Part A - MPEG 1 Layer 2

Ibrahim Hamada 201800739 -Radwa Elsawy 201800901-Salma Mahran 201801012
Sohaila Islam 201800998 - Yara Atef 201801745

MPEG-1 Layer II Encoder:

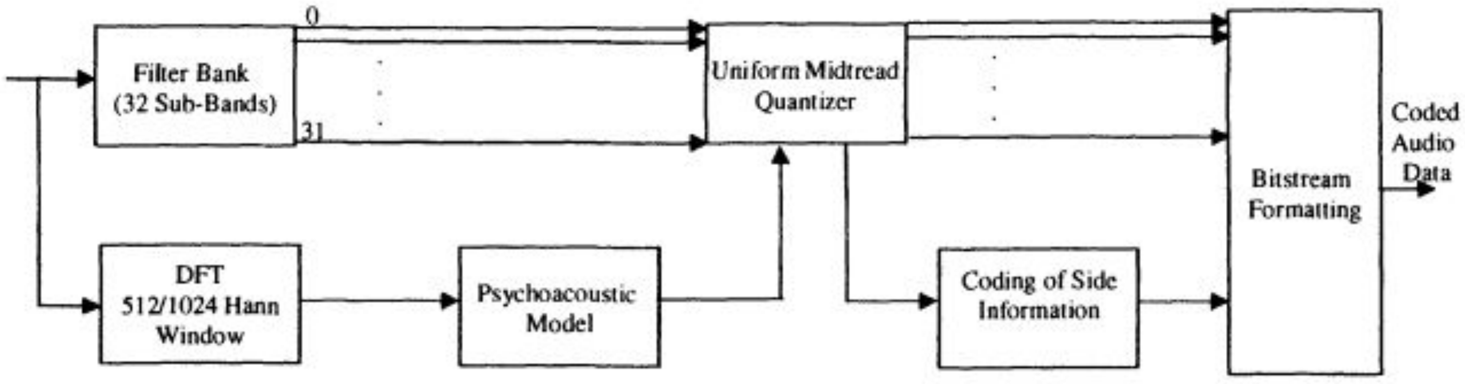


Figure 3. Block diagram of Layers I and II (single channel mode)

MPEG Layer:

•**Three layers of MPEG audio :**

where each higher layer is better in compression but is more complex (at encoder) in the psychoacoustic model.

• MPEG layers generates a good tradeoff between quality and bit-rate.

• **Layer 1** 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 2** has more complexity; was proposed for use in Digital Audio Broadcasting

• **Layer 3** (MP3) is most complex, and was originally aimed at audio transmission over ISDN lines

MPEG Audio Strategy:

• MPEG 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 1 & 2: "quadrature-mirror"; Layer 3: adds a DCT; psychoacoustic model: Fourier transform)

Time to frequency mapping:

- We can often reduce the redundancy in an audio signal by subdividing its content into its frequency components.
- Frequency domain coding techniques have the advantage over time domain techniques since the **number of bits** used to encode each frequency component is **adaptable**.
- This allows us to control the level of quantization noise in each component to ensure that we have the highest coding accuracy in the frequency components that most need it based on the **PSYCHOACOUSTIC MODELS**.

PQMF Filter Banks:

- The filters are used for Time-to-frequency mapping of the signal.
- This filter divides the frequency spectrum into **32** equally spaced frequency sub-bands.
- For Layers I and II the output of the PQMF represents the signal spectral data to be quantized.
- The basic idea is to take a narrow low-pass filter (Prototype filter) and modulate copies of it to span the frequency domain.
- The PQMF filter bank consists of K (**32 in this case**) channels, each of which is a low-pass **filter h[n] modulated by a cosine**.
- The K channels, therefore, lay down **2K** copies of H(f) to divide up the frequency spectrum between **-Fs/2 and Fs/2**.

Analysis filter:

- The filters used has a specific format to allow for perfect reconstruction of the signal at the decoder side:

$$h_k[n] = h[n]\cos\left[\left(k+\frac{1}{2}\right)(n-16)\frac{\pi}{32}\right]$$

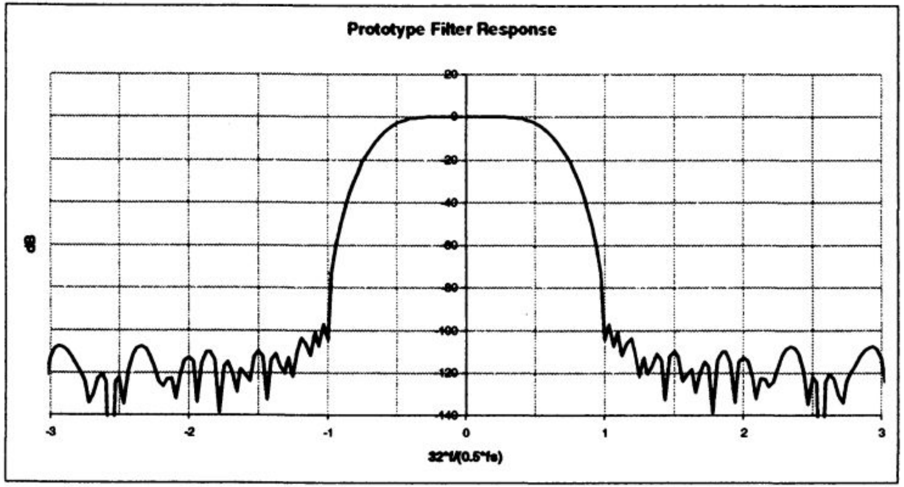


Figure 9. MPEG Audio PQMF prototype filter frequency response in units of $F_s/64$.

