



universidade de aveiro
theoria poiesis praxis

Transmission Systems, Coding and Error Control

Redes de Comunicações I

**Mestrado Integrado em Engenharia de Computadores e
Informática**

DETI-UA, 2023/2024

Transmission systems

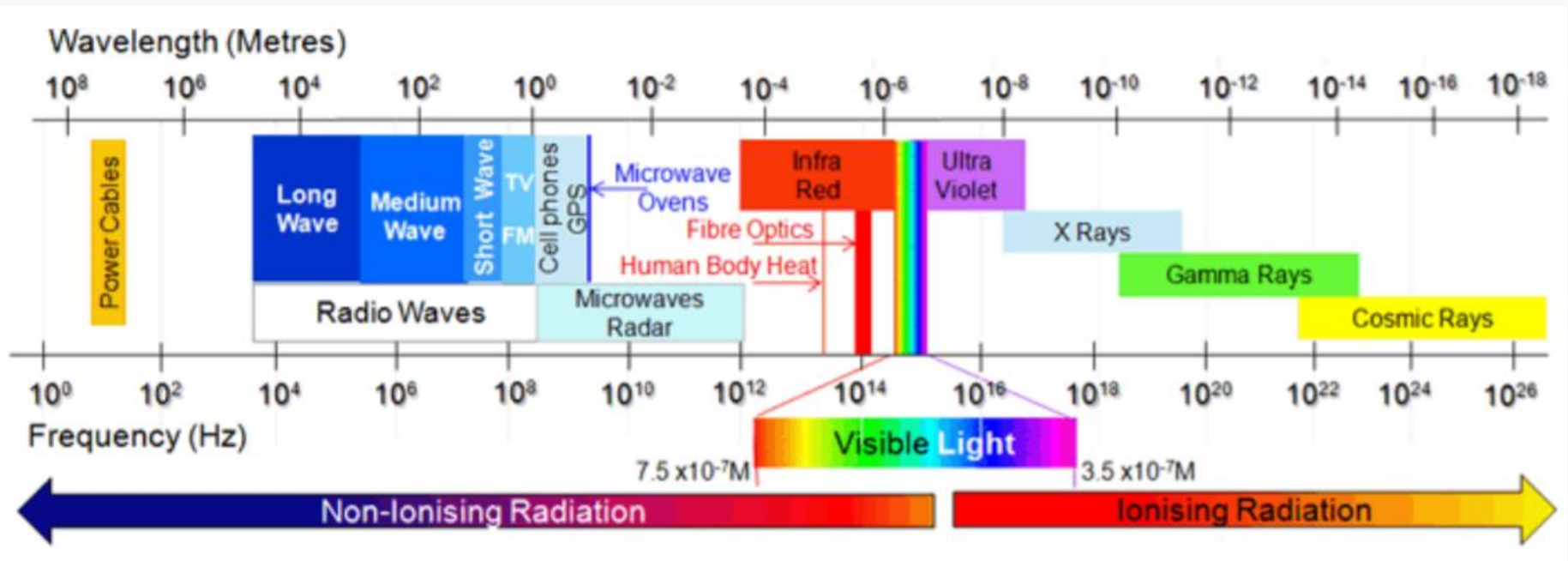
- **Physical layer of communication networks**
 - Responsible for transmission of signals over a physical connection
- **Terminal equipments: the sources and destinations of information**
 - PCs, mobile terminals, servers, ...
- **Physical connection**
 - Transmission medium between two remote points
 - Between terminal equipment
 - Example: a shared medium connecting PCs
 - Between network equipment
 - Example: a direct cable between routers
 - Between terminal and network equipment
 - Example: in a shared medium, the communication between a PC and the router that is its default gateway
- **Physical connections**
 - Point-to-point connections
 - Shared connections

Guided/Unguided Transmission Systems

- **Guided:** a signal travels through a bounding physical medium (copper cable, optical fibre, ...)
- **Unguided:** a signal travels through a boundless medium (air, water, vacuum, ...)
 - Can be directional or omni-directional.

