ELEC S347F Multimedia Technologies

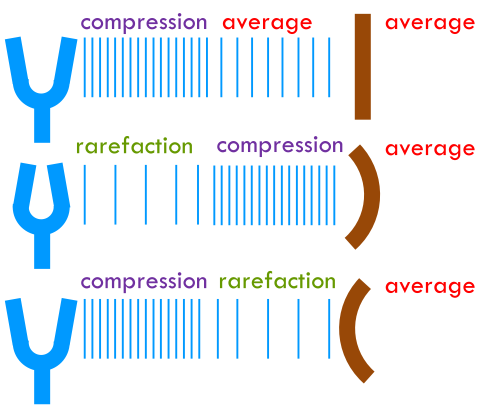
Audio Representation and Compression

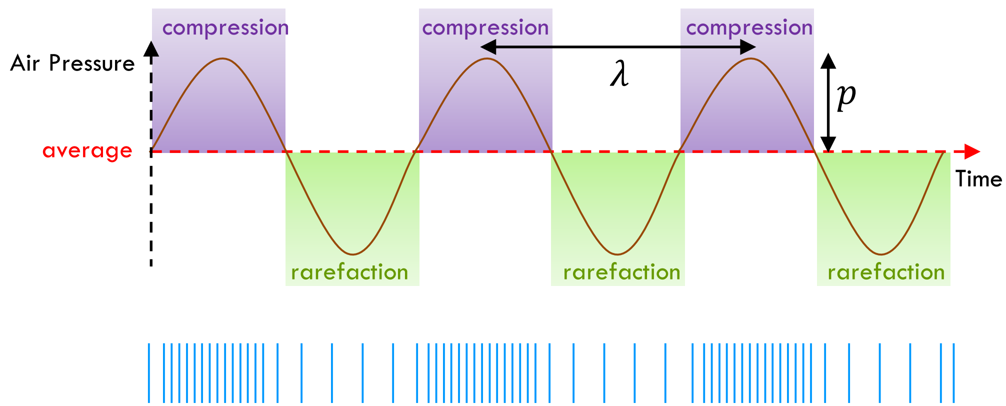
Outline

- Sound Basics
  - Sensitivity, Psychoacoustics, Critical Bands
  - Frequency Masking, Temporal Masking
- Digital Audio Representation
  - Sampling Rate, Quantization, Pulse Code Modulation
  - Silence Compression, DPCM, ADPCM
- Industrial Standards
  - MPEG-1 Audio, MPEG-2 Audio
  - MIDI, GM, HD-MIDI
  - MPEG-4 Structured Audio, MPEG-4 SLS
  - RTP-MIDI, OSC, HTML5 Audio

Sound Basics

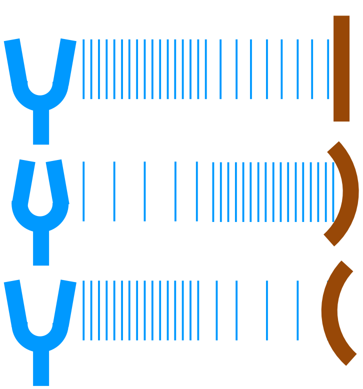
- Sound is a pressure wave produced by a vibrating source
  - The vibrations disturb air molecules and produce variations in air pressure
  - Lower than average pressure: Rarefactions
  - Higher than average pressure: Compressions

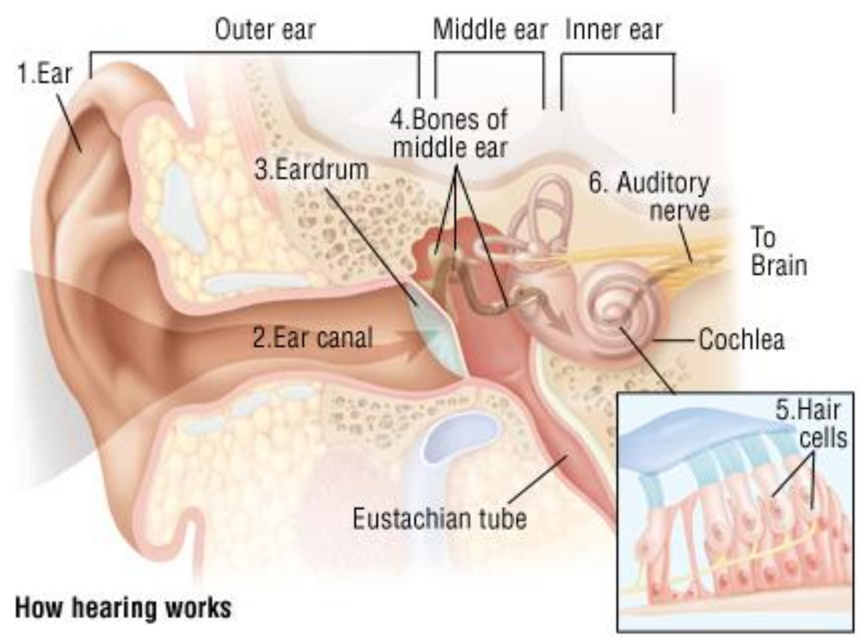


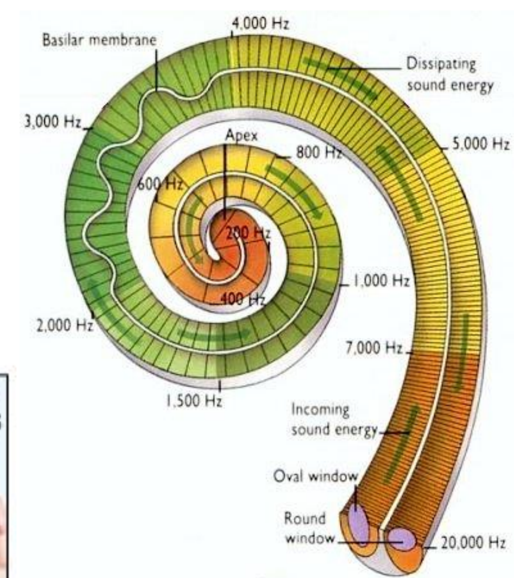


Sound Basics

- When a sound wave impinges on a surface
  - It causes the surface to vibrate in sympathy
  - In this way acoustic energy is transferred from a source to a receptor







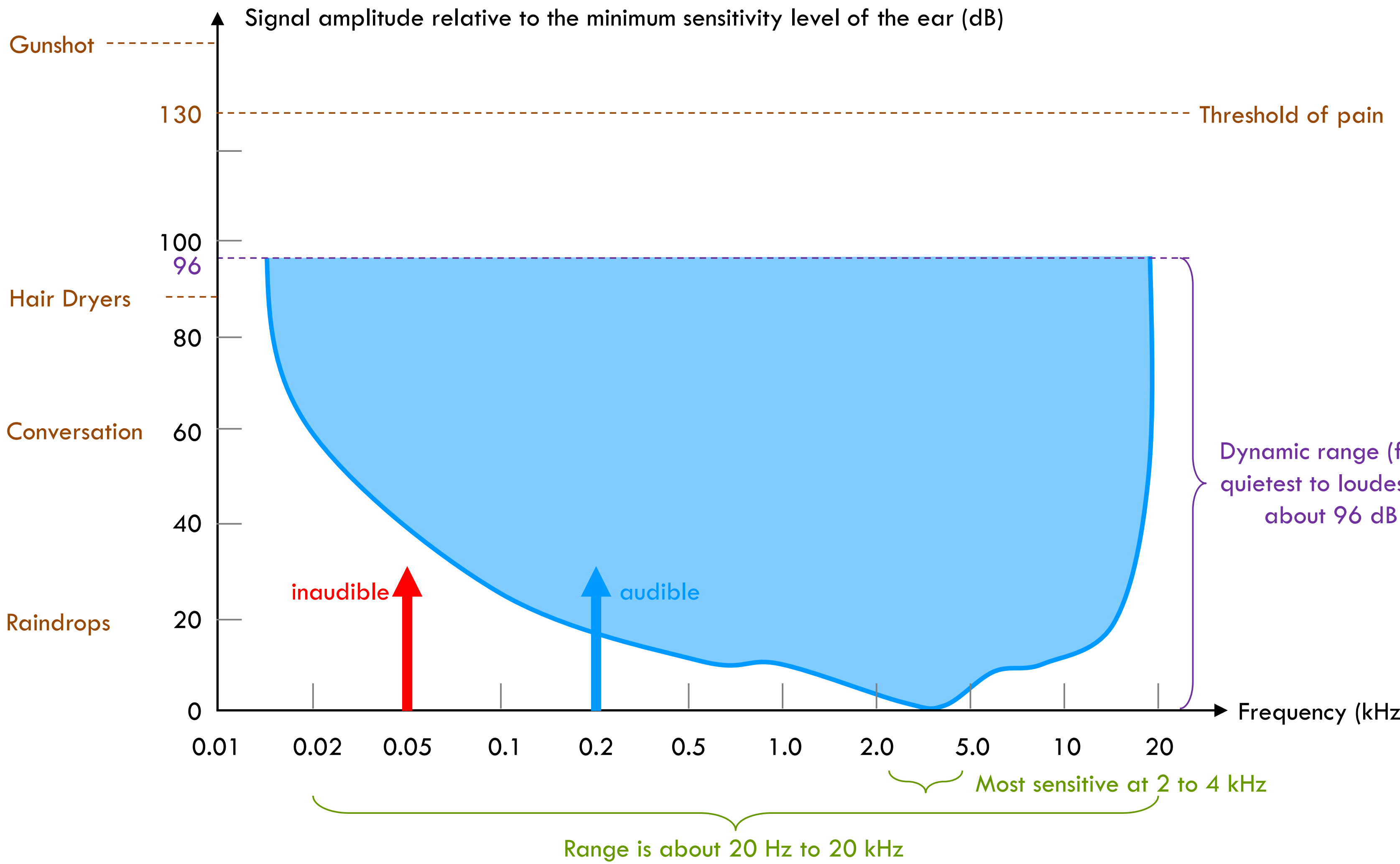
How Human Hearing Works?

