

# Codifica Audio

*Corso di Fondamenti di Telecomunicazioni*

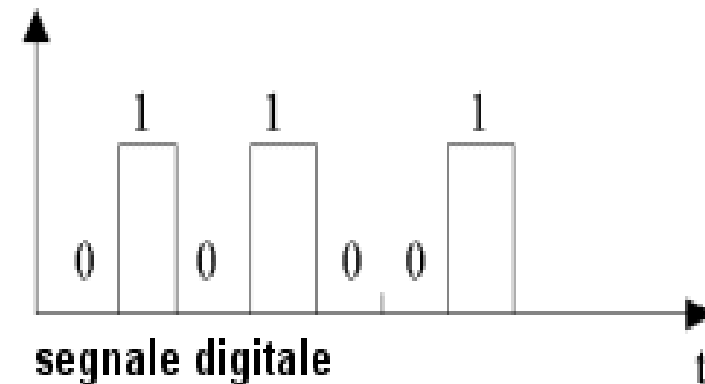
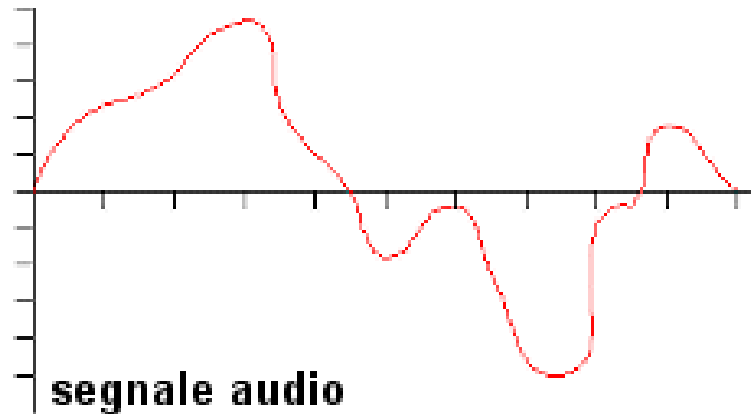
*Anno Accademico 2009/2010*

# Contenuti

- Caratteristiche del segnale audio
- Codifica Audio (Voce), standard 3G
- Codifica Audio (Musica), standard DVB
- Codifica Musica Elettronica, standard MIDI

# Il segnale audio

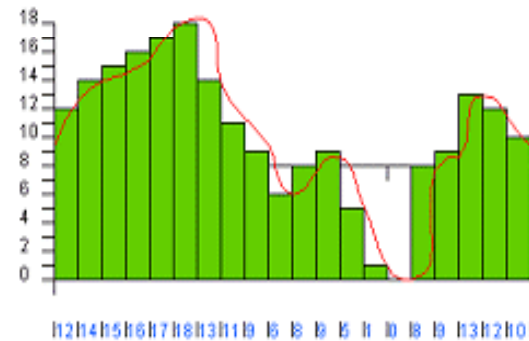
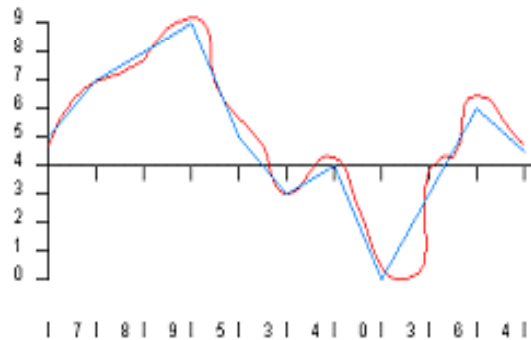
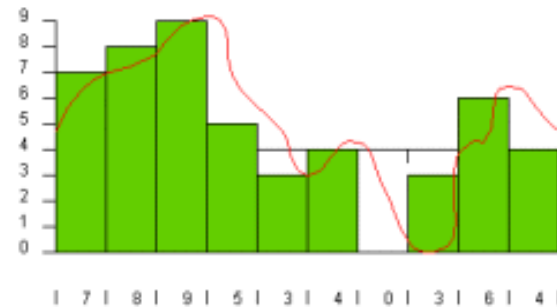
Il segnale audio è per sua natura un segnale analogico, ovvero un segnale che varia in modo continuo nel tempo.



Riuscire a rappresentare un segnale audio con un segnale digitale preservando l'informazione è un argomento che è alla base della teoria dell'informazione.

# Campionamento

La risposta sta nel campionare il segnale, ossia prelevare, ad intervalli regolari, il valore del segnale audio.



All'aumentare della risoluzione, ovvero del numero di rettangoli (passo di campionamento), l'errore di conversione diminuisce.

# Qualità CD

Le tracce audio digitali di un CD sono immagazzinate in file binari ed in ogni secondo vengono prelevati ben 44100 campioni e ogni campione è quantizzato con 16 bit.

Considerando due canali, sinistro e destro, si avrà un bit-rate complessivo pari a:  $44100 * 16 * 2 = 176400$  bytes/secondo valore che corrisponde alla velocità di lettura "1x" dei normali lettori CD audio.

Tale bit-rate, a partire dalla codifica finora discussa, porta alla qualità sonora nota come "qualità CD", che è lo standard di riferimento quando si valutano i risultati prodotti dagli algoritmi di compressione.

# Codifica audio

Nel caso dell'audio, delle immagini e dei filmati, un certo livello di degradazione è un compromesso accettabile per ridurre (e di molto) l'occupazione o la banda richiesta dal file.

