

# From Waveforms to Bits

Diogo Fonseca  
Professor: João Dias

UALG  
MESTRADO EM ENGENHARIA INFORMÁTICA

8 de Novembro de 2025

# Table of contents

- 1 Theoretical Basis
  - Fourier Analysis
  - Bandwidth-Limited Signals
- 2 The Maximum Data Rate of a Channel
- 3 Digital Modulation
  - Baseband Transmission
  - Bandwidth Efficiency
  - Clock Recovery
  - Balanced Signals
  - Passband Transmission

# Table of contents

# Fourier Analysis

## Fourier Series

$$g(t) = \frac{1}{2}c + \sum_{n=1}^{\infty} a_n \sin(2\pi nft) + \sum_{n=1}^{\infty} b_n \cos(2\pi nft)$$

## Fourier Series

$$g(t) = \frac{1}{2}c + \sum_{n=1}^{\infty} a_n \sin(2\pi nft) + \sum_{n=1}^{\infty} b_n \cos(2\pi nft)$$

$$a_n = \frac{2}{T} \int_0^T g(t) \sin(2\pi nft) dt$$

## Fourier Series

$$g(t) = \frac{1}{2}c + \sum_{n=1}^{\infty} a_n \sin(2\pi nft) + \sum_{n=1}^{\infty} b_n \cos(2\pi nft)$$

$$a_n = \frac{2}{T} \int_0^T g(t) \sin(2\pi nft) dt$$

$$b_n = \frac{2}{T} \int_0^T g(t) \cos(2\pi nft) dt$$

## Fourier Series

$$g(t) = \frac{1}{2}c + \sum_{n=1}^{\infty} a_n \sin(2\pi nft) + \sum_{n=1}^{\infty} b_n \cos(2\pi nft)$$

$$a_n = \frac{2}{T} \int_0^T g(t) \sin(2\pi nft) dt$$

$$b_n = \frac{2}{T} \int_0^T g(t) \cos(2\pi nft) dt$$

$$c = \frac{2}{T} \int_0^T g(t) dt$$

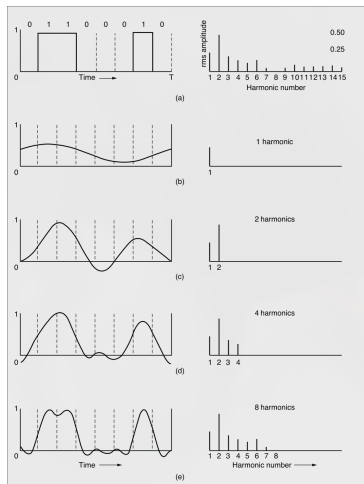


# Bandwidth-Limited Signals

## Some definitions

- Bandwidth (analog)
- Bandwidth (digital)
- Cutoff
- Baseband
- Passband

# Bandwidth-Limited Signals



**Figure 1:** (a) A binary signal and its root-mean-square Fourier amplitudes. (b)-(e) Successive approximations to the original signal.

# Bandwidth-Limited Signals

<b>Bps</b>	<b>T (msec)</b>	<b>First harmonic (Hz)</b>	<b># Harmonics sent</b>
300	26.67	37.5	80
600	13.33	75	40
1200	6.67	150	20
2400	3.33	300	10
4800	1.67	600	5
9600	0.83	1200	2
19200	0.42	2400	1
38400	0.21	4800	0

Table 1: Relation between data rate and harmonics.

# Maximum Data Rate of a Channel

# Maximum Data Rate of a Channel

Assume a perfect channel for communication with **no noise or interference**. Furthermore, assume an arbitrary signal that has been run through a low-pass filter of bandwidth  $B$ .

# Maximum Data Rate of a Channel

Assume a perfect channel for communication with **no noise or interference**. Furthermore, assume an arbitrary signal that has been run through a low-pass filter of bandwidth  $B$ .

## Nyquist's sample rate proof

The filtered signal can be completely reconstructed by making exactly  $2B$  samples per second.

# Maximum Data Rate of a Channel

Assume a perfect channel for communication with **no noise or interference**. Furthermore, assume an arbitrary signal that has been run through a low-pass filter of bandwidth  $B$ .

## Nyquist's sample rate proof

The filtered signal can be completely reconstructed by making exactly  $2B$  samples per second.