- Outer Ear
  - Ear canal: focuses the incoming sound
  - Eardrum: upon receiving the waveform, the eardrum vibrates in sympathy
- Middle Ear
  - Bones of middle ear: amplify the force of sound vibrations
- Inner Ear
  - Cochlea: filled with fluid
    - It transforms mechanical ossicle forces into hydraulic pressure
  - Stereocilia (Hair Cells): inner surface of the cochlea
    - Tight at one end, looser at the other
    - Differ in length by minuscule amounts
    - So have different degrees of resiliency to the fluid which passes over them
    - Increased vibrational amplitude induces the cell to release an electrical impulse which passes along the auditory nerve towards the brain
- Brain
  - Interprets the sound upon reception of these electric nerve impulses

Sensitivity of Human Hearing

- Frequency is measured in Hertz (Hz in short) which represents the number of vibration cycles per second
- The average human ear can perceive sound frequencies between 20 Hz to 20,000 Hz
  - Sounds with frequencies from 250 Hz to 6 kHz are important for communication in human beings
  - All phonemes in human speech fall within this range
  - For example, frequencies around 4 kHz are mostly composed of the speech sounds of “f”, “k” and “th”
  - An approximate rule of thumb: as your body ages, it loses 1 Hz of sensitivity from the top end of the hearing range every day (however, some will have negligible hearing loss as they age)

Sensitivity of Human Hearing

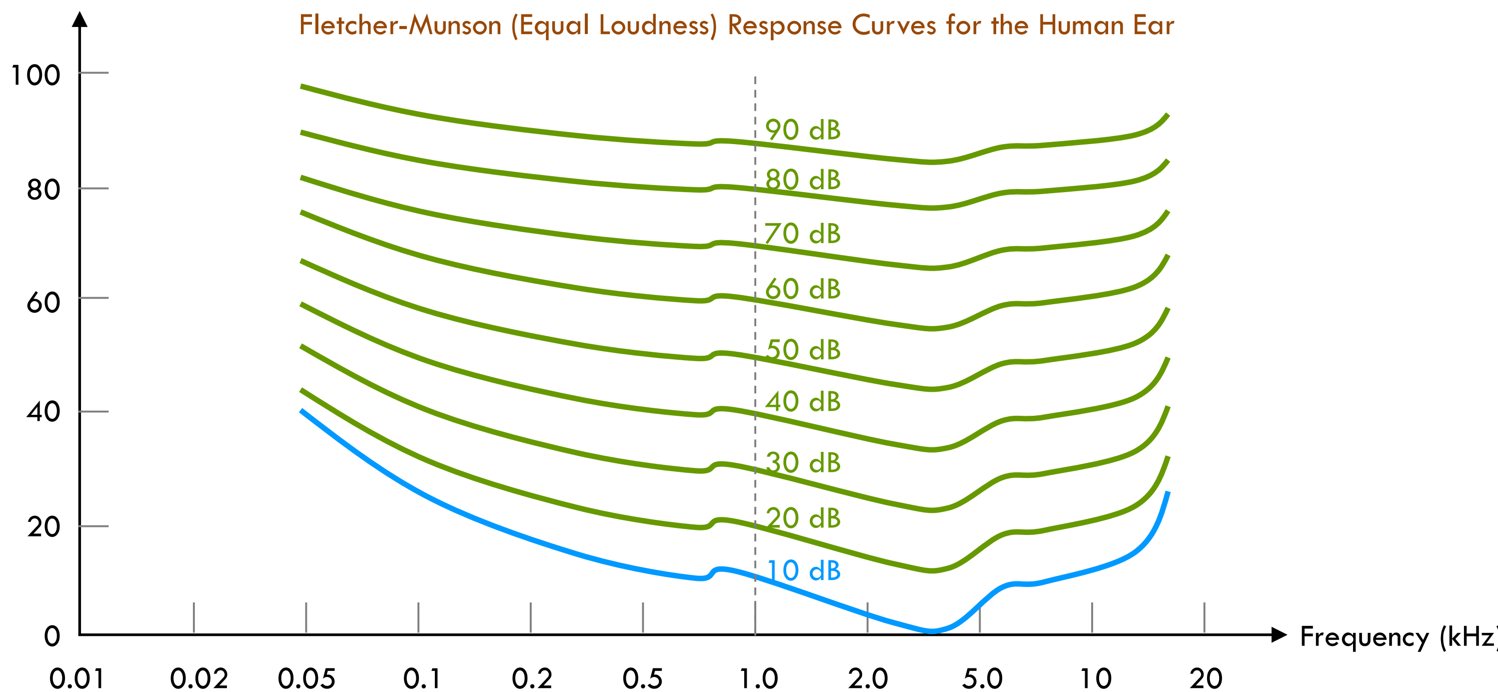


Interpretation of Decibel Scale

- Decibel (dB): a logarithmic measurement of sound
  - Defined as $10\log_{10}(P/P_0)$
  - 0 dB: $P = P_0$, threshold of hearing (TOH)
  - 10 dB: $P = 10P_0$, 10 times more intense than TOH
  - 20 dB: $P = 10^2P_0$, 100 times more intense than TOH
  - 30 dB: $P = 10^3P_0$, 1000 times more intense than TOH
- An increase in 10 dB means that the intensity of the sound increases by a factor of 10
  - If a sound is 10^n times more intense than another
  - It has a sound level that is $10n$ more decibels than the less intense sound

Fletcher-Munson Response Curves

- Pure tone stimuli producing the same perceived loudness "Phons" in dB



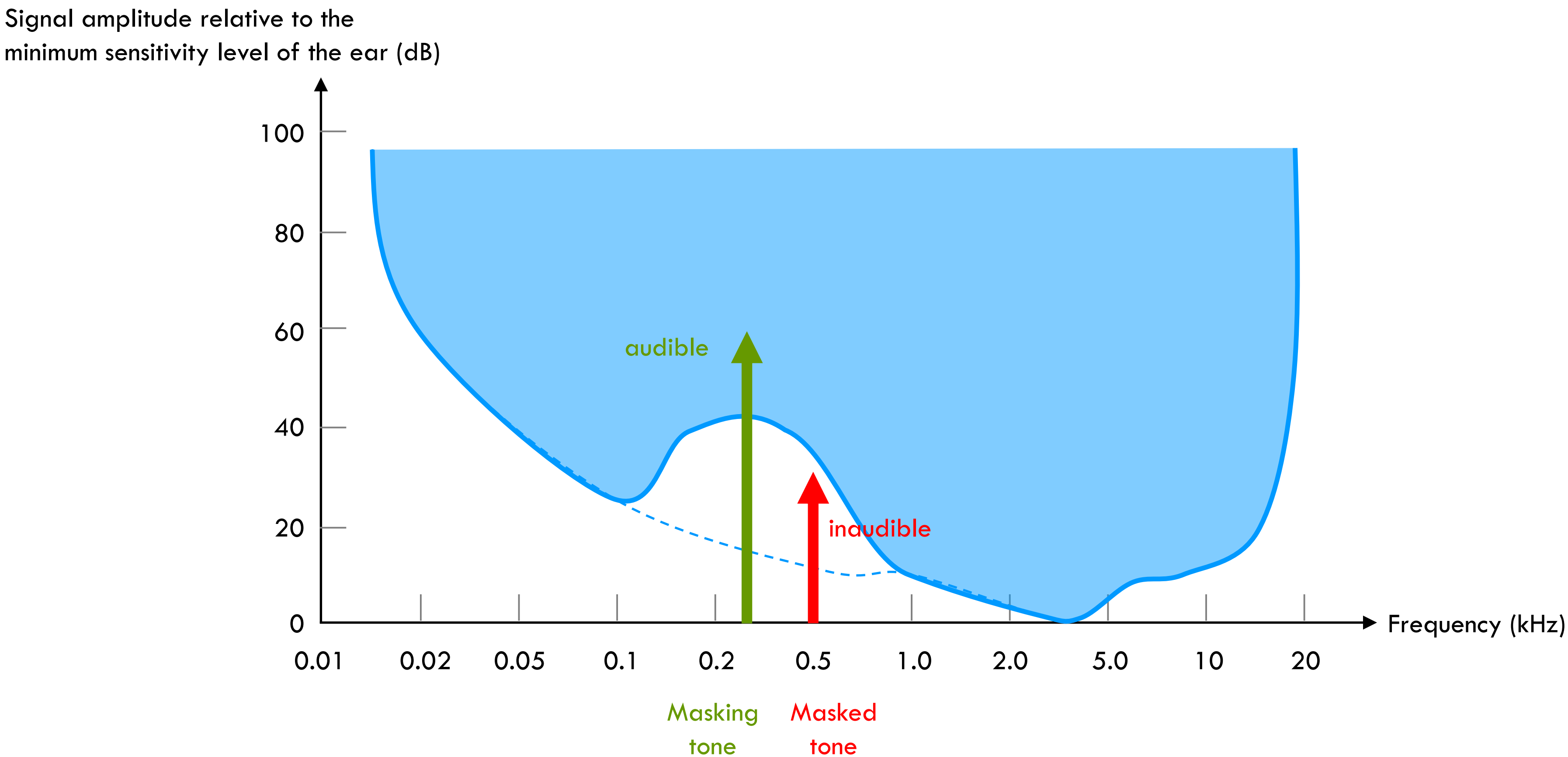
Physiological Implications

- Curves indicate perceived loudness as a function of both the frequency and the level
  - Each contour of the curves expresses how much a sound level must be changed as the frequency varies, to maintain an equal perceived loudness
- The curves accentuated at frequency range to coincide with speech
  - The ability to hear sounds of the accentuated range is thus vital for speech communication