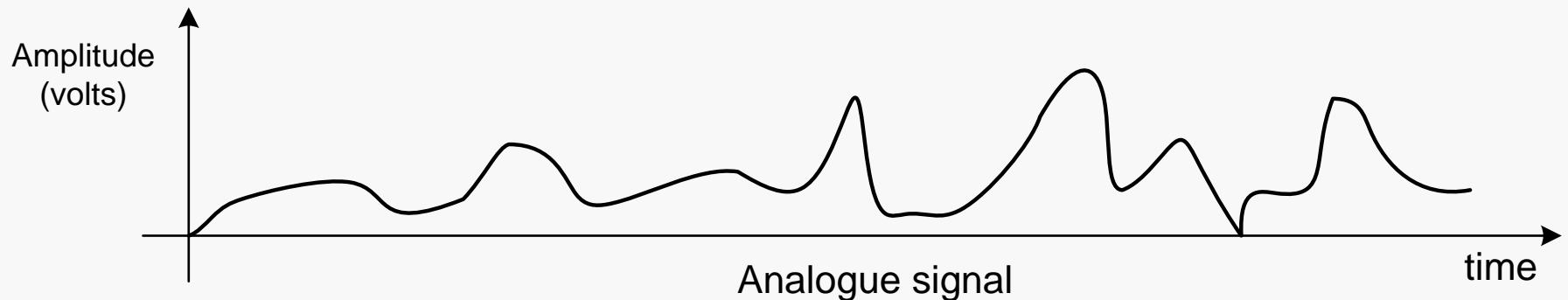
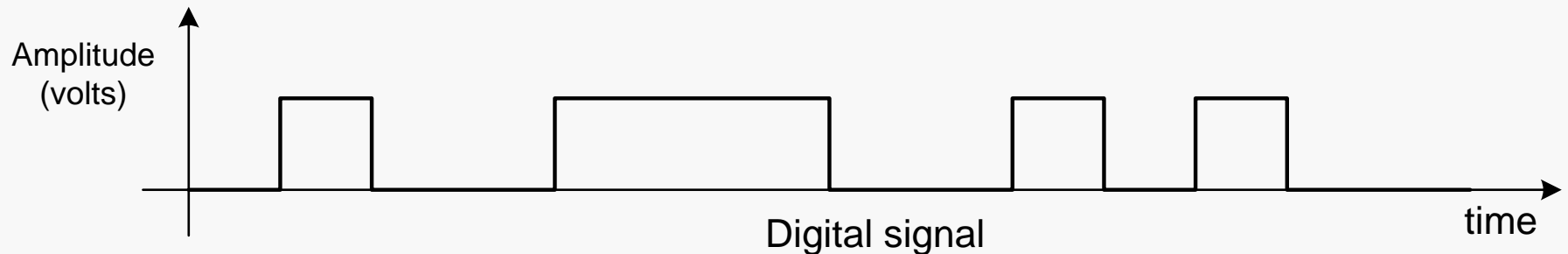


Electromagnetic spectrum



Transmission systems

- **Analogue electrical signals:** the signal amplitude varies continuously over time
- **Digital electrical signals:** the signal amplitude assumes a discrete set of values

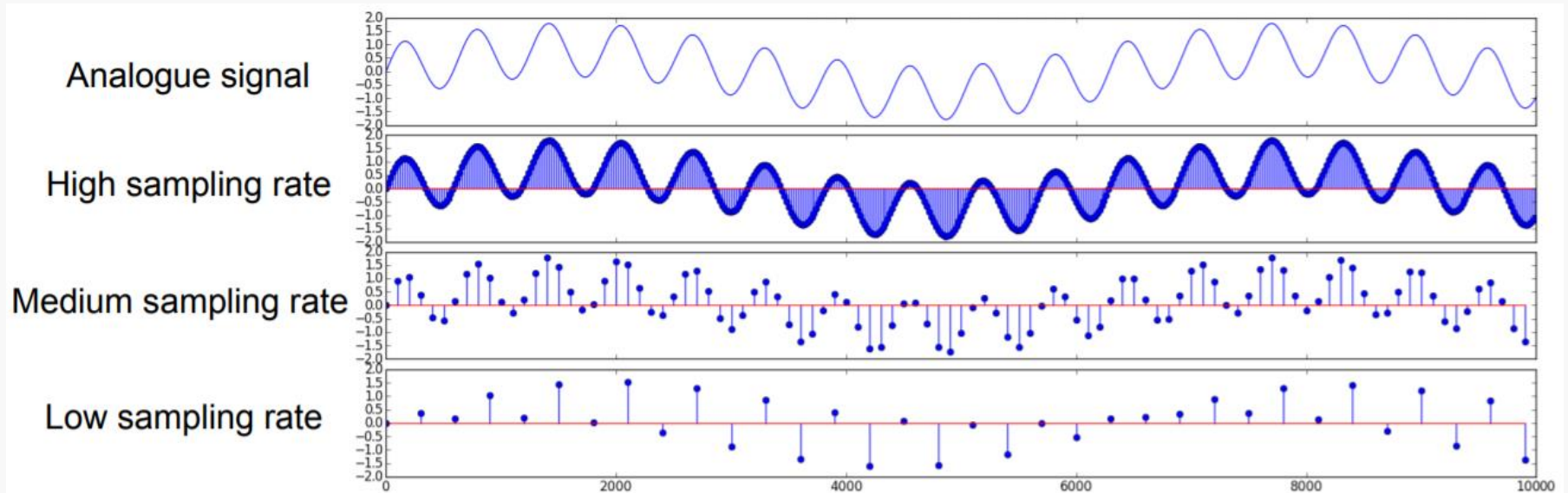
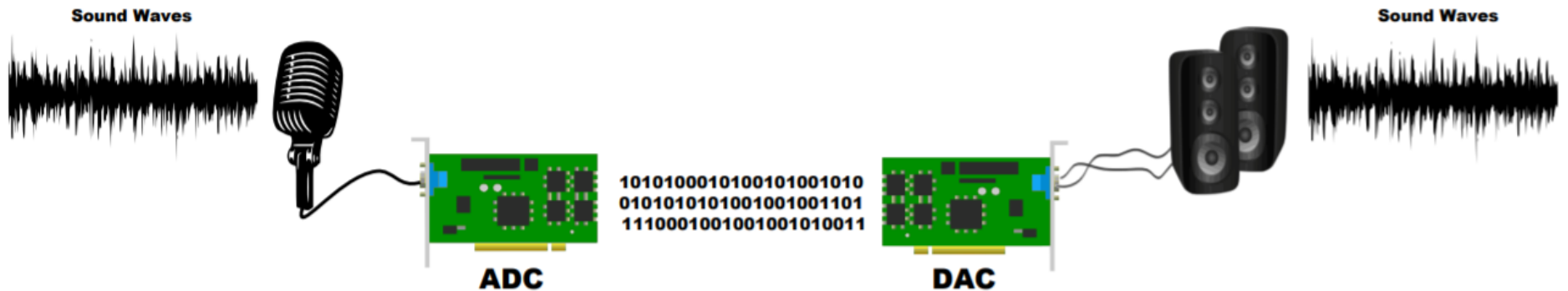