And given  $V$ , the number of discrete states a signal can transmit at each sampling instant:



# Maximum Data Rate of a Channel

Assume a perfect channel for communication with **no noise or interference**. Furthermore, assume an arbitrary signal that has been run through a low-pass filter of bandwidth  $B$ .

## Nyquist's sample rate proof

The filtered signal can be completely reconstructed by making exactly  $2B$  samples per second.

And given  $V$ , the number of discrete states a signal can transmit at each sampling instant:

## Nyquist's theorem

Maximum data rate of channel =  $2B \log_2(V)$  bits/sec

# What about interference?

# Maximum Data Rate of a Channel

Assume a channel analogous to the last, with a signal power  $S$  and a noise power  $N$ .

# Maximum Data Rate of a Channel

Assume a channel analogous to the last, with a signal power  $S$  and a noise power  $N$ .

SNR (Signal-to-Noise Ratio)

$$\frac{S}{N}$$

# Maximum Data Rate of a Channel

Assume a channel analogous to the last, with a signal power  $S$  and a noise power  $N$ .

SNR (Signal-to-Noise Ratio)

$$\frac{S}{N}$$

Capacity of a real channel

Maximum data rate of channel =  $B \log_2(1 + \frac{S}{N})$  bits/sec

# Digital Modulation

# Baseband transmission

**NRZ scheme** (Non-Return-to-Zero)

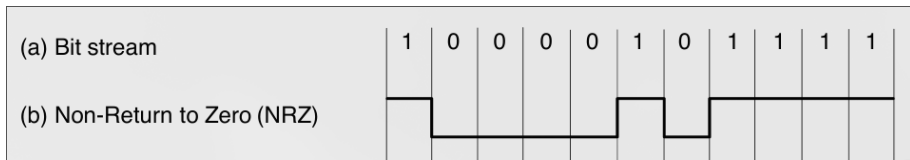


## **NRZ scheme** (Non-Return-to-Zero)

- Signal follows the data. (i.e. Positive voltage for one, negative voltage for zero)
- Sender sends the signal.
- Receiver samples the signal at regular intervals of time.
- Receiver maps signal samples to the closes symbols (i.e.  $\{0, 1\}$ )

## NRZ scheme (Non-Return-to-Zero)

- Signal follows the data. (i.e. Positive voltage for one, negative voltage for zero)
- Sender sends the signal.
- Receiver samples the signal at regular intervals of time.
- Receiver maps signal samples to the closes symbols (i.e.  $\{0, 1\}$ )



# Bandwidth Efficiency

## NRZ scheme

## NRZ scheme

- For a data rate of  $B$  bits/sec we need a bandwidth of at least  $\frac{B}{2}$  Hz.

## NRZ scheme

- For a data rate of  $B$  bits/sec we need a bandwidth of at least  $\frac{B}{2}$  Hz.
- Given this is a fundamental limit, we cannot run NRZ faster without using additional bandwidth.

## NRZ scheme

- For a data rate of  $B$  bits/sec we need a bandwidth of at least  $\frac{B}{2}$  Hz.
- Given this is a fundamental limit, we cannot run NRZ faster without using additional bandwidth.
- One strategy to increase efficiency is to use more than two signaling levels.

## NRZ scheme

- For a data rate of  $B$  bits/sec we need a bandwidth of at least  $\frac{B}{2}$  Hz.
- Given this is a fundamental limit, we cannot run NRZ faster without using additional bandwidth.
- One strategy to increase efficiency is to use more than two signaling levels.
- By using 4 different voltages, for instance, we can send 2 bits at once as a single **symbol**.



## NRZ scheme

- For a data rate of  $B$  bits/sec we need a bandwidth of at least  $\frac{B}{2}$  Hz.
- Given this is a fundamental limit, we cannot run NRZ faster without using additional bandwidth.
- One strategy to increase efficiency is to use more than two signaling levels.
- By using 4 different voltages, for instance, we can send 2 bits at once as a single **symbol**.
- This works as long as the signal at the receiver is sufficiently strong to distinguish all the levels.