Frequency Masking

- When an audio signal consists of multiple frequencies
  - The sensitivity of the ear changes with the relative amplitude of the signals
- If the frequencies are close and the amplitude of one is less than the other close frequency
  - The weaker frequency may not be heard
  - The phenomenon is known as <u>frequency masking</u>
- Why?
  - The stereocilia are excited by air pressure variations
  - Different stereocilia respond to different ranges of frequencies
  - After excitation by one frequency, further excitation by a less strong similar frequency of the same group of cells is not possible

Frequency Masking

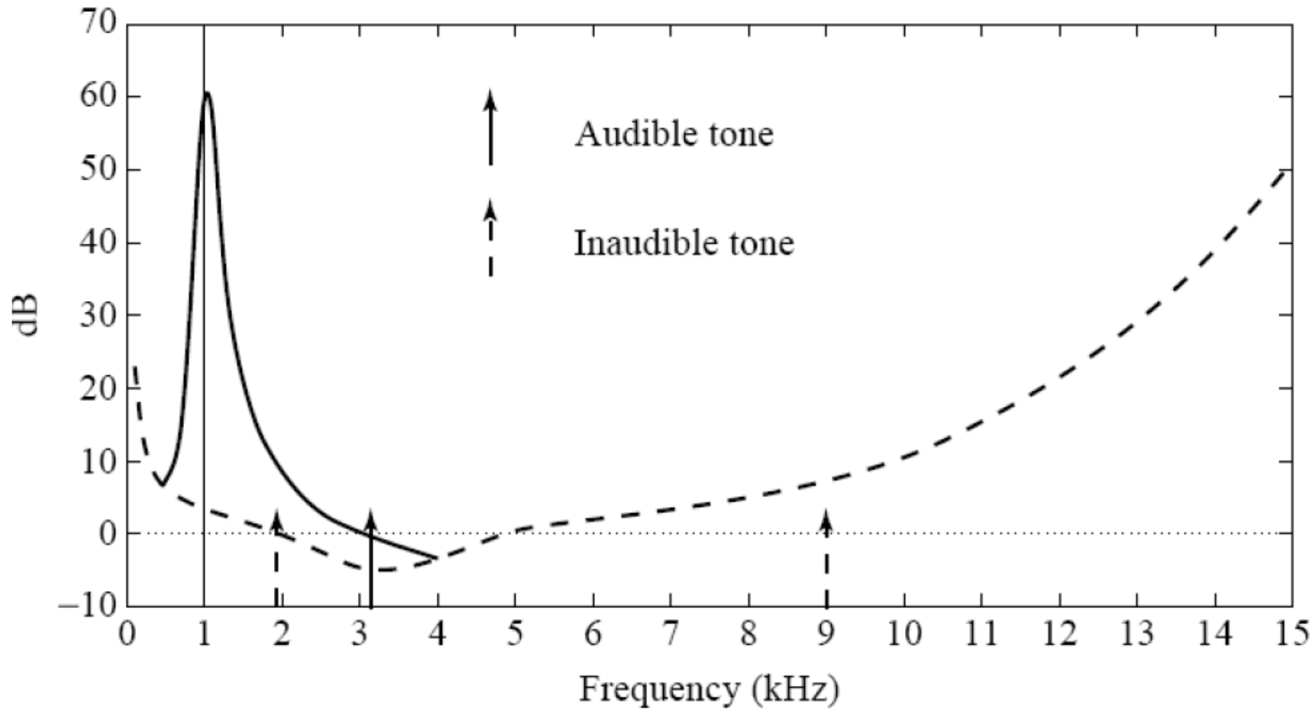


Frequency Masking

- Remarks:
  - A lower tone can effectively mask a higher tone played simultaneously
  - But a higher tone does not mask a lower tone that well
  - The greater the power in the masking tone, the wider is its influence, the broader the range of frequencies it can mask
  - If two tones are widely separated in frequency then little masking occurs

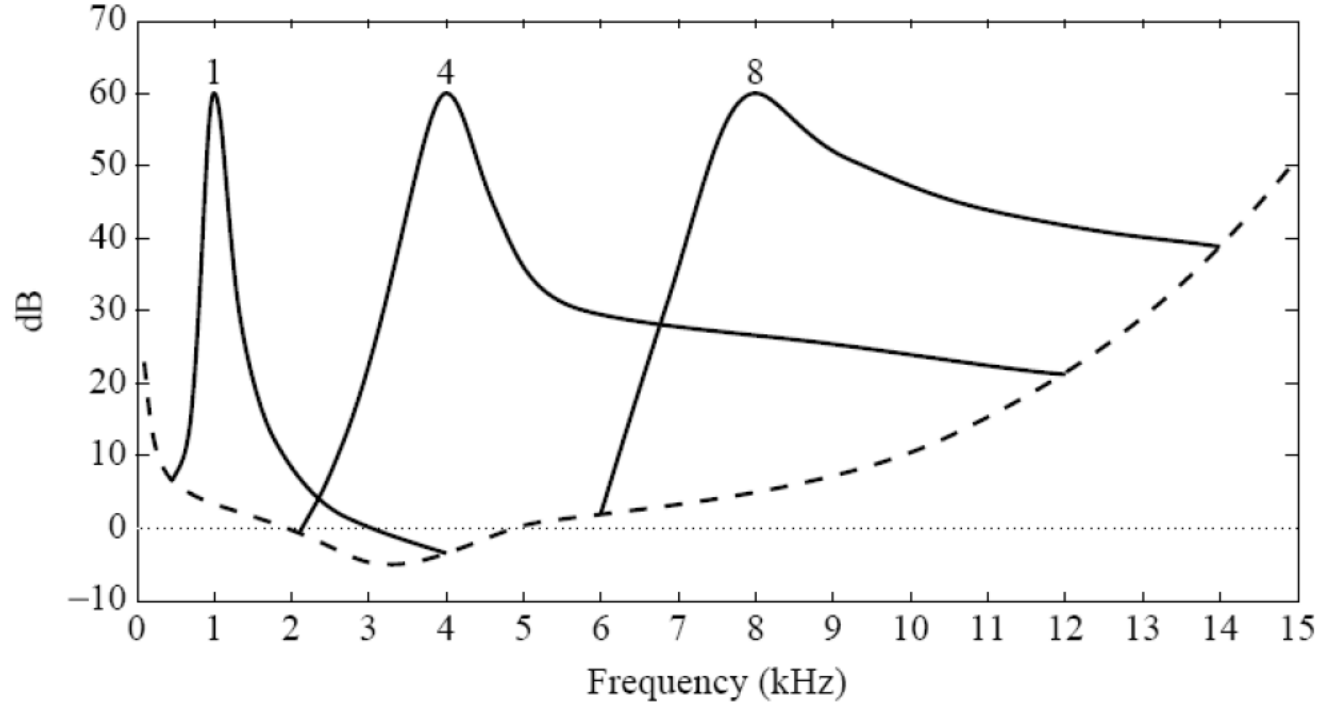
Frequency Masking

- Frequency masking due to 1 kHz signal:



Frequency Masking

- Frequency masking due to 1, 4, 8 kHz signals:

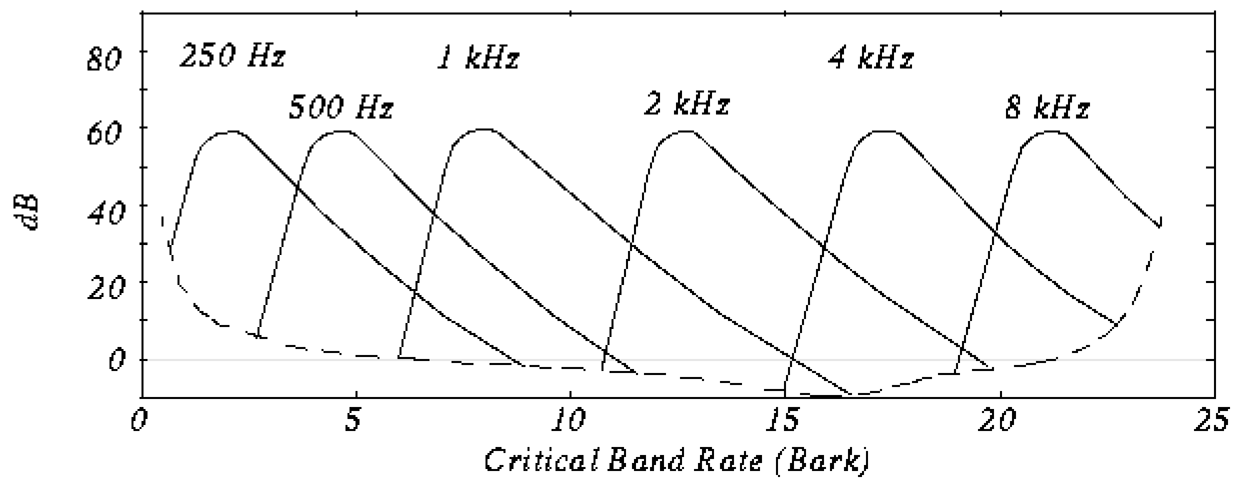


Critical Bands

- Human auditory system has a limited, frequency-dependent resolution
  - The perceptually uniform measure of frequency can be expressed in terms of the width of the Critical Bands
  - It is less than 100 Hz at the lowest audible frequencies, and more than 4 kHz at the high end
  - Altogether, the audio frequency range can be partitioned into 25 critical bands
- The number associated with a critical band is bark
  - For frequency < 500 Hz, it is $f/100$
  - For frequency > 500 Hz, it is $9+4\log_2(f/1000)$

