



Digitalizzazione

Parte 2

Prof. Filippo Milotta
milotta@dmi.unict.it



Quantizzazione - Definizione

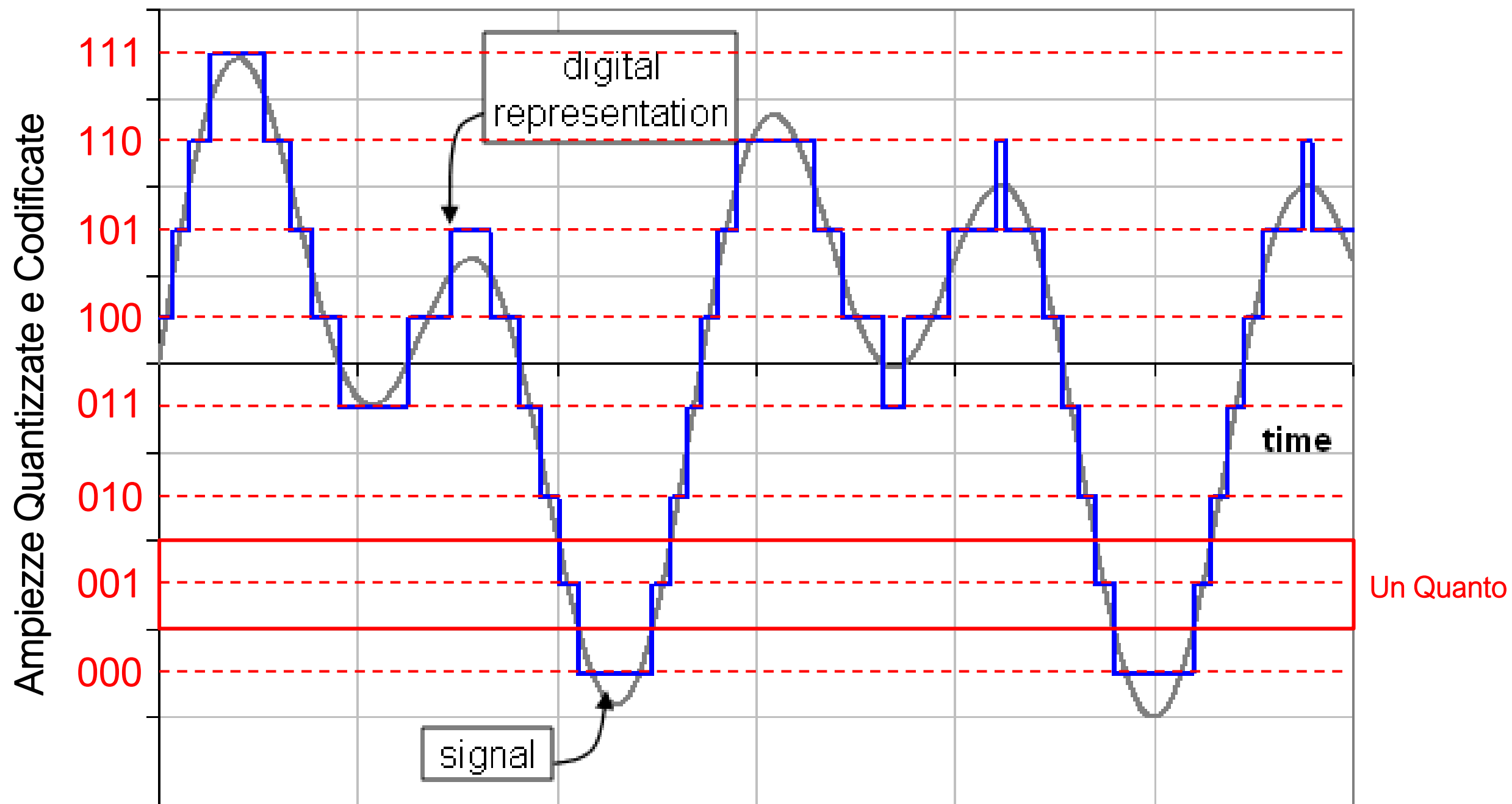
La trasformazione di un segnale a valori continui in un segnale a valori discreti, prende il nome di **quantizzazione**.

