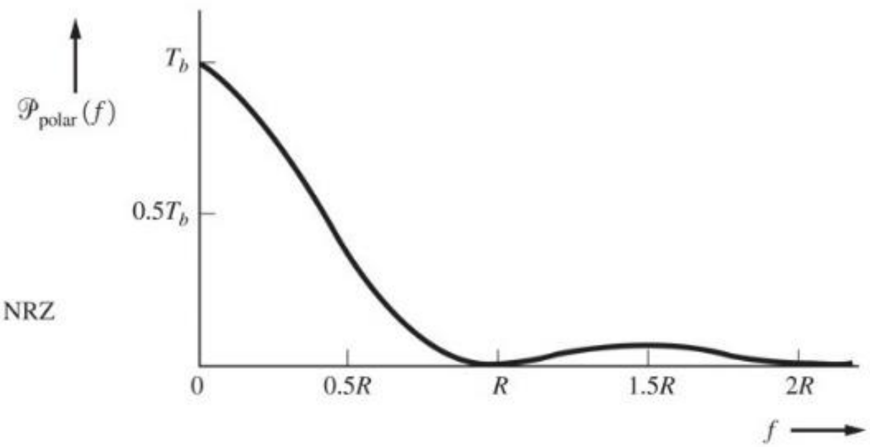
- Substitute equation (3-10) and (3-8) into (3-6a),

$$P_{polar\ NRZ}(f) = A^2 T_b \left(\frac{\sin \pi f T_b}{\pi f T_b} \right)^2 \qquad (3\text{-}11)$$

Queen Mary University of London

Polar NRZ signalling - 2

- If A is selected so that the normalised average power of the polar NRZ signal is unity, then A=1, and the resulting PSD is shown in Figure (b).
- The polar signal has the disadvantage of having a large PSD near DC.
- On the other hand, polar signals are relatively easy to generate, although positive and negative power supplies are required, unless special-purpose integrated circuits are used that generate dual supply voltage from a single supply.
- The probability of bit error performance is superior to that of other signalling methods.



(b) Polar NRZ

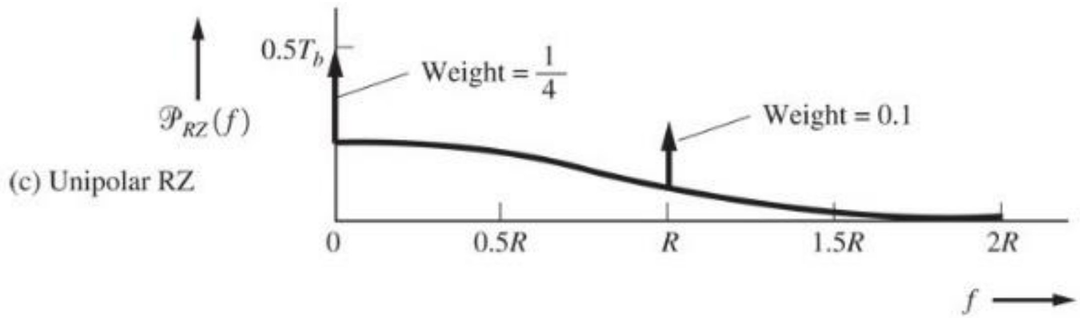
Unipolar RZ signalling

- The autocorrelation for unipolar data was calculated previously and is given by equation (3-7). For RZ signalling, the pulse duration is $T_b/2$ instead of T_b, as used in NRZ signalling. That is, for RZ,

$$F(f) = \frac{T_b}{2}\frac{\sin \pi f (T_b/2)}{\pi f T_b/2} \qquad (3\text{-}12)$$

- Referring to equation (3-6a) and (3-7), we get the PSD for the unipolar RZ line code:

$$P_{unipolar\ RZ}(f) = \frac{A^2 T_b}{16}\left(\frac{sin\pi f T_b/2}{\pi f T_b/2}\right)^2\left[1 + \frac{1}{T_b}\sum_{n=-\infty}^{\infty}\delta\left(f - \frac{n}{T_b}\right)\right] \qquad (3\text{-}13)$$



Bipolar RZ signalling - 1

- The PSD for a bipolar signal can also be obtained using equation (3-6a).
- The permitted values for a_n are +A, -A, and 0 where binary 1's are represented by alternation +A and –A values, a binary 0 is represented by $a_n = 0$.
- For $k = 0$, the products $a_n a_n$ are A^2 and 0, where each of these products occurs with a probability of 50%. Thus,

$$\mathrm{R}(0) = \frac{1}{2}A^2$$

- For $k = 1$, (the adjacent-bit case) and the data sequences (1,1), (1,0), (0,1) and (0,0), the possible $a_n a_{n+1}$ products are $-A^2$, 0, 0 and 0. Each of this sequence occurs with a probability of 25%. Thus,

$$\mathrm{R}(1) = \sum_{i=1}^{4} (a_n a_{n+k})_i \, P_i = -\frac{A^2}{4}$$

- For $k > 1$, the bits being considered are not adjacent, and the $a_n a_{n+1}$ products are $\mp A^2$, 0, 0 and 0. These occur with a probability of 12.5%, then,

$$\mathrm{R}(k > 1) = \sum_{i=1}^{8} (a_n a_{n+k})_i \, P_i = A^2 \frac{1}{8} - A^2 \frac{1}{8} = 0$$

Queen Mary University of London

Bipolar RZ signalling - 2

- Overall autocorrelation,

$$R_{bipolar}(k) = \begin{cases} \dfrac{A^2}{2} & k = 0 \\ -\dfrac{A^2}{4} & |k| = 1 \\ 0, & |k| > 1 \end{cases} \qquad (3\text{-}14)$$

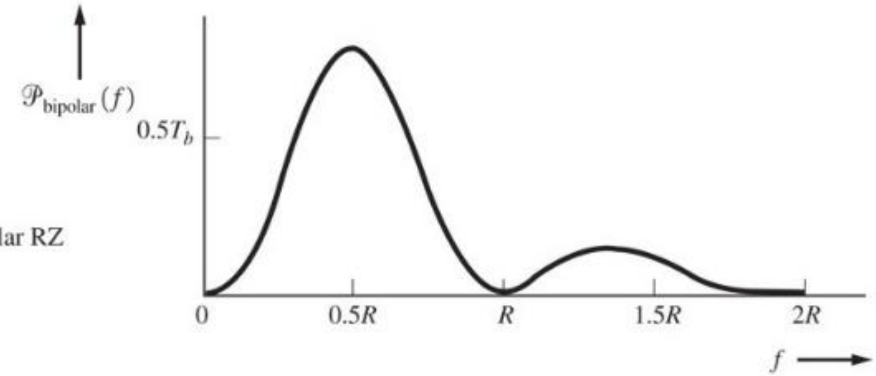
- Using equation (3-14) and (3-12) in equation (3-6a), where $T_s = T_b/2$, we find that the PSD for the bipolar RZ line code is

$$P_{bipolar\ RZ}(f) = \frac{A^2 T_b}{8}\left(\frac{sin\pi f T_b/2}{\pi f T_b/2}\right)^2 [1 - cos(2\pi f T_b)]$$

or

$$P_{bipolar\ RZ}(f) = \frac{A^2 T_b}{4}\left(\frac{sin\pi f T_b/2}{\pi f T_b/2}\right)^2 [sin^2(\pi f T_b)] \qquad (3\text{-}15)$$

where A=2 if the normalized average power is unity. PSD is plotted in Figure (d).



