CMPN206: Multimedia

Lecture 11: Audio Compression

Mohamed Alaa El-Dien Aly
Computer Engineering Department
Cairo University
Spring 2014

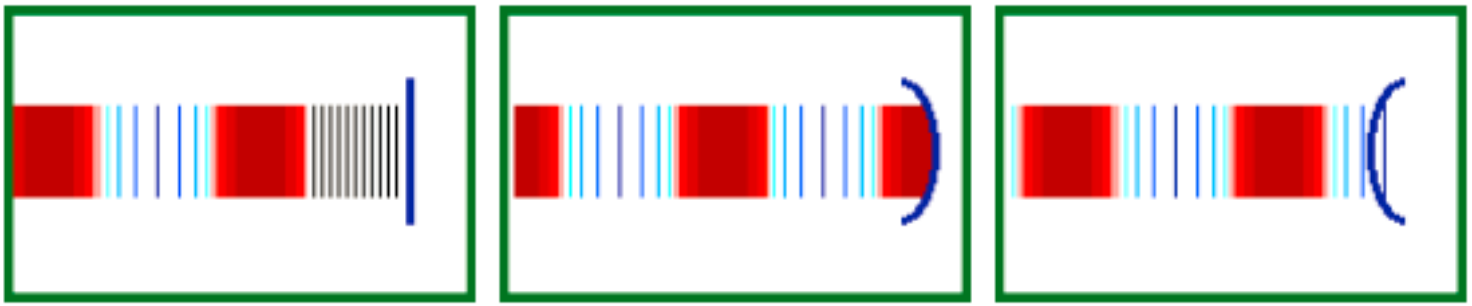
Agenda

- Sound
- Speech Compression
  - DPCM
  - ADPCM
  - Vocoders
- Audio Compression
  - Psychoacoustic
  - MPEG Layer I, II, and III
  - Other coders

Acknowledgments: Most slides are adapted from Richard Ladner, from Li and Drew, from Khaled Sayood, and from David Marshall and Kirill Sidorov.

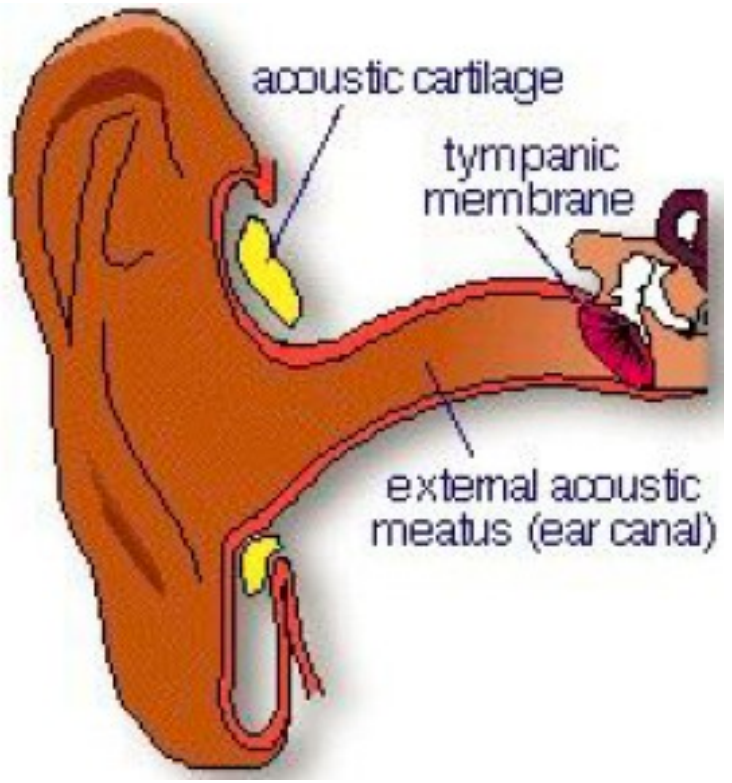
Sound

- Sound is produced by a *vibrating* source
- The vibrations disturb air molecules.
- This produces variations in *air pressure*: lower than average pressure, *rarefactions*, and higher than average, *compressions*. This produces sound waves.
- When a sound wave impinges on a surface (e.g. eardrum or microphone) it causes the surface to vibrate accordingly
- In this way *acoustic energy* is transferred from a source to a receptor



Human Hearing

- Upon receiving the waveform, the *ear drum* vibrates
- Acoustic energy then transfers from the ear drum through the *auditory nerve* to the brain
- The brain *perceives* the signals and interprets the sound



Sound Signals

- Sound signals are analog, and need to be digitized:
  - *Sampling*: digitization in the *time* domain to get a discrete set of *samples*
  - *Quantization*: digitization in the *range* domain to get a discrete set of sample *values*

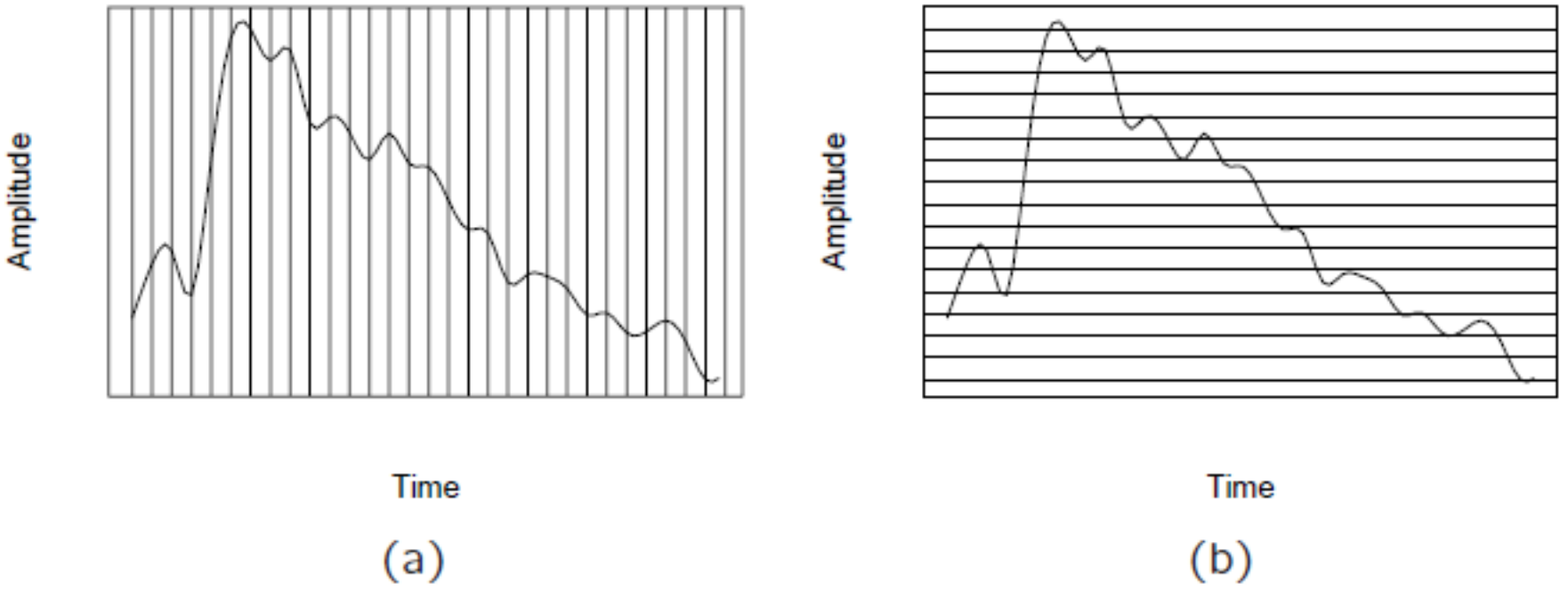


Fig. 6.2: Sampling and Quantization.

Sound Signals

Table 6.2: Data rate and bandwidth in sample audio applications

Quality	Sample Rate (KHz)	Bits per Sample	Mono/ Stereo	Data Rate (uncompressed) (kB/sec)	Frequency Band (KHz)
Telephone	8	8	Mono	8	0.200-3.4
AM Radio	11.025	8	Mono	11.0	0.1-5.5
FM Radio	22.05	16	Stereo	88.2	0.02-11
CD	44.1	16	Stereo	176.4	0.005-20
DAT	48	16	Stereo	192.0	0.005-20
DVD Audio	192 (max)	24 (max)	6 channels	1,200.0 (max)	0-96 (max)

Speech Signals

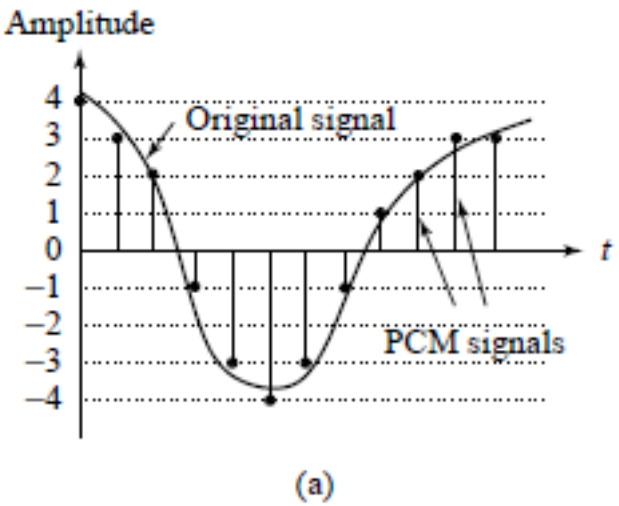
- *Speech* signals are a special kind of audio signals
  - They have a much limited *bandwidth*: from 50 Hz to about 10 kHz
  - They have special *source models*: we can use a mathematical model for the process of *generating* the speech signal by the vocal tract, nose, and mouth. This can be used to devise special techniques for coding speech signals efficiently

Speech Compression

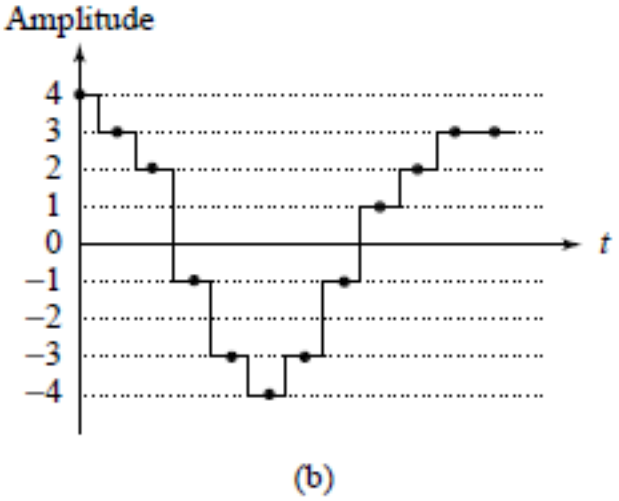
- Pulse Code Modulation (PCM)
- Differential PCM (DPCM)
- Adaptive DPCM (ADPCM)
- Voice Coders (Vocoders)

PCM

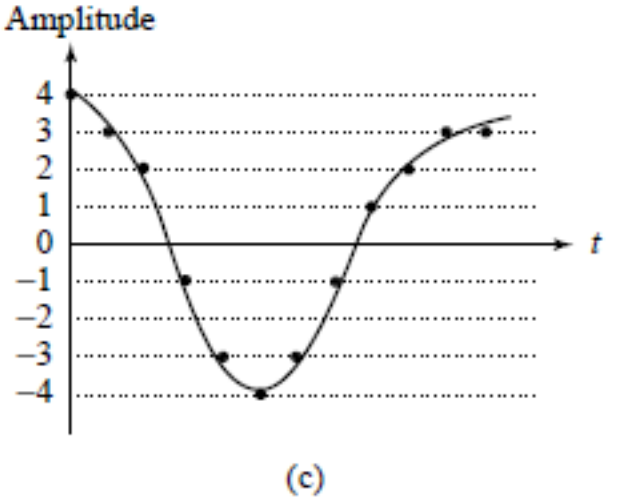
- Telephony assumes voice signals have a bandwidth of 4 kHz
- Thus they are sampled with a *Nyquist* frequency of 8 kHz
- Using 8 bits per sample, this becomes 64 kbps = 8 kB/s



Original and PCM signal



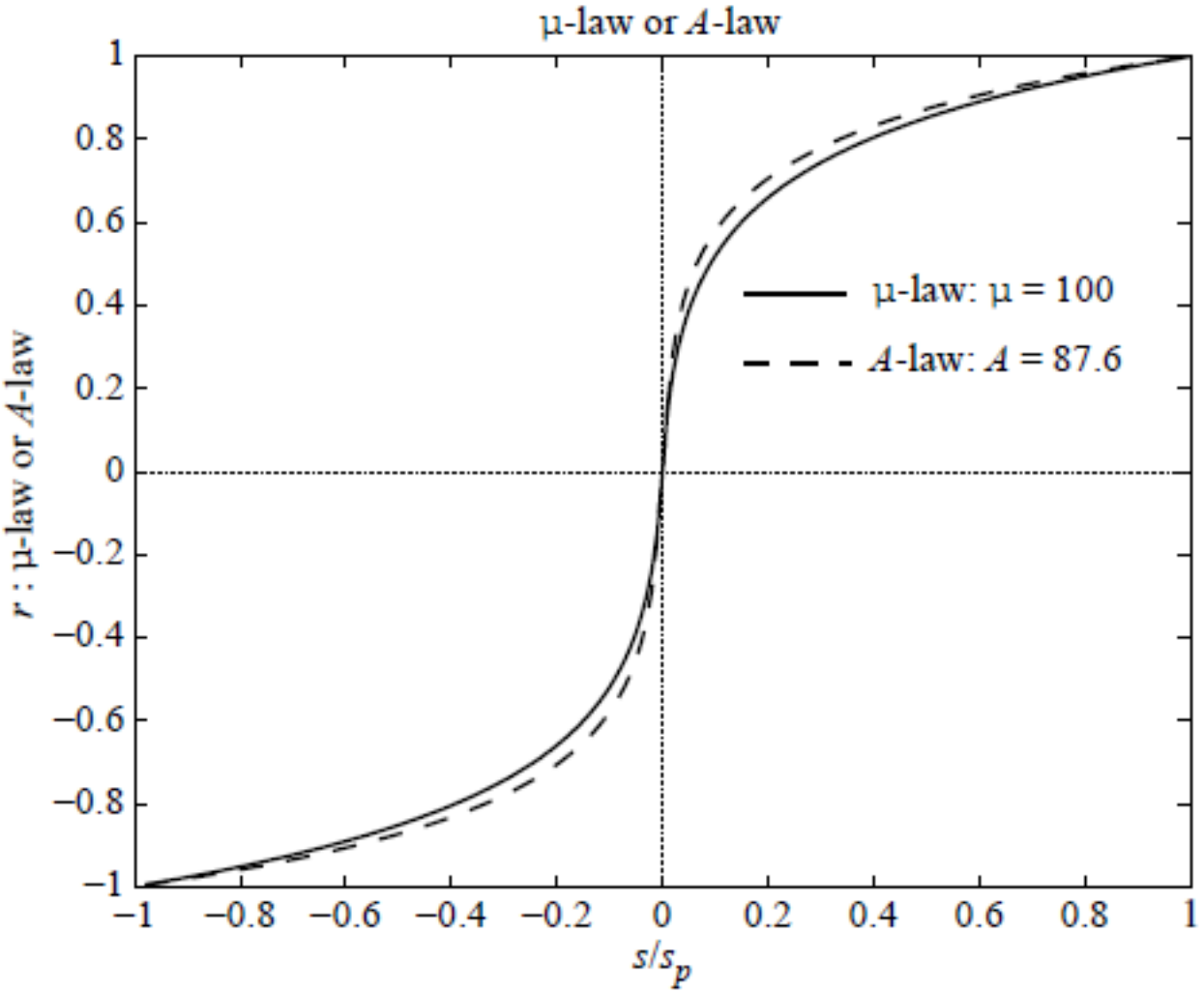
Decoded stair-case signal



Reconstructed low-pass filtered signal

PCM

- PCM also uses *companding* with uniform quantization to achieve *non-uniform quantization* for better compression