Ad ogni valore originale sarà associato un nuovo valore in un insieme discreto di livelli. Questo introdurrà un certo errore, poiché valori originariamente differenti possono collassare nello stesso livello, divenendo indistinguibili.

La **precisione** di quantizzazione è legata alla minima variazione nella grandezza originale che induce un passaggio da un livello ad un altro nel dominio quantizzato. Più è piccola la minima variazione necessaria, più sarà precisa la quantizzazione.



Quantizzazione - Esempio





Quantizzazione – Definizioni

Uniforme o lineare: la quantizzazione è uniforme quando ad intervalli di ampiezza uguale nel dominio originale, corrispondono un numero uguale di livelli di quantizzazione.

Non Uniforme o non lineare: la quantizzazione è non uniforme quando ad intervalli di ampiezza uguale nel dominio originale, corrispondono un numero diverso di livelli di quantizzazione. In altre parole si è più precisi nel quantizzare certi intervalli e meno precisi per altri.

Gamma dinamica (Valore di Fondo Scala): è la dimensione del range che si vuole rappresentare. Ad esempio, se si vuole rappresentare una tensione tra 5 e 20 volt:

$$VFS = 20 - 5 = 15 \quad \text{ossia} \quad VFS = Max - Min$$



Quantizzazione uniforme – Errore

La quantizzazione è un processo di approssimazione che come tale introduce una perdita e quindi una distorsione. Di norma, si associa ai valori che cadono nello stesso intervallo il valore di mezzo di tale intervallo. Per cui, nel caso di quantizzazione uniforme, **l'errore massimo di quantizzazione** E_{max} vale:

$$E_{max} = \frac{VFS}{2Q} \text{ dove } Q \text{ è uguale al numero di livelli di quantizzazione.}$$

Nel caso dell' audio digitale, il numero massimo di livelli (uniformi o meno) dipenderà dal **numero di bit** che decideremo di utilizzare per rappresentare i livelli di ampiezza (**profondità in bit**). Dunque se N è il numero di bit utilizzati:

$$Q = 2^N$$



Quantizzazione uniforme

Distorsione SQNR

Come nel caso dell'audio analogico è possibile stimare la distorsione introdotta dal trattamento del segnale originale. Un indice che tiene conto della distorsione introdotta dalla **quantizzazione** è il **Signal to Quantization Noise Ratio**. L'**SQNR medio** nel caso di **quantizzazione uniforme** può essere calcolato come segue:

$$SQNR = 2^N \times \sqrt{\frac{3}{2}}$$

Dove N è la **profondità in bit**.

In decibel:

$$SQNR_{dB} = 10 \log_{10} SQNR^2 = 6,02N + 1,7609$$

E' evidente che utilizzando un numero superiore di bit si quantizzi in maniera più precisa e si introduca una più bassa distorsione. Infatti un SQNR alto è indice di maggiore qualità.



Quantizzazione uniforme

Distorsione SQNR (dal testo)

In decibel:

$$SQNR_{dB} = 10 \log_{10} SQNR^2 = 6,02N + 1,7609$$

Questa relazione stabilisce che ogni bit contribuisce per circa 6dB al SQNR.

Es.:

Dato $N=8$, l'SQNR sarà circa $6 \cdot 8 = 48\text{dB}$

Dato un SQNR pari a circa 120dB, allora $N = 120/6 = 20$



Quantizzazione uniforme – Distorsione

SQNR

Il valore SQNR così calcolato è inteso per l'errore di quantizzazione medio!
Cosa vuol dire?