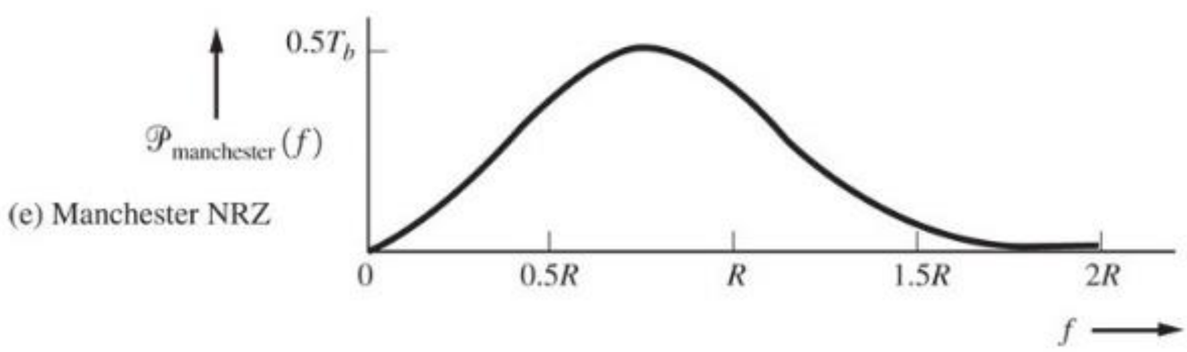
(d) Bipolar RZ

Matlab code (bipolar_RZ_line_code.m)
For R=1

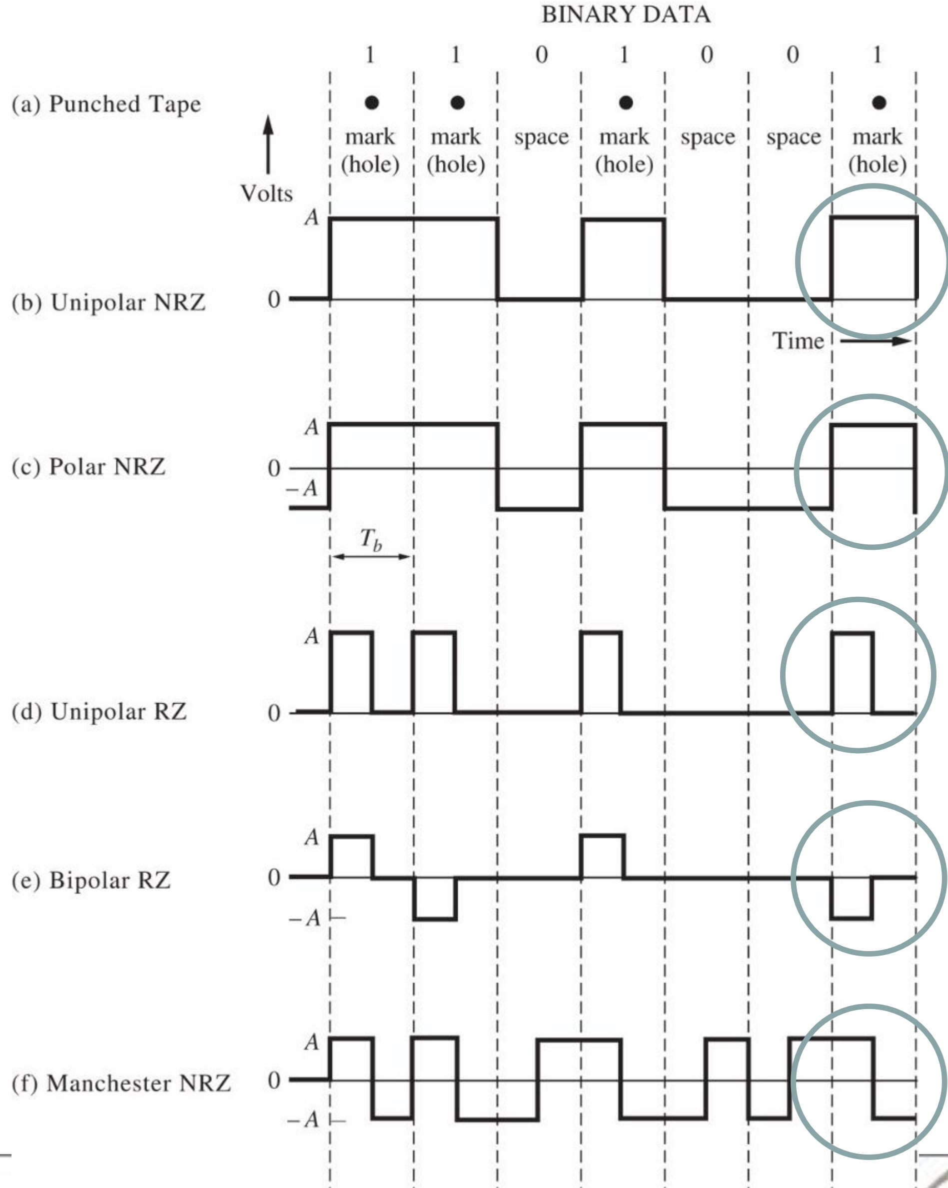
Manchester NRZ signalling

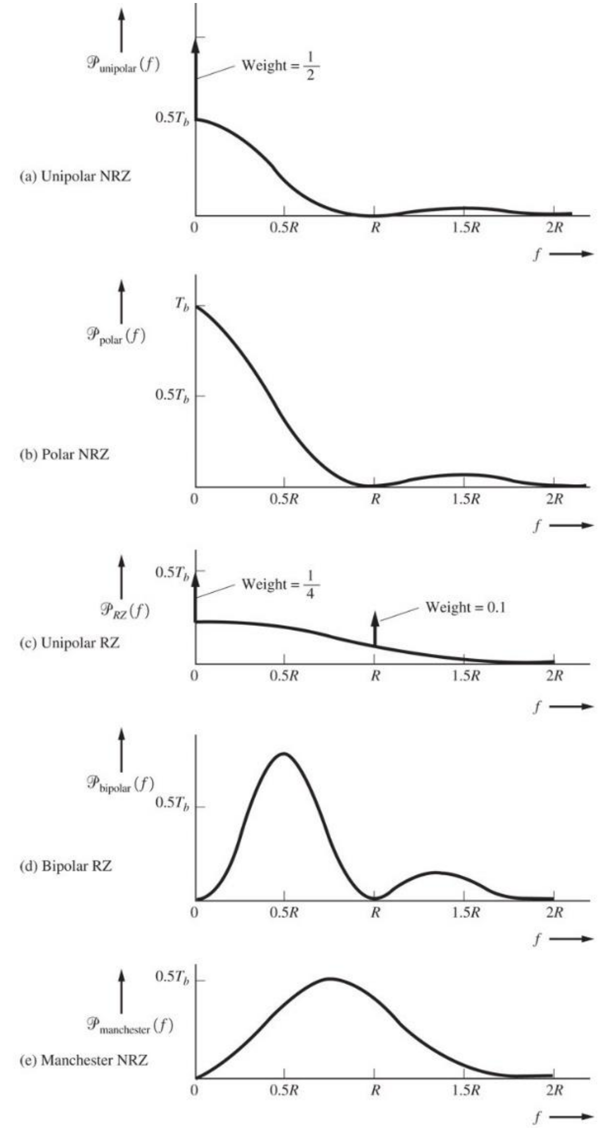
- The PSD for Manchester

$$P_{manchester\ NRZ}(f) = A^2 T_b \left(\frac{sin\pi f T_b/2}{\pi f T_b/2}\right)^2 [sin^2(\pi f T_b/2)] \qquad (3\text{-}16)$$



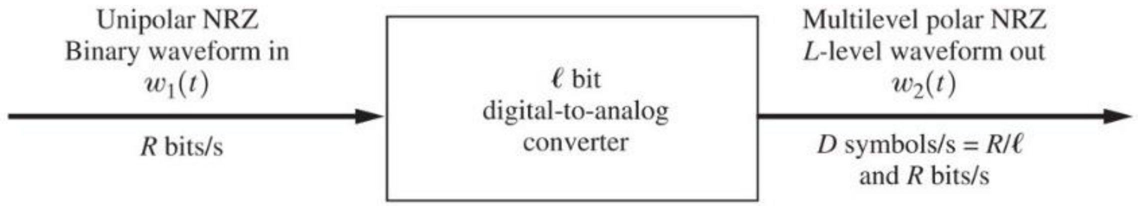
PSD for different line codes



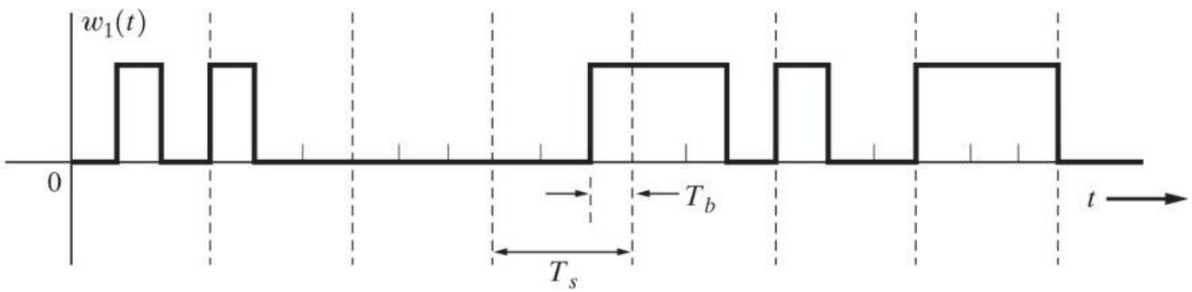


Power spectra for multilevel polar NRZ signal - 1

- Multilevel signalling provides reduced bandwidth compared with binary signalling.
- Figure (a) and (b) shows how a binary signal is converted into a multilevel polar NRZ signal, where an l-bit DAC is used to covert the binary with data rate R bits/s to an $L = 2^l$-level multilevel polar NRZ signal.
- $l = 3$-bit DAC is used, so that $L = 2^3 = 8$ levels.
- Figure (b) illustrate a typical input waveform.



(a) ℓ Bit Digital-to-Analog Converter

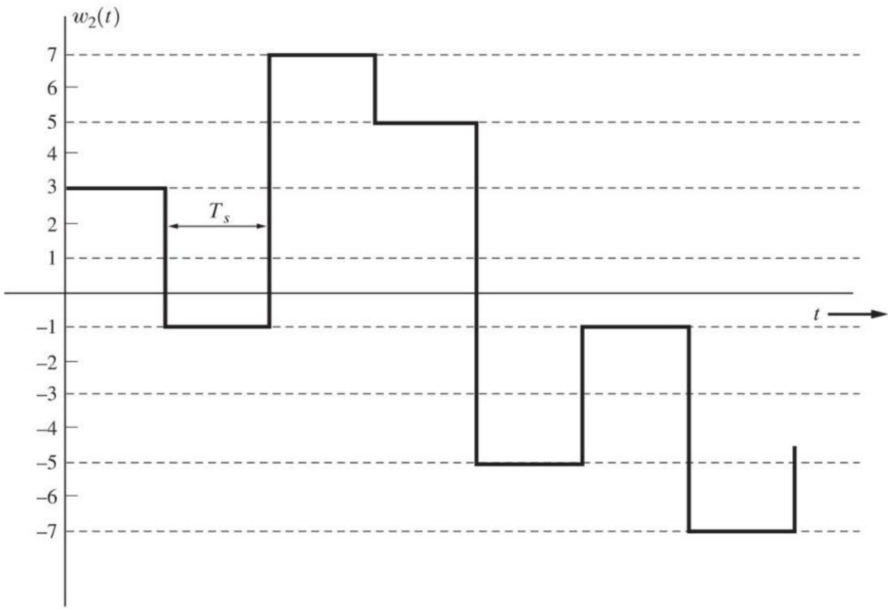


(b) Input Binary Waveform, $w_1(t)$

Binary-to-multilevel polar NRZ signal conversion

Power spectra for multilevel polar NRZ signal - 2

- Figure (c) shows the corresponding eight-level multilevel output waveform, where Ts is the time it takes to send on multilevel symbol.