- Where :
- h[n] is the coefficients of the prototype low pass filter response shown in fig(9).

Analysis filter:

- Each filter copy has a nominal bandwidth of Fs/64.
- The analysis filter ith output at time t is given by:

$$y_t[i] = \sum_{n=0}^{511} x[t-n] \times h_k[n]$$

- $h_k[n]$ response is shifted to frequencies

$$f_k = \pm \frac{(k+\frac{1}{2})}{K} F_s / 2$$

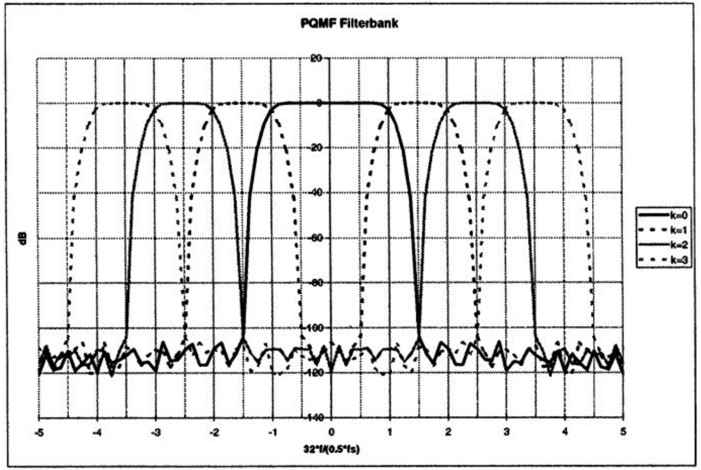
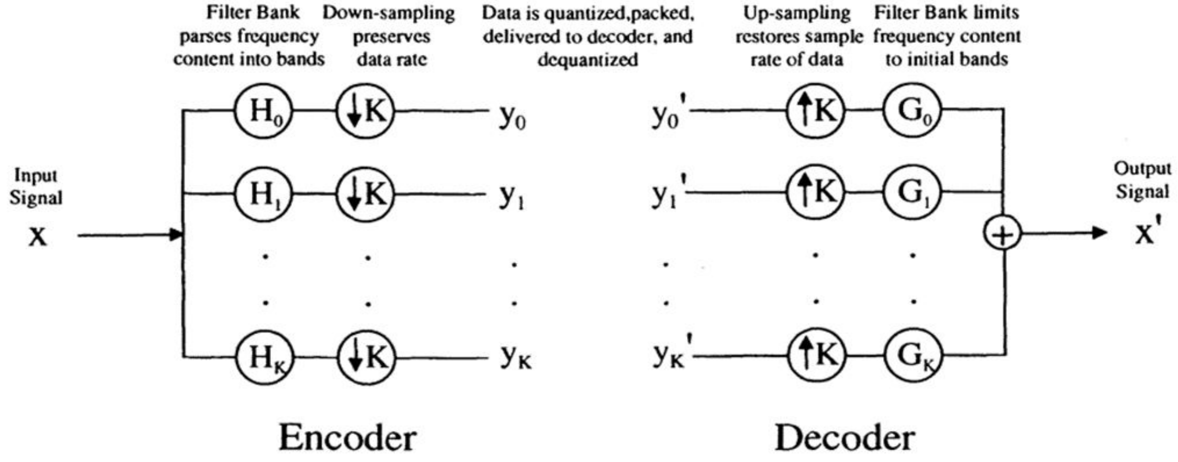


Figure 10. Frequency response of the first four bands of the MPEG Audio PQMF in units of $F_s/64$

Time to frequency mapping (Filters Bank):

- By splitting the signal into 32 parallel bands, we have multiplied our data rate by a factor of 32. To avoid raising the data rate when passing the signal through the encoder filter bank, we **down sample** by a factor of **32**.
- Each subband filter produces **1 sample** out for every **32 samples in**.



Implementation of the filter banks:

- When implementing the filter banks, the following equation is used because it has less computational complexity than the regular convolution equation

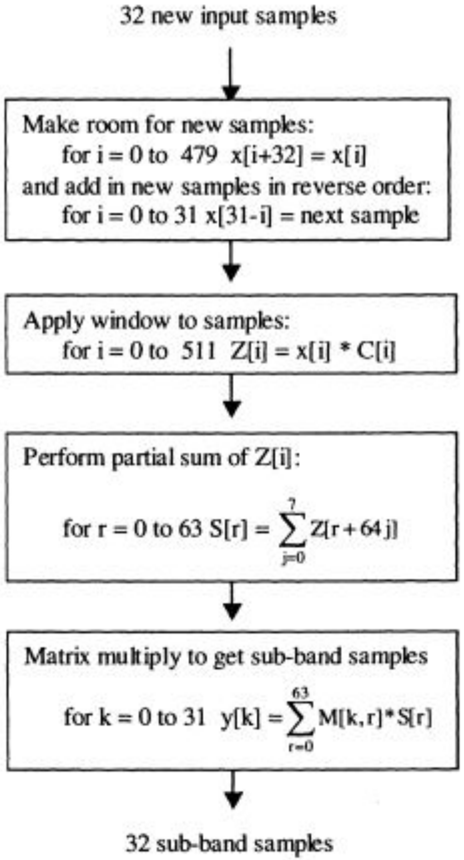
$$y_m[k] = \sum_{r=0}^{63} M[k,r] * \sum_{p=0}^{7} \left[C[r+64p] * x_m[r+64p] \right] \quad \text{for all m and k=0,\ldots,31}$$

- Where:

$$M[k,r] = \cos\left(\left(k + \frac{1}{2} \right)(r-16)\frac{\pi}{32} \right) \qquad C[n] = (-1)^{\mathrm{int}\left(\frac{n}{64}\right)} h[n]$$

- Ym[k] is the output of the kth analysis filter after processing the mth block of 32 new input samples.