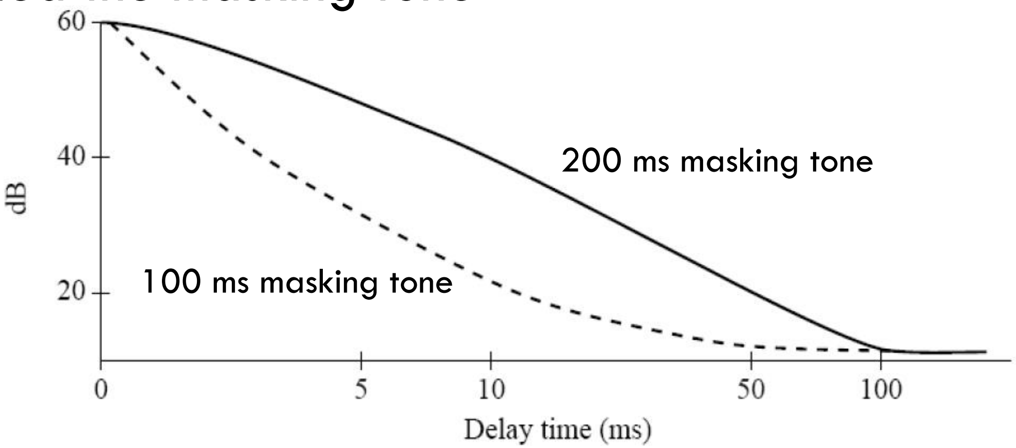
Critical Bands

- Frequency masking on critical band scale



Temporal Masking

- After the ear hears a loud tone, it takes a further short while before it can hear a quiet tone close in frequency
  - The longer the loud tone being played, the longer it takes for the quiet tone to be heard
  - The phenomenon is known as <u>temporal masking</u>
- Experiment: Play a 1 kHz (A) tone at 60 dB and a 1.1 kHz (B) tone at 40 dB
  - B tone cannot be heard (B tone is being masked)
  - A is called the masking tone



Temporal Masking

- Why a stronger tone can reduce the ear's sensitivity to a follow-on weak tone?
  - The stereocilia vibrate with corresponding force of input sound stimuli
  - If the stimuli is strong, the stereocilia will be in a high state of excitation and get fatigued and require time to recover
  - Prolonged exposure of loud tones would permanently damage the stereocilia (called deafness)

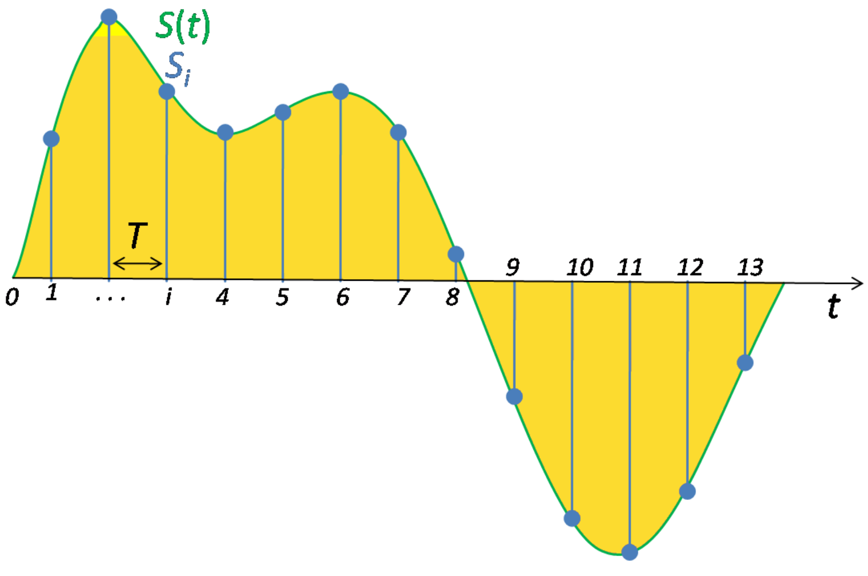
Psychoacoustics

- How to exploit psychoacoustics to remove the redundancy?
  - Whenever the presence of a strong audio signal makes a temporal or spectral neighborhood of weaker audio signals imperceptible
  - Remove the acoustically irrelevant parts of audio signals

Digital Audio Representation

Digital Sampling

- Sampling process basically involves
  - Measuring the analog signal at regular discrete intervals
  - Recording the value at these points

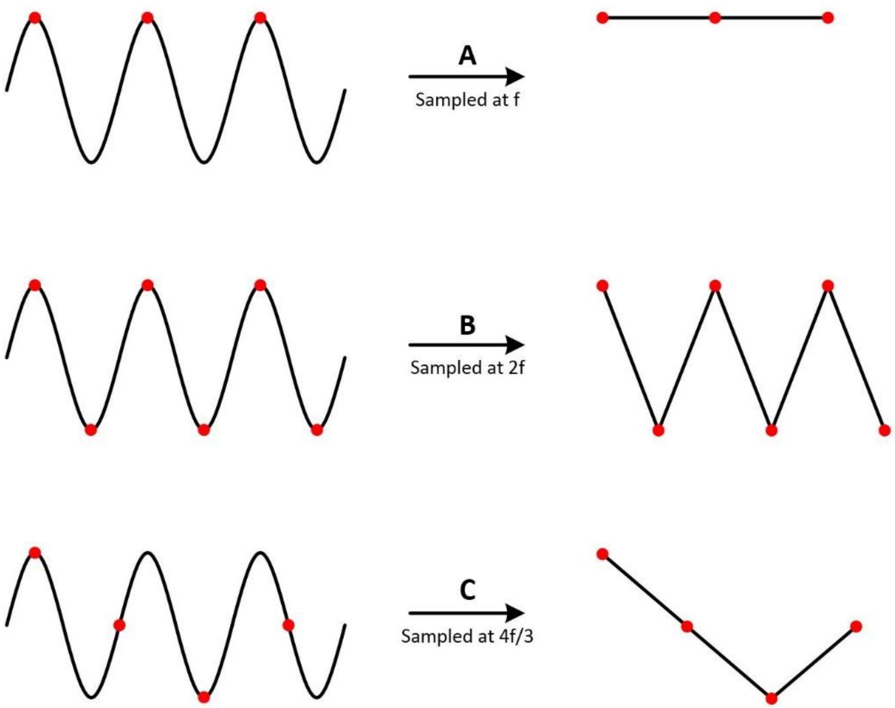


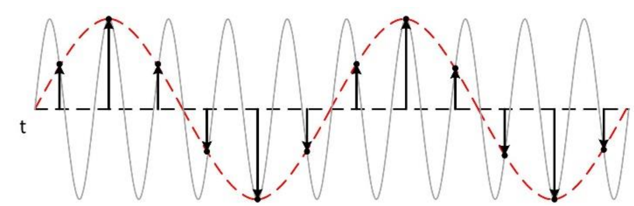
Sampling Rate

- How many samples to take?
  - Telephone: 8 kHz
  - CD Quality: 44.1 kHz
- Why 44.1 kHz?
  - Upper range of human hearing is around 20 to 22 kHz
  - Apply Nyquist's Sampling Theorem
    - The sampling frequency for a signal must be greater than twice the highest frequency component in the signal
    - For the accurate reproduction of a digital version of an analog waveform

Sampling Rate

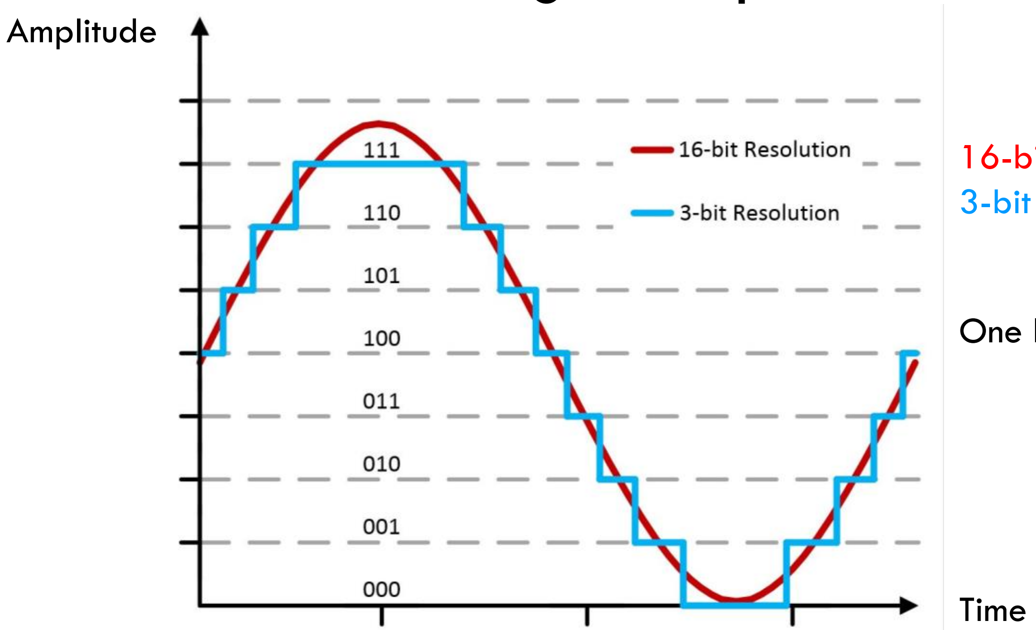
- If the sampling rate is not greater than the Nyquist rate
  - Artefacts arise: the effect is known as aliasing