- The code in Table 3-2 is used to obtain the waveform.
- $D = \frac{1}{T_s} = \frac{1}{3T_b} = R/3$ or in general $D = \frac{R}{l}$



(c) $L = 8 = 2^3$ Level Polar NRZ Waveform Out

Table 3-2 DAC code

Digital Word	Output Level, $(a_n)_i$
000	+7
001	+5
010	+3
011	+1
100	−1
101	−3
110	−5
111	−7

Power spectra for multilevel polar NRZ signal - 3

- The PSD for the multilevel polar NRZ wave from in Figure (c) can be obtained by the use of equation (3-6a).
- Evaluating $R(k)$ for the case of equally likely levels a_n as shown in Table 3-2, We have for the case of $k = 0$,

$$R(0) = \sum_{i=1}^{8} ({a_n}^2)_i P_i = 21$$

where $P_i = \frac{1}{8}$ for all the eight possible values. For $k \neq 0$, $R(k) = 0$. Then, from equation (3-6a), the PSD for $w_2(t)$ is

$$P_{w_2}(t) = \frac{|F(f)|^2}{T_s}(21 + 0)$$

Where the pulse width (or symbol width) is $T_s = 3T_b$. For the rectangular pulse shape of width $3T_b$, this becomes

$$P_{w_2}(t) = 63T_b \left(\frac{sin3\pi f T_b}{3\pi f T_b}\right)^2$$

Spectral efficiency - 1

- The spectral efficiency of a digital signal is given by the number of bits per second of data that can be supported by each hertz of bandwidth. That is,

$$\eta = \frac{R}{B} \ (bits/s)/Hz \qquad (3\text{-}17)$$

where R is the data rate and B is the bandwidth.