Transmission systems



- **Transmitter** – adapts the signal to the transmission channel characteristics.
 - Typical operations: modulation, coding.
- **Transmission channel** – medium connecting transmitter to receiver.
 - Examples: copper twisted pairs, coaxial cable, optical fibres, free space.
 - Degradation factors: attenuation, distortion, interference, noise.
- **Receiver** – processes the received signal in order to compensate the transmission channel degradation factors.
 - Typical operations: amplification, demodulation, decoding, filtering.

Analogue to Digital: Sampling

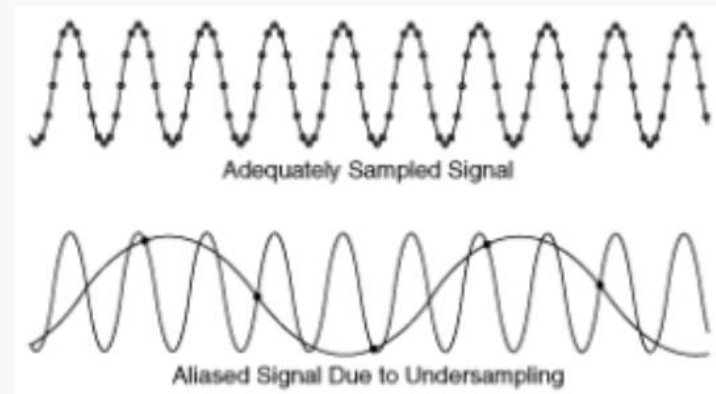
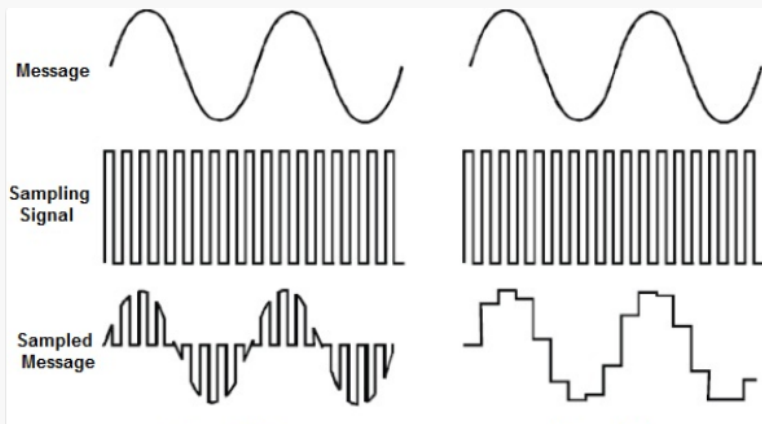


Sampling Frequency

- **The sampling process measures and quantifies the analogue signal at equally space time intervals.**
- **The sampling process must be able to capture the main characteristics of the original analogue signal.**
- **The sampling rate determines the amount of information that is transferred to the digital signal.**
- **To reconstruct a signal from the samples, the sampling frequency must be high enough to capture the relevant signal information (frequency components).**