PCM Encoder

- A *low-pass filter* is applied to limit the bandwidth to 4 kHz
- The signal is *compressed* by a compressor
- Linear PCM is applied:
  - sampling
  - uniform quantization and coding

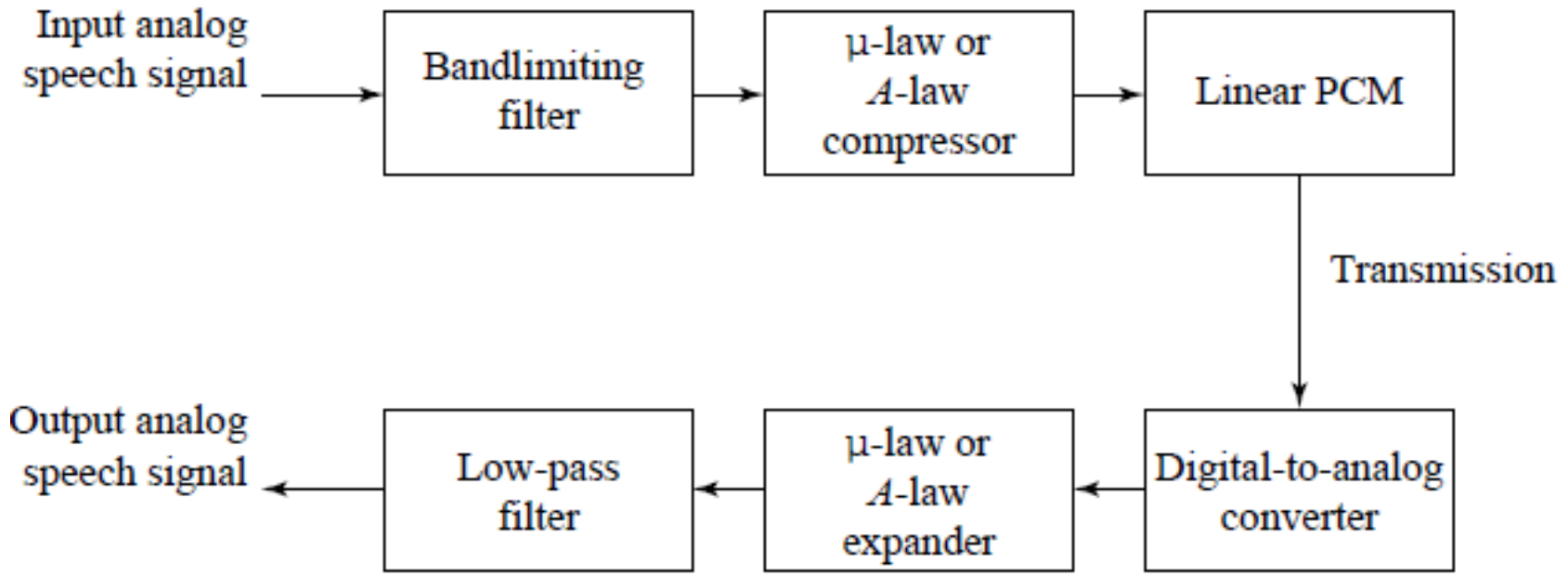


Fig. 6.14: PCM signal encoding and decoding.

PCM Decoder

- The *stair-case* signal is reconstructed
- The signal is *expanded* by the inverse of the compressor
- A *low-pass filter* is applied to remove *aliasing* noise

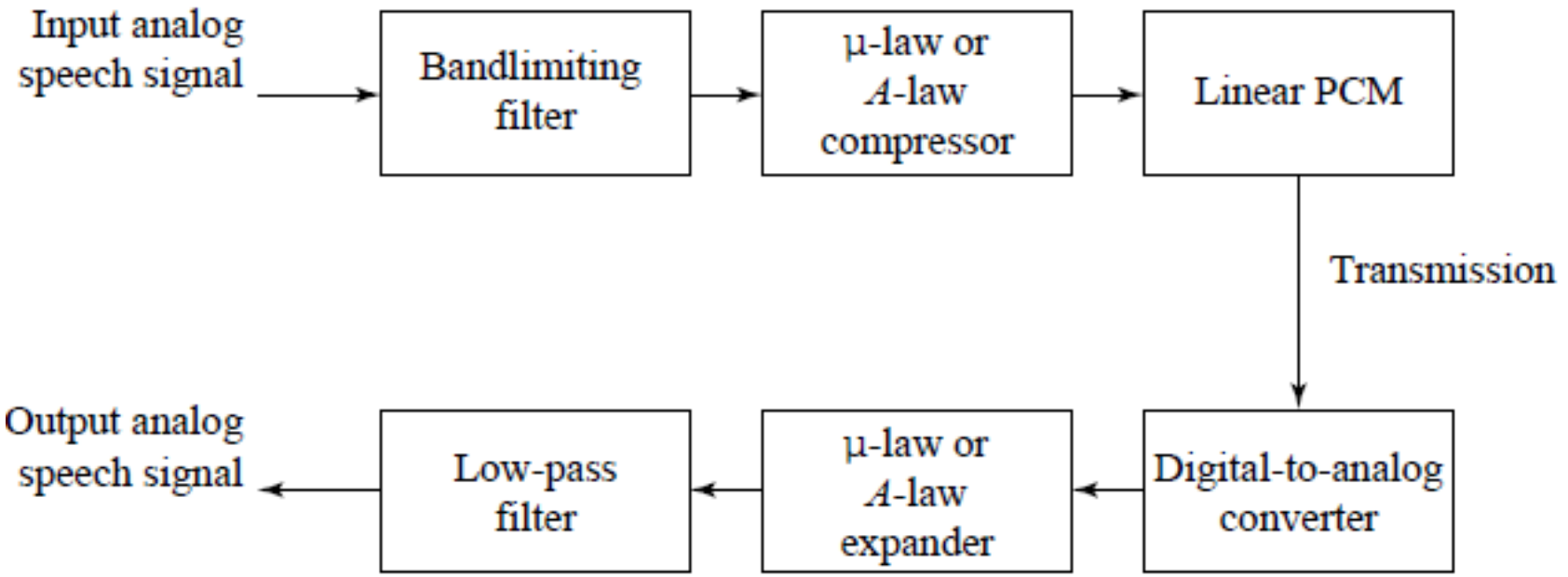
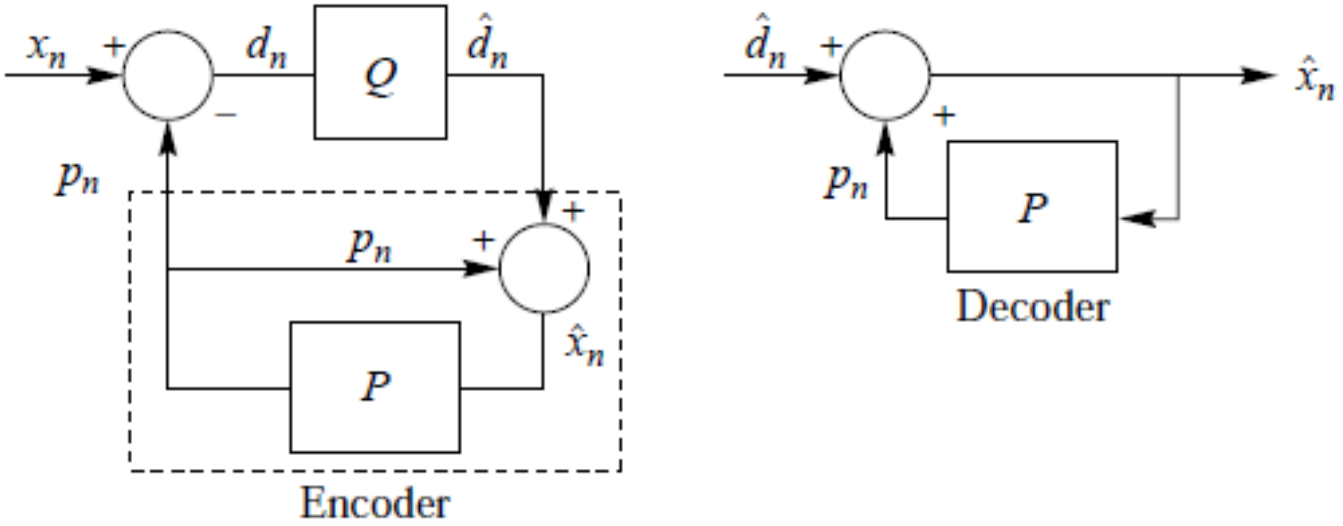


Fig. 6.14: PCM signal encoding and decoding.

DPCM

- Utilizes the *temporal redundancy* in the audio signal to achieve better compression
- Encodes the *difference* between a *sample* and its *prediction*
- Consists of two main components:
  - *Predictor*: predicts the inputs
  - *Quantizer*: quantizes the difference

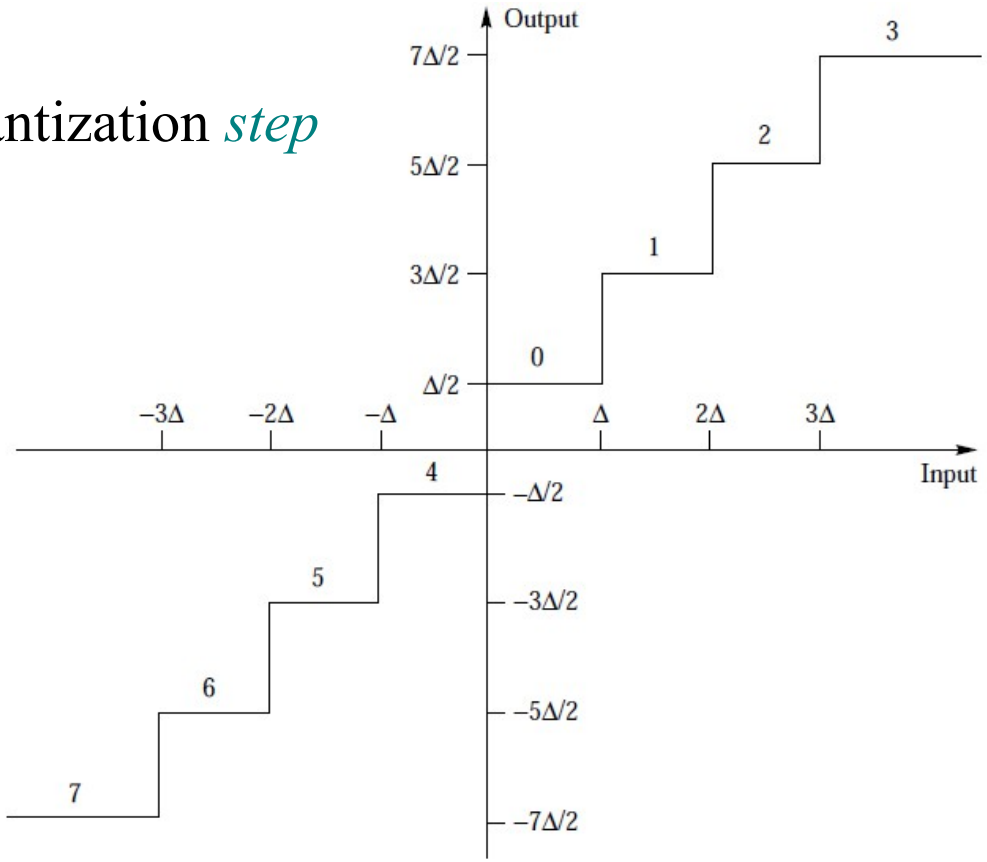


ADPCM

- Makes the DPCM components (quantizer and/or predictor) *adaptive* i.e. change their parameters in response to the *input* (*forward*) or the *output* (*backward*)
- The heart of many speech compression standards e.g. ITU G.721, G.723, G.726, G.727
- The difference between these standards involves the bitrate and various algorithmic details

ADPCM

- *Adaptive Quantization*:
  - Change the parameters of the quantizer (e.g. the quantization step for a uniform quantizer) in response to the input or the output
- Example: *Jayant Quantizer* (backward adaptive i.e. depends on its output)
  - The main idea is to change the quantization *step* to reduce the quantization noise
  - When the quantized level is in the outer levels (away from zero), *increase* the quantization step because otherwise the quantization noise is *unbounded*
  - When the quantized level is in the inner levels (close to zero), *decrease* the quantization step



ADPCM

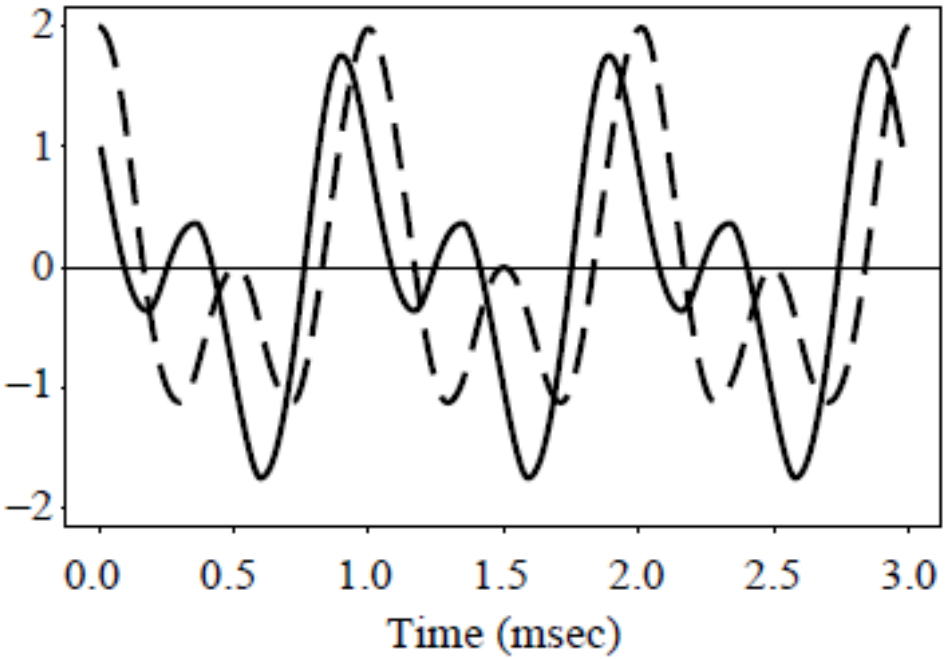
- *Adaptive Prediction*:
  - Change the parameters of the predictor, e.g. the coefficients of the linear predictor, in response to the input or the output.
- Example:
  - Solve Wiener-Hopf equations for each block of inputs (*forward* adaptation) to compute the optimal predictor coefficients
  - Update the predictor coefficients based on the reconstructed values to minimize the squared error (*backward* adaptation)

Vocoders

- All previous techniques are *not* specific to *speech* signals
- Voice Coders (*Vocoders*) are only applicable to speech/voice signals
- The basic idea:
  - The coder works with a mathematical *parametric model* of *speech generation*
  - The *encoder analyzes* the source output to extract the model parameters, and then codes and transmits these parameters to the decoder
  - The *decoder synthesizes* (generates) an approximation of the source output using the model and the received parameters
- The encoded data is not a direct representation of the samples, but rather *instructions* for the decoder on how to re-generate the data

Phase Insensitivity

- Humans are insensitive to *phase* changes (time delays) in the sound signals
- The two sound signals below would be *perceived* the same
- The *energy* of the signal is what *only* matters



$$f_1(t) = \cos(\omega t) + \cos\left(2\omega t + \frac{\pi}{2}\right)$$

$$f_2(t) = \cos(\omega t) + \cos(2\omega t)$$

Fig. 13.3: Solid line: Superposition of two cosines, with a phase shift. Dashed line: No phase shift. The wave is very different, yet the sound is the same, perceptually.