- A parità di errore di quantizzazione, le ampiezze più piccole in valore assoluto sono distorte maggiormente (in proporzione), rispetto alle ampiezze più grandi.
- Per essere precisi bisognerebbe calcolare la distorsione per ogni possibile valore di ampiezza.
- L'indice SQNR è calcolato considerando l'ampiezza media quadratica di un segnale (RMS)



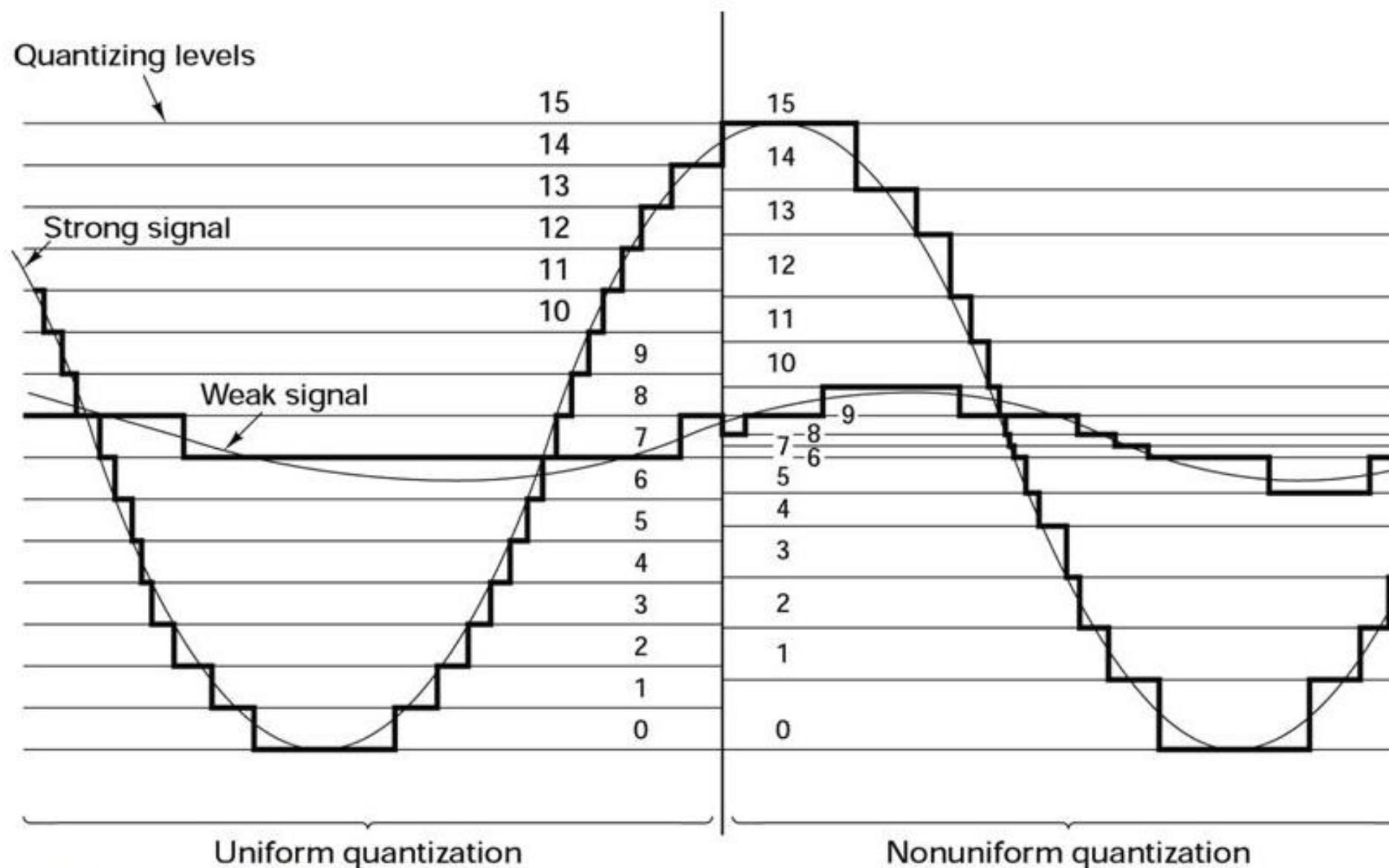
Quantizzazione non uniforme

- Le considerazioni fatte per l'errore di quantizzazione, che pesa più sulle ampiezze piccole ci suggeriscono un'idea.
- Se a queste uniamo il fatto che gli esseri umani percepiscono meglio variazioni di volume che interessano le ampiezze basse...
- ... capiamo che la quantizzazione non uniforme potrebbe aiutarci a migliorare la qualità complessiva a parità di bit utilizzati.