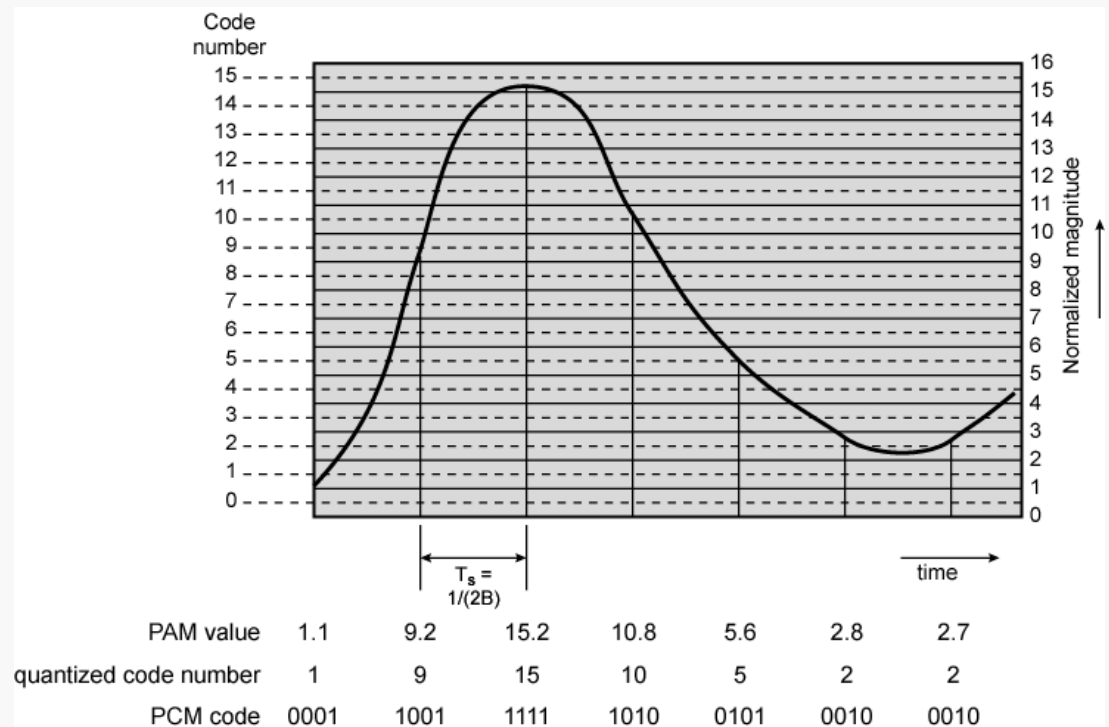
Sampling Frequency

- For a signal where the highest (relevant) frequency is f_m , the sampling frequency (f_s) must be higher than two times f_m
- $f_s > 2 * f_m$ is called the Nyquist rate.



PCM (*Pulse Code Modulation*) system

- The digital transmission of analogue signals requires:
 - an analogue-digital converter (ADC) in the source, and
 - a digital-analogue converter (DAC) in the destination.



PCM

- Assuming that $|x(t)| \leq z$, the difference between rounding levels is z/q and the maximum rounding error is $z/(2q)$. In this case:

$$q = M^v \qquad v = \log_M q$$

v symbols, and the transmission rate is $v \times f_s$ baud.

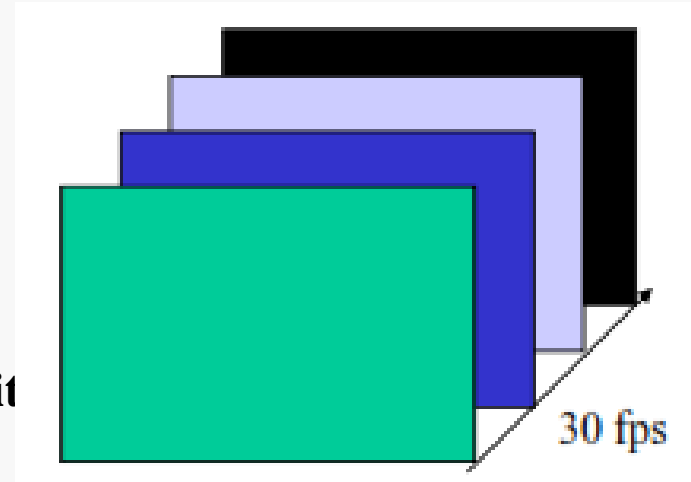
- Lowering the rounding distortion implies
 - enlarging the number of rounding levels
 - and, consequently, enlarging the transmission rate.
- In the case of voice telephony:
 - $f_s = 8000$ samples/s, $q = 256$ levels, $M = 2$, $v = 8$:
 $v = \log_2 256$
 - it results in a voice channel bit rate of 64 Kb/s ($v \times f_s = 8 \times 8000$).