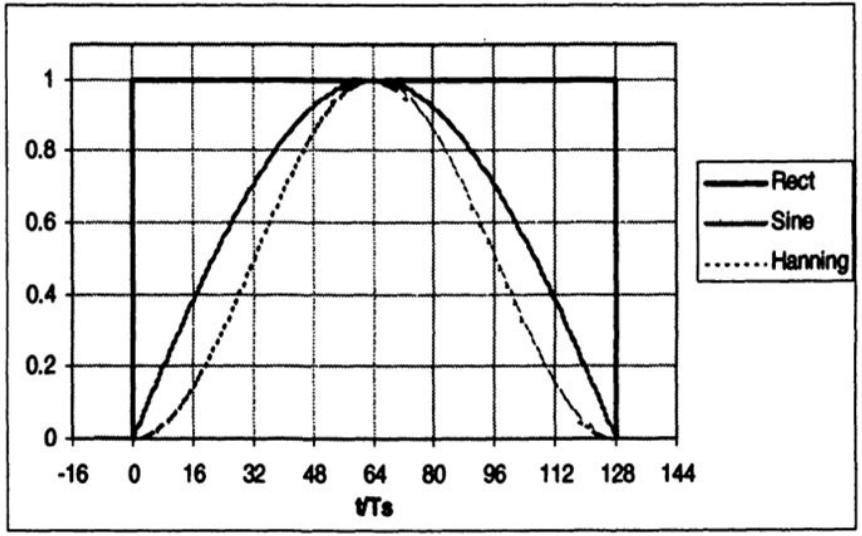
- A direct implementation of the PQMF filter bank results about **512 multiplications and additions per sample**.
- Using the implementation given in the previous slide, about **80 multiplications and additions per sample** only would be required.



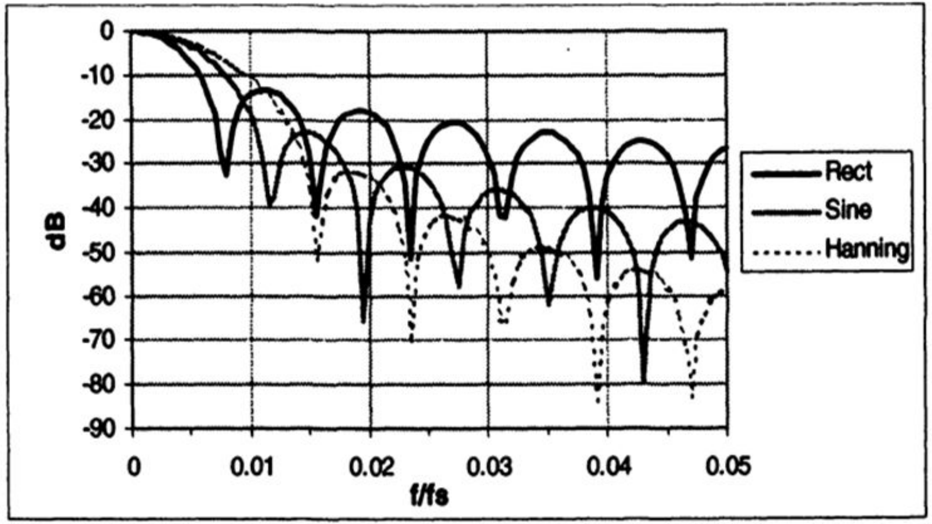
Time to frequency mapping (DFT):

- Another approach to the time to frequency mapping of audio signals.
- If we have K data samples in the time domain, they can be transformed into K data samples in the frequency domain using the Discrete Fourier Transform, DFT.
- Fast and efficient algorithms such as FFT were developed for carrying out these transforms for large block sizes.
- We would like to work with a finite extent of sampled data and be able to map it into discrete frequencies in a finite range without any loss of information.
- This can be reached with a careful choice of windows.

Windowing the Signal in the Time Domain:



Time domain comparison of the rectangular, sine and Hanning windows for T=128 * Ts



Frequency domain comparison of the rectangular, sine and Hanning windows for T=128*Ts (Note: Windows are normalized to have integral equal to I prior to graphing.)

Windowing the Signal in the Time Domain:

- Convolution theorem tells us that windowing in the time domain is equivalent to a convolution in the frequency domain.

$$w_H(t) = ½(1-\cos(2\pi t/T))$$

1. Hanning Window:

$$W_H(f) = \int_{-\infty}^{\infty} w_H(t)e^{-j2\pi ft}dt = \int_0^T \tfrac{1}{2}(1-\cos(2\pi t/T))e^{-j2\pi ft}dt \quad = e^{-j\pi fT}\frac{\sin(\pi fT)}{\pi f}\left(\frac{1/2}{1-(fT)^2}\right)$$

- Doesn't have the sudden change in derivative at the edges like sine and rect windows.
- The drop-off is much faster for the Hanning window (good to avoid aliasing)

DFT:

- For a signal x(t) that was Hanning windowed to finite length by a windowed signal that is also band-limited.
- Then, we select an adequate sample rate Fs = 1/Ts and that the signal duration is T = N*Ts. Since the windowed signal is finite length, we can work only with discrete frequency components.
- The Fourier series tells us that we can get signal values as a sum over these frequency components; so we can write:

$$x[n] \equiv x(nT_s) = \frac{1}{N}\sum_{k=0}^{N-1} X[k]e^{j2\pi kn/N} \qquad n = 0,\ldots,N-1$$

- Likewise, we can write the frequency components as a Fourier series sum over time samples as:

$$X[k] \equiv F_s X(kF_s/N) = \sum_{n=0}^{N-1} x[n]e^{-j2\pi kn/N} \qquad k = 0,\ldots,N-1$$

- This pair of transform is known as DFT
- The psychoacoustic analysis stage in Layer II is performed with a 1024-point FFT.