Le codifiche di compressione dell'audio sono numerose ed utilizzano tecniche anche molto differenti l'una dall'altra:

- le *codifiche nel dominio del tempo* (lossless),
- le *codifiche per modelli* (lossy),
- le *codifiche nel dominio della frequenza* (lossy).

Generalmente le prime due categorie di algoritmi vengono usate per comprimere il segnale vocale mentre alla terza appartengono algoritmi come MP3, WMA, AAC ottimi per la compressione della musica.

# Codifiche temporali

Algoritmi che elaborano il segnale campionato direttamente, senza estrarre le informazioni spettrali (frequenze).

L'obiettivo è quello di trovare correlazioni tra i campioni e/o proprietà dalla sorgente e della destinazione che permettano di ridurre il numero di bit usati per descrivere il valore di un campione audio.

Sono storicamente le prime ad essere state elaborate, hanno bassa efficienza e sono state ampiamente superate dai nuovi algoritmi. Le più note sono il **DPCM** e l'**ADPCM** (evoluzioni della PCM).

# Codifiche per modelli

Le codifiche per modelli sono tecniche legate ad una particolare sorgente sonora (in questo caso la voce) che si tenta di emulare tramite un modello più o meno semplificato.

Le corde vocali e la gola hanno delle ben precise caratteristiche fisiche, il loro comportamento sarà quindi predicibile sulla base di un modello.

Queste codifiche rappresentano una scelta ottimale per la compressione della voce, tanto che vengono utilizzate nella telefonia mobile (GSM, UMTS) e anche su Internet. Le più famose sono **LPC** e **CELP**.



# Linear Predictive Coding



**FINORA:** campionamento della forma d'onda e invio dei campioni (PCM) o delle differenze (DPCM).

## **LPC:**

**ENCODER:** analisi delle caratteristiche del segnale audio (perceptual features) e invio al decoder.

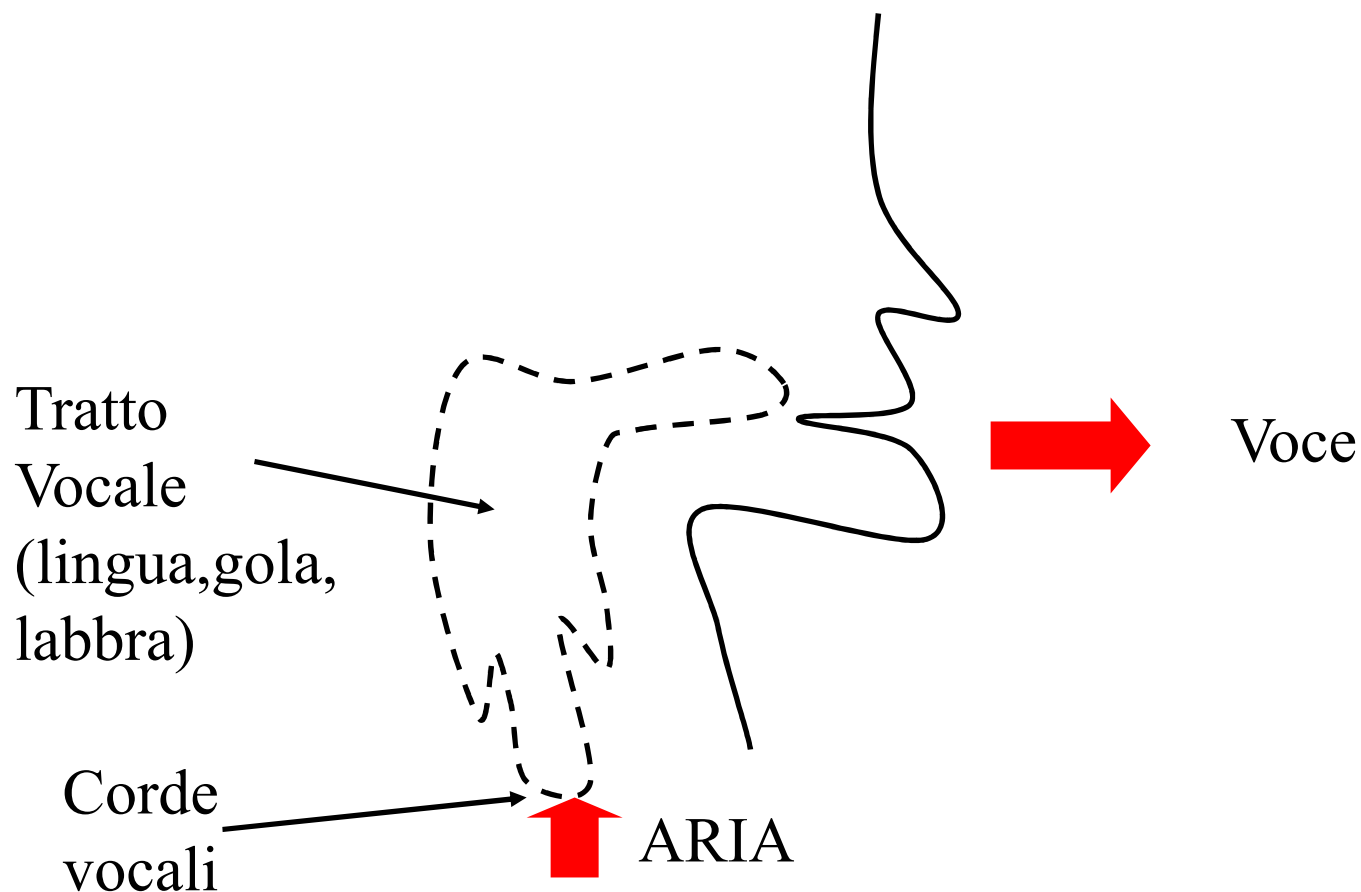
**DECODER:** uso di un sintetizzatore audio con le perceptual features come parametri di ingresso.