Sampling Size

- How many bits is used for each sample value?
  - Called quantization
  - Each sample is quantized to the nearest value within a range of quantization steps



16-bit quantization: 16 bits per sample (65536 values)
3-bit quantization: 3 bits per sample (8 values)

One less bit of quantization introduces ~6 dB of noise

Raw Size of Digital Audio

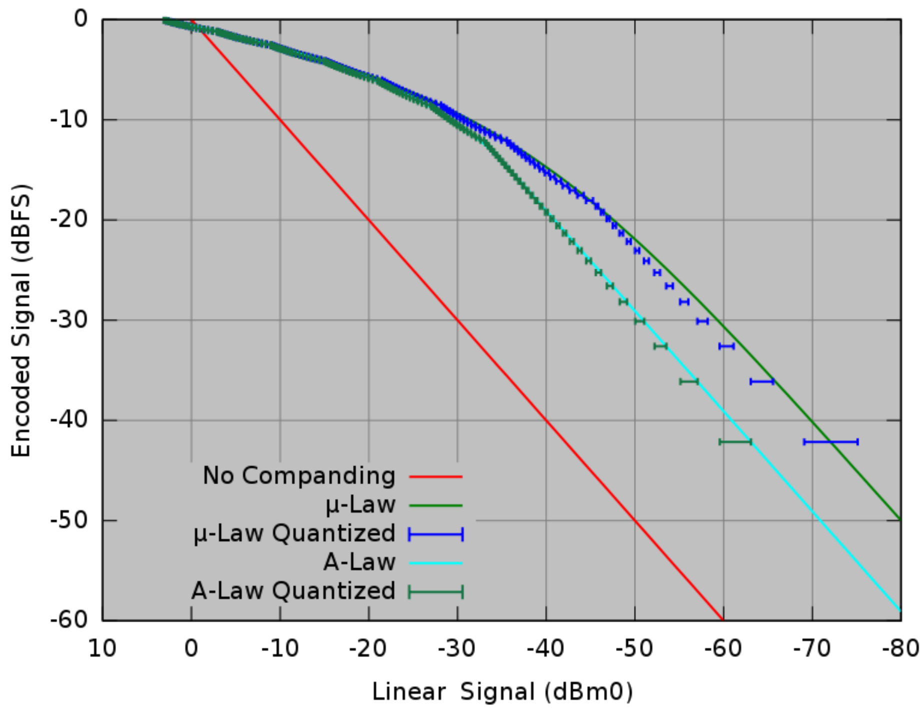
- Quantization size does not only affect the quality of audio, but also the size of audio
- Raw bitrate of telephone quality audio
  - 8-bit mono@8 kHz: 8x8 kHz = 64 kbps
- Raw bitrate of CD quality digital audio
  - 8-bit mono@44.1 kHz: 8x44.1 kHz = 352 kbps
  - 16-bit mono@44.1 kHz: 16x44.1 kHz = 705 kbps
  - 16-bit stereo@44.1 kHz: 2x16x44.1 kHz = 1.411 Mbps
  - Mono: single channel; Stereo: two channels

Linear/Non-Linear Quantization

- If the quantization levels are linearly uniform
  - Called linear pulse code modulation (LPCM)
- However, ears do not respond to sound in a linear fashion
  - Better to use non-linear quantization levels
  - In telephony, US and Japan: μ-law PCM
  - Europe: A-law PCM

μ-Law and A-Law Quantization

- Idea: map a 13- or 14-bit linear sampled input (x) to a 8-bit value (y)



μ-law encoding

$$y = F(x) = sgn(x)V\frac{log\left(1+\frac{\mu|x|}{V}\right)}{\log(1+\mu)}$$

μ-law expansion

$$x = F^{-1}(y)$$
$$= sgn(y)\frac{V}{\mu}\left(e^{\frac{|y|\log(1+\mu)}{V}} - 1\right)$$

The μ-law is used in the G.711 (Pulse Code Modulation of voice frequencies)

The μ-law algorithm provides a slightly larger dynamic range than the A-law at the cost of worse proportional distortion for small signals

Various PCM Representations

- Silence Compression
  - Detect the silence and use run-length encoding (RLE) to compress the silence
  - Similar to image, detect the background color and use RLE to compress the background color
- Differential Pulse Code Modulation (DPCM)
  - The difference in amplitude in successive samples is small
  - Based on the previous sample, predict the next sample and encode the residual between the actual value and the predicted value
  - Require fewer bits than storing the actual value

Various PCM Representations

- Adaptive DPCM (ADPCM)
  - A refinement on DPCM
  - If the change between successive sample is rapid, use large quantization steps
  - Otherwise if the change is slow, use small quantization
  - Map a series of 8-bit μ-law PCM samples into a series of 4-bit ADPCM samples
  - Half the sampling size of G.711
  - Used in G.723, G.726 standards, Voice over IP (VoIP) applications
  - Used in Apple's ACE (Audio Compression/Expansion) proprietary format (achieved 2:1 compression)

MPEG Audio

Motion Pictures Experts Group

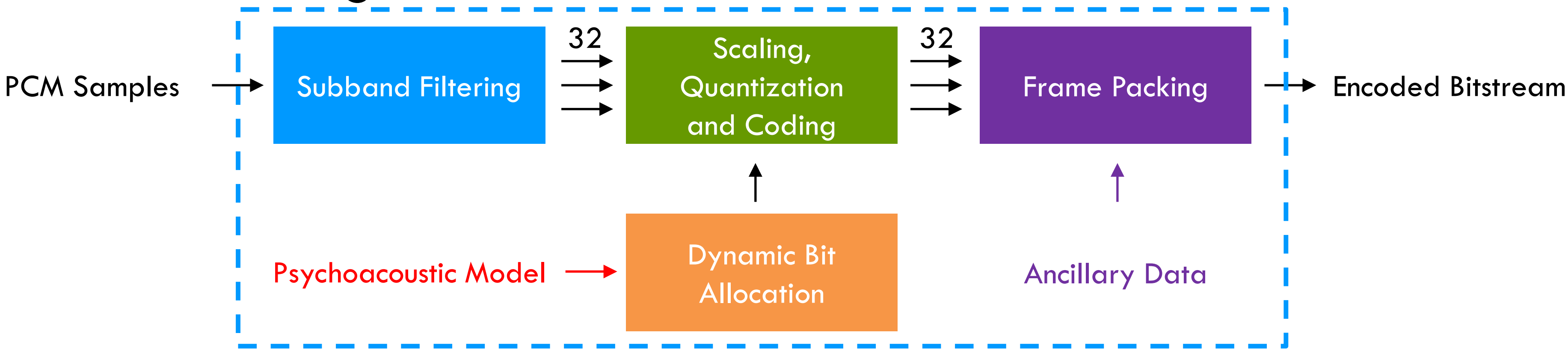
- The Motion Pictures Experts Group (MPEG)
  - A working group that works on standards for video systems
  - Form from the existing Joint Photographic Experts Group (JPEG)
  - Since motion pictures are often accompanied by sound, MPEG also defined a standard for encoding audio information
- For example, a loud orchestra easily masks the sounds of some individual instruments playing softly
  - The masked instruments will not be audible to the listeners
  - MPEG audio drops the inaudible areas of data

MPEG-1 Audio

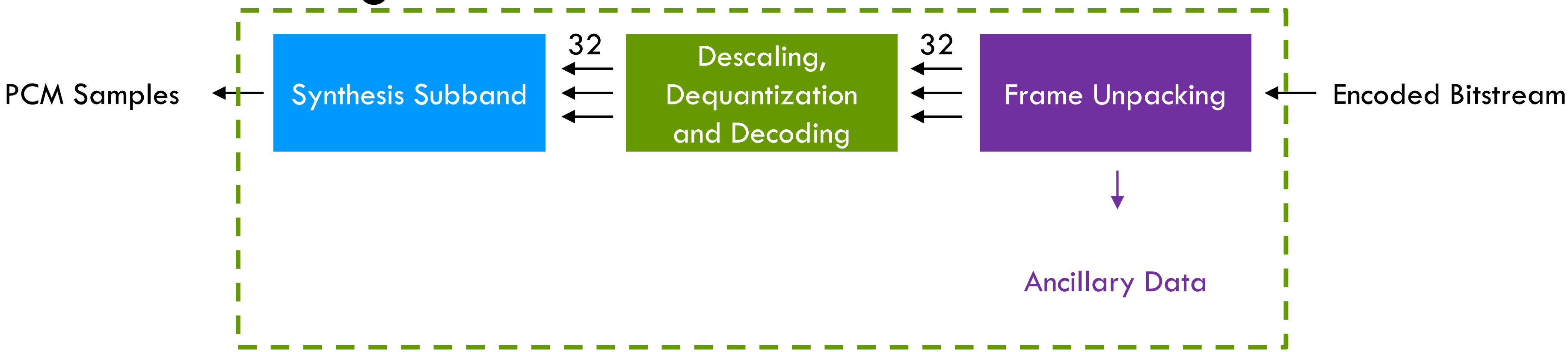
- In 1993, the MPEG-1 standard for audio and video encoding was published as ISO/IEC 11172
- The standard consists of 3 parts, in which
  - Part 1: ISO/IEC 11172-1 (known as MPEG-1 Systems)
  - Part 2: ISO/IEC 11172-2 (known as MPEG-1 Video)
  - Part 3: ISO/IEC 11172-3 (known as MPEG-1 Audio)
  - Part 4 and 5 were added in 1995 and 1998 respectively
- MPEG-1: 1.5 Mbps for audio and video
  - 1.2 Mbps for video
  - 0.3 Mbps for audio
    - Supports sampling frequencies of 32, 44.1 and 48 kHz