Sound Pitch

- The general *frequency* or *period* of the sound signal, generally two categories:
  - low or bass (pronounced /base/)
  - high or tenor

Sound Excitation

- Whether the signal is *voiced* or *unvoiced*:
  - *Voiced*:
    - Air is forced trough the vocal cords
    - The signal looks *periodic* or pseudo-periodic
    - Examples: sounds like /a/, /e/, /m/, /b/, …
    - Can be approximated by a *pulse generator* with the correct energy and pitch

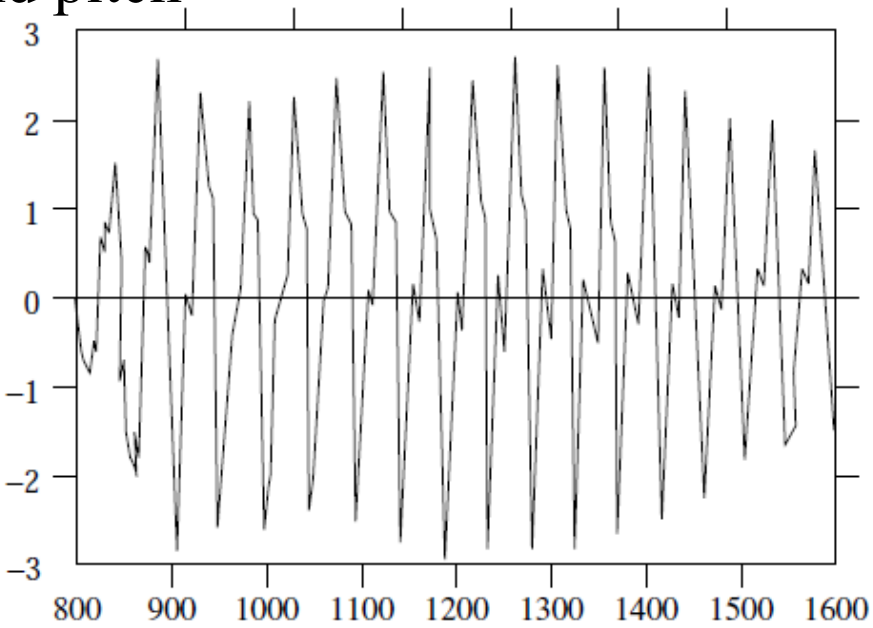


FIGURE 17. 2 **The sound /e/ in *test*.**

Sound Excitation

- Whether the signal is *voiced* or *unvoiced*:
  - *Unvoiced*:
    - The signal looks like *noise*
    - Examples: sounds like /s/, /f/, /sh/, …
    - Can be approximated by a *noise generator* with the correct energy

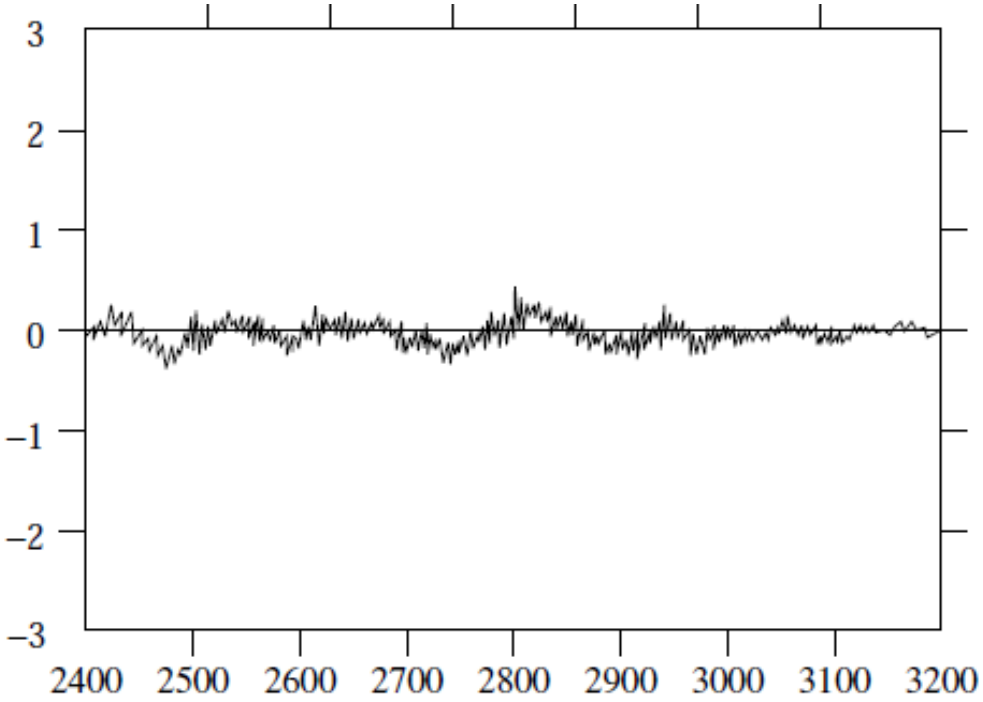


FIGURE 17. 3 **The sound /s/ in *test*.**

Types of Vocoders

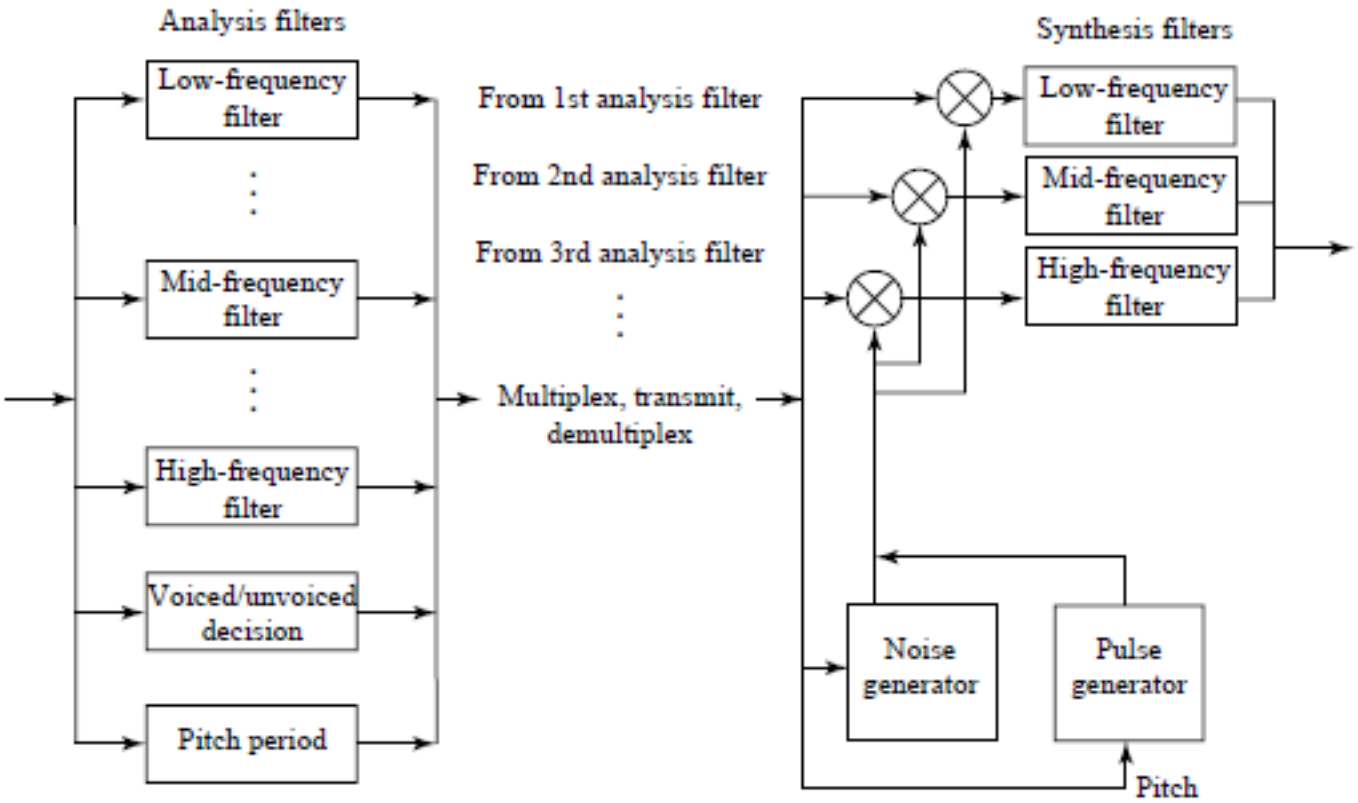
- Channel Vocoder
- Linear Predictive Coder (LPC)
- Code Excited Linear Prediction (CELP)
- Mixed Excited Linear Prediction (MELP)
- …

Channel Vocoder

- One of the oldest vocoders
- Developed by Homer Dudley in Bell Labs in 1928
- Uses the idea of subband coding
- Can achieve an intelligible but *synthetic* voice at a rate of 2.4 kbps

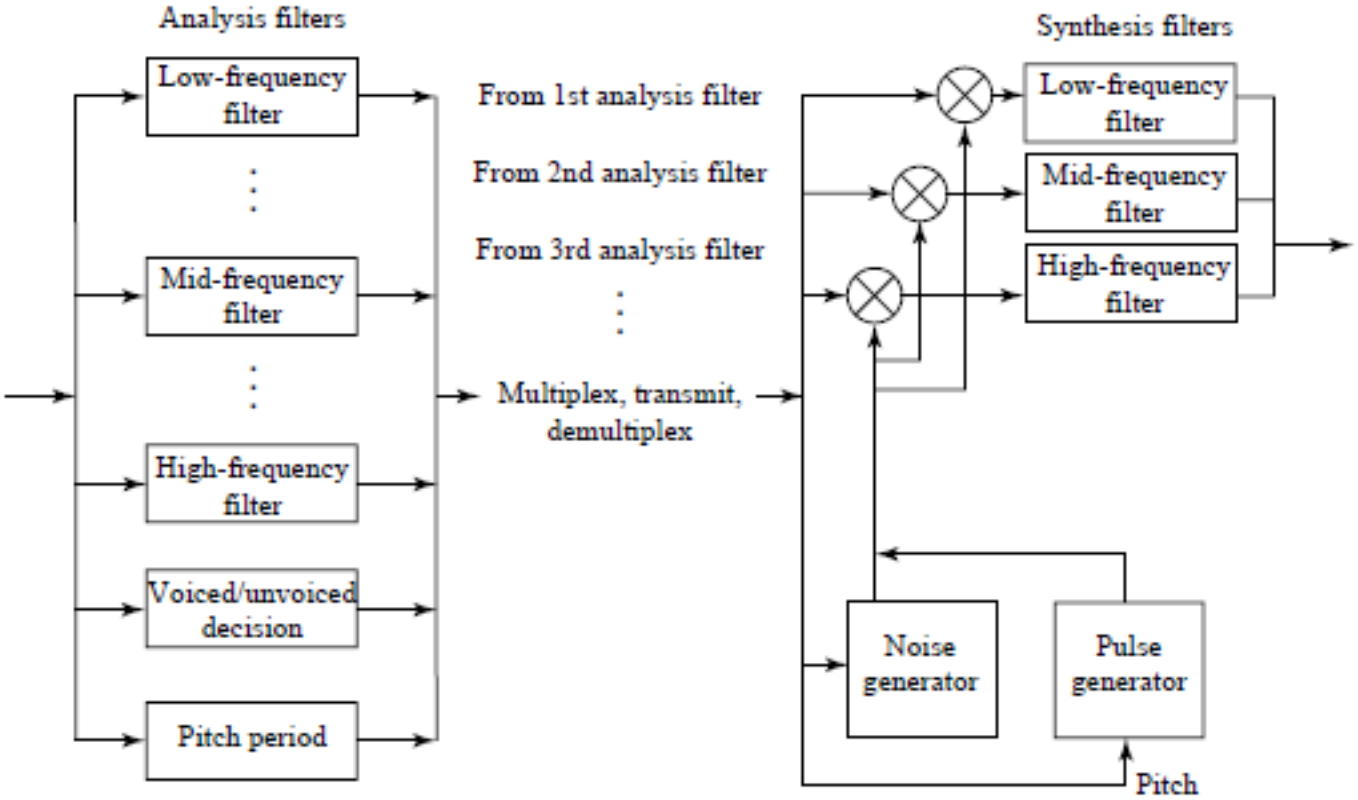
Channel Vocoder

- *Encoder*:
  - Applies the *analysis filter bank* on the input signal to analyze the different frequency bands
  - Estimate the *energy*, *excitation* (voiced or unvoiced), and *pitch* of the signal in each subband



Channel Vocoder

- *Decoder*:
  - From the received parameters, generate the inputs for the synthesis filters:
    - voiced: use pulse generator with estimated pitch
    - unvoiced: use noise generator
  - Apply the *synthesis* filters and combine their outputs



LPC Vocoder

- Works on the signal in the *time domain* directly
- The vocal tract model is a *single filter* depending on previous outputs and the current input
- It models a *segment* of the signal using a set of *coefficients* (model parameters) and sends the coefficients to the receiver
- The model is of the form

$$s(n) = \sum_{i=1}^{p} a_i s(n-i) + Gf(n)$$

where $f(n)$ is the input (pulse for voiced segments and noise for unvoiced), G is the *gain* coefficient, a_i are the filter coefficients, $s(n)$ is the filter outputs