Psychoacoustics:

- The hearing threshold, or threshold in quiet, represents the lowest sound level that can be heard at a given frequency.
- Even in extremely quiet conditions, the human ear cannot detect sounds at SPLs below the threshold in quiet.

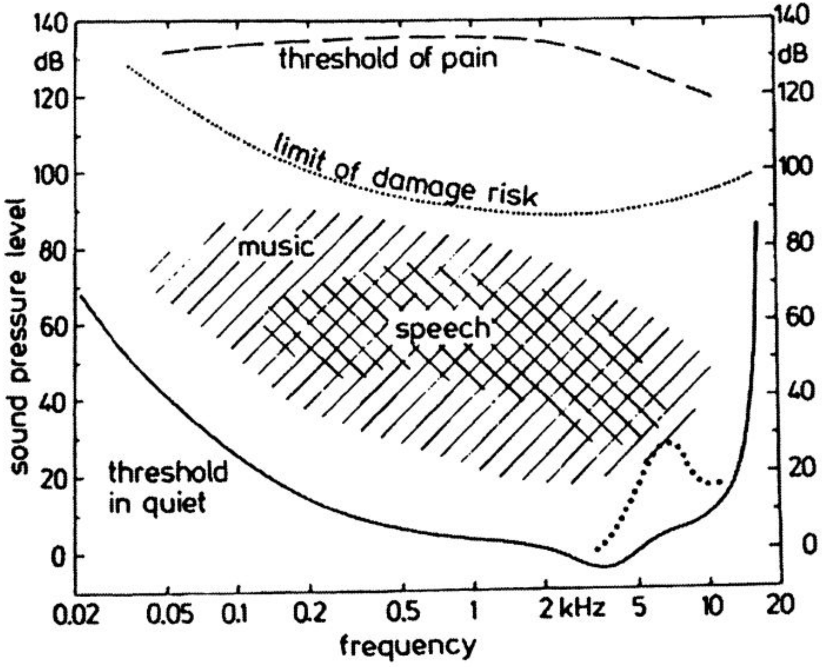


Figure 2. Hearing area from [Zwicker and Fastl 90]

Psychoacoustics:

- This fact can be used in auditory coding. Since frequency components in a signal that fall below this level are irrelevant to our perception of sound and therefore they do not need to be transmitted.
- Human ears can't hear signals (such as quantization noise) under the condition of Auditory masking.
- **Auditory masking** occurs when the perception of one sound is affected by the presence of another sound.

Frequency (Simultaneous) masking:

- It occurs whenever the presence of a strong audio signal makes **a spectral neighborhood** of weaker audio signals imperceptible.
- In fig(6) , a loud signal is masking two other signals at nearby frequencies.
- Signals or frequency components with SPL below the masking threshold will not be heard when the masker is present.

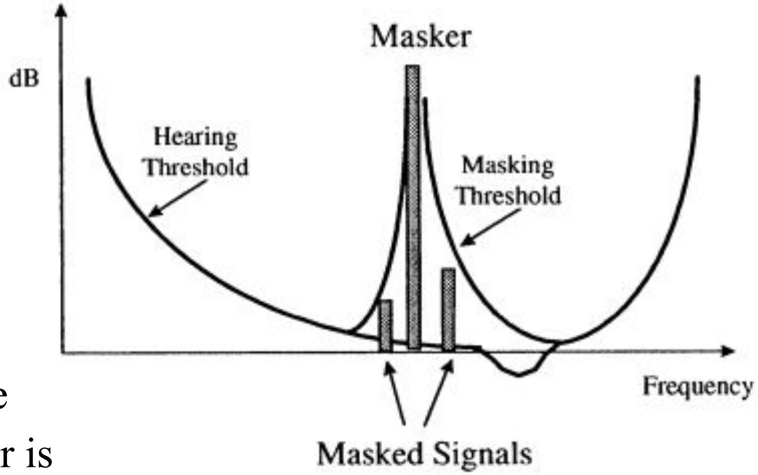


Figure 6. Example of frequency masking

Frequency (Simultaneous) masking:

- Just like with the threshold in quiet, we can exploit the masking thresholds in coding to identify signal components that do not need to be transmitted and to determine how much inaudible quantization noise is allowed for signal components that are transmitted.
- The masking threshold differ depending on the factors, as nature and other characteristics of the masker and the prope (test signal)
- In the frequency masking, the prope and the masker can be sinusoidal tone or narrow band noise of extended duration.

Psychoacoustic Model 1:

- MPEG compresses the signal by removing acoustically irrelevant parts of the signal.
- The first step in both MPEG psychoacoustic models is to time-align the audio data used by
- the psychoacoustic model stage with the main path audio data.
- Fig(8) shows the block diagram for MPEG Psychoacoustic model 1.