Il timbro di voce è scarsamente riconoscibile nel segnale riprodotto: il suono è artificiale ma si raggiungono elevati livelli di compressione (qualche kbps).

# Perceptual Features

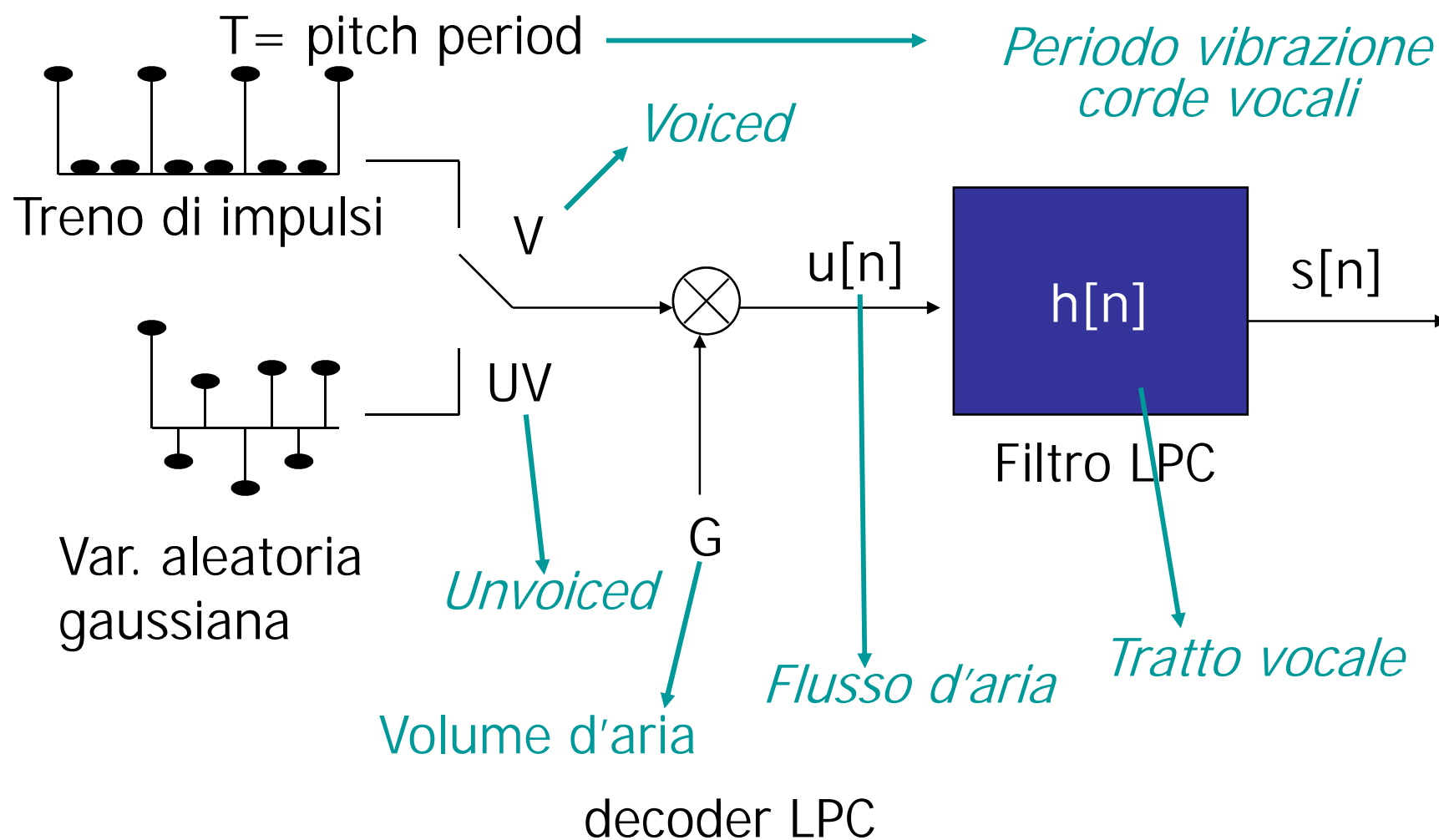
1. TIMBRO (PITCH). Frequenza del segnale. L'orecchio è molto sensibile al range 2-5 kHz piuttosto che al range superiore o inferiore.
2. PERIODO. Durata del segnale
3. VOLUME. Ammontare di energia nel segnale
4. PARAMETRI DI ECCITAZIONE DEL TRATTO VOCALE:
  - SUONI VOICED. Generati attraverso le corde vocali aria + vibrazione corde ad una certa frequenza (pitch): We, Yes
  - SUONI UNVOICED. Generati tenendo le corde vocali aperte: flusso d'aria modulato dal tratto vocale (corde non vibrano): Say, Tutto

# Modello Fisico



La forma del tratto vocale cambia lentamente (ogni 10-100 ms).

# Modello Matematico



# Compressione con LPC

Cosa deve trasmettere l'encoder?