- The maximum possible spectral efficiency is limed by the channel noise if the error is to be small. It can be given by Shannon's channel capacity formula,

$$\eta_{max} = \frac{C}{B} = log_2\left(1 + \frac{S}{N}\right) \ (bits/s)/Hz \qquad (3\text{-}18)$$

- The spectral efficiency for multilevel polar NRZ signalling is obtained by substituting $B_{null} = R/l$ to equation (3-18), so that

$$\eta = l \ (bits/s)/Hz \qquad \text{(multilevel polar NRZ signalling)}$$

Matlab code for spectral_efficiency.m

Spectral efficiency - 2

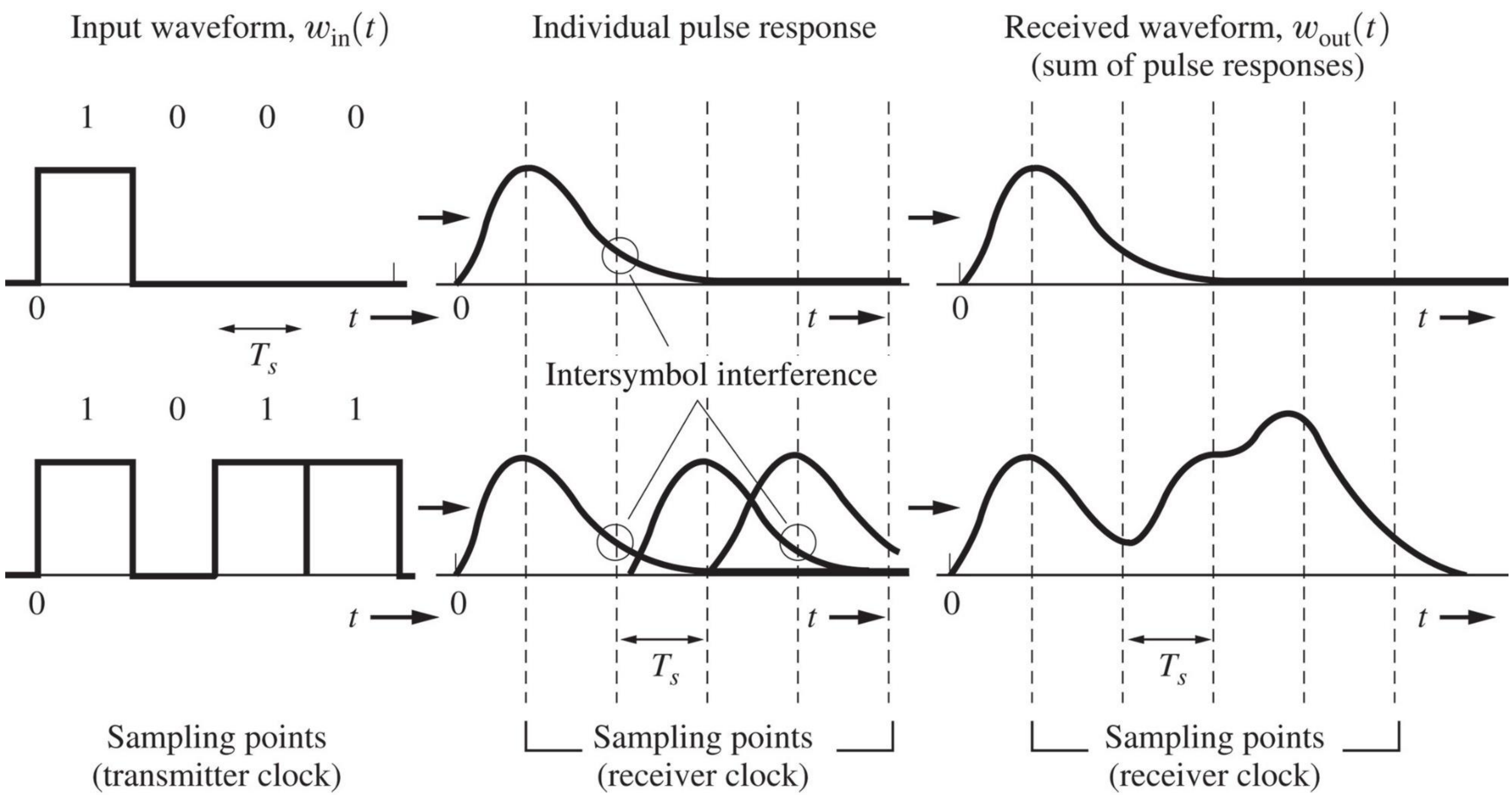
Table 3-3 spectral efficiencies of line codes

Code Type	First Null Bandwidth (Hz)	Spectral Efficiency $\eta = R/B$ [(bits/s)/Hz]
Unipolar NRZ	R	1
Polar NRZ	R	1
Unipolar RZ	$2R$	$\frac{1}{2}$
Bipolar RZ	R	1
Manchester NRZ	$2R$	$\frac{1}{2}$
Multilevel polar NRZ	R/ℓ	ℓ

Inter-symbol Interference (ISI)

Queen Mary
University of London

Baseband pulse-transmission system

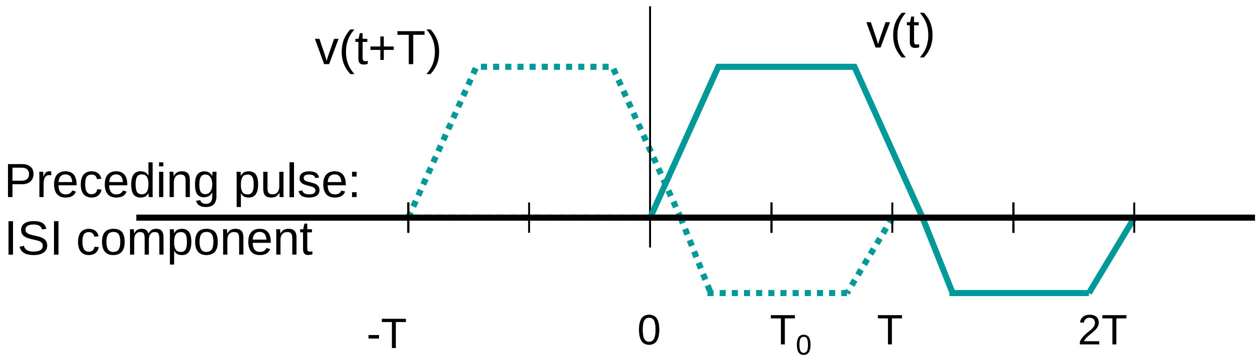


Inter-Symbol Interference (ISI)