LPC Vocoder

- The input speech is 8000 samples per second
- It is divided into *segments* of 22.5 msec each (180 samples)
- For each segment the filter coefficients are estimated by a *least squares* method to minimize the squared error, and sent to the receiver
- Can achieve rates of 2.4 kbps
- A US Government Standard LPC-10 (with $p = 10$)

$$s(n) = \sum_{i=1}^{p} a_i s(n-i) + Gf(n)$$

Audio Compression

- Previous techniques, e.g. ADPCM and LPC, mainly target telephony and speech compression
- We will now consider methods that target general audio compression, including music, movies, and broadcast TV
- Instead of modeling the signal *source*, we use properties of the *receiver* i.e. a *psychoacoustic* model of the human hearing system, to perform audio compression
- This is called *perceptual coding*
- This is similar to the JPEG coding, which used properties of the human visual system to achieve higher compression rates

Psychoacoustics

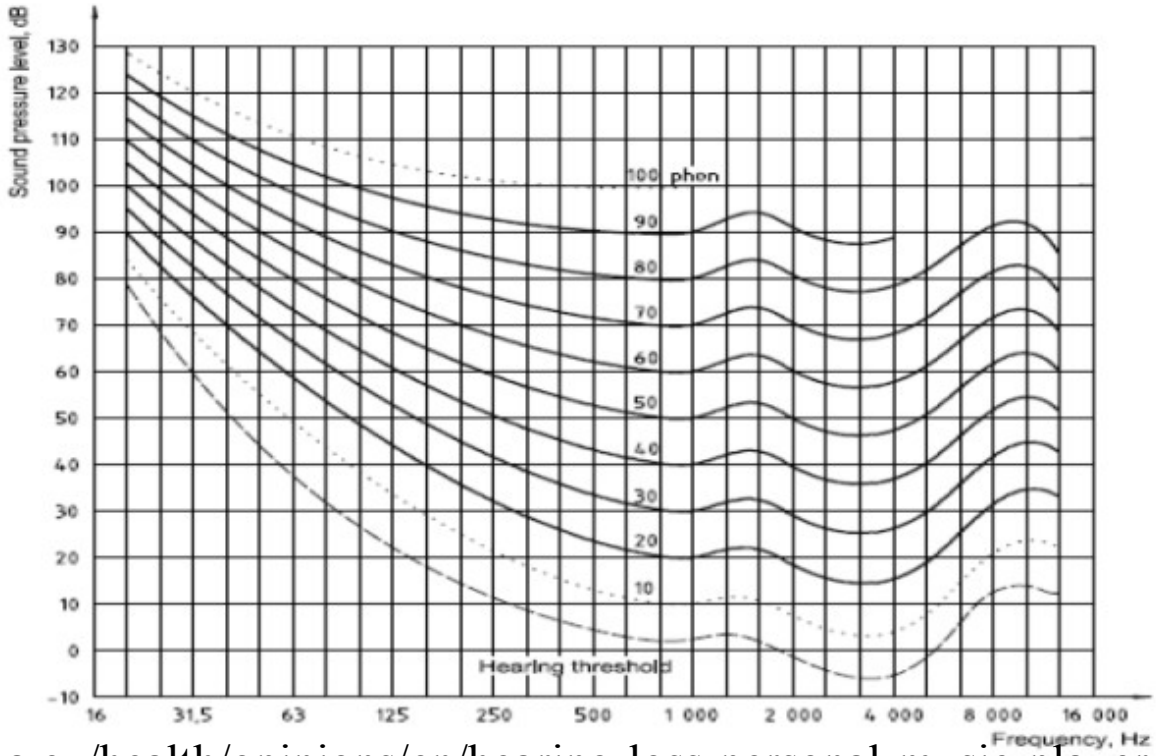
- Humans hear sounds in the range 20 Hz to 20 kHz
- Sound levels measured in *decibels* dB, which is the ratio to the *quietest sound* we can hear

$$dB = 10\log_{10}\frac{V_1^2}{V_2^2} = 20\log\frac{V_1}{V_2}$$

Threshold of hearing	0
Rustle of leaves	10
Very quiet room	20
Average room	40
Conversation	60
Busy street	70
Loud radio	80
Train through station	90
Riveter	100
Threshold of discomfort	120
Threshold of pain	140
Damage to ear drum	160

Equal Loudness Contours

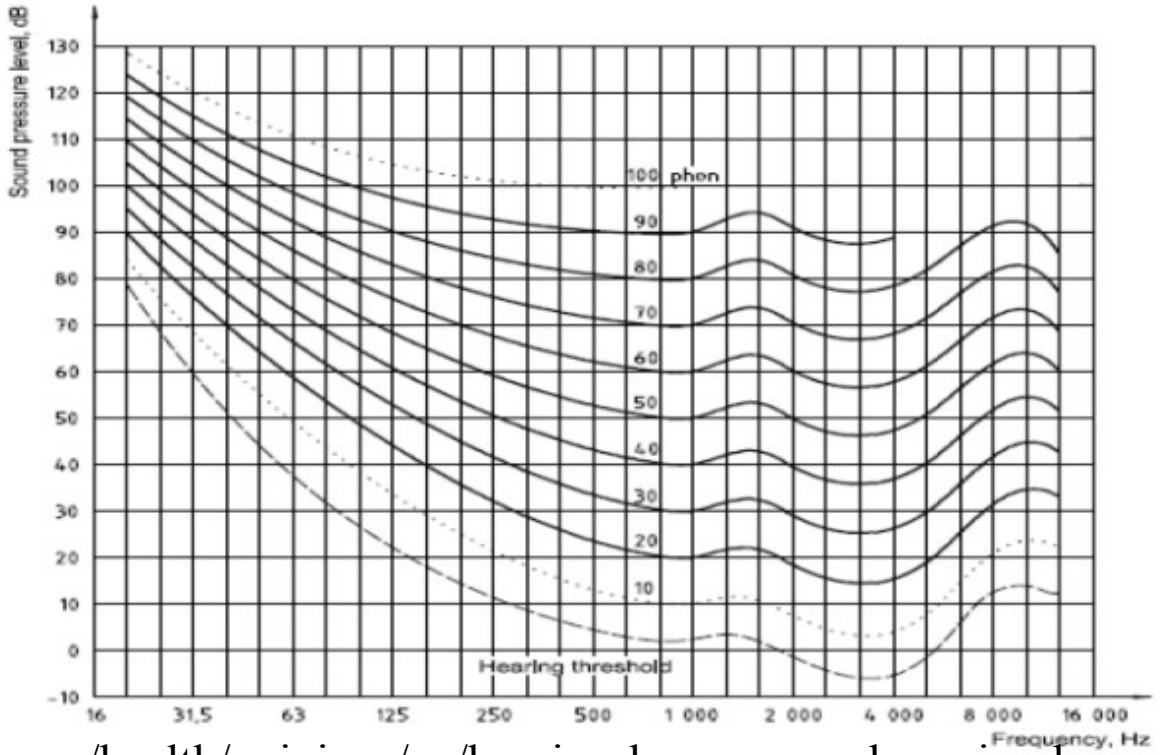
- The *perceived loudness* of a sound depends on its *frequency*
- The human ear is most sensitive in the range 2-5 kHz, and less sensitive at higher and lower frequencies
- The equal loudness contours plot sounds that have the same *perceived loudness* across different frequencies
- The loudness is relative to a sound at 1 kHz



[http://ec.europa.eu/health/opinions/en/hearing-loss-personal-music-player-mp3/images/figure-1.png]

Equal Loudness Contours

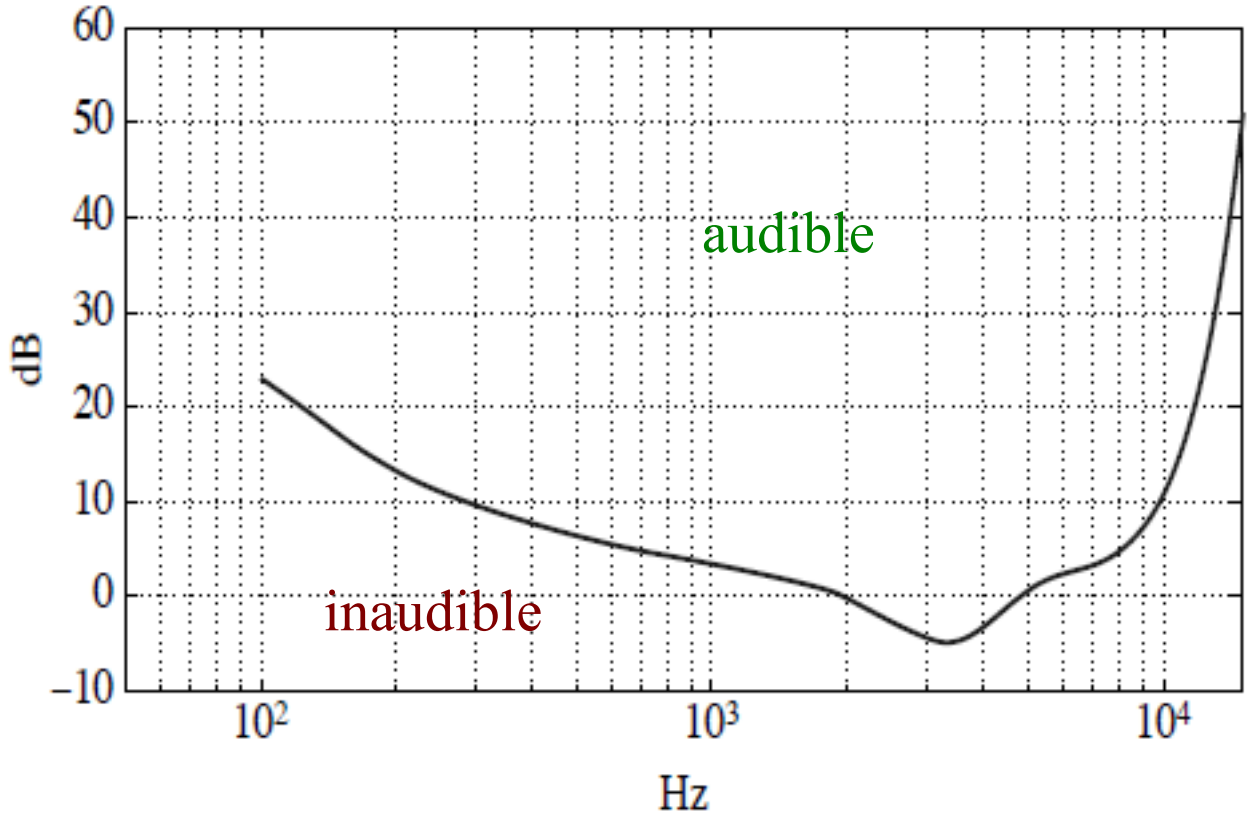
- For example:
  - a sound at 1 kHz with actual loudness of 10 dB would be perceived as having a loudness of 10 dB
  - a sound at 4 kHz with actual loudness of 2 dB would be perceived as having a loudness of 10 dB
  - a sound at 10 kHz with actual loudness of 20 dB would be perceived as having a loudness of 10 dB



[http://ec.europa.eu/health/opinions/en/hearing-loss-personal-music-player-mp3/images/figure-1.png]

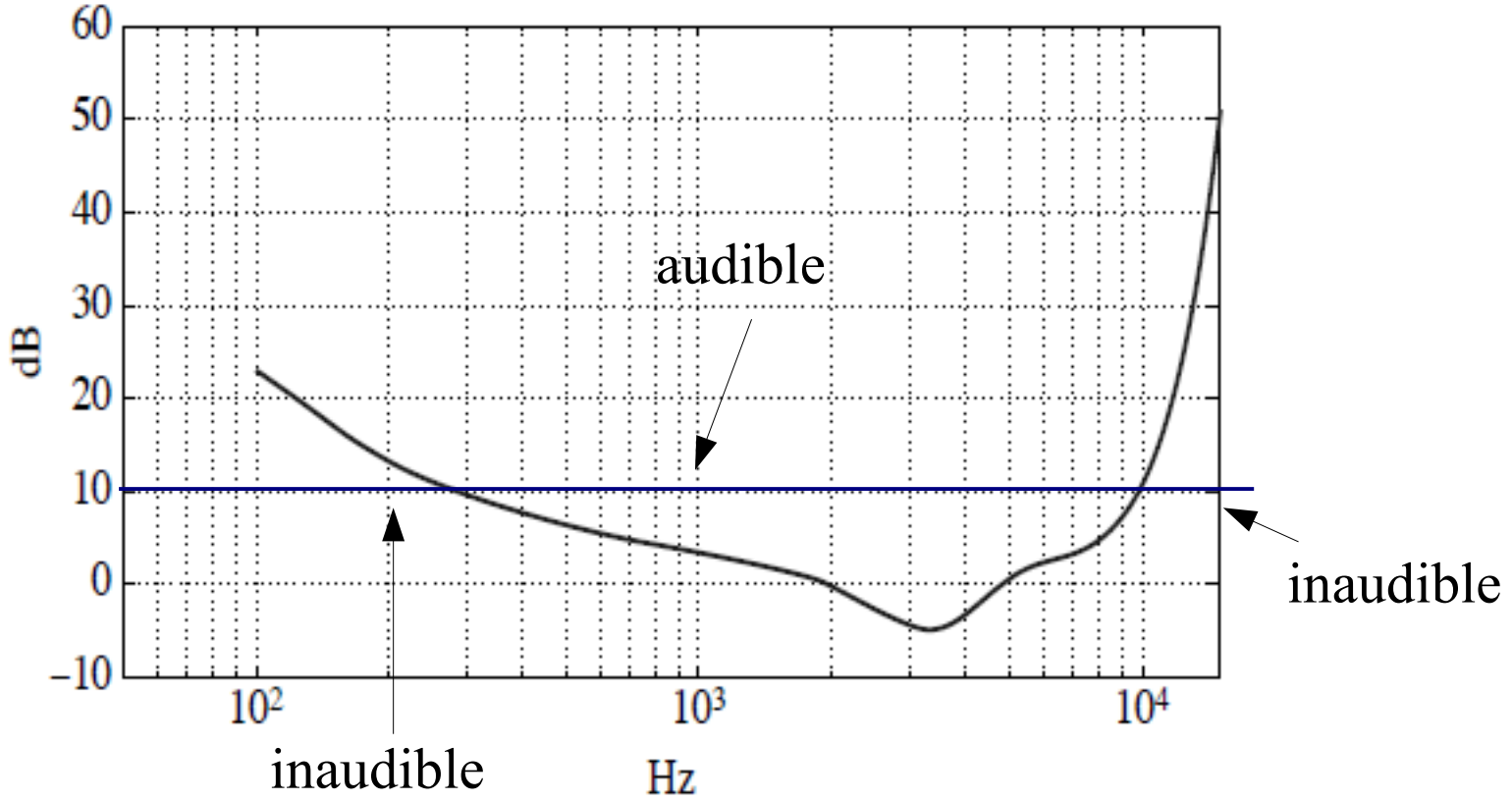
Threshold of Human Hearing

- Similarly, the *threshold* of hearing, i.e. the *minimum* loudness for a sound to be audible, depends on the *frequency*
- If the sound is *above* the threshold curve, it's *audible*
- Otherwise, it's *inaudible*