Quantizzazione non uniforme

Uniform and Non-uniform Quantization





Audio digitale

A questo punto abbiamo tutte le grandezze necessarie a caratterizzare un segnale audio digitale:

- ❑ **Numero di canali;**
- ❑ **Tasso di campionamento** (numero di campioni al secondo);
- ❑ **Profondità in bit per campione** (bit di quantizzazione per valore).

Quanti bit sono necessari in memoria per rappresentare un segnale audio digitale?



Audio digitale – Spazio in memoria

Sia f_c la frequenza di campionamento, N la profondità in bit, D la durata del flusso audio e C il numero di canali, allora il numero di bit necessari a rappresentare il segnale (senza compressione) si calcola:

$$Size = f_c \times N \times D \times C$$

Il numero di bit che fluisce nell'unità di tempo (un secondo) prende il nome di **bit rate**. Si misura in **bps** (bit per secondo).

$$bitrate = f_c \times N \times C$$

E' chiaro che l'obiettivo è garantire una buona qualità utilizzando la minima quantità di memoria. I metodi di compressione rappresentano un passo successivo che permette di abbassare il **bit rate** preservando la qualità.



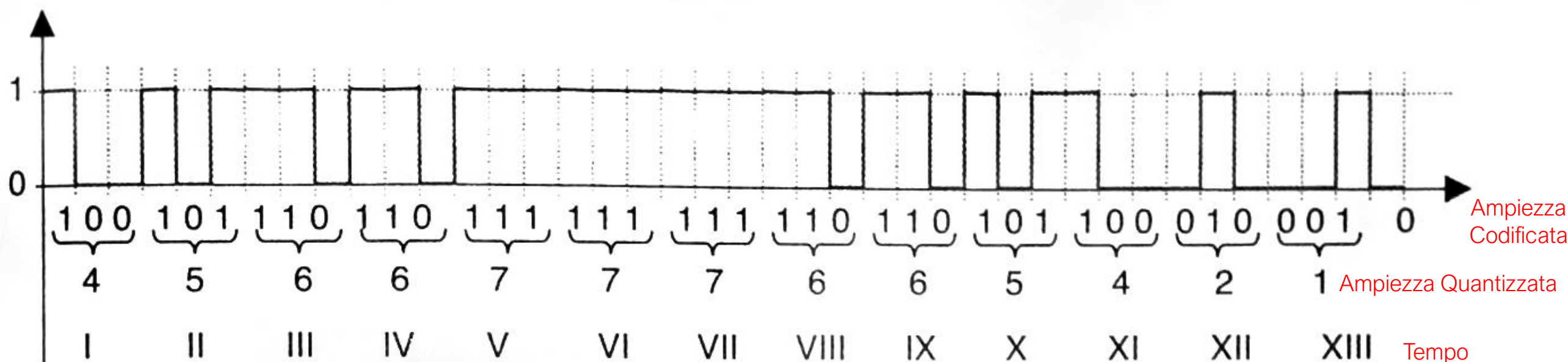
Audio digitale – Qualche Standard

Supporto	Frequenza di campionamento (Hz)	Bit per campione	SQNR (dB)	Canali	Bitrate (KBps)
Telefono	8000	8	48	1	8.00
Radio AM	11025	8	48	1	11.05
Radio FM	22050	16	96	2	88.20
CD Audio	44100	16	96	2	176.40
Digital Audio Tape (DAT)	48000	16	96	2	192.00
DVD Audio	192000	24	144	6	1200.00



Audio Digitale – Codifica PCM

La **Pulse Code Modulation** (PCM), è forse la più semplice tecnica di codifica di un audio digitale. In effetti non si fa altro che considerare ogni singolo campione come un impulso e associarvi una parola binaria che ne rappresenta l'ampiezza. La lunghezza delle parole binarie dipende ovviamente dai bit di quantizzazione (lineare) utilizzati.