- A signal is being filtered by a number of linear systems, and non-ideal system transfer function causing "smearing"
  - Filters and linear amplifiers in transmitter propagation medium
  - Filters and linear amplifiers in the front end of the receiver
  - Filters used in the demodulation process
- Electrical noise and interference produced by a variety of sources
  - Galaxy and atmospheric noise
  - Switching transients
  - Intermodulation noise
- $H(f) = H_t(f)H_c(f)H_r(f)$
  - $H_t(f)$: transmitting filter;
  - $H_c(f)$:Filtering within a channel;
  - $H_r(f)$: receiving filter

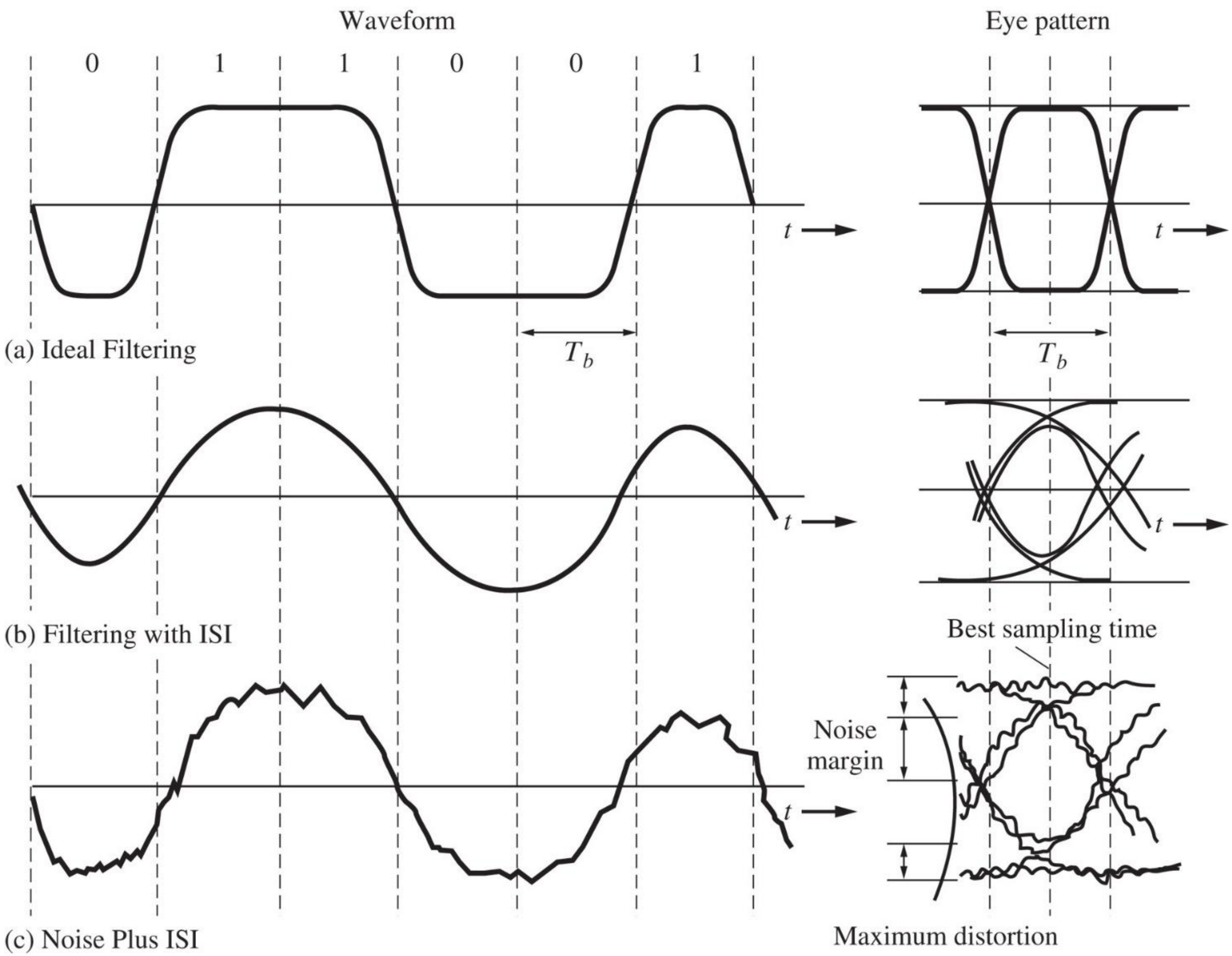
Eye diagram

- Use eye diagram to make an engineering judgement on the likely performance and source of degradation in a data communications link
- It is used to determine the peak distortion and other measures of performance in a system
- Consider a very simple pulse response v(t)



Use sampling time T_0 to obtain the decision statistic for the 0^{th} pulse

Distorted polar NRZ waveform and corresponding eye pattern



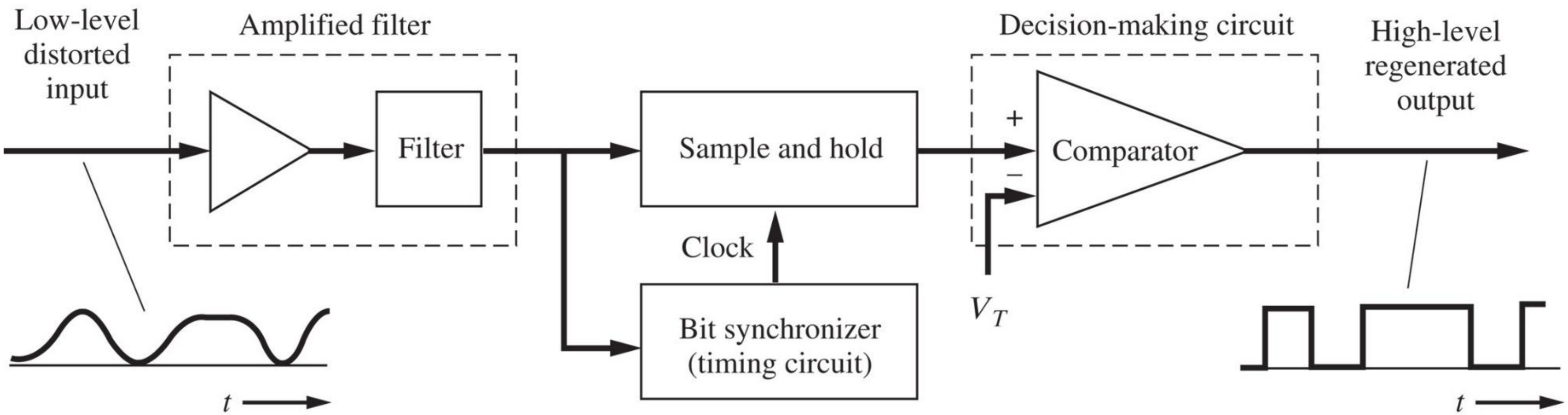
The eye patterns provide the following information

- The *timing error* allowed on the sampler at the receiver is given by the width inside the eye, called the eye opening. Of course, the preferred time for sampling is at the point where the vertical opening of the eye is largest.

- The sensitivity to timing error is given by the slope of the open eye (evaluated at, or near, the zero-crossing point).