MPEG-1 Audio Encoding/Decoding

- Encoding



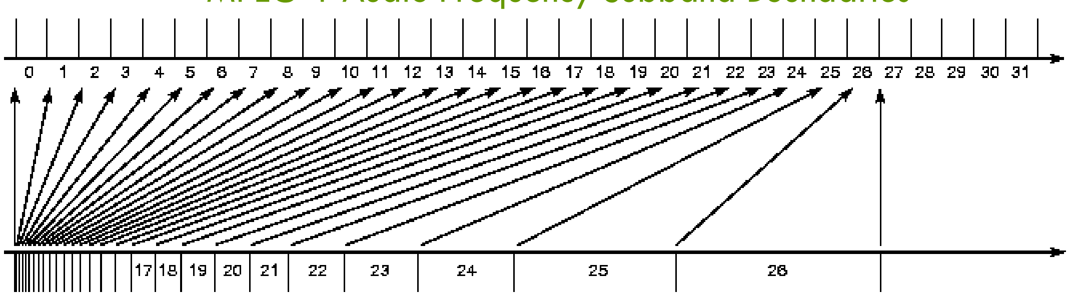
- Decoding



MPEG-1 Encoding

- The audio signal is first sampled and quantized using PCM
  - The sampling rate and quantization levels are application dependent
  - The PCM samples are then divided up into 32 equal-width frequency subband which approximate the 32 critical bands

MPEG-1 Audio Frequency Subband Boundaries



Critical Band Boundaries

MPEG-1 Encoding

- The maximum amplitude of 12 subband samples in each subband is then determined
  - Called the scaling factor of the subband
  - The subband scaling factors is then passed to psycho-acoustic modeler and quantizer blocks
- Example:

Band	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	..	32
Level (dB)	0	8	12	10	6	2	10	60	35	20	15	2	3	5	3	1	..	0

MPEG-1 Encoding

- Psychoacoustic Modeler
  - Employ frequency masking
  - Determine the amount of masking for each band caused by nearby bands
  - If the power in a band is below the masking threshold, do not encode it
  - Otherwise determine the no. of bits (from scaling factors) needed to represent the coefficient such that noise introduced by quantization is below the masking effect
  - One less bit of quantization introduces ~6 dB of noise

MPEG-1 Encoding

- Example

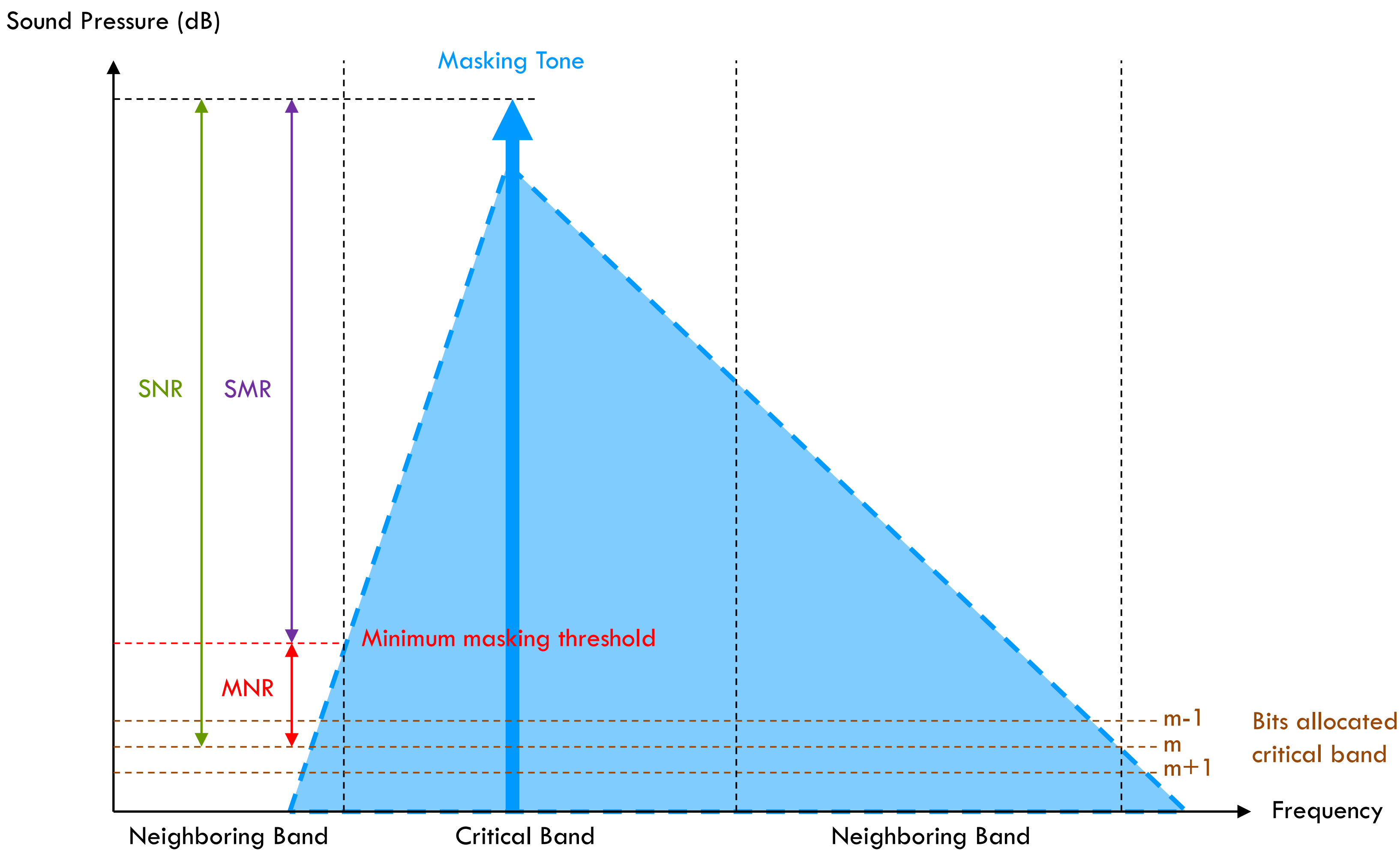
Band	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	..	32
Level (dB)	0	8	12	10	6	2	10	60	35	20	15	2	3	5	3	1	..	0

  - If the level of the 8th band is 60 dB
  - According to the Psycho-acoustic model, it gives a masking of 12 dB in the 7th band, 15 dB in the 9th band
  - Level in 7th band is 10 dB (< 12 dB), so ignore it
  - Level in 9th band is 34 dB (>= 15 dB), so send it

MPEG-1 Bit Allocation

- Bit allocation is a process that determines the no. of code bits for each subband so as to minimize the audibility of noise
- The mask-to-noise is defined as
  - $\mathrm{MNR_{dB}} = \mathrm{SNR_{dB}} - \mathrm{SMR_{dB}}$
  - where $\mathrm{MNR_{dB}}$ is the mask-to-noise ratio,
  - $\mathrm{SNR_{dB}}$ is the signal-to-noise ratio, given in the MPEG audio standard as a lookup table
  - $\mathrm{SMR_{dB}}$ is the signal-to-mask ratio from the psycho-acoustic model

MPEG-1 Bit Allocation



MPEG-1 Bit Allocation

- Bit allocation algorithm
  - For each subband, use the psycho-acoustic model to calculate the $\mathrm{SMR}_{\mathrm{dB}}$
  - Then the $\mathrm{MNR}_{\mathrm{dB}}$ is computed for all subbands
  - While there are still code bits for allocation
    - The subband with the lowest $\mathrm{MNR}_{\mathrm{dB}}$ is determined
    - The no. of code bits allocated to this subband is incremented
    - Then a new estimate of the $\mathrm{SNR}_{\mathrm{dB}}$ is made

MPEG-1 Encoding

- The output bitstream of MPEG-1

Header	Side	Subband Samples	Ancillary Data

- Header
  - Contain information such as the sampling frequency and quantization
- Side
  - Bit allocation, quantized scaling factors
- Subband Samples
  - 12 samples in each subband
  - (36 samples in each subband for layer II and III)
- Ancillary Data [Optional]
  - To carry additional code samples associated with special broadcast format, cyclic redundancy code for error checking, etc

MPEG-1 Decoding

- Dequantize the subband samples after demultiplexing the coded bitstream into subbands
- Synthesis bank decodes the dequantized subband samples to produce PCM stream
- This involves inverse Fast Fourier transform (IFFT) on each substream and multiplexing the channels to give the PCM bit stream