Nell'esempio si può osservare la codifica PCM a 3 bit di un segnale audio. I 13 campioni assumono valori tra 0 e 7.



Codifiche del segnale (dal testo)

- La differenza fondamentale dal punto di vista della rappresentazione binaria è se la codifica è con segno o senza segno

ID Quanto	Offset binary	Comp. a 2	Segno e Magnitudo
7	$111_2 (7_{10})$	$011_2 (+3_{10})$	$011_2 (+3_{10})$
6	$110_2 (6_{10})$	$010_2 (+2_{10})$	$010_2 (+2_{10})$
5	$101_2 (5_{10})$	$001_2 (+1_{10})$	$001_2 (+1_{10})$
4	$100_2 (4_{10})$	$000_2 (+0_{10})$	$000_2 (+0_{10})$
3	$011_2 (3_{10})$	$111_2 (-1_{10})$	$100_2 (-0_{10})$
2	$010_2 (2_{10})$	$110_2 (-2_{10})$	$101_2 (-1_{10})$
1	$001_2 (1_{10})$	$101_2 (-3_{10})$	$110_2 (-2_{10})$
0	$000 (0_{10})$	$100_2 (-4_{10})$	$111_2 (-3_{10})$



Error Correcting Codes (ECC)

I bit di parità

0 0 0 1 0 1 0 0 -> 2 bit hanno valore 1 -> bit di parità impostato a (0)

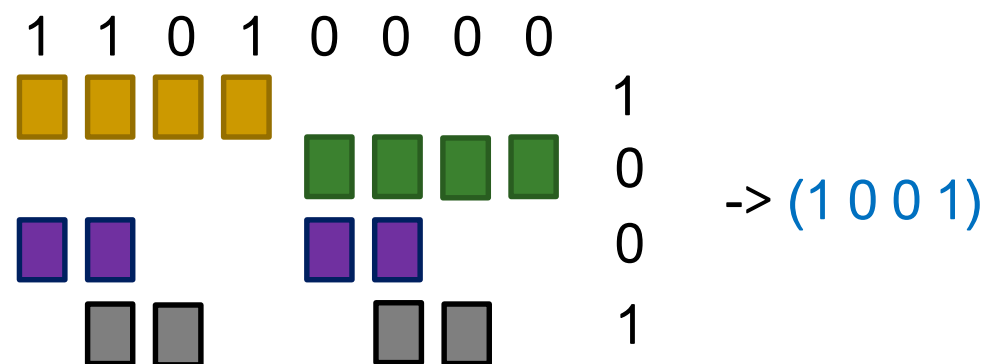
0 0 0 1 0 1 1 0 -> 3 bit hanno valore 1 -> bit di parità impostato a (1)

Si verifica 1 errore...

0 0 0 1 1 1 1 0 (1) -> 4 bit hanno valore 1 -> mi aspetto (0) ma ho (1) -> Errore rilevato! Ma... dove?

Il bit di parità è una tecnica perfetta? Che problemi potrebbero insorgere?

Vediamo una configurazione più complessa con 4 bit di parità:





Esercitazione Pratica (dal testo)

■ 3.11.6 – Equalizzatore grafico

In un editor audio registrare una frase usando un microfono

- Accedere allo strumento Equalizzatore
 - Su Audacity, si trova sotto il menù Effetti...
- Applicare un filtro *midcut* (a V)
- Applicare un filtro *midpass* (inverso del midcut)
- Applicare un filtro telefono