- The noise margin of the system is given by the height of the eye opening

Regenerative repeater for unipolar NRZ signalling



Synchronisation Signals

- Synchronisation signals are clock-type signals that are necessary within a receiver (or repeater) for detection (or regeneration) of the data from the corrupted input signal.
- Digital communications usually need at least three types of synchronisation signals:
  - Bit sync, to distinguish one bit interval form another;
  - Frame sync, to distinguish groups of data;
  - Carrier sync, for bandpass signalling with coherent detection.
- Systems are designed so that the sync is derived either directly from the corrupted signal or from a separate channel that is used only to transmit the sync information.

ISI

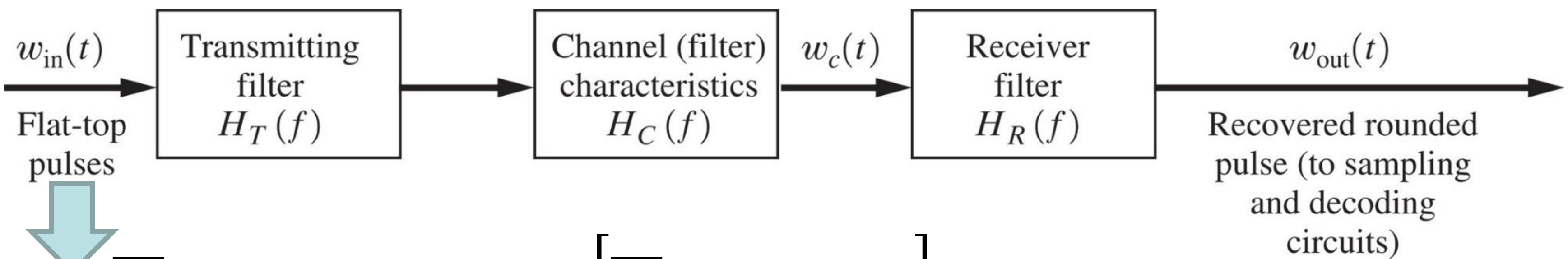
- ISI can be reduced by increasing the channel bandwidth.
- Or
- Reshape the pulse from the transmitter to minimise the ISI

Baseband pulse-transmission system - 1

- Consider a digital signalling system as shown in Figure below, in which the flat-topped multilevel signal at the input is

$$w_{in}(t) = \sum_n a_n h(t - nT_s) \tag{3-18}$$

where $h(t) = Rect\ (t/T_s)$, and a_n may take on any of the allowed L multilevel (L=1 for binary signalling). The symbol rate is $D = 1/T_s$ pulses/s.

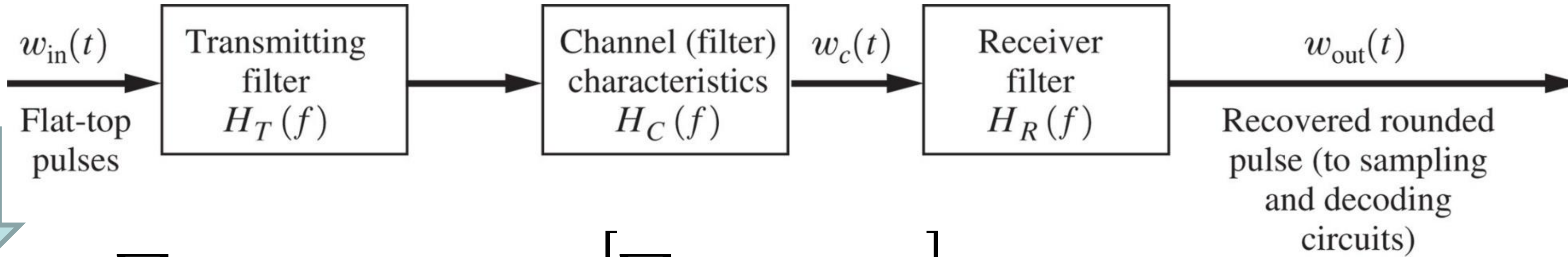


$$w_{in}(t) = \sum_n a_n h(t) * \delta(t - nT_s) = \left[\sum_n a_n \delta(t - nT_s)\right] * h(t) \tag{3-19}$$

The output of the linear system would be just the input impulse train convolved with the equivalent impulse response of the overall system.

$$w_{out}(t) = \left[\sum_n a_n \delta(t - nT_s)\right] * h_e(t) \tag{3-20}$$

Baseband pulse-transmission system - 2



$$w_{in}(t) = \sum_n a_n h(t) * \delta(t - nT_s) = \left[\sum_n a_n \delta(t - nT_s)\right] * h(t)$$

$$w_{out}(t) = \left[\sum_n a_n \delta(t - nT_s)\right] * h_e(t)$$

$h_e(t)$ is also the pulse shape that will appear at the output of the receiver filter when a single flat-top pulse is fed into the transmitter filter.

$$h_e(t) = h(t) * h_T(t) * h_C(t) * h_R(t) \qquad (3\text{-}21)$$

- The equivalent system transfer function is

$$H_e(f) = H(f)H_T(f)H_C(f)H_R(f) \qquad (3\text{-}22)$$

$$H(f) = F\left[Rect\left(\frac{t}{T_s}\right)\right] = T_s\left(\frac{\sin \pi T_s f}{\pi T_s f}\right) \qquad (3\text{-}23)$$

- The receiving filter is given by

$$H_R(f) = \frac{H_e(f)}{H(f)H_T(f)H_C(f)} \qquad (3\text{-}24)$$

$H_e(f)$ is the overall filtering characteristic

Baseband pulse-transmission system - 3

- When $H_e(f)$ is chosen to minimize the ISI, $H_R(f)$, obtained from equation (3-24) is called an ***equalising filter***.
- The equalising filter characteristic depends on $H_C(f)$, the channel frequency response, as well as on the required $H_e(f)$.
- The output of the receiving filter is

$$w_{out}(t) = \sum_n a_n h_e(t - nT_s) \qquad (3\text{-}25)$$

- The output pulse shape is affected by
  - the input pulse shape (flat-topped in this case)
  - The transmitter filter
  - The channel filter
  - The receiving filter
- In practice, the channel filter is already specified, the problem is to determine the transmitting filter and the receiving filter that will minimize the ISI on the rounded pulse at the output of the receiving filter.

Nyquist's First Method (Zero ISI) -1

- Nyquist's First Method for eliminating ISI is to use an equivalent transfer function, such that the impulse response stratifies the condition:

$$h_e(kT_s + \tau) = \begin{cases} C, & k = 0 \\ 0, & k \neq 0 \end{cases} \qquad (3\text{-}26)$$

where k is an integer,

T_s is the symbol (sample) clocking period,

τ is the offset in the receiver sampling clock times compared with the clock times of the input symbols,

C is a nonzero constant.

- Hence, for a single flat-top pulse of level a present at the input to the transmitting filter at $t = 0$, the received pulse would be $ah_e(t)$. It would have a value of aC at $t = \tau$ but would not cause interference at other sampling times because $h_e(kT_s + \tau)$ for $k \neq 0$.