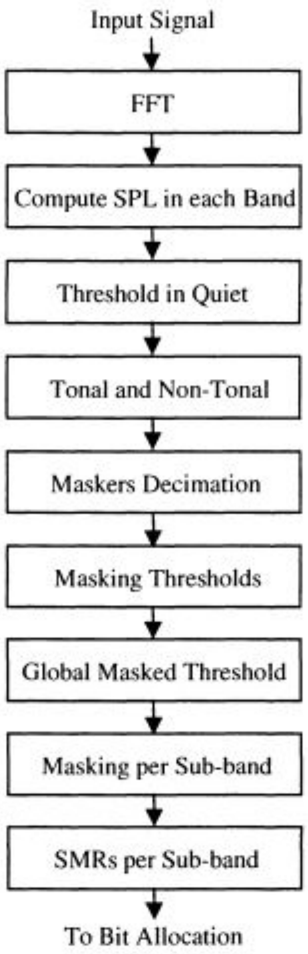


Figure 8. Block diagram of MPEG Psychoacoustic Model 1

Time signal alignment:

- The audio data sent to the psychoacoustic model must be concurrent with the audio data to be coded.
- The model must consider the delay of the audio data through the filter bank and a data offset so that the relevant data is centered within the analysis window.
- Fourier transform is computed in parallel to the main audio path time to frequency mapping stage to provide finer frequency resolution for the psychoacoustic block.

Calculate SPL in each band:

- For each subband, the SPL is calculated for each spectral line L_k

$$L_k = 96\,\text{dB} + 10\log_{10}\left(4 / N^2 \,|X[k]|^2\, 8/3\right) \quad \text{for } k=0,\ldots,N/2-1$$

- X[k] is FFT amplitude of the corresponding spectral line.

- Then Sound pressure of a subband:

$$L_{sb}[m] = \max\{L_k, 20\log_{10}(scf_{max}[m]\, 32{,}768) - 10\}\text{dB}$$

Tonal and non-tonal Components:

- To calculate the SMR , masking threshold is computed for each subband.
- Since noise is a better masker than tones, we need to identify and separate the tonal and noise components of the audio signal.
- Assuming that the local maximum of the critical band represents the tonal components of the signal, then The local maximum L_k , is included in the list of the tonal components if :

$$L_k - L_{k+j} \geq 7 \text{ dB}$$

- The power of the remaining spectral components is summed to form SPL of the noise masker L_N for that critical band.
- The noise masker components for each critical band are centered at the geometric mean of the FFf spectral line indices for each critical band

Maskers Decimation:

- Reducing the number of maskers.
- The masking thresholds are then computed for the remaining maskers by applying a spreading function and shifting the curves down by a certain amount of dB which depends on whether the masker is tonal or noise-like and the frequency position of the masker.
- The masking thresholds are evaluated at only a sub-sampling of the frequency lines.
- In layer 2 the number of sub-samples lines ranging from 126 to 132.

Masked Threshold:

- The global masked threshold MG(Zj) is computed in Model I by summing the power of the individual masking thresholds and the threshold in quiet as follows:

$$M_G(z_i) = 10\log_{10}[10^{\frac{M_q(z_i)}{10}} + \sum_{j=1}^{m} 10^{\frac{M_{T_j}(L_j, z_i, z_j)}{10}} + \sum_{k=1}^{n} 10^{\frac{M_{N_k}(L_k, z_i, z_k)}{10}}]$$

- Where:

Mq represents the threshold in quiet

MTj themasking threshold from the lh tonal masker,

MNk the masking threshold from the kth noise-like masker,

SMR Computation:

- The signal to mask ratio, for each sub-band, that would be transmitted to the bit allocation block is calculated using:

$$SMR(sb) = L(sb) - M_{Gmin}(sb)$$

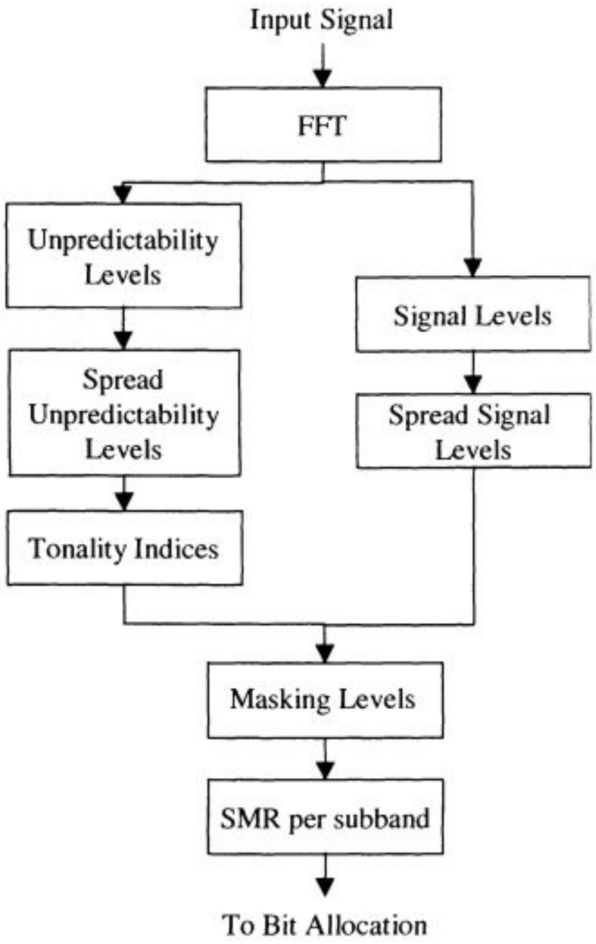
- where :

L(sb) is the sound pressure level for the sub band.

MGmin(sb) is the minimum global masking threshold in a subband. Since some PQMF sub-bands span more than one critical band, MGmin(sb)is determined based on the minimum masking level at the sub-sampled lines in that sub-band.