1. guadagno  $G$
2. il flag  $V/UV$
3. il *pitch period*  $T$
4. i coefficienti del filtro LPC (es. 10 in LPC-10)

Anziché i campioni del segnale di ingresso



compressione

# Parametri LPC

- A 8 kHz, 20 ms corrispondono a 160 campioni
- Il segnale di ingresso viene diviso in *finestre* da 20 ms
- In quei 20 ms, l'insieme dei segnali

$$A = (a_1, a_2, \dots, a_{10}, G, V/UV, T)$$

È equivalente a

$$S = (s(0), s(1), s(2), \dots, s(159))$$

Cioè anziché 160 valori solo 13!!!!

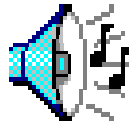
LPC SYNTHESIS:

passare da A ad S (mediante filtering)

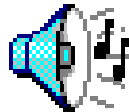
LPC ANALYSIS:

passare da S ad A (mediante metodi matematici)

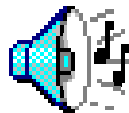
# Esempi di Compressione



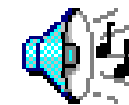
- Codifica PCM (64 kbit/s)



- Codifica LPC (MELP) (4.8 kbit/s)



- Codifica GSM half-rate ( $\approx 6$  kbit/s)



- Codifica GSM full-rate ( $\approx 12$  kbit/s)

# Codifiche in Frequenza

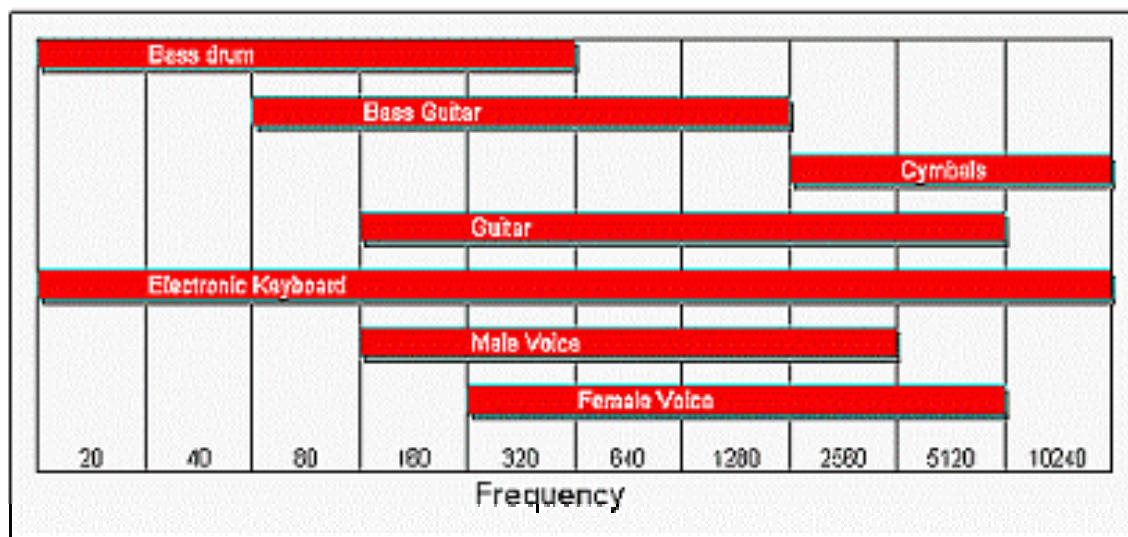
Questi algoritmi sono accomunati dal fatto di esaminare e processare il segnale non nel dominio del tempo, ma nel dominio della frequenza.

Ogni strumento musicale, ogni suono e anche la voce ha una propria impronta spettrale caratteristica costituita da una combinazione di frequenze contenute in uno spettro più o meno ampio.

Lavorando su tale spettro è possibile comprimere il segnale in misura molto maggiore di quanto non si riesca a fare nel dominio del tempo.



# Voce e Musica

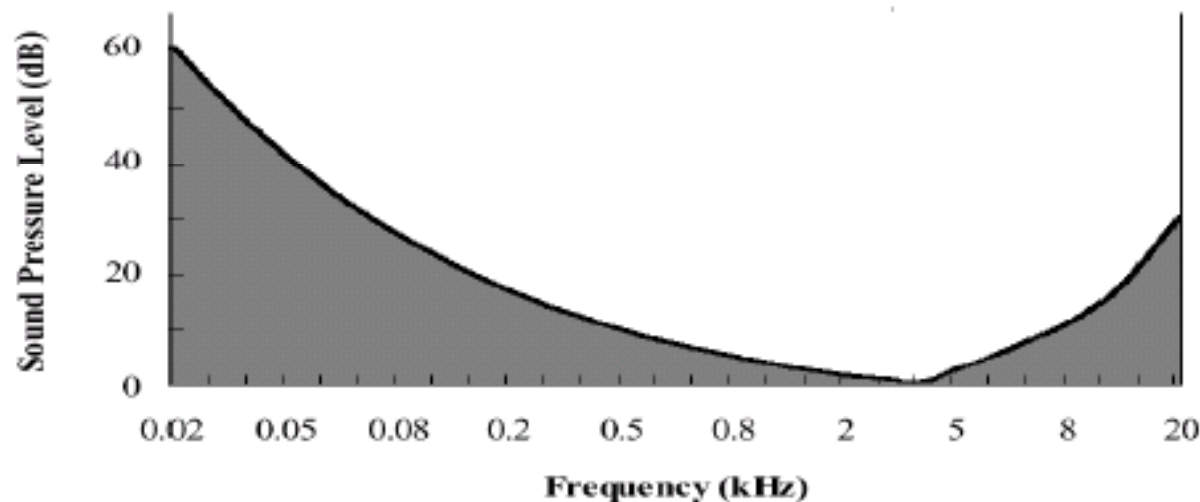


La voce umana occupa solo certe frequenze mentre gli strumenti musicali spaziano su range diversi a seconda del tipo di strumento impiegato.