## NRZ scheme

- For a data rate of  $B$  bits/sec we need a bandwidth of at least  $\frac{B}{2}$  Hz.
- Given this is a fundamental limit, we cannot run NRZ faster without using additional bandwidth.
- One strategy to increase efficiency is to use more than two signaling levels.
- By using 4 different voltages, for instance, we can send 2 bits at once as a single **symbol**.
- This works as long as the signal at the receiver is sufficiently strong to distinguish all the levels.
- The rate at which the signal changes (**symbol rate**) is then half the bit rate, so bandwidth has been reduced.

## NRZ scheme

- For a data rate of  $B$  bits/sec we need a bandwidth of at least  $\frac{B}{2}$  Hz.
- Given this is a fundamental limit, we cannot run NRZ faster without using additional bandwidth.
- One strategy to increase efficiency is to use more than two signaling levels.
- By using 4 different voltages, for instance, we can send 2 bits at once as a single **symbol**.
- This works as long as the signal at the receiver is sufficiently strong to distinguish all the levels.
- The rate at which the signal changes (**symbol rate**) is then half the bit rate, so bandwidth has been reduced.
- Bitrate can be interpreted as (**symbol rate**  $\times$  **bits per symbol**).

# Clock Recovery

## The problem

## The problem

- The receiver must know when one symbol ends and the next symbol appears.

## The problem

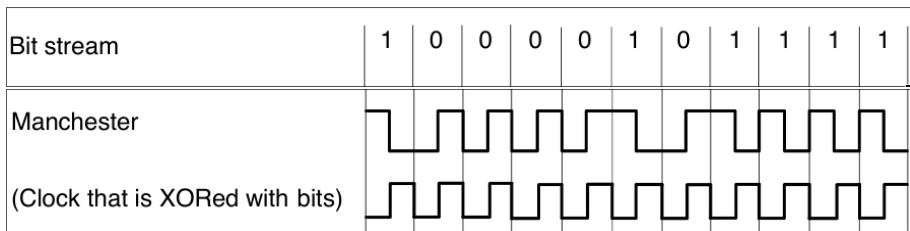
- The receiver must know when one symbol ends and the next symbol appears.
- After a while it's hard to tell the bits apart, 15 zeroes look much like 16 zeroes unless you have a very accurate clock.

## The problem

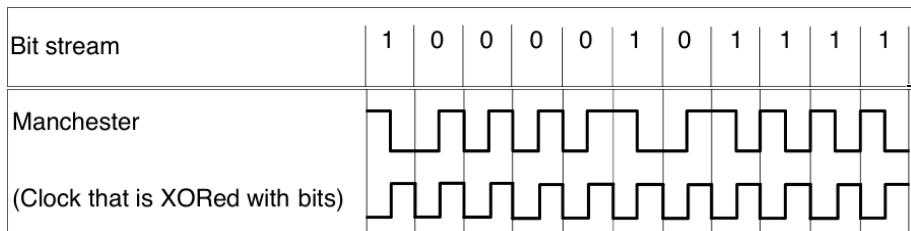
- The receiver must know when one symbol ends and the next symbol appears.
- After a while it's hard to tell the bits apart, 15 zeroes look much like 16 zeroes unless you have a very accurate clock.
- Very accurate clocks are expensive.



## Manchester encoding

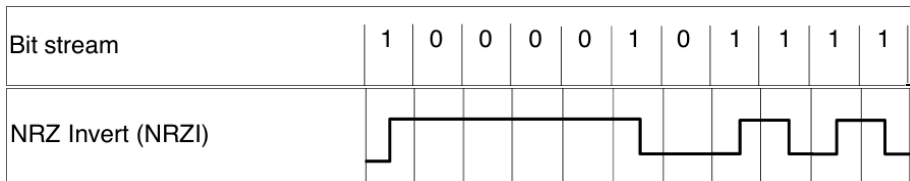


## Manchester encoding

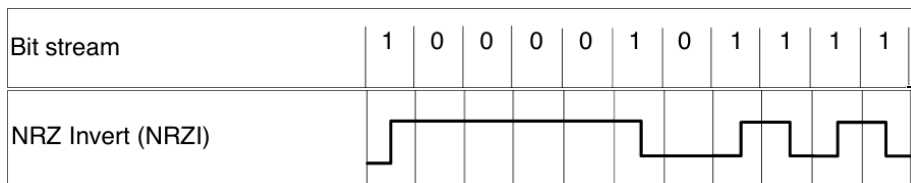


- Requires twice the bandwidth.
- Used in classic Ethernet.