Differences in Model 2:

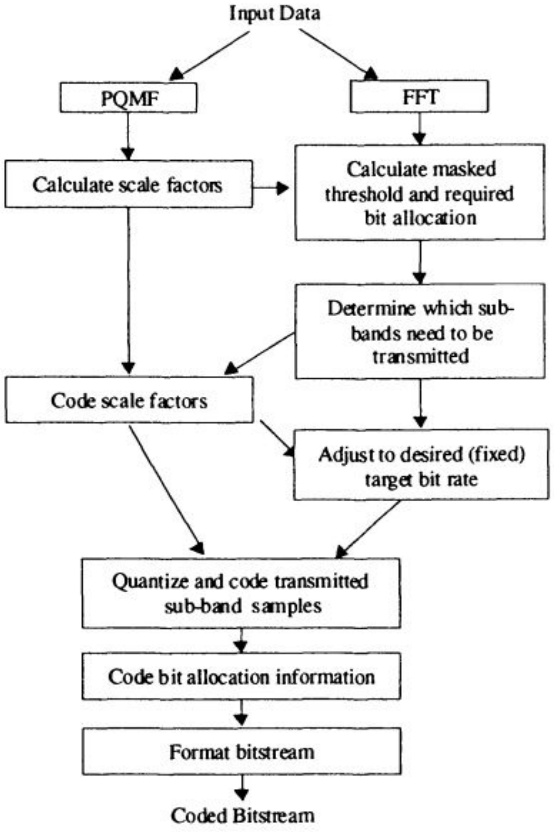
- The **tonal and non tonal are not separated**, instead of that the model computes a tonality index as a function of frequency.
- The total masking energy for the signal frame is computed by first convolving a spreading function with each of the maskers in the signal. Where the used **spreading function differ** from that of Model 1.
- In finding the masking threshold subbands, the **higher frequency subbands has the same accuracy** as the lower frequency subband



Encoding:

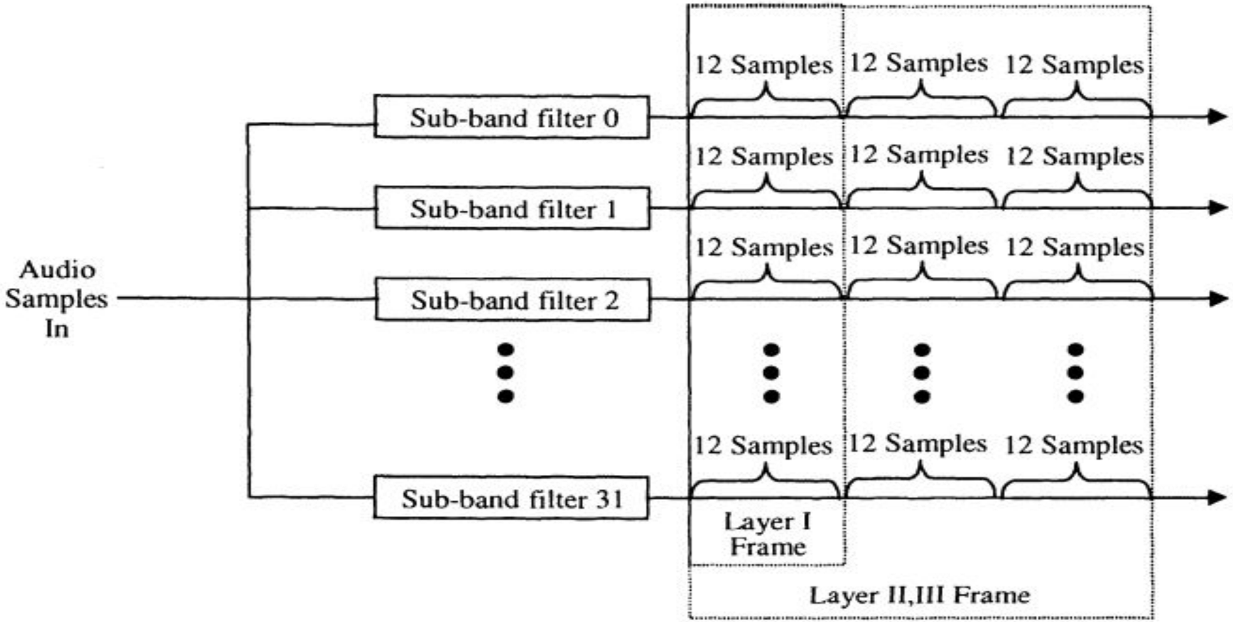
- Scale factor calculation
- Scale factors select information.
- Bit Allocation.
- Quantization.

Basic structure of the encoding process



Scale factors:

- The scale factor is transmitted only if the bit allocation for that band is non-zero.
- The scale factors in Layer II can be shared among the three consecutive granules. The scale factors are computed in the same manner as in Layer I (using tables).
- The audio encoded data in each frame represent **24 ms** of data at a **48 kHz sampling rate.**



In which the scale factor is represented in 6 bits, since it is 63 different value which is 2^N-1.

Scale Factors Computation:

- One scale factor is computed for each 12 sub-band frequency samples (called a "granule") and it is represented using 6 bits. The dynamic range covered by the scale factors is **120 dB.**
- Layer 2 may be stereo mode or mono mode.
- **The maximum loudness** of a sequence granule mapped into a scale factor value via lookup table.