Example 1

- An analogue signal $x(t)$ limited in amplitude such that $|x(t)| < 5$ Volts and with effective bandwidth of 6 MHz is converted into a binary PCM stream for transmission. What is the minimum data rate that guarantees a rounding error not higher than 0.05 Volts?
 - Maximum rounding error = $10/2q$ (q is the number of rounding levels)
 - q must be the lower power of 2 (binary signal) higher or equal to $10/(2 \times 0.05) = 100$; therefore, $q = 128$
 - We need $\log_2(128) = 7$ bits per sample
 - Minimum sampling rate: $2 \times 6 \text{ MHz} = 12 \times 10^6 \text{ samples/s}$
 - Minimum data rate: $7 \times 12 \times 10^6 = \underline{84 \text{ Mbits/s}}$

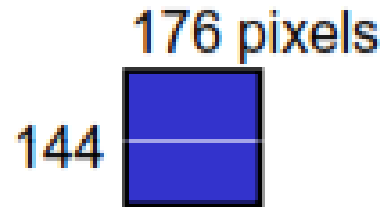
Video rate

- **Sequence of image frames**
 - Each image is digitized and compressed
- **Frame rate**
 - 10-30-60 frames/sec depending on the quality
- **Frame resolution**
 - Small frames for videoconference
 - Standard frames for standard broadcast TV
 - HDTV frames
- **BitRate = 3colours x (WxH) pixels/frame x B bits/pixel x F frames/sec**



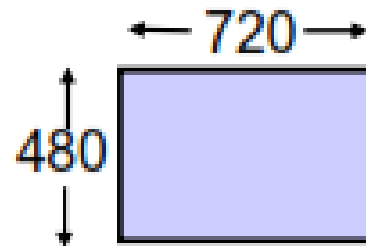
Video frames

QCIF videoconferencing



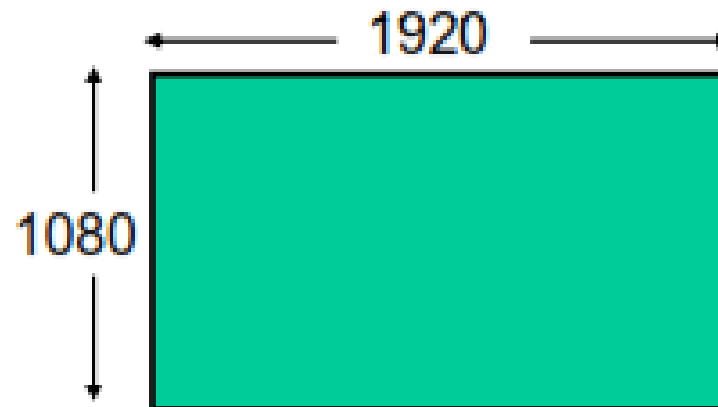
at 30 frames/sec =
760,000 pixels/sec

Broadcast TV



at 30 frames/sec =
 10.4×10^6 pixels/sec

HDTV

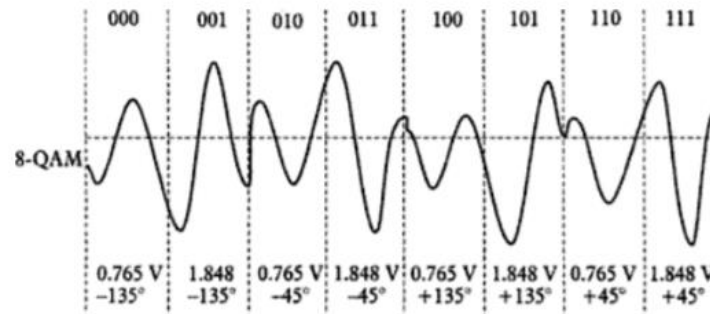
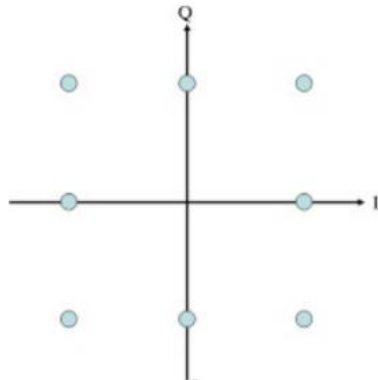