Comprimere una musica generica utilizzando un approccio basato su modelli (LPC) risulterebbe estremamente complesso. La musica dovrà essere "ascoltata da un orecchio" quindi conoscere fino in fondo quello che un uomo medio riesce a sentire o non riesce a sentire può rivelarsi sorprendentemente utile.

# Modello Percettivo

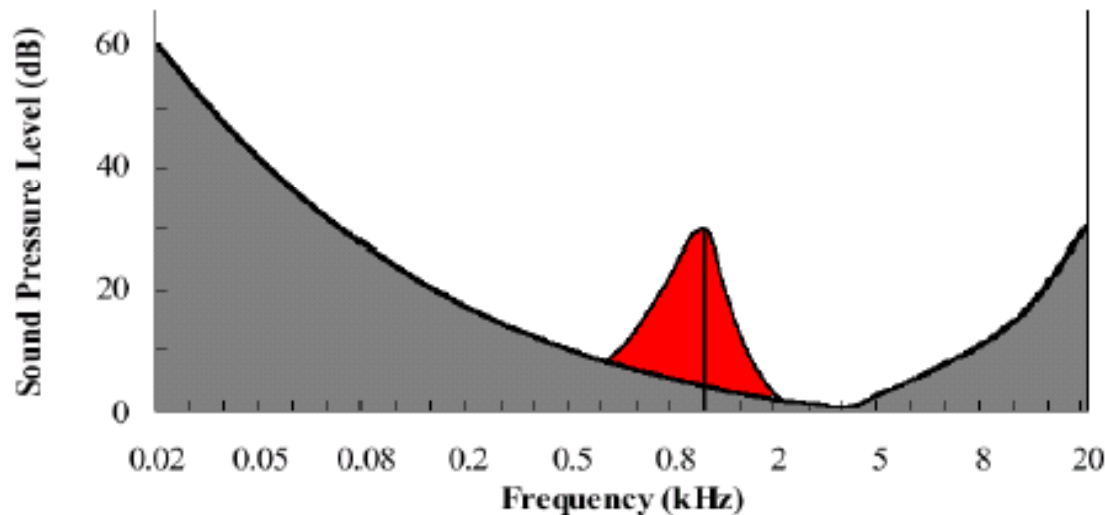
Il nostro orecchio non è uno strumento lineare, non percepisce tutti i suoni e soprattutto non li percepisce nello stesso modo. Da qui l'idea di eliminare tutte quelle componenti frequenziali che non possiamo udire.



Dal grafico emerge che l'orecchio umano è maggiormente sensibile alle frequenze comprese fra 2 e 4 KHz. In sostanza si tagliano le alte frequenze e le bassissime frequenze.

# Mascheramento Audio

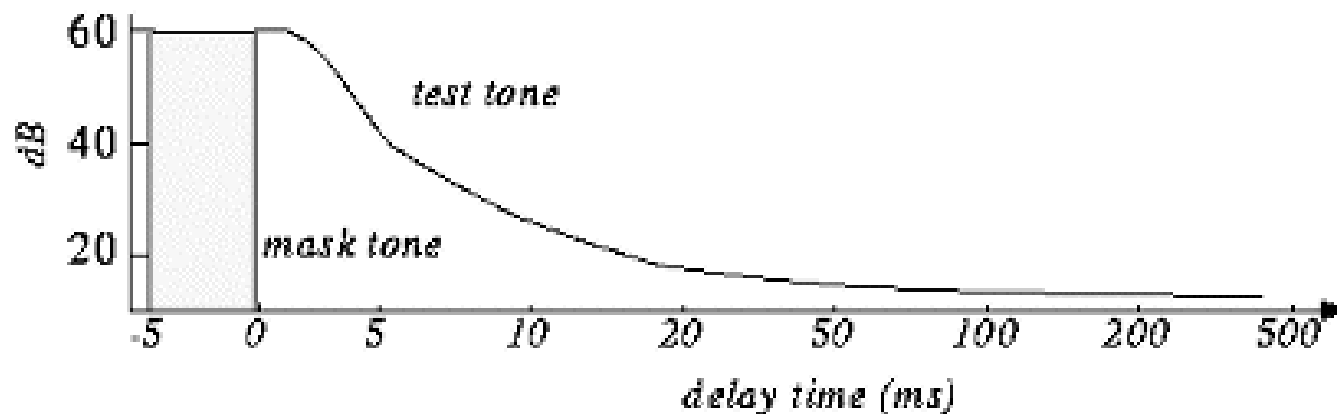
La percezione del suono non è costante nel tempo, varia in funzione di ciò che ascoltiamo. Nella figura è riportata la variazione della caratteristica in presenza di un suono.



In pratica un tono forte copre i suoni d'intensità minore non solo ad una determinata frequenza ma anche in quelle vicine.