MPEG-1 Layers

- MPEG-1 defines 3 layers of processing layers for audio
  - Layer I: exploits frequency masking
  - Layer II: exploits temporal masking
  - Layer III: exploits stereo redundancy

  - MPEG-1 layer I is commonly known as MP1
  - MPEG-1 layer II is commonly known as MP2
  - MPEG-1 layer III is commonly known as MP3

MPEG-1 Layer II

- Layer I: each frame contains 384 samples
  - 12 samples x 32 subbands per frame
- Layer II: group 3 frames together
  - Before, current and next: a total of 1152 samples
  - Model the effect of temporal masking
  - Allow more compact coding of scale factors and quantized samples
  - Better audio quality as the bits saved by temporal masking could be assigned to quantized subband values
- Layer I: uses a 512-point FFT for the subband filtering
  - Layer II: uses a 1024-point FFT

MPEG-1 Layer III

- Layer I & II use equal-width frequency subband filter to approximate the critical bands
  - Layer III uses non-equal frequency subband filter (more accurate)
- Layer I & II use FFT for subband filtering
  - Layer III uses hybrid subband filtering
  - Both FFT and modified discrete cosine transform (MDCT)
- Layer I: block length of 12 samples
  - Layer III: block length of 18 (long) or 6 (short) samples
  - Long for better frequency resolution and short for better temporal resolution
  - Exploit temporal masking by 50% overlap between successive transform windows which gives window sizes of 36 or 12

MPEG-1 Layer III

- Layer I & II: linear quantization
  - Layer III: non-linear quantization
- Layer I & II use bit allocation algorithm
  - Layer III uses Huffman coding on quantized samples for better result

MPEG-1 Audio

- MPEG-1 Audio supports one or two audio channels in one of the four modes
  - 1: Monophonic: single audio channel
  - 2: Dual-monophonic: two independent channels
  - 3: Stereo: two non-independent channels that share bits
  - 4: Joint-Stereo: use joint-stereo coding (takes advantages of the correlations between stereo channels)

MPEG-1 Layer III

- Layer I and II use Intensity Stereo Coding
  - At upper-frequency subbands (> 2 kHz), encode summed signals instead of independent signals from left and right channels
  - Assign independent left and right scalefactors (directional information)
  - Since human auditory system is insensitive to the signal phase at frequencies above approximately 2 kHz
  - A lossy coding method, primarily useful at low bitrates
- Layer III introduces Middle/Side (MS) Stereo Coding
  - Middle: sum of left and right channels ($L + R$)
  - Side: difference of left and right channels ($L - R$)
  - Encoder uses specially tuned threshold values to compress the side channel signal further
  - Useful for high bitrates

MPEG-1 Audio Summary

- MPEG audio compression basically works by:
  - Dividing the audio signal up into a set of 32 frequency subbands by Fourier Transform
  - Subbands approximate critical bands of human hearing
  - Each band quantized according to the 'audibility' of quantization noise
  - Exploit Frequency Masking: near frequencies not heard in same time frame
  - Exploit Temporal Masking: near frequencies not heard close to some short time frame between frequencies
- The key reasons why MPEG Audio is lossy
  - Quantization, frequency masking (and temporal masking)

MPEG-1 Audio Summary

- Compression rate
  - Layer I: 25% (about 4:1)
  - Layer II: 16%-12% (about 6:1 to 8:1)
  - Layer III: 10%-8% (about 10:1 to 12:1)
- Audio bitrate (size) for perceptually lossless quality
  - Raw: ~1.4 Mbps (~31 MB, for 44.1 kHz, 16-bit, stereo, 3 mins)
  - Layer I: ~384 kbps (~8 MB)
  - Layer II: ~192 kbps (~4 MB)
  - Layer III: ~128 kbps (~2.8 MB)
- The higher the layer,
  - The greater the compression ratios
  - The greater the computational complexity
  - Backward compatible to lower layers

MPEG-2 Audio

MPEG-2 Audio BC

- MPEG-2 Part 3 was published in 1995
  - ISO/IEC 13818-3 (commonly known as MPEG-2 Audio BC)
  - Backward compatible with MPEG-1 standard
  - MPEG-2 Layers I, II, III are similar to those of MPEG-1
  - MP3 is now referred to both MPEG-1 Audio Layer III and MPEG-2 Audio Layer III
- MPEG-2 Audio's additional features
  - Capable of using lower sampling frequencies (16 kHz, 22.05 kHz and 24 kHz) to support wideband speech to medium band audio
  - Support up to 5.1 multichannel coding
    - 2/0 (L, R), 3/0 (L, C, R), 3/1 surround sound (L, C, R, S)
    - 3/2 (L, C, R, Ls, Rs), 3/2 with woofer 5.1 channels (L, C, R, Ls, Rs, LFE)

MPEG-2 Audio (AAC)

- MPEG-2 Part 7 was published in 1997
  - ISO/IEC 13818-7
  - Known as MPEG-2 Advanced Audio Coding (AAC)
  - Same framework as MPEG-1, with some enhancements
  - Not backward compatible with MPEG-1 standard
  - Default audio format for YouTube, Apple iDevices, etc

AAC vs. MP3

- Sampling frequency
  - AAC supports sampling rate of 8 to 96 kHz
  - MP3 supports 16 to 24 kHz (MPEG-2 Layer III) and 32 to 48 kHz (MPEG-1 Layer III)
- Channels
  - AAC supports up to 48 channels, 16 low frequency effects (120 Hz) (e.g. woofer), 16 data streams
  - MP3 supports up to 2 channels (MPEG-1 Layer III) and 5.1 channels (MPEG-2 Layer III)

Audio Synthesis

Audio Synthesis

- Instead of recording the sound samples
  - Encoder records the way of making the sound
  - Using high-level descriptions to represent signals and control the signals, e.g.
    - Which notes to play
    - How loud to play them
    - What tempo to play them at
    - How long do they last
    - When to fade in/out them, etc
  - Much greater reduction than directly encoding the audio
- Decoder (called synthesizer) reproduces the sound (called synthesis) by following the parameters described by the encoder

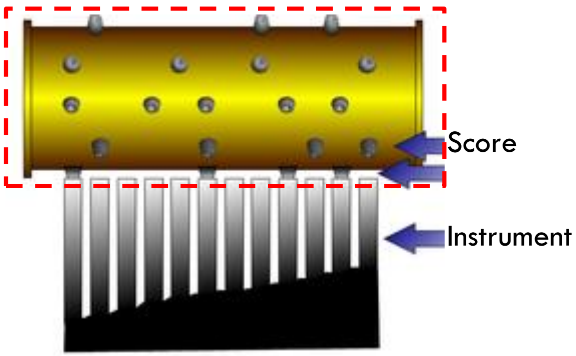
High Level Representation of Audio

Sheet/Score

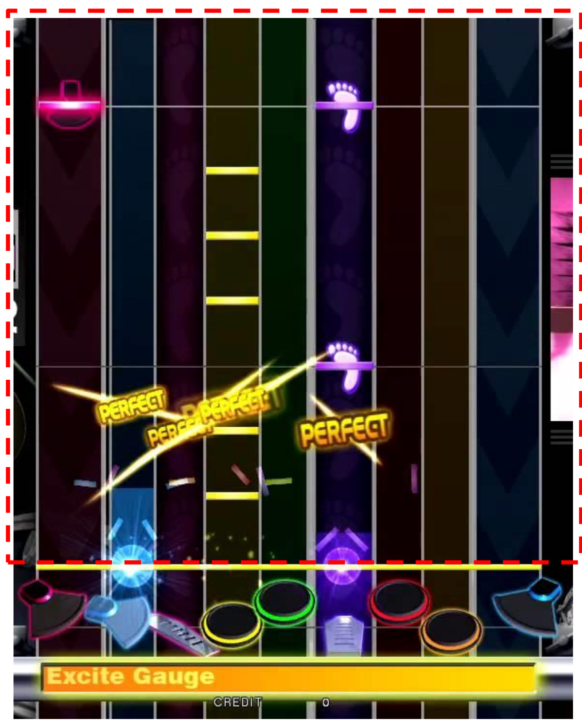
- Indicate pitches, rhythms, volume, tempo, key, etc of a music
- The medium is not necessary a paper

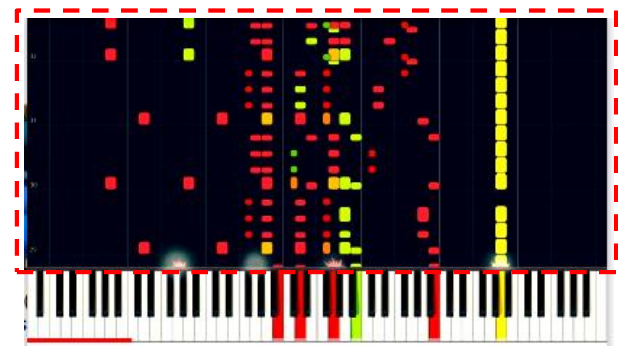
High Level Representation of Audio

Other forms of score



Music Box





Audio Synthesis

- Audio synthesizer
  - Hardware: MIDI devices (e.g. keyboards), PC soundcard, etc
  - Software: synthesizer software, browser, etc
- Common audio synthesis standards
  - Musical Instrument Digital Interface (MIDI)
  - MPEG-4 Structured Audio
  - Web Audio API