at 30 frames/sec =
 67×10^6 pixels/sec

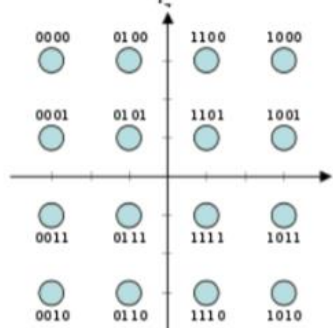
Examples of Modulation

- Quadrature Amplitude Modulation (QAM)
- Uses 2-Dimensional signalling

8-QAM

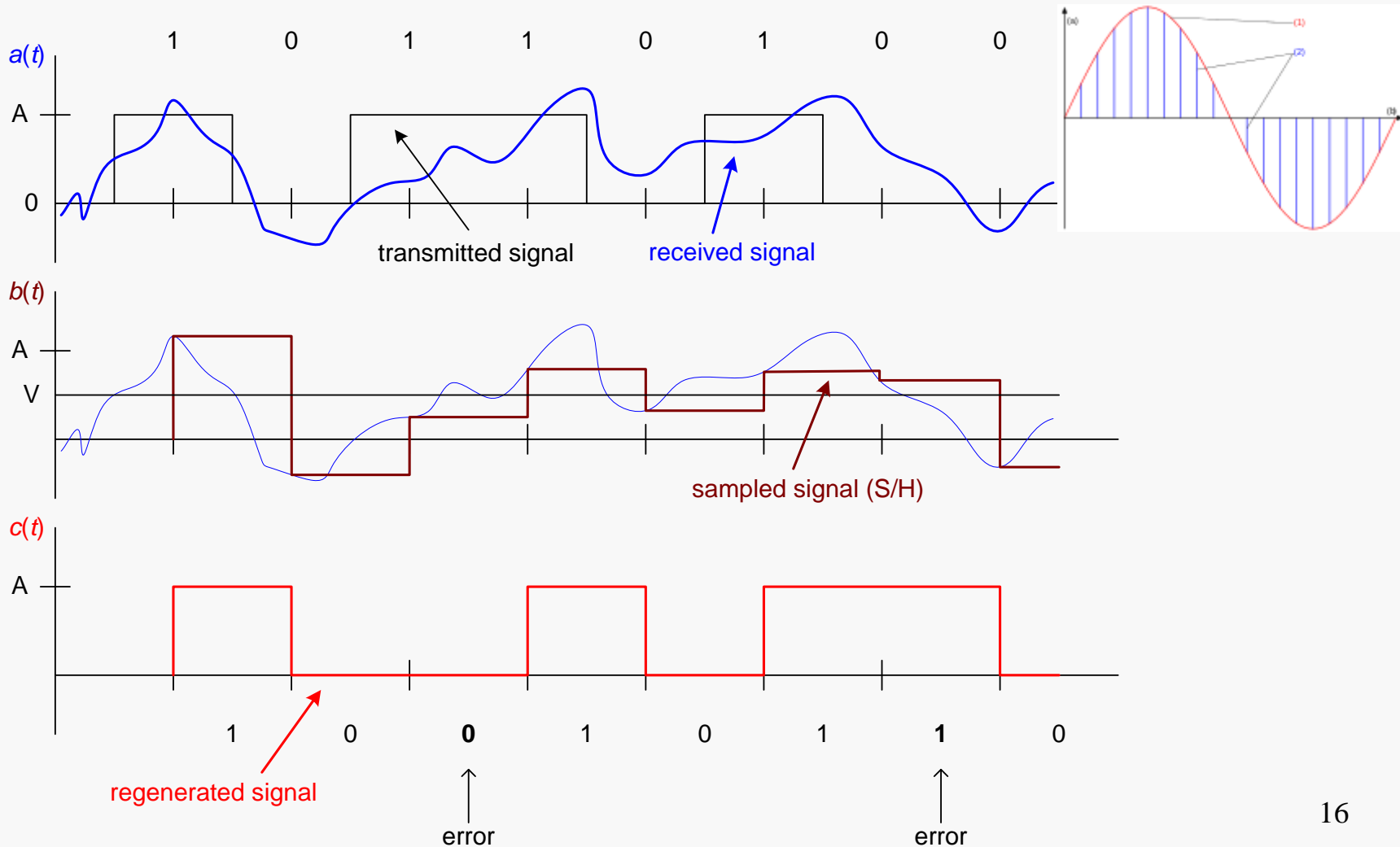
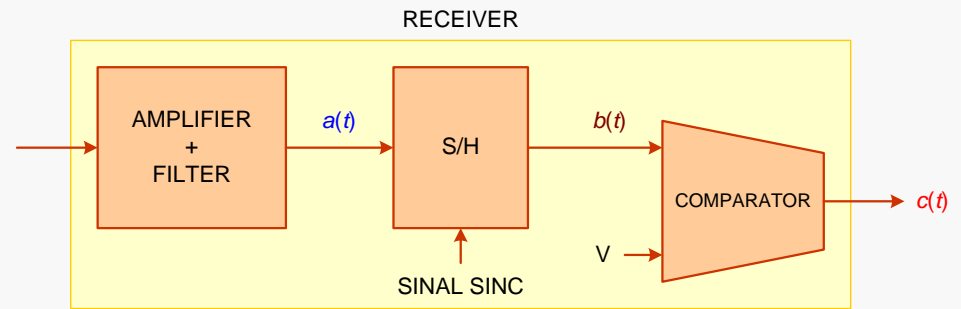