## NRZI (Non-Return-to-Zero Inverse)



## NRZI (Non-Return-to-Zero Inverse)



- Long streaks of 0 still have the same problem.
- Used in USB.

## 4B/5B

Data (4B)	Codeword (5B)	Data (4B)	Codeword (5B)
0000	11110	1000	10010
0001	01001	1001	10011
0010	10100	1010	10110
0011	10101	1011	10111
0100	01010	1100	11010
0101	01011	1101	11011
0110	01110	1110	11100
0111	01111	1111	11101

Table 2: 4B/5B mapping.

## 4B/5B

Data (4B)	Codeword (5B)	Data (4B)	Codeword (5B)
0000	11110	1000	10010
0001	01001	1001	10011
0010	10100	1010	10110
0011	10101	1011	10111
0100	01010	1100	11010
0101	01011	1101	11011
0110	01110	1110	11100
0111	01111	1111	11101

Table 2: 4B/5B mapping.

- 25% overhead.

## Scrambling

Scrambles the data by XORing it with a pseudorandom sequence.  
Receiver XORs the data with the same sequence.

## Scrambling

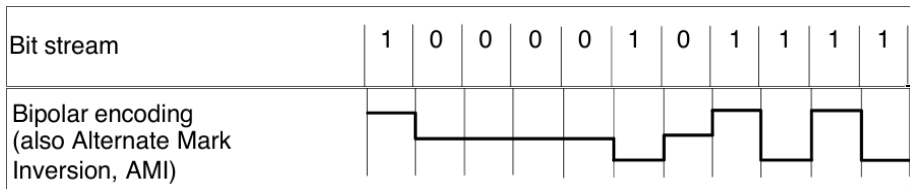
Scrambles the data by XORing it with a pseudorandom sequence.  
Receiver XORs the data with the same sequence.

- Used in early versions of IP over SONET.
- Still possible to get long streaks of identical symbols.
- Possible malicious usage, "killer packets".

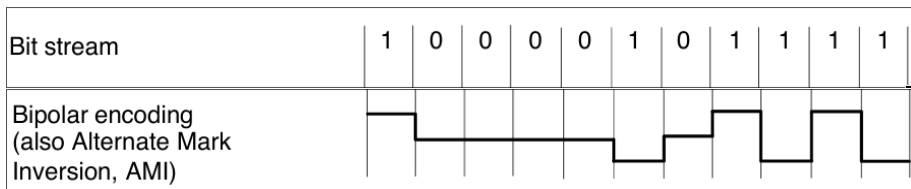


# Balanced Signals

## Bipolar Encoding



## Bipolar Encoding



- Adds a voltage level.
- Average of 0.
- Indirectly helps with clock recovery.