Nyquist's First Method (Zero ISI) -2

- Suppose that we choose a $\sin(x)/x$ function for $h_e(t)$. In particular let $\tau = 0$, and choose

$$h_e(t) = \frac{sin\pi f_s t}{\pi f_s t} \qquad (3\text{-}27)$$

where $f_s = 1/T_s$. This impulse response satisfies Nyquist first criterion for zero ISI in equation (3-26).

- If the transmit and receive filters are designed so that overall transfer function is

$$H_e(f) = \frac{1}{f_s} Rect\left(\frac{f}{f_s}\right) \qquad (3\text{-}28)$$

- Then, there will be no ISI.
- However, $\sin(x)/x$ type of overall pulse shape has two practical difficulties:
  - The overall amplitude transfer characteristic $H_e(f)$ has to be flat over the whole bandwidth and zero elsewhere.
  - The synchronisation of the clock in the decoding sampling circuit has to be almost perfect, since the $\sin(x)/x$ pulse decays only as $1/x$ and is zero in adjacent time slots only when t is at the exactly correct sampling time. Thus, inaccurate syc will cause ISI.

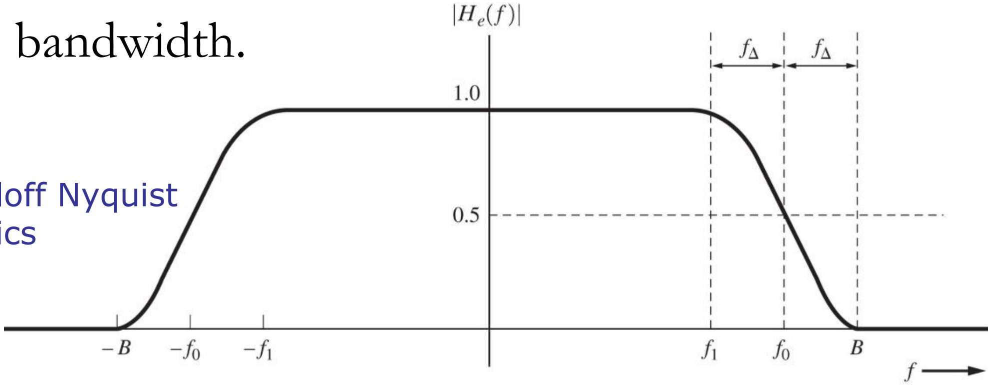
Raised cosine-rolloff Nyquist filtering

- The raised cosine-rolloff Nyquist filter has the transfer function

$$H_e(f) = \begin{cases} 1, & |f| < f_1 \\ \frac{1}{2}\left\{1 + cos\left[\frac{\pi(|f| - f_1)}{2f_\Delta}\right]\right\}, & f_1 < |f| > B \\ 0 & |f| > B \end{cases} \qquad (3\text{-}29)$$

where B is the absolute bandwidth.

Raised cosine-rolloff Nyquist filter characteristics



- This rolloff factor is defined to be

$$r = \frac{f_\Delta}{f_0} \qquad (3\text{-}30)$$

- The impulse response is $h_e(t) = F^{-1}[H_e(f)] = 2f_0\left(\frac{sin2\pi f_0 t}{2\pi f_0 t}\right)\left\{\frac{cos2\pi f_\Delta t}{1-(4f_\Delta t)^2}\right\}$

Frequency and time response for different rolloff factors

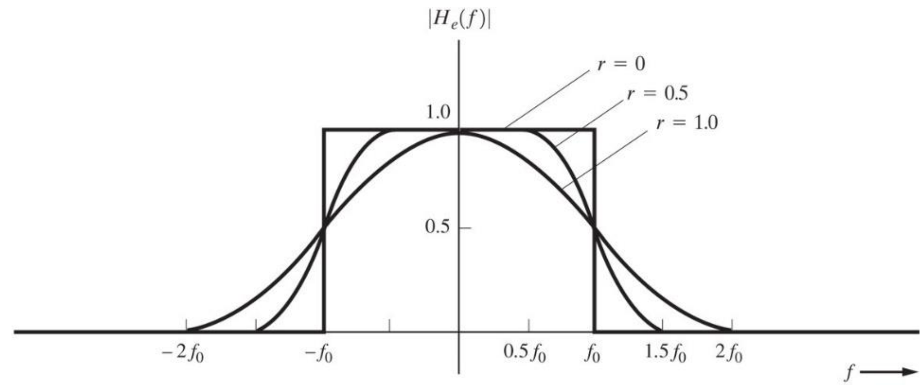
- Plots of the frequency response and impulse response for rolloff factors r=0, r=0.5 and r=1.

- The baud rate of the raised cosine-rolloff system with ISI free is

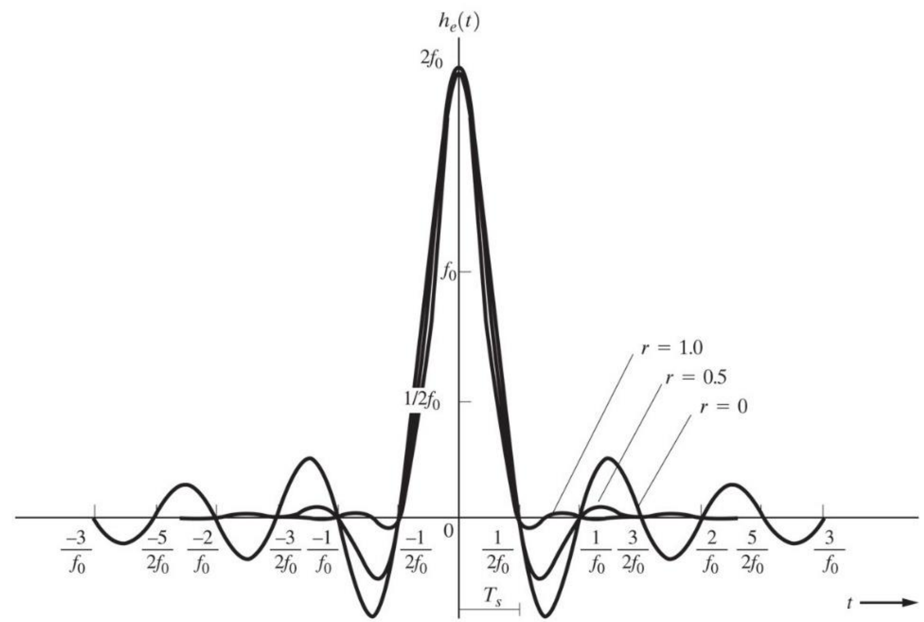
$$D = \frac{2B}{1+r}$$

Where B is the absolute bandwidth of the system and r is the system rolloff factor