# Mascheramento Temporale

Supponiamo di avere al solito due toni, uno forte e l'altro, che gli è vicino in frequenza, piuttosto debole. Dall'analisi vista prima percepiremo solo il tono più forte (tono maschera).



Se improvvisamente questo tono maschera cessa di esistere, la membrana del nostro timpano deve assestarsi. Impiegheremo quindi del tempo prima di riuscire a sentire il tono più debole anche dopo che il tono forte è morto.

# MPEG-1

La tecnica di codifica permette di selezionare diverse opzioni, primo fra tutti il numero di canali e la loro funzionalità:

- a canale singolo;
- a due canali;
- stereo semplice;
- stereo congiunta.

Le frequenze di campionamento possono essere di 32 kHz, 44.1 kHz o 48 kHz, il bitrate varia da 16 a 320 kbit/sec.

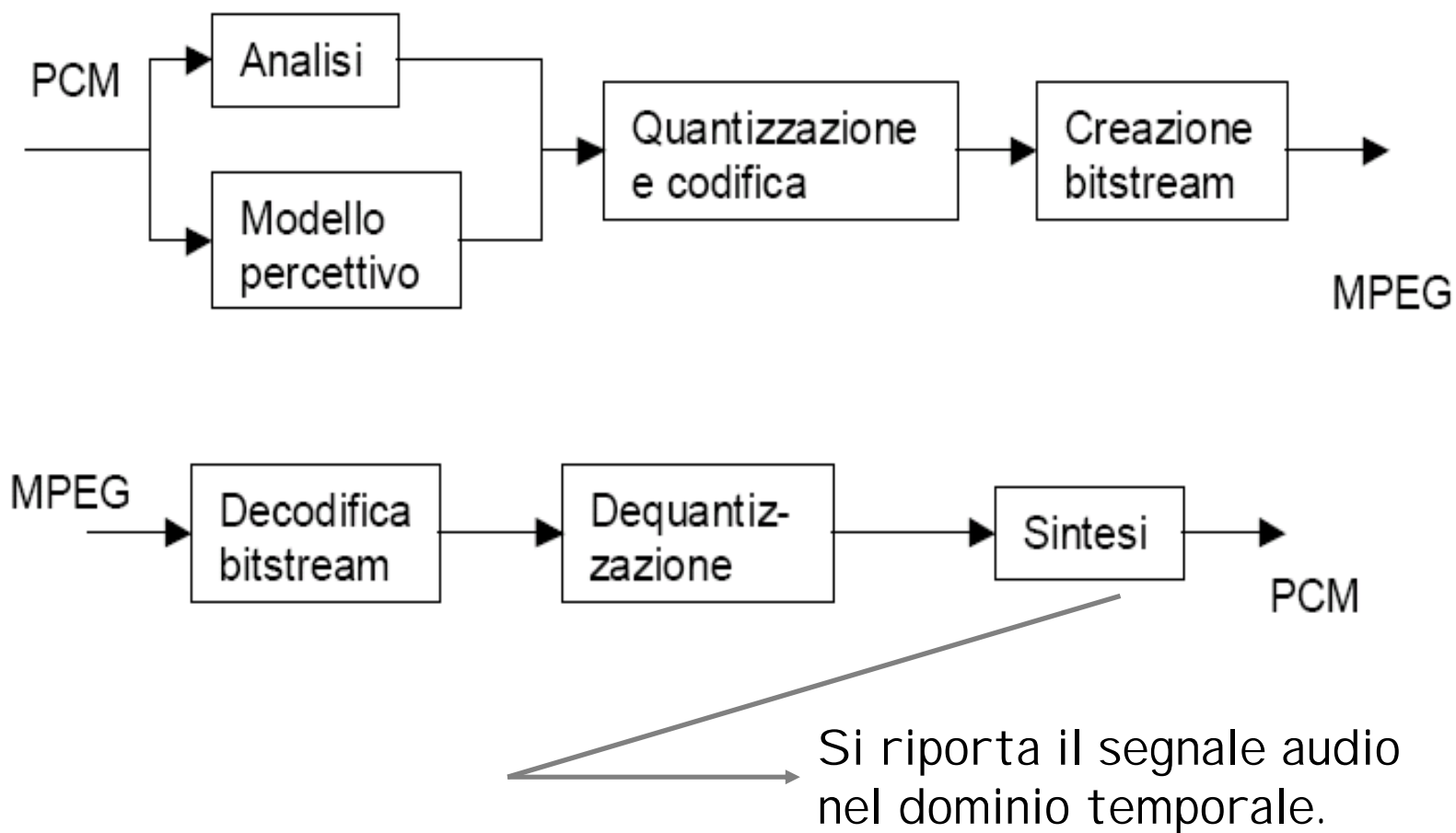
# MPEG-1

**Layer I:** è il più semplice dei tre ed è studiato per avere le migliori prestazioni con bitrate sopra 128 kbit/s per canale. Fornisce fattori di compressione di circa 1 a 4.

**Layer II:** superiore per complessità al primo è adatto per bitrate intorno a 128 kbit/s per canale. I fattori di compressione vanno da 1 a 6 fino a 1 a 8.

**Layer III (MP3):** è il più complesso dei tre ed offre ottime prestazioni con bitrate di circa 64 kbit/s per canale. Riesce a ridurre la dimensione fino a 12 volte.