Threshold of Human Hearing

- The threshold is *higher* at lower and higher frequencies, where the human ear is *less* sensitive
  - At lower frequencies, e.g. 200 Hz, a sound of 10 dB will *not* be heard
  - At 1 kHz, a sound of 10 dB *will* be heard
  - At higher frequencies, e.g. 20 kHz, it will *not* be heard

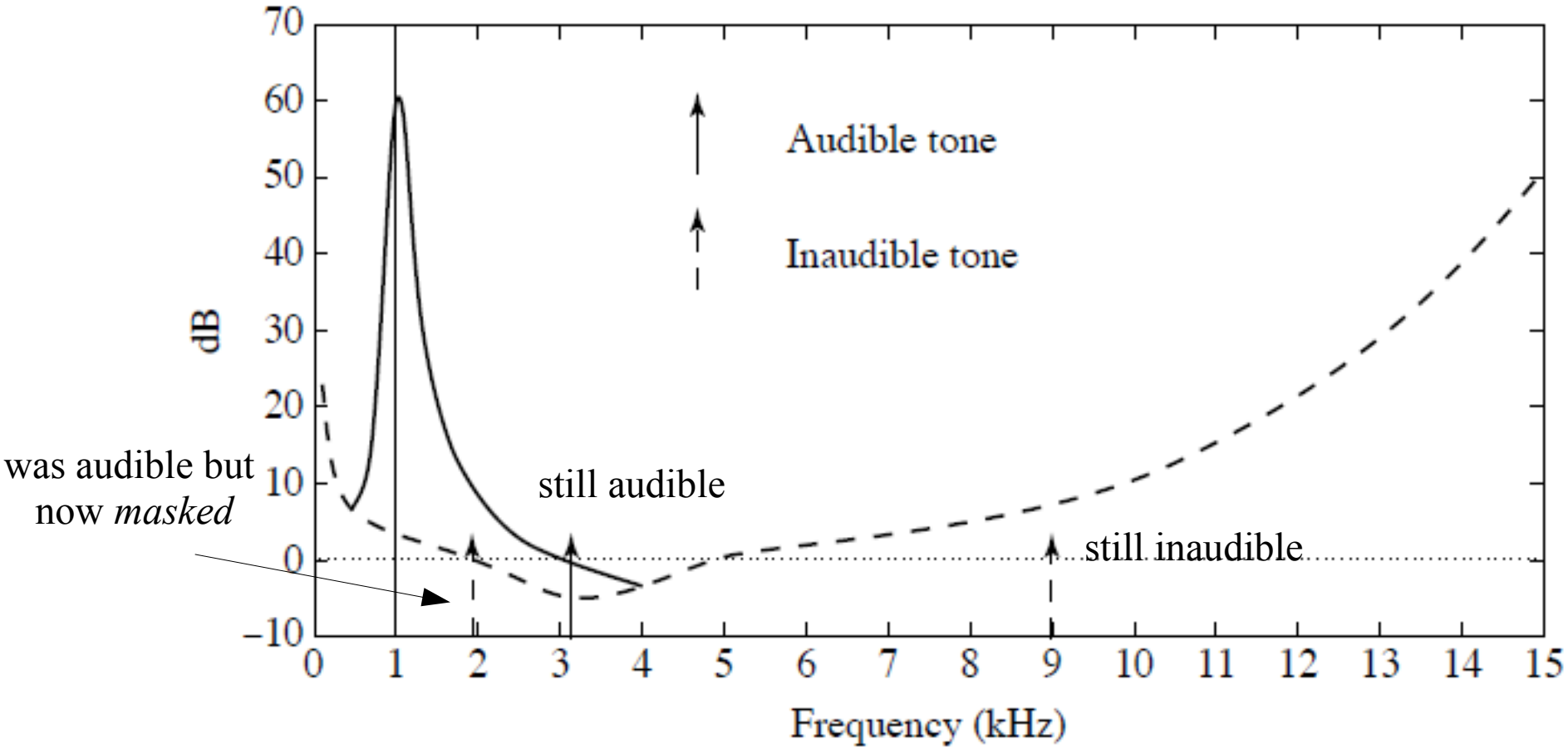


Frequency/Spectral Masking

- A sound at a certain frequency can *mask* or *hide* sounds at nearby frequencies i.e. make them *inaudible*
  - A *lower tone* (lower frequency sound) can mask *higher* tones
  - The converse is not true, higher tones don't always mask lower tones
  - The *louder* the masking tone, the *wider* its range of influence
- This observation is used in MPEG audio compression to *reduce* the amount of data that need to be encoded, without affecting the *perceived* quality of the audio i.e. the loss of information is *not* noticeable

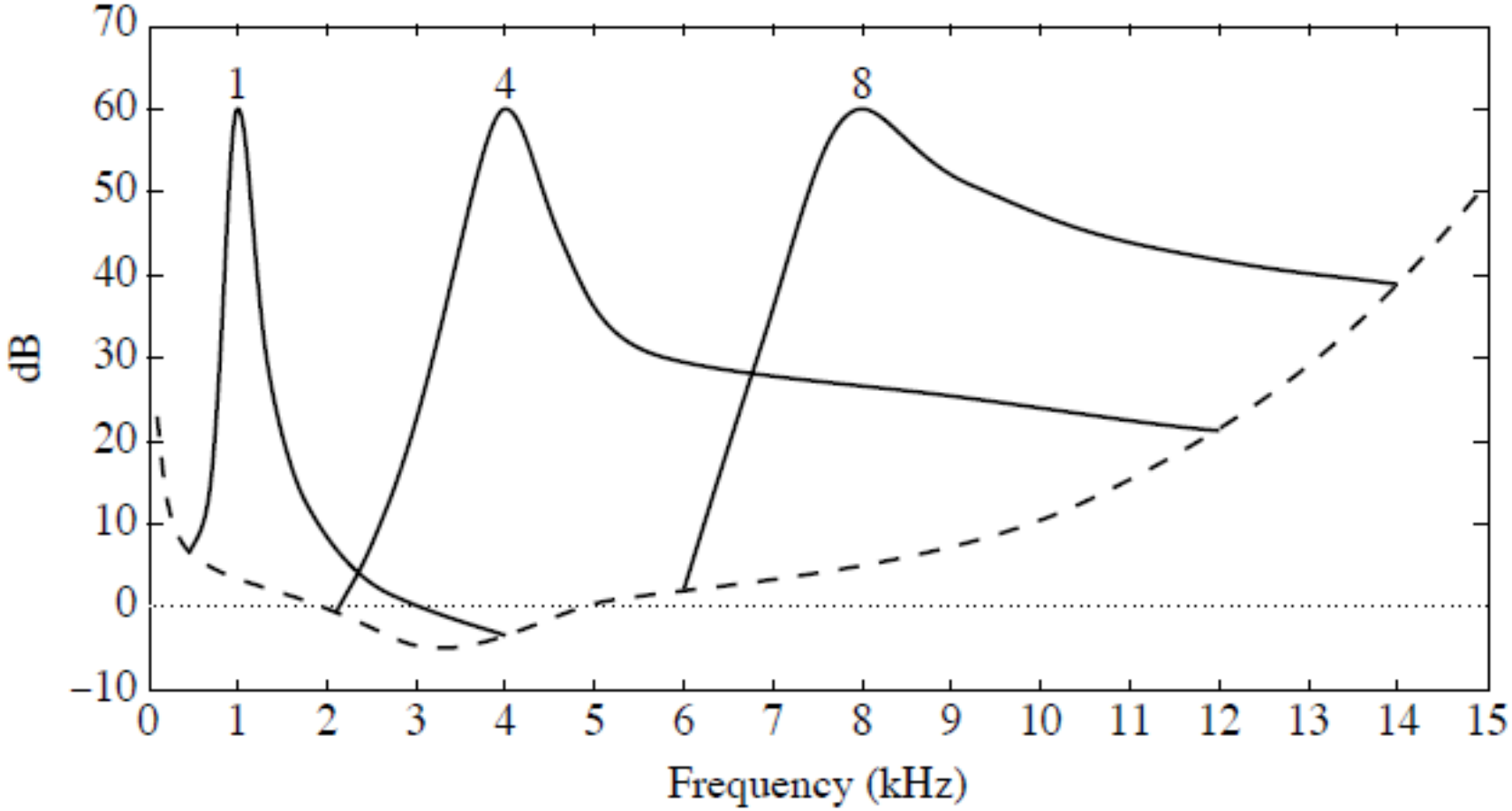
Frequency Masking

- A *masking tone raises* the threshold of hearing in neighboring frequencies
- For example, a masking tone at 1 kHz would mask the tone at 2 kHz that was audible before



Frequency Masking

- Masking tones at different frequencies have different curves
- For example, these are curves for masking tones of 1, 4, and 8 kHz



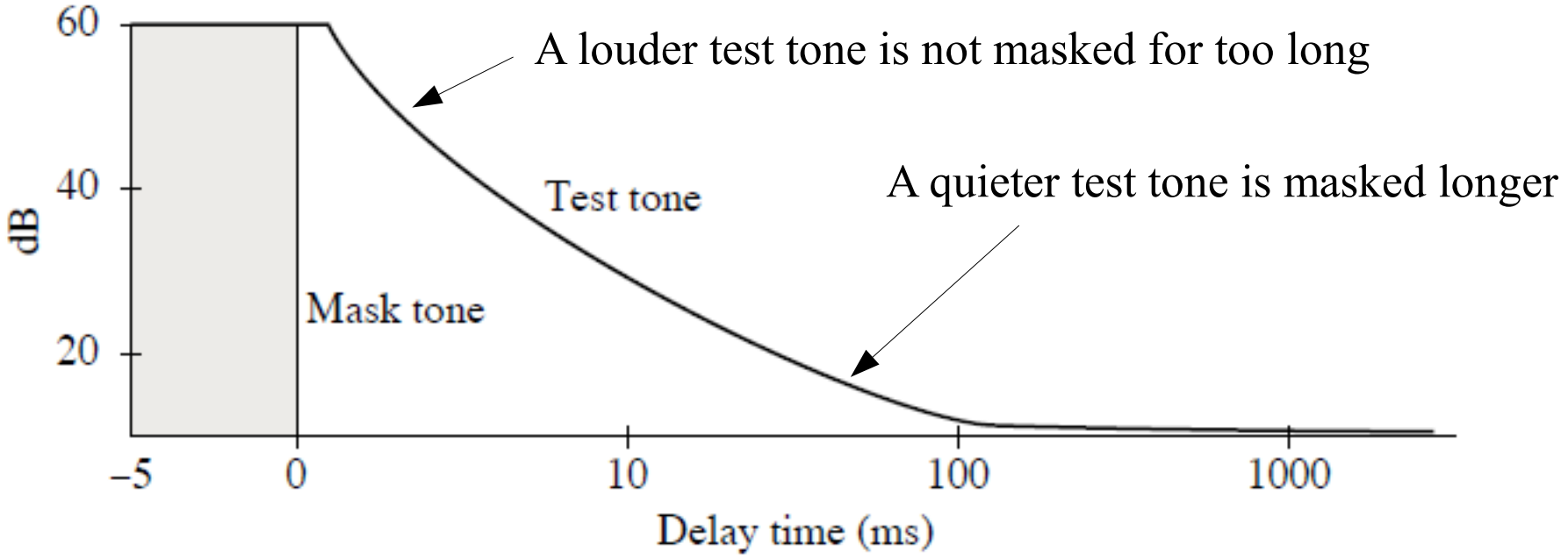
Critical Bands

- The range of audible frequencies is naturally divided into a set of *critical bands* or ranges of frequencies
- Two sounds with two *different* frequencies within the *same* critical band will *not* be distinguished from each other
- The *bandwidth* of a critical band depends on the frequency, with *smaller* bands (around 100 Hz) at *lower* frequencies, and *wider* bands (up to 4 kHz) at *higher* frequencies
- The human ear has about 24 or 25 critical bands

Temporal Masking

- A loud tone causes the *receptors* in the human ear to *saturate* i.e. *decrease* their sensitivity
- Therefore, after a loud tone is stopped, the ear takes some *time* to go back to normal, and thus some other tones can be *masked*. This is called *temporal masking*.

The plot shows the amount of time delay it takes our ears to hear a *test tone* of varying loudness after a *mask tone* of 60 dB has been stopped



Temporal Masking

- The *temporal masking* depends on both the *frequency* of the *test tone* and on the *time delay* after stopping the *mask tone*
- The closer the *test tone* in either *frequency* or *time*, the more likely it will be *temporally masked*

A 60 dB mask tone with frequency of 1 kHz

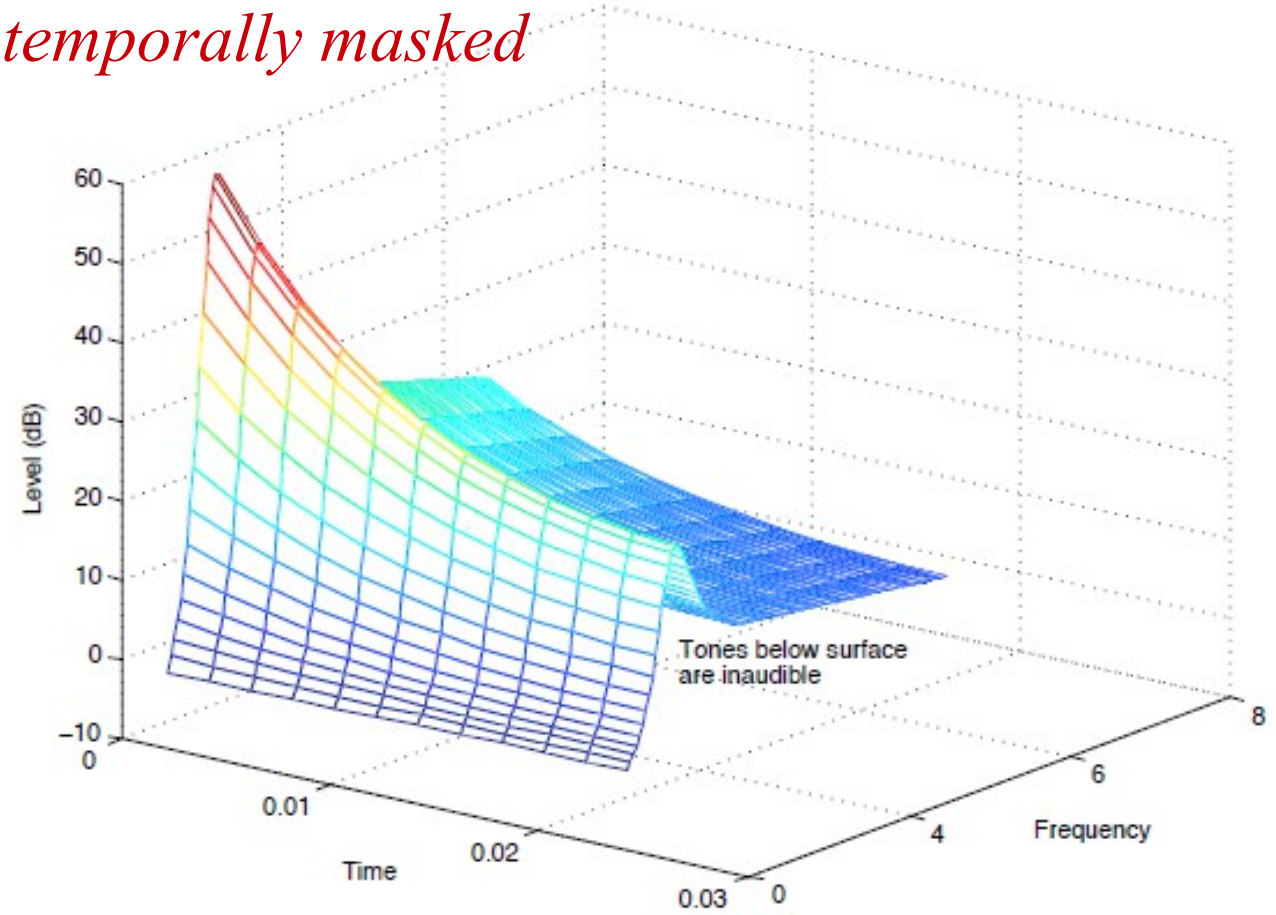


Fig. 14.7: Effect of temporal and frequency maskings depending on both time and closeness in frequency.