Matlab code raised_cosine_rolloff.m



(a) Magnitude Frequency Response



(b) Impulse Response

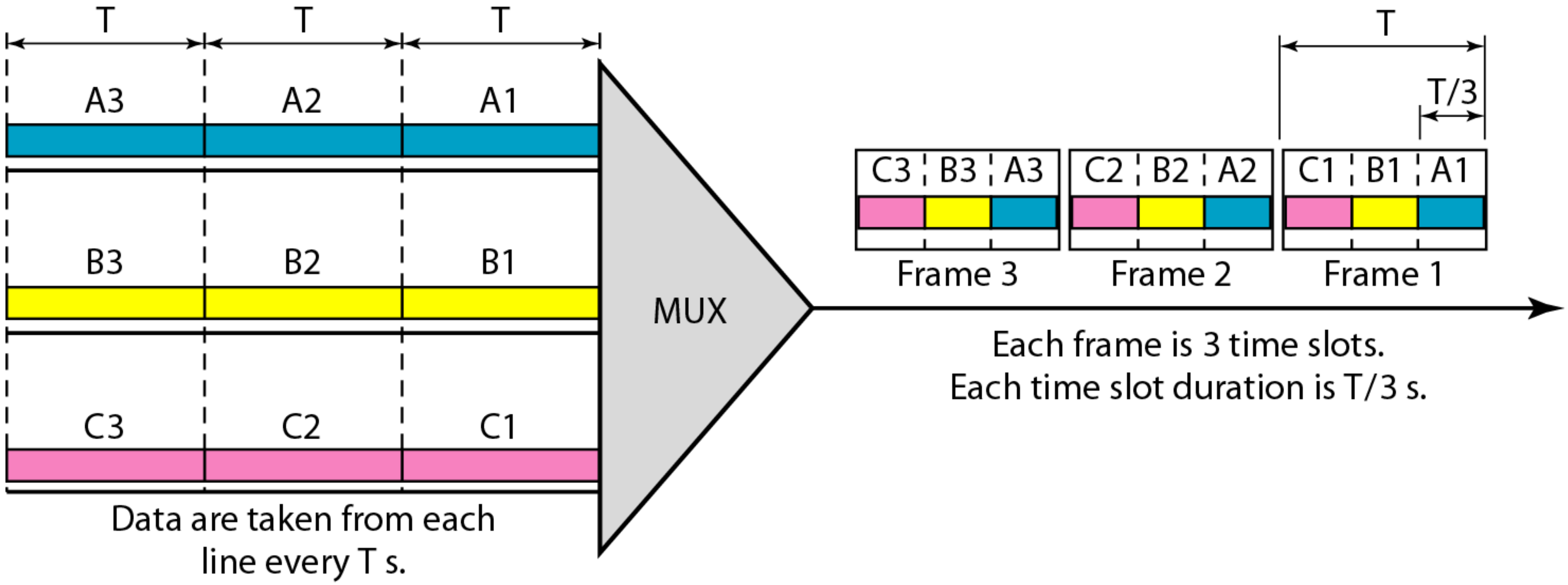
Time-Division Multiplexing

Queen Mary
University of London

TDM

- Time-division multiplexing (TDM) is the time interleaving of samples from several sources so that the information from these sources can be transmitted serially over a single communication channel.
- TDM is a digital multiplexing technique for combining several low-rate digital channels into one high-rate one.
- In synchronous TDM, the data rate of the link is *n* times faster, and the unit duration is *n* times shorter.

Figure: *Synchronous time-division multiplexing*



Example 6.5

In the Figure of the previous slide, the data rate for each one of the 3 input connection is 1 kbps. If 1 bit at a time is multiplexed (a unit is 1 bit), what is the duration of (a) each input slot, (b) each output slot, and (c) each frame?

Solution

We can answer the questions as follows:

a. The data rate of each input connection is 1 kbps. This means that the bit duration is 1/1000 s or 1 ms. The duration of the input time slot is 1 ms (same as bit duration).

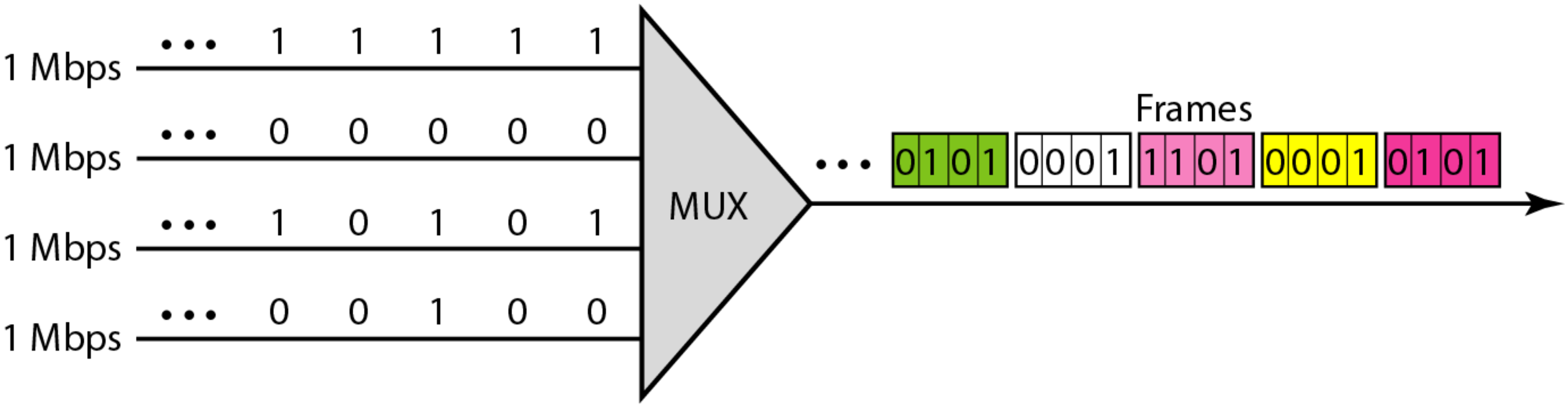
Example 6.5 (continued)

b. The duration of each output time slot is one-third of the input time slot. This means that the duration of the output time slot is 1/3 ms.

c. Each frame carries three output time slots. So the duration of a frame is 3 × 1/3 ms, or 1 ms.

Note: The duration of a frame is the same as the duration of an input unit.

Figure 6.14 *Example 6.6*



Example 6.6

Figure 6.14 shows synchronous TDM with 4 1Mbps data stream inputs and one data stream for the output. The unit of data is 1 bit. Find (a) the input bit duration, (b) the output bit duration, (c) the output bit rate, and (d) the output frame rate.

Solution

We can answer the questions as follows:

a. The input bit duration is the inverse of the bit rate: 1/1 Mbps = 1 μs.

b. The output bit duration is one-fourth of the input bit duration, or ¼ μs.

Example 6.6 (continued)

c. The output bit rate is the inverse of the output bit duration or 1/(4μs) or 4 Mbps. This can also be deduced from the fact that the output rate is 4 times as fast as any input rate; so the output rate = 4 × 1 Mbps = 4 Mbps.

d. The frame rate is always the same as any input rate. So the frame rate is 1,000,000 frames per second. Because we are sending 4 bits in each frame, we can verify the result of the previous question by multiplying the frame rate by the number of bits per frame.