# Schema di co/decodifica



# Codificatore

- **Analisi:** si trasforma il segnale rappresentandolo nel dominio delle frequenze.
- **Modello percettivo:** indica quali componenti spettrali possono essere scartate.
- **Quantizzazione e codifica:** incrementa o decrementa il valore d'ogni campione fino a portarlo a livelli standard e poi lo codifica con un certo numero di bit.
- **Creazione dello stream:** ai dati relativi alla finestra si aggiunge un'intestazione con le informazioni necessarie alla decodifica.



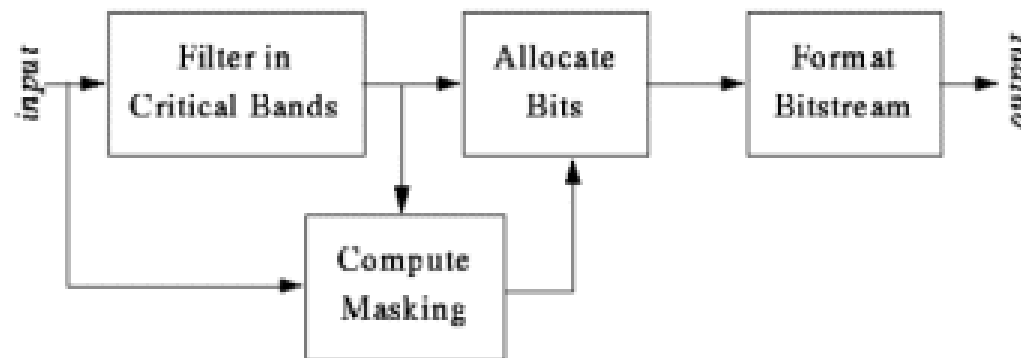
# Quantizzazione e Codifica

**Modello percettivo:** genera per ogni pacchetto la Threshold Mask, che sarà utilizzata in fase di campionamento per eliminare le componenti frequenziali non necessarie.

**Quantizzazione:** operazione che inevitabilmente introduce rumore. Il numero di livelli di quantizzazione è ottenuto, per ogni componente spettrale, da un algoritmo d'allocazione dinamica dei bit. Questo sceglie il quantizzatore all'interno di quelli disponibili, lavorando in modo da ottenere il bitrate richiesto e da rispettare le imposizioni della Threshold Mask.

# MP3 - Funzionamento

Per ognuna delle sottobande, viene calcolata l'entità del mascheramento causata dalle bande adiacenti. Se la potenza in una sottobanda è sotto la soglia di mascheramento, allora non viene codificata in uscita l'informazione poiché sarebbe inudibile.



Altrimenti, occorre calcolare il numero di bit necessari per rappresentare l'informazione della sottobanda facendo attenzione che il rumore introdotto stia sotto la soglia.

# MP3-Esempio

Consideriamo per semplicità solo 16 delle 32 sottobande:

Banda	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Livello(dB)	0	8	12	10	6	2	10	60	35	20	15	2	3	5	3	1

Dai dati del nostro modello psicoacustico, sappiamo che l'ottava banda, se presenta una intensità di 60dB, genera una maschera di 12dB nella settima e di 15dB nella nona. La settima ha un livello pari a 10 (<12dB), ed è perciò mascherata e tagliata via dall'uscita. La nona è a 35dB (>15) così passa in uscita.

# MP3 - Prestazioni

Sound quality	Bandwidth	Mode	Bitrate	Reduction ratio
better than short-wave	4.5 kHz	mono	16 kbit/s	48:1
better than AM radio	7.5 kHz	mono	32 kbit/s	24:1
similar to FM radio	11 kHz	stereo	56..64 kbit/s	26..24:1
near-CD	15 kHz	stereo	96 kbit/s	16:1
CD	>15 kHz	stereo	112..128 kbit/s	14..12:1

I dati non sono “assoluti”, la riduzione varia in funzione del suono da comprimere e del codificatore utilizzato. I dati riportati sono stati recuperati dal *Fraunhofer Institut* e si riferiscono al loro codificatore.

# MPEG-2



Con il passare del tempo è divenuto chiaro che due canali non sono sufficienti a coprire tutte le applicazioni, per permettere la codifica multicanale è stato sviluppato il secondo standard.

L'impulso principale è stato dato dalla necessità di codificare l'audio che rispondesse alle richieste del settore cinematografico-televisivo, con almeno 5 canali: destro, sinistro, centrale, destro e sinistro surround.

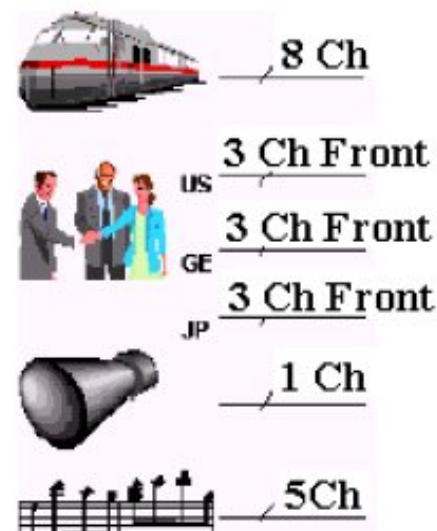
Il progetto MPEG-2 si articola in due fasi successive, nella prima si è privilegiata la compatibilità con gli standard precedenti. Nella seconda questa è stata sacrificata, si era compreso che abbandonare la compatibilità era l'unica via per ottenere un notevole incremento nelle prestazioni.