Temporal Masking

- Temporal masking can happen *before* the masking tone starts (*premasking*), as well as *after* the masking tone stops (*postmasking*)
- In general, postmasking is much *longer* (50-200 msec) than premasking (2-5 msec)

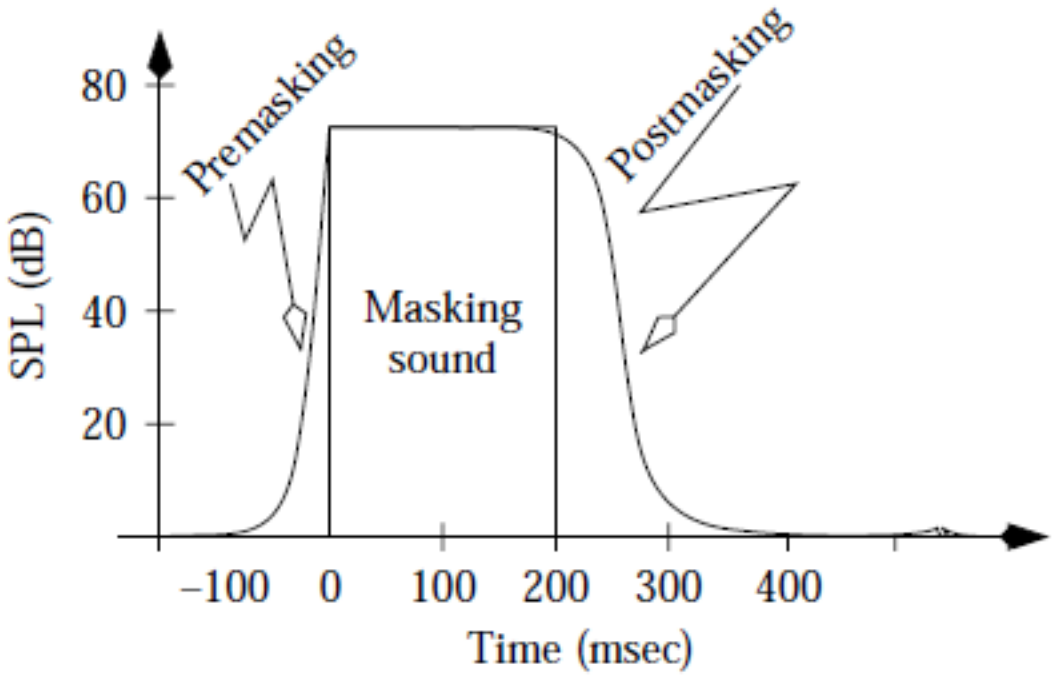


FIGURE 16. 3 **Change in the audibility threshold in time.**

MPEG Audio Compression

- MPEG audio compression takes advantage of the *psychoacoustic* model to reduce the number of bits allocated to *masked* components
- It works in three steps:
  - Applies a *filter bank* to the input to break it into its frequency components
  - In parallel, a psychoacoustic model is applied to the data for bit allocation purposes, where fewer bits are allocated to less important components
  - The number of bits allocated are used to quantize the info from the filter bank. This provides the compression

MPEG Layers

- MPEG-1 and -2 audio offers *three* compatible *layers*: Layer I, Layer II, and Layer III:
  - Each layer is able to understand the lower layers e.g. a Layer II decoder can also decode a Layer I stream
  - Each layer offering more complexity in the psychoacoustic model and better compression for a given level of audio quality
  - Each layer, with increased compression effectiveness, accompanied by extra delay
- The objective of MPEG layers: a good tradeoff between *quality* and *bit-rate*

MPEG Layers

- Layer I quality can be quite good provided a comparatively high bit-rate is available
  - Digital Audio Tape typically uses Layer 1 at around 192 kbps
- Layer II has more complexity; was proposed for use in Digital Audio Broadcasting
- Layer III (MP3) is the most complex, and was originally aimed at audio transmission over ISDN lines, but now used almost everywhere, specially for music over the Internet

- Most of the complexity increase is at the *encoder*, not the decoder – accounting for the popularity of MP3 players (decoders)

MPEG Audio Strategy

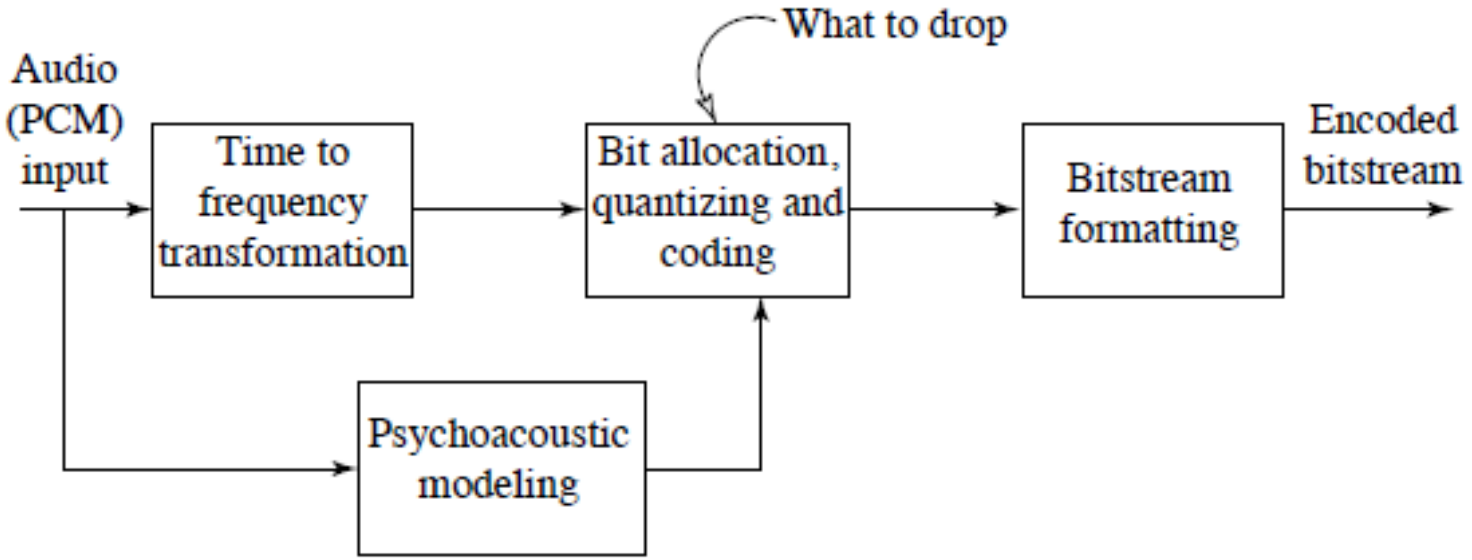
- MPEG approach to compression relies on:
  - Quantization
  - Human auditory system is not accurate within the width of a critical band (perceived loudness and audibility of a frequency)
- MPEG encoder employs a *bank of filters* to:
  - Analyze the frequency (*spectral*) components of the audio signal by calculating a frequency transform of a *window* of signal values
  - Decompose the signal into *subbands* by using a bank of filters (Layer I & II: use *quadrature-mirror filters*; Layer III: adds a DCT)

MPEG Audio Strategy

- *Psychoacoustic Model*:
  - *Fourier Transform* is applied to the signal
  - Estimates the *masking thresholds* in each subband (minimum amplitude for audible sounds), and thus the amount of tolerable noise, and the signals to be discarded
  - Sets the quantization step and the number of bits after discarding signals below the threshold
- *Masking*:
  - *Frequency Masking* used in all layers to remove inaudible frequency components
  - *Temporal Masking* used in Layers II and III

Basic MPEG Audio: Encoder

- Divides the input into 32 frequency *subbands* via a filter bank (a *linear operation* taking 32 PCM samples and producing 32 frequency coefficients)
- The frequency components are grouped together into *frames*
  - In Layer I, the coefficients are grouped into frames of 384 samples, 12 samples from each subband
  - This introduces an inherent time lag in the coder, equal to the time to accumulate 384 (i.e., 12×32) samples
  - In Layer II and III, the frame has 1,152 samples



Basic MPEG Audio: Encoder

- In Layer I, the coefficients are grouped into frames of 384 samples
- In Layer II and III, the frame has 1,152 samples

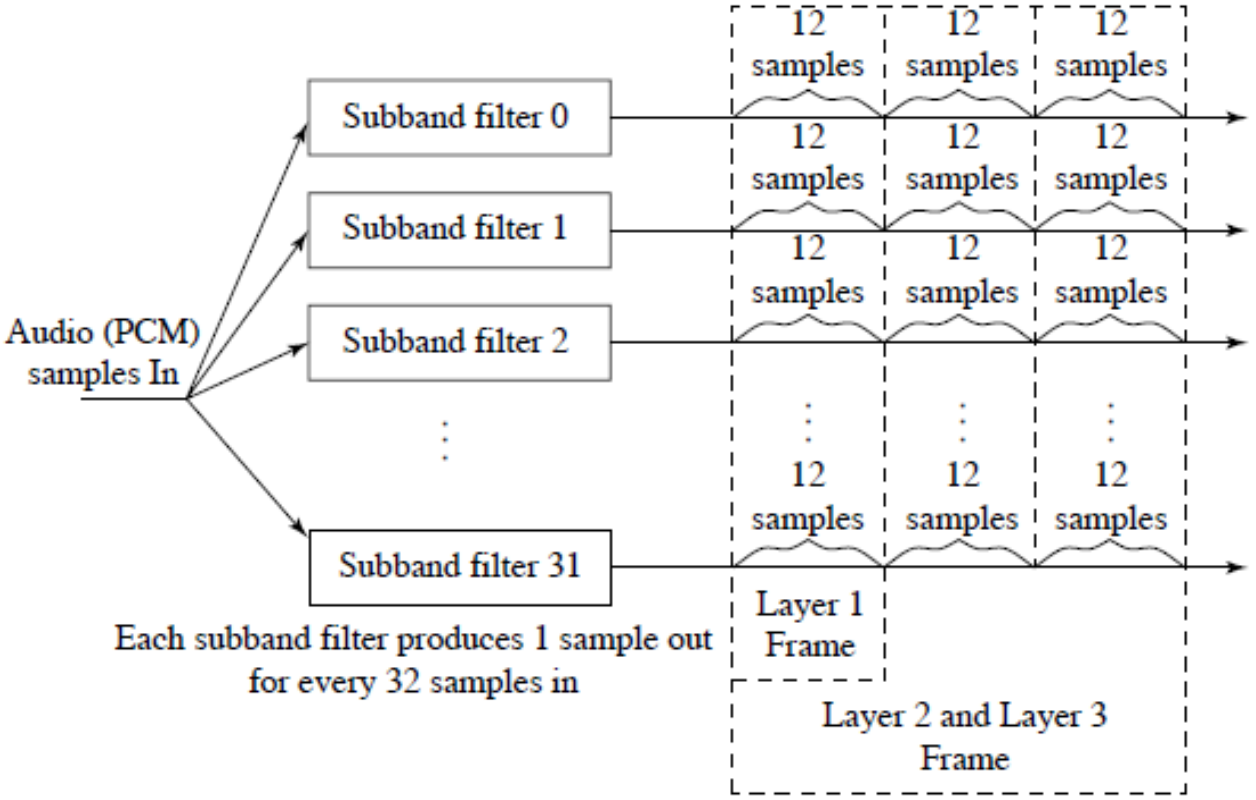


Fig. 14.11: MPEG Audio Frame Sizes

Basic MPEG Audio: Encoder

- For each subband in the frame, a *scaling factor* is computed which is the maximum value in the subband, e.g. in Layer I, a scaling factor is computed for each 12 samples
- This is passed to the bit allocation and the quantization block that apportions bits to minimize the quantization noise