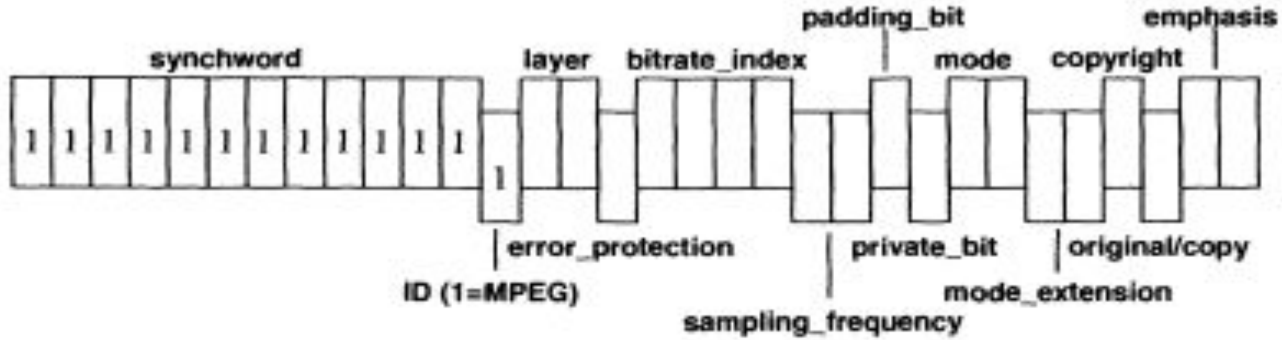


Figure 20. MPEG-1 Audio Header

LAYER I	Header (32)	CRC (0,16)	Bit Allocation (128-256)	Scale Factors (0-384)	Samples	Ancillary Data

LAYER II	Header (32)	CRC (0,16)	Bit Allocation (26-188)	SCFSI (0-120)	Scale Factors (0-1080)	Samples	Ancillary Data

Scale factor select information:

- stores how the loudness changes on three subsequent groups; a value of 0 indicates no change, so the loudness is stable, a value of 2 indicates a transient change, and the values 1 indicate an unstable change
- The bits used for SCFSI vary between zero and **30 * 2 * 2 = 120 bits** for each frame and the bits used for scale factors vary between zero and **6 * 3 * 30 * 2 = 1080 bits per frame.**

Bit allocation:

- **The bit allocation** routine is an **iterative process**,in each iteration additional bits are allocated to the sub-band with the **highest noise-to-mask ratio.**
- **Bit allocation tables** are now specified in the standard that **determines the possible quantization levels** that can be used to quantize the samples in any sub-band **depending on the sample rate** and the target data rate.
- Bit allocation codes range **from zero to 4 bits per sub-band,** where a greater number of bits are used to code the bit allocations of lower frequency sub-bands.
- In all cases, no bits are allocated to the two highest frequency sub-bands in Layer II.

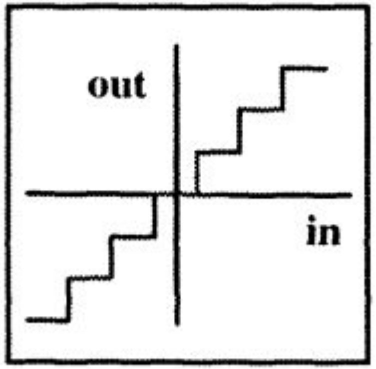
Bit allocation:

- In other sub-bands, the number of quantization levels can reach that of 16-bit quantization.
- The total number of bits employed to describe the bit allocation for one frame of the **stereo signal varies between 26 and 188 depending on which table is applicable.**

Midtread Uniform Quantizer:

- Midtread quantizers are **able to pass a zero output** and, due to the symmetry between how positive and negative signals are quantized, necessarily have an **odd** number of output steps.
- With R number of bits the midtread quantizer allows for **2^R -1** different codes versus the **2^R** codes allowed by the midrise quantizer.
- In spite of the smaller number of codes allowed, in general, given the distribution of audio signal amplitudes, midtread quantizer yield better results.

Midtread Uniform Quantizer:



Midtread

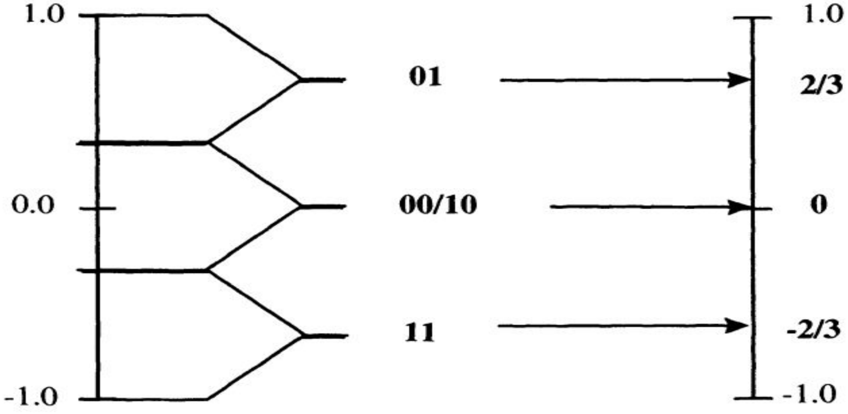


Figure 4. A two-bit uniform midtread quantizer

Quantization:

A Quantization procedure for an R-bit uniform midtread quantizer is given by

Quantize:

$$\text{code}(\text{number}; R) = [s][|\text{code}|]$$

where

$$s = \begin{cases} 0 & \text{number} \geq 0 \\ 1 & \text{number} < 0 \end{cases}$$

$$|\text{code}| = \begin{cases} 2^{R-1} - 1 & \text{when } |\text{number}| \geq 1 \\ \text{INT}(((2^R - 1)|\text{number}| + 1) / 2) & \text{elsewhere} \end{cases}$$

Quantization in Layer II:

- In Layer II, quantization is carried out by dividing each sub-band sample by its scale factor and quantizing using a midtread uniform quantizer having a number of steps determined by the bit allocation for that sub-band.
- In general Layer II represents bit allocation, scale factors, and quantized samples in a more compact way than Layer I, so that more bits are available to represent the coded audio data.
- Layer II provides higher quality than Layer I at any given data rate.