# MPEG-2 AAC

La seconda fase di MPEG-2, denominata AAC (Advanced Audio Coding), parte con l'obiettivo di sviluppare un tool per migliorare le prestazioni nella codifica multicanale.

Il risultato ottenuto è notevole, si possono codificare fino a 48 canali con una frequenza di campionamento che va da 8 a 96 kHz per canale. Per comprendere il livello raggiunto si tenga presente che alcuni test dimostrano come, a parità di qualità e nel caso di codifica a 5 canali, l'AAC in pratica dimezza il bitrate rispetto a MPEG-2 Layer II.

# MPEG-4



Seguendo l'evoluzione informatica, MPEG-4 introduce il concetto di “oggetto” nel settore audiovisivo. Ogni file multimediale è una combinazione di diversi oggetti che, pur potendo esistere separatamente sono armonizzati per ottenere l'effetto complessivo.

# Estensione .mid

Nato come standard per la comunicazione tra strumenti musicali, si è in seguito evoluto in una maggiore standardizzazione denominata General MIDI.

In un file MIDI sono contenute istruzioni che comunicano alla scheda audio di modulare la frequenza in modo da produrre una particolare nota che abbia una timbrica simile a quella di un pianoforte e che duri una certa quantità di tempo.

Un intero brano musicale, con tutte le parti, della durata di svariati minuti, può occupare qualche decina di Kbyte, dal momento che ogni singolo evento MIDI occupa soltanto 11 byte.



# La Storia

Nonostante siano passati più di vent'anni, il protocollo MIDI rimane indiscutibilmente il linguaggio informatico che ha più cambiato il modo di eseguire e produrre musica.

Il MIDI, acronimo di **musical instruments digital interface**, nasce principalmente da due fattori:

1. ricerca di compatibilità tra strumenti diversi e di diverse marche;
2. crescente disponibilità di tecnologia digitale.

Prima, ogni costruttore ottemperava a proprie specifiche, nella definizione delle caratteristiche elettriche, che quasi mai corrispondevano a quelle definite da altri costruttori.

# Cronologia di Eventi

1981: fu presentato un progetto di *Universal Synthesizer interface*.

1982: fu posto in commercio il *Prophet 600*, primo synth al mondo dotato del nuovo modo di dialogare in musica.

1983/1984: diffusione di massa dei sintetizzatori digitali MIDI.

1985: pubblicazione delle specifiche dettagliate dello standard MIDI 1.0, debitamente riviste e corrette.

Le appendici più significative sono state poi pubblicate nell'autunno 1986 e nella primavera 1987, ma ancora adesso si sta lavorando alla definizione di nuovi messaggi destinati a consentire lo scambio universale di file di dati MIDI.

# Il Protocollo

- Stabilisce le specifiche sia *hardware* (interfaccia, cavi, connettori), sia *software* (linguaggio informatico, modalità di trasmissione, tipologia dei messaggi) che ogni apparecchiatura deve “capire” se vuole essere veramente MIDI compatibile.

Scopo: trasmettere comandi.

Quando sulla nostra tastiera premiamo un tasto, dalla presa MIDI out uscirà un comando digitale che indicherà ad un'apparecchiatura in ricezione che è stato premuto quel determinato tasto (numero di nota) con una determinata forza (velocity).

# Il Protocollo

Il MIDI **non** trasmette nessun tipo di suono ma unicamente comandi che verranno eseguiti dall'apparecchiatura in ricezione.

Ogni movimento fatto dall'esecutore su di una tastiera MIDI verrà codificato in un modo univoco secondo il protocollo: abbassare o rilasciare un tasto e con che forza, il muovere una rotella o un altro controllo, il cambiare timbro...

Questo sistema permette di comandare con una sola tastiera più generatori sonori collegati (*concerto*) o la possibilità di memorizzazione attraverso computer o sequencer dedicato dei dati di esecuzione di una partitura suonata su una tastiera MIDI (registrazione in studio).

# L'Hardware

Il componente principale è l'**UART** (**universal asynchronous receiver transmit**) un microprocessore appositamente costruito alla comprensione e decodificazione MIDI che lavora in maniera asincrona cioè solo quando qualcosa appare alle porte MIDI.

Le prese MIDI sono tre:

- ingresso** (MIDI In, dati in ricezione)
- uscita** (MIDI Out, dati in uscita dalla macchina)
- Midi Thorough**: si limita a ripetere fedelmente in uscita tutti i dati che appaiono all'ingresso MIDI In.

# La Trasmissione Dati

Per il sistema MIDI, anche se la trasmissione parallela è molto più veloce, è stato scelto il tipo di trasmissione seriale, per semplificare i collegamenti e aumentarne l'affidabilità.

**Vantaggi:** basta un solo filo per trasmettere l'informazione, per cui il collegamento risulta economico, pratico e affidabile.

**Efficienza:** la velocità della trasmissione deve essere abbastanza alta (*Trasmissione dati a 31.25 Kbaud, cioè 31.250 bit al secondo*).

**Trasmissione Asincrona:** l'inizio e la fine di ogni byte devono essere "annunciati" ogni volta da due bit speciali, lo *start bit* e lo *stop bit*, posti rispettivamente davanti e dietro gli otto bit del byte MIDI da trasmettere (è necessario trasmettere 10 bit per inviare un byte MIDI, quindi tale byte richiederà 320 msec prima di essere ricevuto).