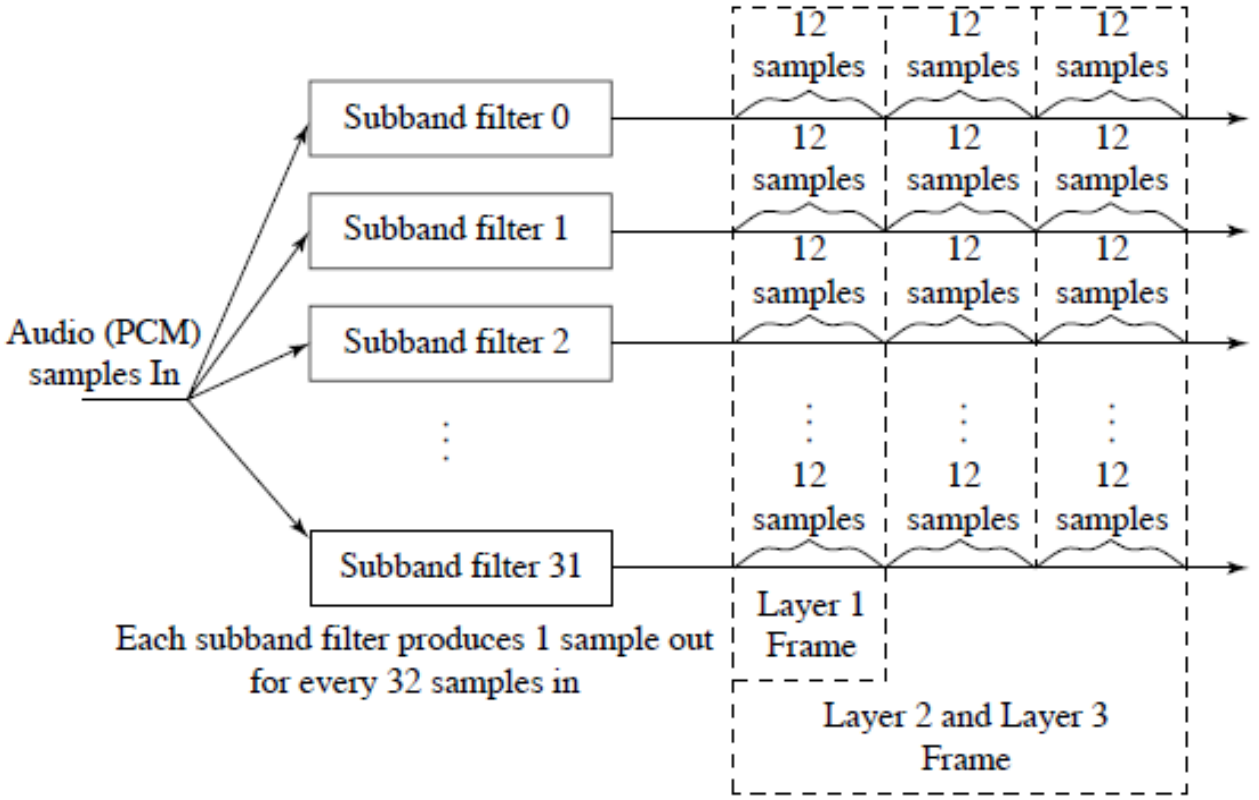
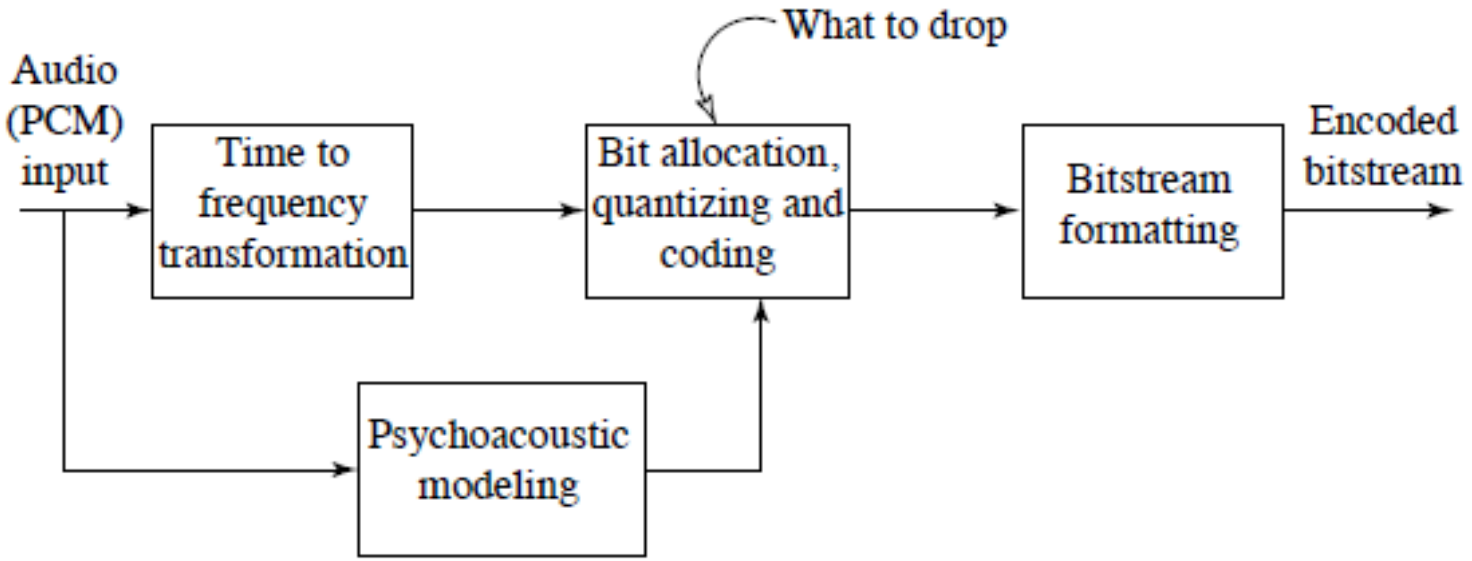


Fig. 14.11: MPEG Audio Frame Sizes

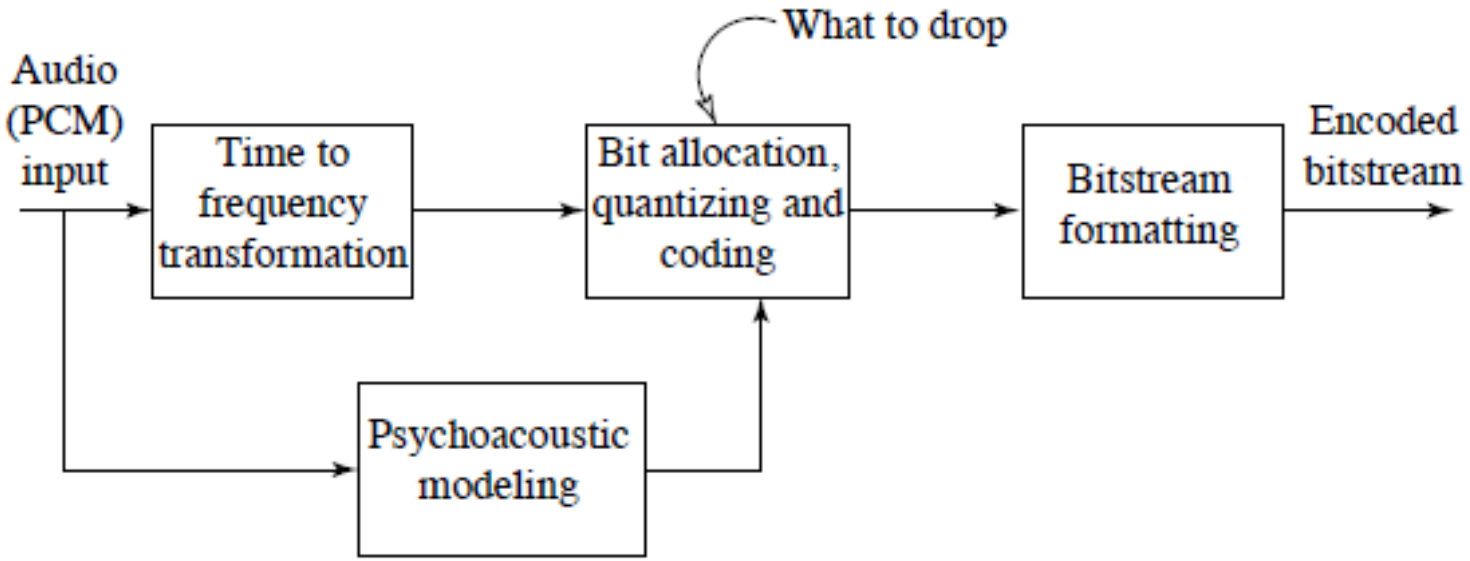
Basic MPEG Audio: Encoder

- The *psychoacoustic* model is composed of lookup tables and other data that models the human hearing system
- It is basically used for *spectral* and *temporal* masking
- In Layer I:
  - A decision is made whether the subband contains a *tone* or *noise*
  - Based on that, and the scaling factor, a *masking threshold* is computed
  - This is then compared to the *hearing threshold*



Basic MPEG Audio: Encoder

- The output of the *psychoacoustic* model and the scale factors are fed to the *bit allocation and quantization block*
- It allocates bits to each *subband* based on its *masking threshold* and *scale factors*, such that:
  - The number of bits per *frame* remains below the maximum rate required
  - The *quantization noise* is *below* the masking threshold (so not audible)



Basic MPEG Audio: Encoder

- The output is then passed to the *bitstream formatting block* that adds the header and side information and formats the frame according to the standard

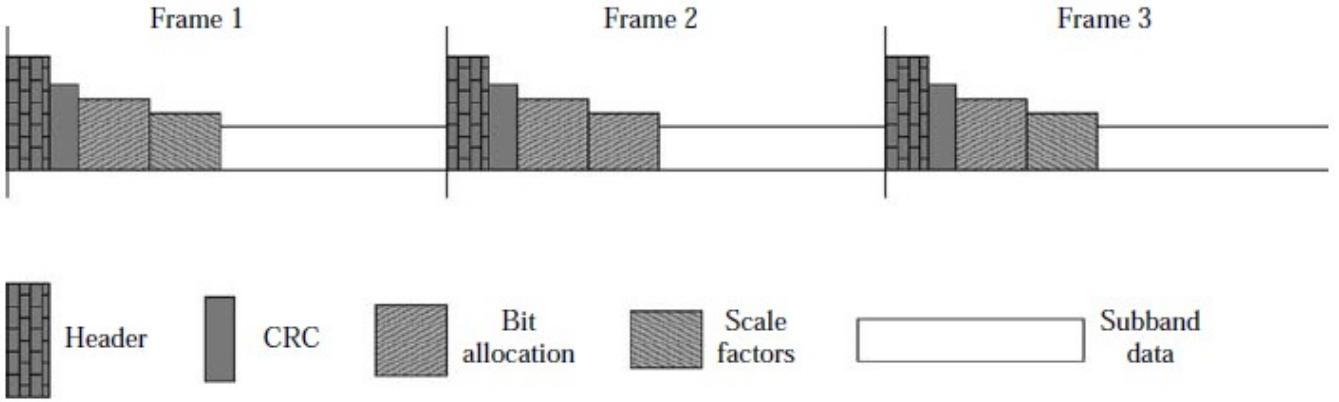
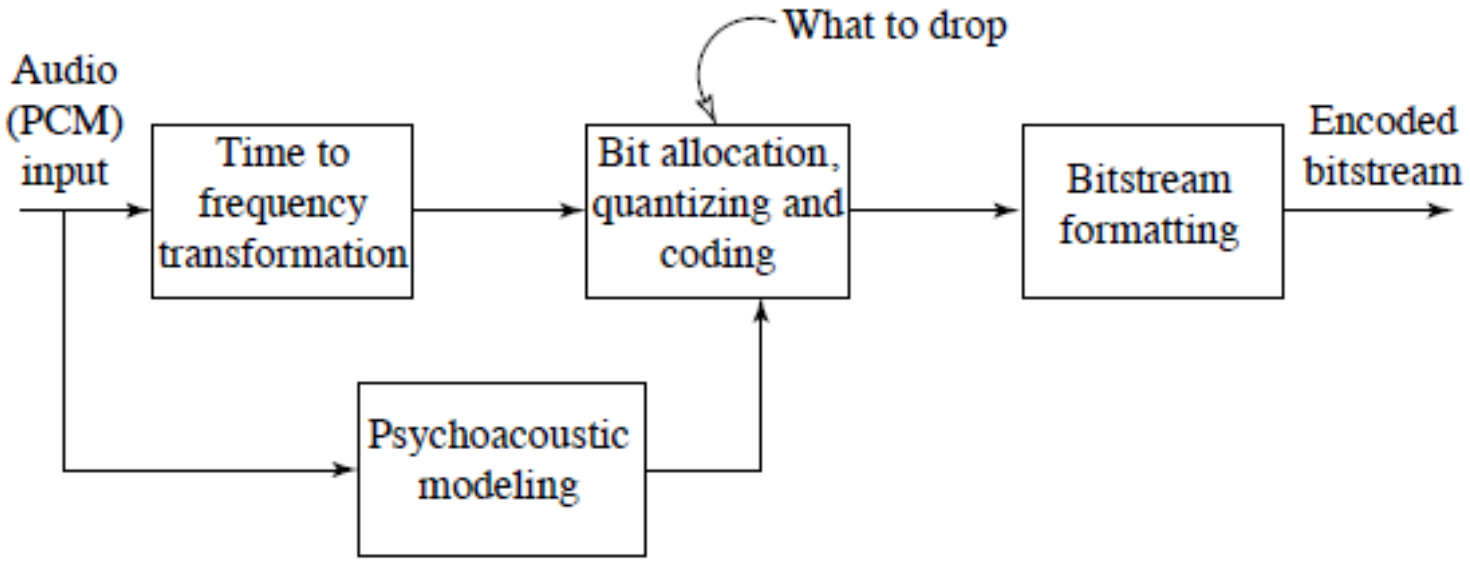
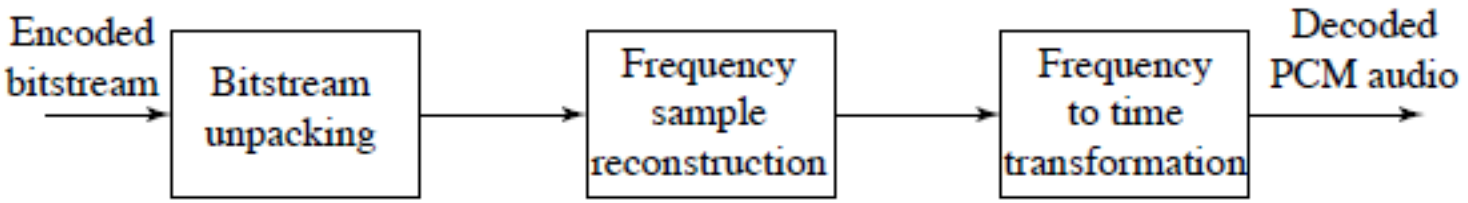


FIGURE 16. 5 Frame structure for Layer 1.