Vector Coding:

- Instead of quantizing one sample at a time, Layer II code three consecutive samples as a code word.
- This vector coding approach saves bits by allocating bits to enumerate possible triplets of coded values rather than to each quantized value individually
- Example of a code word consisting of 7 bits **V5 =25z + 5y + x**
- where x, y, z, are the quantization levels corresponding to three consecutive sub-band samples
- 3 consecutive 5-step quantized values would require 3*3 = **9 bits** but the 5*5*5 = 125 possible code triplets can be enumerated with a **7-bit code.**

Uniform Quantizers errors:

- The uniform quantizer has a maximum round-off error equal to half of the bin width at any, non-overload input level.
- However, this error level could be huge relative to a very low amplitude signal.
- Since the perception of round-off distortion is more related to the relative error in amplitude than to the absolute size of the error, this means that uniform quantizers perform significantly worse on low power input signals than they do on higher power signals.

Decoder:

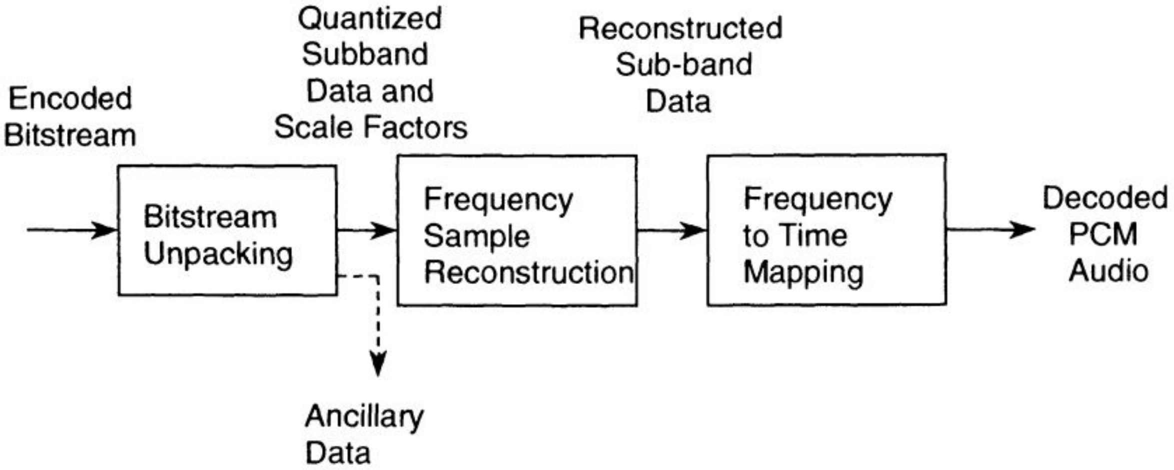


Figure 2. MPEG-1 Audio decoder basic building blocks

Reconstruction:

- In this step, we need to restore the values of the samples before they were quantized.
- This will be done by using a midtread uniform quantizer having a number of steps determined by the bit allocation for that sub-band (so that quantization and dequantization are done using the same number of levels for each subband).
- After dequantization, each subband sample is multiplied by its scale factor.

Reconstruction:

- A dequantization procedure for an R-bit uniform midtread quantizer is given by:

Dequantize:

$$\text{number}(\text{code}; R) = \text{sign} * |\text{number}|$$

where

$$\text{sign} = \begin{cases} 1 & \text{if } s = 0 \\ -1 & \text{if } s = 1 \end{cases}$$

$$|\text{number}| = 2.|\text{code}| / (2^R - 1)$$

Synthesis Filter:

- To set up the filter banks so that there is no aliasing of the signal, the synthesis filters are defined in terms of the analyzing filters:

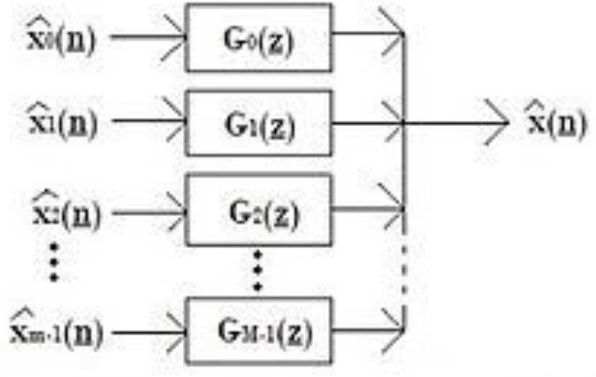
$$g_k[n] = h_k[N-1-n]$$

- Substituting with $h_k[n]$ for the PQMF :

$$g_k[n] = 32\,h[n]\cos\left[\left(k+\frac{1}{2}\right)(n+16)\frac{\pi}{32}\right]$$

Synthesis Filter:

- The data is up-sampled by a factor of 32 prior to recombining the sub-band signals to space it out back to the original data rate.



Multidimensional Synthesis Filter Banks

Performance measure:

- Compression Ratio is the ratio of uncompressed data size to the compressed data size

$$CR = \frac{\text{uncompressed signal}}{\text{compressed signal}}$$

Thanks