16-QAM



Regeneration of Pulse Amplitude Modulation (PAM) digital signals

Example: NRZ unipolar



DETEÇÃO DE ERROS

Introdução

- **Problemas no canal de comunicação**
 - Pacotes corrompidos (recebidos com erros)
 - Pacotes perdidos
 - Pacotes recebidos fora de ordem
- **Deteção de erros**
 - O recetor ser capaz de detetar pacotes recebidos com erros
- **Métodos a abordar:**
 - Código de verificação de paridade simples
 - Código de verificação de paridade em blocos
 - CRC (*Cyclic-Redundancy Check*)

Código de verificação de paridade simples

- Junta-se um bit adicional, o bit de paridade, à palavra binária a transmitir
- Bit de paridade escolhido por forma a garantir que a palavra completa tem um número par de 1s (paridade par) ou um número ímpar de 1s (paridade ímpar)
- Exemplo (paridade par):

0 1 0 0 0 0 0 0 1 caracter ASCII A com bit de paridade par

0 1 0 0 0 0 0 1 1 com 1 bit errado

0 1 0 0 0 0 1 1 1 com 2 bits errados


- Detecta todos os erros com um nº ímpar de bits errados; não detecta nenhum erro com um nº par de bits errados

Código de verificação de paridade em blocos

- Utilizado na transmissão de blocos de palavras binárias
- É formado um bit de paridade em cada palavra individual (na horizontal) e também sobre o bloco de palavras (na vertical). É adicionado um caráter designado por *Block-Check Character* (BCC) no fim do bloco.
- Exemplo:

	bit de paridade								
	↓								
A	0	1	0	0	0	0	0	0	1
B	0	1	0	0	0	0	0	1	0
C	1	1	0	0	0	0	0	1	1
BCC	1	1	0	0	0	0	0	0	0

BCC	C	B	A
11000000	110000011	010000010	010000001



sequência de bits transmitidos

Código de verificação de paridade em blocos

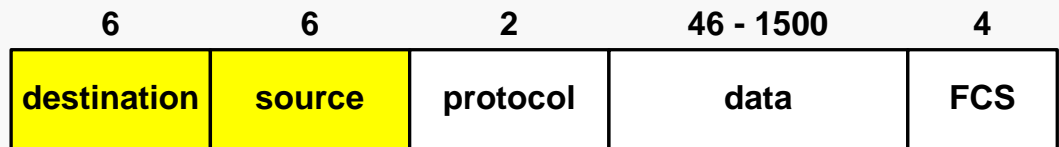
							erros		
							↓	↓	
A	0	1	0	0	0	0	1	1	1
B	0	1	0	0	0	0	0	1	0
C	1	1	0	0	0	0	0	1	1
BCC	1	1	0	0	0	0	0	0	0
							↑	↑	
							erros detetados		

							erros		
							↓	↓	
A	0	1	0	0	0	0	1	1	1
B	0	1	0	0	0	0	1	0	0
C	1	1	0	0	0	0	0	1	1
BCC	1	1	0	0	0	0	0	0	0
							↑	↑	
							erros não detetados		

CRC (*Cyclic-Redundancy Check*)

- Mensagem a transmitir: **57268**
- Emissor e recetor combinam divisor: **84**
- No emissor executa-se $57268 / 84 = 681 + 64/84$
- O emissor transmite **5726864**

- A mensagem chega com erros ao recetor: **5754864**
- Agora $57548 / 84 = 685 + 8/84$
- Como resto é diferente de 64 o erro é detetado!



Ethernet II

CRC (*Cyclic-Redundancy Check*)

- Represente-se a sequência de bits

$$b_{n-1}, b_{n-2}, \dots, b_3, b_2, b_1, b_0$$

através de um polinómio

$$b_{n-1}x^{n-1} + b_{n-2}x^{n-2} + \dots + b_3x^3 + b_2x^2 + b_1x + b_0$$

- A geração de um CRC para uma mensagem $M(x)$ com m bits segue os seguintes passos:
 1. O emissor e o recetor acordam num polinómio gerador $G(x)$, com pelo menos dois termos não-nulos, x^r e 1, onde r é a ordem de $G(x)$.
 2. O emissor adiciona r zeros no fim da mensagem a ser transmitida. A mensagem fica então com $m + r$ bits correspondendo ao polinómio $x^r M(x)$.
 3. O emissor determina o resto da divisão de $x^r M(x)$ por $G(x)$ (tem sempre r ou menos bits). Este resto designa-se por $R(x)$.
 4. A mensagem transmitida é $T(x) = x^r M(x) + R(x)$.