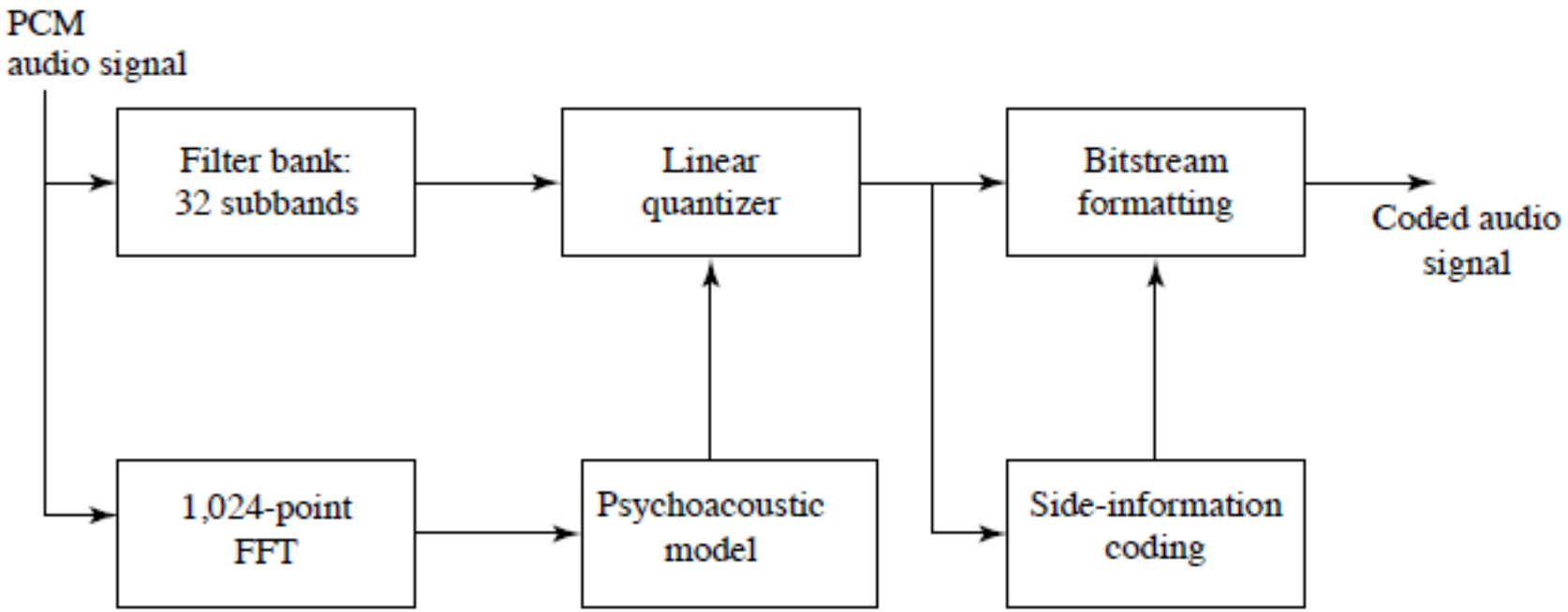
Basic MPEG Audio: Decoder

- The encoded bitstream is unpacked to extract the header, side information, and subband data e.g. bit allocation, scale factors, CRC, …
- The quantized samples are *dequantized*, and then passed through *synthesis filters* to reconstruct their approximations
- The output PCM samples are then passed to the next step
- Note that the psychoacoustic model is *not* needed in the decoder, and that the decoder is much *simpler* than the encoder



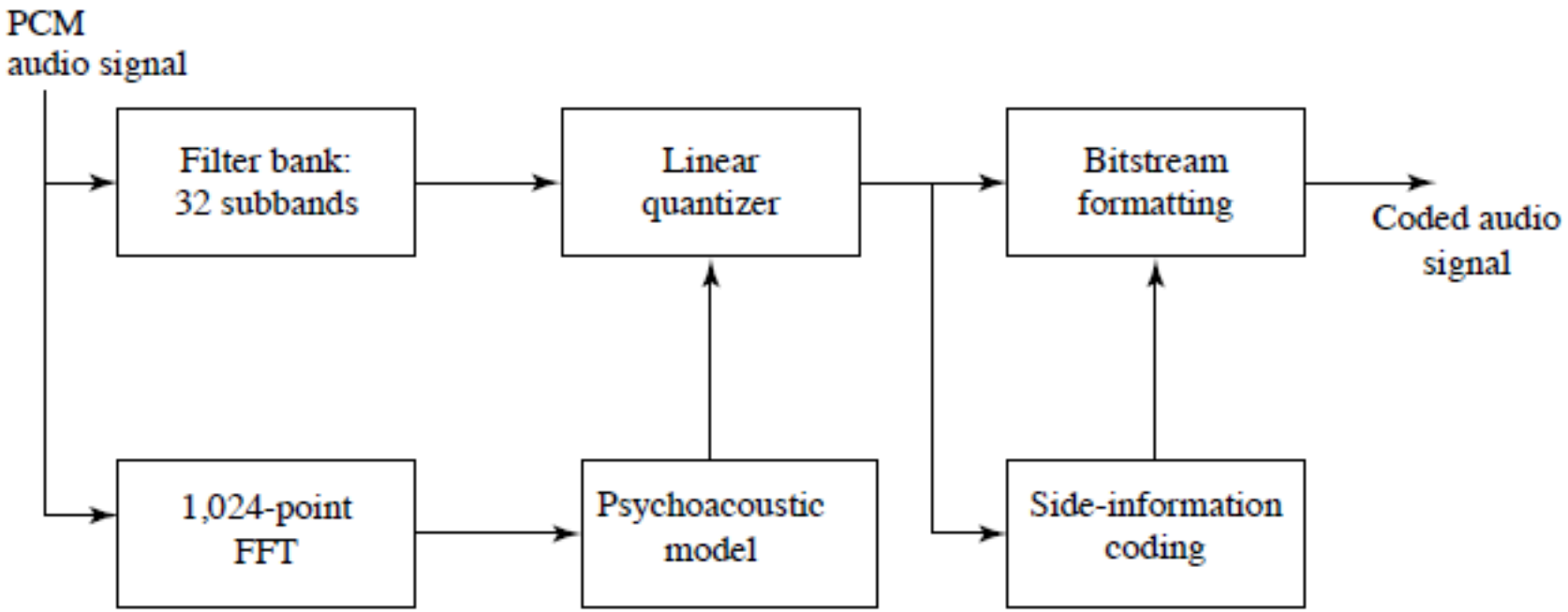
MPEG Layer I

- Frames of 384 samples (12 samples from each subband)
- Uses only frequency masking
- 63 pre-defined scale factors (6 bits to choose any)
- A quantizer is chosen from 16 pre-defined quantizers, each with different step sizes



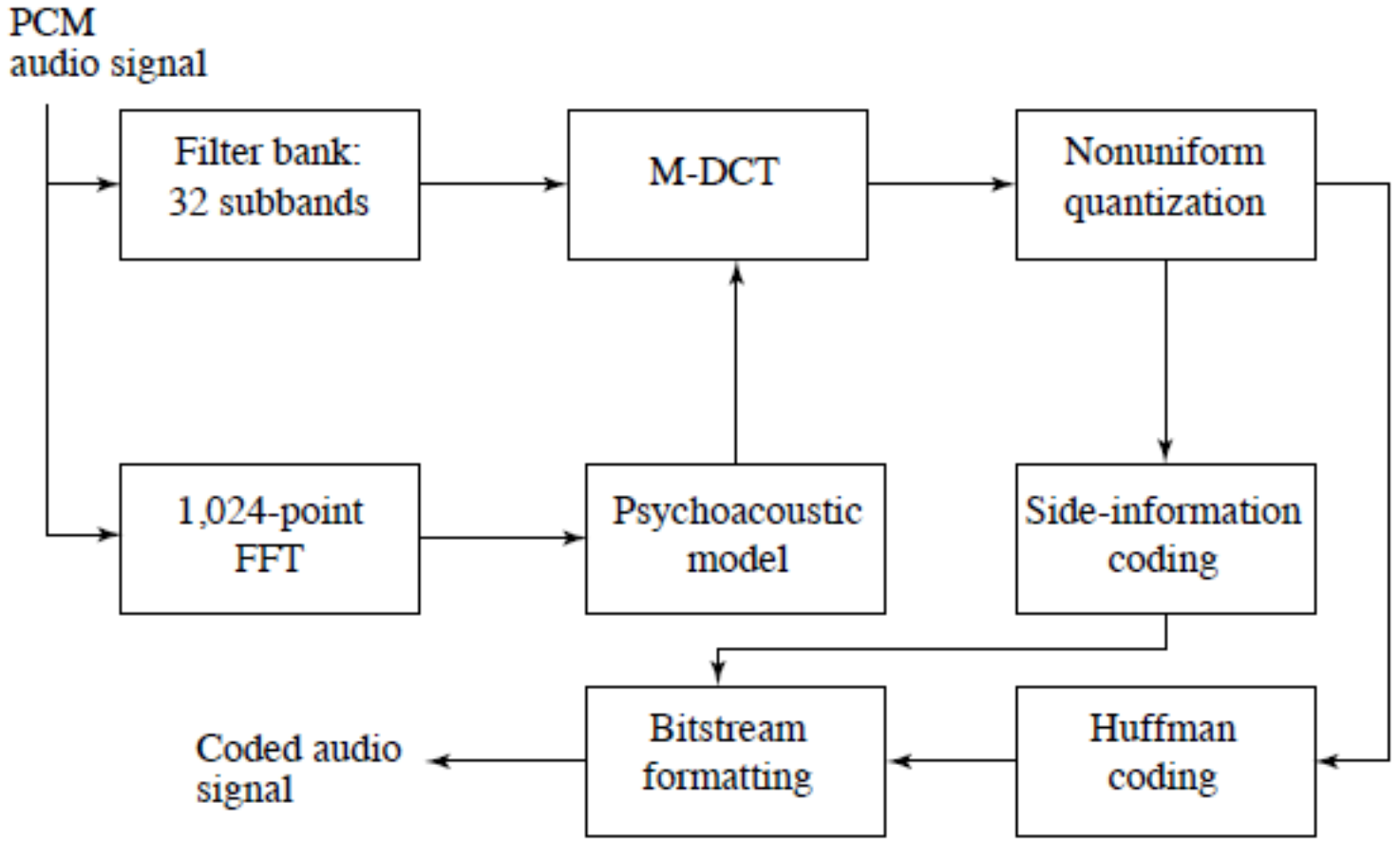
MPEG Layer II

- Frames of 1,152 samples (36 samples from each subband)
- Uses frequency and temporal masking
- 63 pre-defined scale factors (6 bits to choose any)
- The quantizers have higher resolutions (more bits or smaller step size)



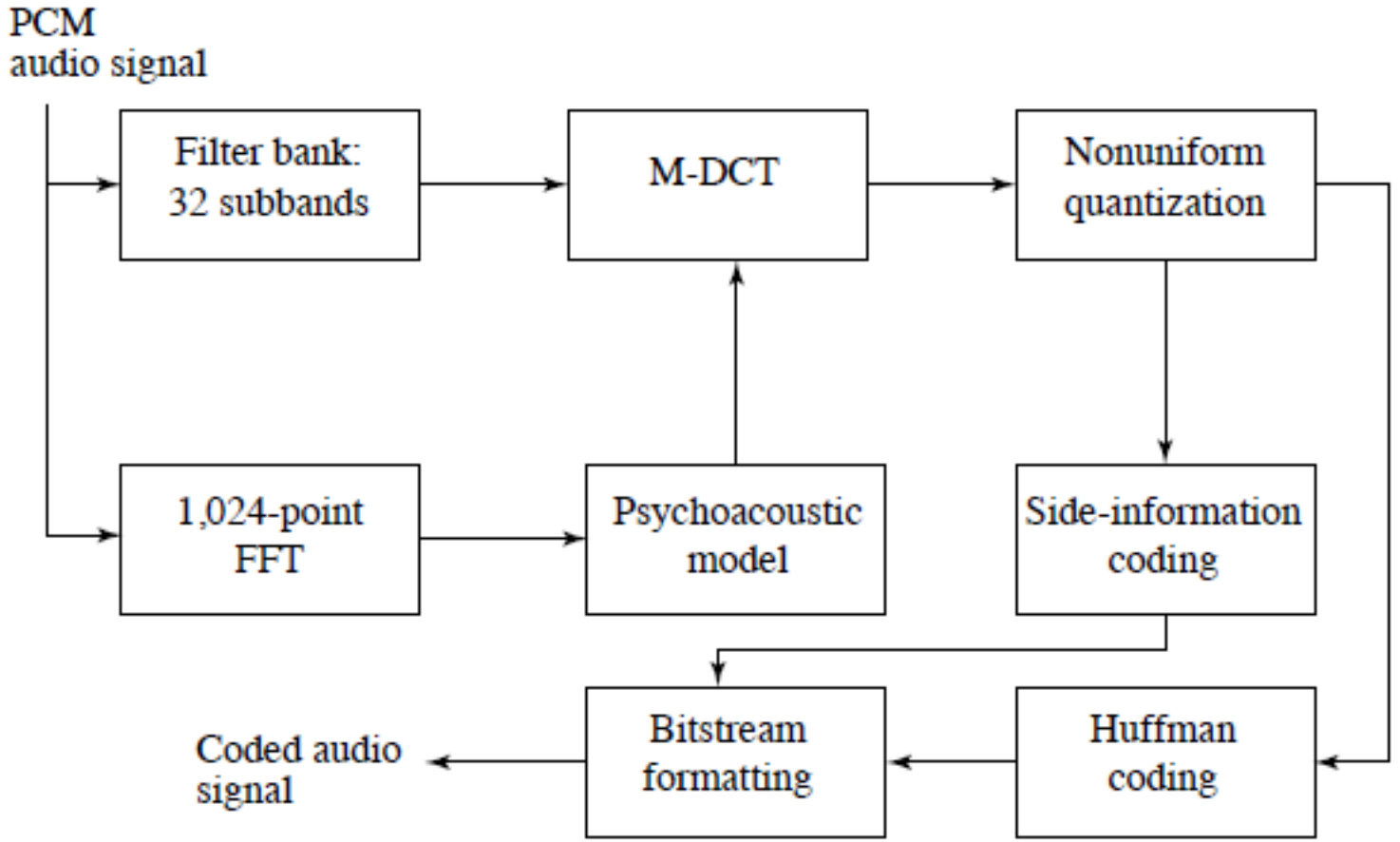
MPEG Layer III

- Frames of 1,152 samples (36 samples from each subband)
- Uses frequency and temporal masking
- Adds a step of *Modified DCT* (MDCT) after the filter bank decomposition, to achieve *higher* spectral resolution for the masking and bit allocation operations



MPEG Layer III

- MDCT coefficients are grouped in non-uniform blocks, based on perceptual critical bands, and the scaling factors are computed from these
- Adds an entropy coding step (Huffman Coding) and uses nonuniform quantization



MPEG Audio Formats

- MPEG supports several audio formats:
  - *Mono*: a single audio channel
  - *Stereo*: two audio channels that are played together but coded separately
  - *Joint Stereo*: two audio channels that are coded together and played together
  - *Dual Channel*: two audio channels that are coded separately and played separately e.g. multi-lingual translation channels

MP3 Compression

Table 14.2: MP3 compression performance

Sound Quality	Bandwidth	Mode	Compression Ratio
Telephony	3.0 kHz	Mono	96:1
Better than Short-wave	4.5 kHz	Mono	48:1
Better than AM radio	7.5 kHz	Mono	24:1
Similar to FM radio	11 kHz	Stereo	26 - 24:1
Near-CD	15 kHz	Stereo	16:1
CD	> 15 kHz	Stereo	14 - 12:1

Other Audio Coders

- *MPEG-2 AAC*
  - Advanced Audio Coding
  - Standard for DVD audio content
  - Based on MDCT
  - Supports three profiles with different complexities and compression ratios
- *MPEG-4 AAC*
  - Enhancement to the MPEG-2 AAC
  - Contains components for speech compression, perceptually based coders, text to speech, ...
- *Dolby AC3*
  - Standard for movies and DVDs
  - Based on MDCT

Recap

- Sound
- Speech Compression
  - DPCM
  - ADPCM
  - Vocoders
- Audio Compression
  - Psychoacoustic
  - MPEG Layer I, II, and III
  - Other coders

- More information: **FM** Ch. 13, 14 and **IDC** Ch. 17, 16.