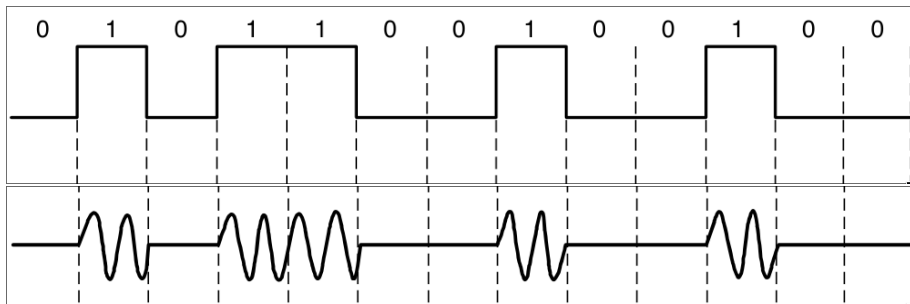
**4B/5B (again) or 8B/10B**

## 4B/5B (again) or 8B/10B

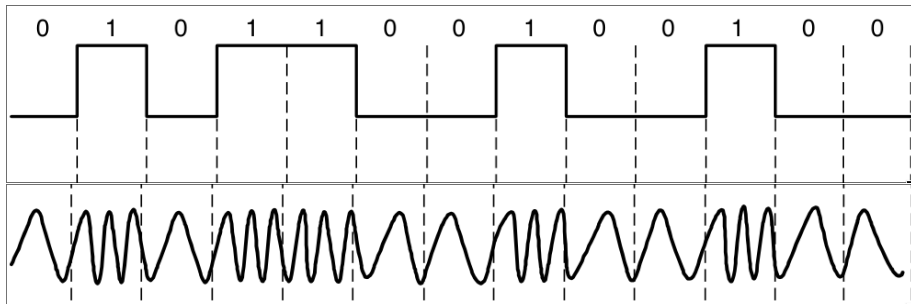
- 8B/10B has at most a disparity of 2 bits.
- 8B/10B is 80% efficient.
- Helps with clock recovery.

# Passband Transmission

## ASK (Amplitude Shift Keying)

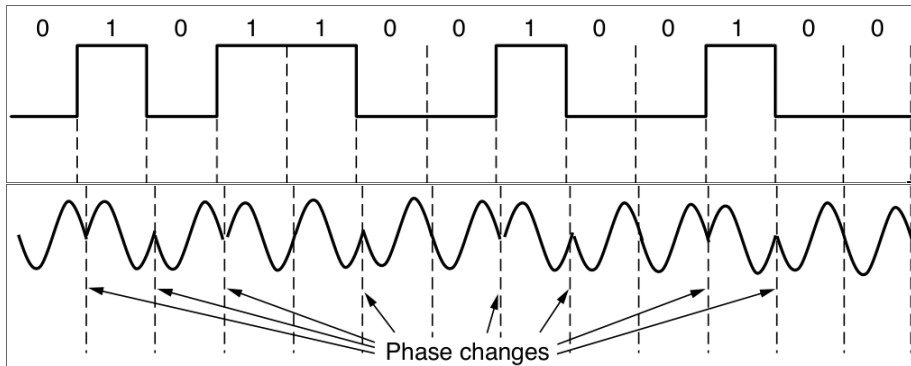


## FSK (Frequency Shift Keying)

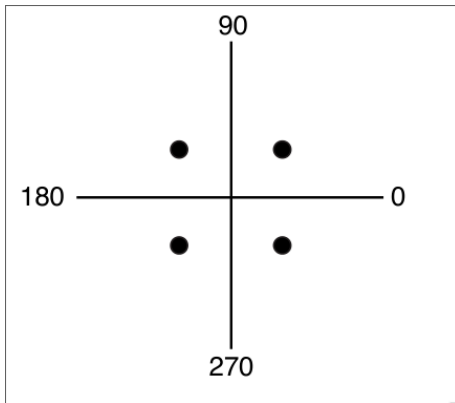




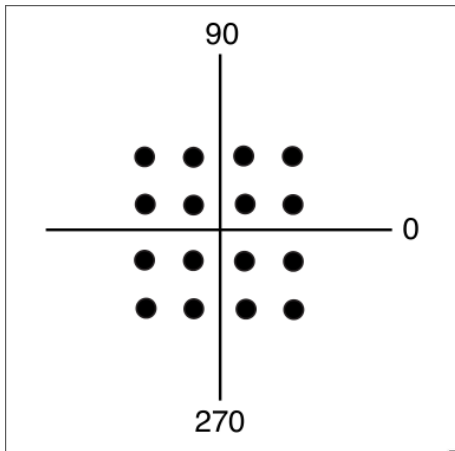
## PSK (Phase Shift Keying)



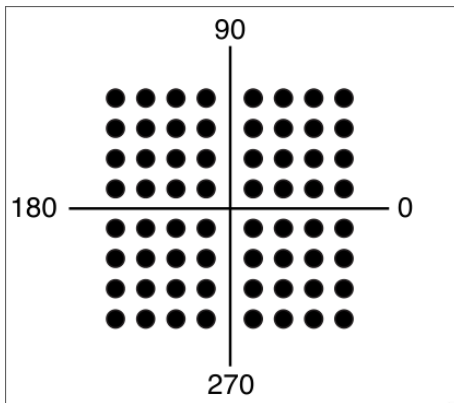
## QPSK (Quadrature Phase Shift Keying)



## QAM-16 (Quadrature Amplitude Modulation - 16)



## QAM-64 (Quadrature Amplitude Modulation - 64)



## Gray-coded QAM-16