CRC (*Cyclic-Redundancy Check*)

- **Exemplo:**

- $M(x) = x^4 + x^3 + x^2 + 1$ ($m = 5$); Mensagem a enviar: 11101
- $G(x) = x^3 + 1$ ($r = 3$);
- $x^r M(x) = x^7 + x^6 + x^5 + x^3$;
- $R(x) = x^2 + x$;
- $T(x) = x^r M(x) + R(x) = x^7 + x^6 + x^5 + x^3 + x^2 + x$. Mensagem enviada: 11101110

- Seja $Z(x)$ o resultado da divisão de $x^r M(x)$ por $G(x)$. Então

$$x^r M(x) = G(x)Z(x) + R(x)$$

e

$$T(x) = x^r M(x) + R(x) = G(x)Z(x)$$

ou seja, todas as palavras transmitidas são divisíveis por $G(x)$.

(Nota: subtração módulo 2 = adição módulo 2).

$$x^4 + x^3 + x^2 + 1 \rightarrow h(x)$$

$$n=3 \in G(x) = x^3 + 1$$

$$x^2 h(x) = x^7 + x^6 + x^5 + x^3$$

$$\begin{array}{r} x^7 + x^6 + x^5 + + x^3 + + + \\ x^7 \\ \hline \end{array}$$

$$\begin{array}{r} x^3 + 1 \\ x^4 + x^3 + x^2 + x \\ \hline \end{array}$$

$$\begin{array}{r} x^6 + x^5 + x^4 + x^3 + 1 \\ x^6 \\ \hline \end{array}$$

$$\begin{array}{r} x^5 + x^4 + 1 \\ x^5 \\ \hline \end{array}$$

$$\begin{array}{r} x^4 + + x^2 \\ x^4 \\ \hline \end{array}$$

$$\boxed{x^2 + x} = R(x)$$

$x^2 h(x) + R(x) \rightarrow 0$ que é enviado para o resto.

\hookrightarrow que é divisível por $x^3 + 1$

CRC (*Cyclic-Redundancy Check*)

- A mensagem recebida pode conter erros, isto é, pode ser $T(x) + E(x)$, onde $E(x)$ é o polinómio que representa os erros.

- O recetor divide a mensagem recebida por $G(x)$, isto é, executa

$$[T(x)+E(x)]/G(x)$$

- Uma vez que

$$\text{Resto de } [T(x)+E(x)]/G(x) = \text{Resto de } E(x)/G(x)$$

então, o recetor decide que não houve erro se o resto for zero e que houve erros caso contrário.

$$\begin{array}{r}
 x^7 + x^6 + x^5 + \cancel{x^4} + x^3 + x^2 + x + 0 \\
 \underline{x^7} \\
 x^6 + x^5 + x^4 + x^3 \\
 \underline{x^6} \\
 x^5 + x^4 + x^3 \\
 \underline{x^5} \\
 x^4 + x^3 \\
 \underline{x^4} \\
 x^3 + x^2 + x + 0 \\
 \underline{x^3} \\
 x^2 + x + 0 \\
 \underline{x^2} \\
 x + 0 \\
 \underline{x} \\
 0 \quad 0 \quad 0 \quad 0 \quad 0
 \end{array}$$

$$\begin{array}{r}
 x^3 + 1 \\
 x^4 + x^3 + x^2 + x
 \end{array}$$

CRC (*Cyclic-Redundancy Check*)

- Os erros não serão detetados, se e só se,

$$E(x) = G(x)Z(x)$$

para algum polinómio não-nulo $Z(x)$.

- Erros detetados pelos CRC:
 - Todos os erros de 1 bit.
 - Todos os erros de 2 bits, quando $G(x)$ tem um fator com pelo menos 3 termos.
 - Qualquer nº ímpar de erros, quando $G(x)$ tem um fator $(x+1)$.
 - Todas as rajadas de erros com um comprimento inferior ao comprimento do CRC.

CRC (*Cyclic-Redundancy Check*)

Norma	Polinómio Gerador $G(x)$
CRC-12	$x^{12}+x^{11}+x^3+x^2+x+1$
CRC-16 (ANSI)	$x^{16}+x^{15}+x^5+1$
CRC-16	$x^{16}+x^{15}+x^2+1$
CRC-CCITT (V.41)	$x^{16}+x^{12}+x^5+1$
CRC-32	$x^{32}+x^{26}+x^{23}+x^{22}+x^{16}+x^{12}+x^{11}+x^{10}+x^8+x^7+x^5+x^4+x^2+x+1$

Exemplos

Considere um emissor com os 4 bytes 01110101.01100101.01000101.00110000 para enviar com um código de paridade ímpar aplicado a palavras de 8 bits. Diga justificadamente qual a sequência de bits enviada.

Considere a recepção da sequência binária “1011100110” gerada com controle de erros através de um CRC com polinómio gerador $x^3 + x^2 + 1$. Determine justificando se o receptor assume que houve erros de transmissão ou não.

Considere que um emissor tem os 2 bytes 00001101.11100111 para enviar e usa um código CRC com o polinómio gerador $x^4 + x + 1$. Indique justificadamente qual a sequência de bits enviada.