Synthetic Audio vs. Recorded Audio

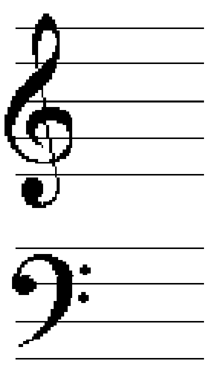
	Synthetic Audio	Recorded Audio
Mechanism	Use high level descriptions to describe how the sound is made	Use regular sampling, then compress the sound samples
Bandwidth	Generally very low data rate	Generally very high data rate
Edit	Easy to edit	Difficult to modify
Sound Reproduction	Decoder requires domain information (e.g. wavetable) to reproduce the signal (but no control on the final quality of the reproduced audio)	Do not require domain information to reproduce the signal
Target Applications	Instrumental Music, Spoken Word Audio (e.g. text to speech)	Nature sound, Non-instrumental Music

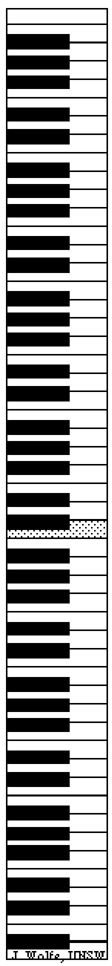
Musical Instrument Digital Interface (MIDI)

MIDI

- Standardized in 1983 by MIDI Manufacturers Association (MMA)
- A protocol and language to represent the parameters of the music notes being played
- Audio file size comparison
  - Typical size of a MIDI file: a few kB
  - Typical size of a MP3 file: a few to tens of MB
  - Typical size of raw audio: tens to hundreds of MB

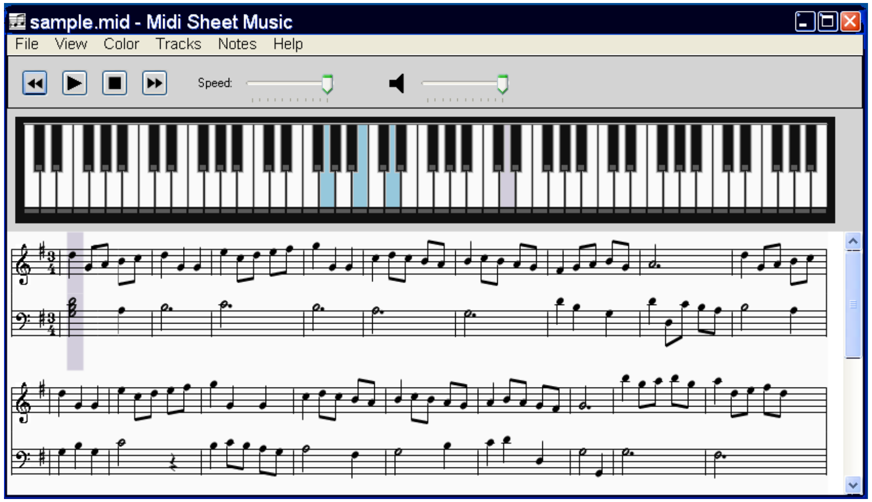
MIDI





Frequency	Note name	MIDI number
4186.0	C8	108
3951.1	B7	107
3729.3		106
3520.0	A7	105
3322.4		104
3136.0	G7	103
2960.0		102
2793.8	F7	101
2637.0	E7	100
2489.0		99
2349.3	D7	98
2217.5		97
2093.0	C7	96
1975.5	B6	95
1864.7		94
1760.0	A6	93
1661.2		92
1568.0	G6	91
1480.0		90
1396.9	F6	89
1318.5	E6	88
1244.5		87
1174.7	D6	86
1108.7		85
1046.5	C6	84
987.77	B5	83
932.33		82
880.00	A5	81
830.61		80
783.99	G5	79
739.99		78
698.46	F5	77
659.26	E5	76
622.25		75
587.33	D5	74
554.37		73
523.25	C5	72
493.88	B4	71
466.16		70
440.0	**A4**	**69**
415.30		68
392.00	G4	67
369.99		66
349.23	F4	65
329.63	E4	64
311.13		63
293.67	D4	62
277.18		61
261.6	**C4**	**60**
246.94	B3	59
233.08		58
220.00	A3	57
207.65		56
196.00	G3	55
185.00		54
174.61	F3	53
164.81	E3	52
155.56		51
146.83	D3	50
138.59		49
130.81	C3	48
123.47	B2	47
116.54		46
110.00	A2	45
103.83		44
97.999	G2	43
92.499		42
87.307	F2	41
82.407	E2	40
77.782		39
73.416	D2	38
69.296		37
65.406	C2	36
61.735	B1	35
58.270		34
55.000	A1	33
51.913		32
48.999	G1	31
46.249		30
43.654	F1	29
41.203	E1	28
38.891		27
36.708	D1	26
34.648		25
32.703	C1	24
30.868	B0	23
29.135		22
27.500	A0	21

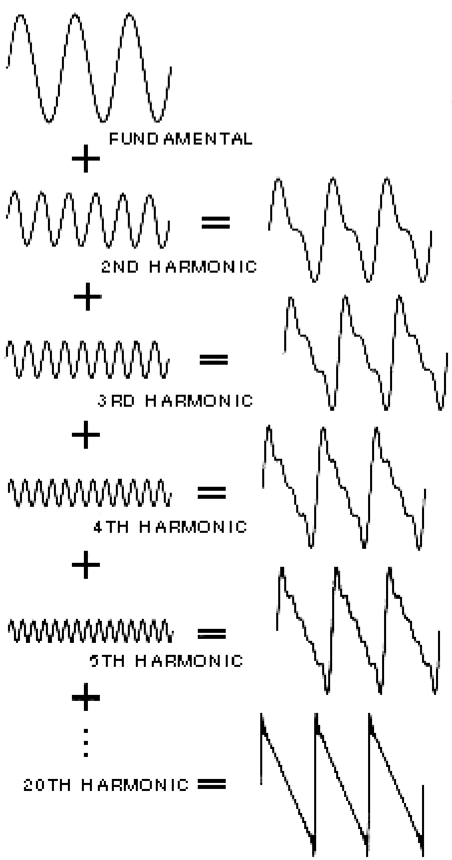
pppp	= 8
ppp	= 20
pp	= 31
p	= 42
mp	= 53
mf	= 64
f	= 80
ff	= 96
fff	= 112
ffff	= 127

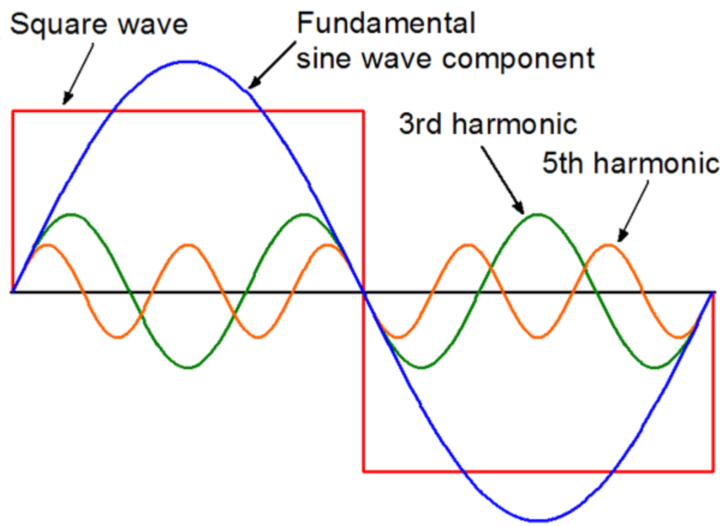


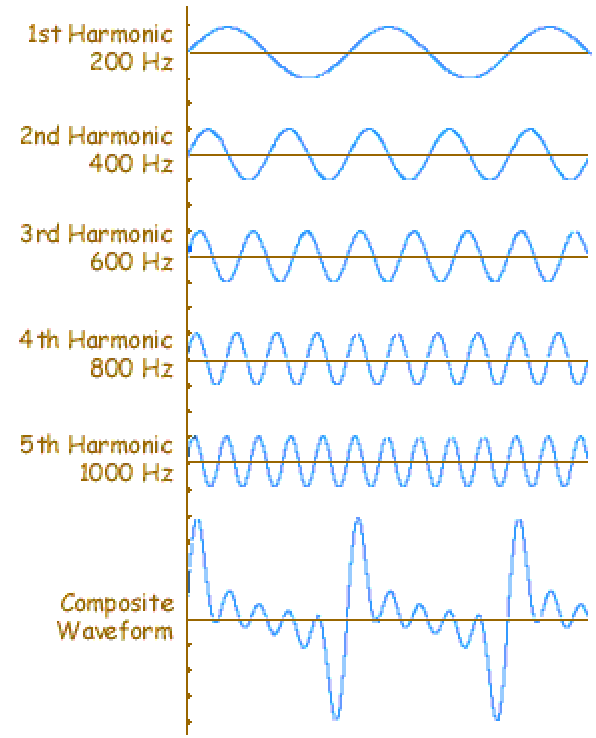
Music Fundamentals

- Human perception of sound is not linear, but logarithmic
  - Equally spaced sound frequencies (e.g. 200 Hz, 400 Hz, 600 Hz, 800 Hz) do not sound as if they are spaced equally
  - But 220 Hz, 440 Hz, 880 Hz do sound as if spaced equally
  - They are respectively the frequencies of notes A3, A4, A5

Harmonic


FUNDAMENTAL
2ND HARMONIC
3RD HARMONIC
4TH HARMONIC
5TH HARMONIC
20TH HARMONIC


Square wave
Fundamental sine wave component
3rd harmonic
5th harmonic

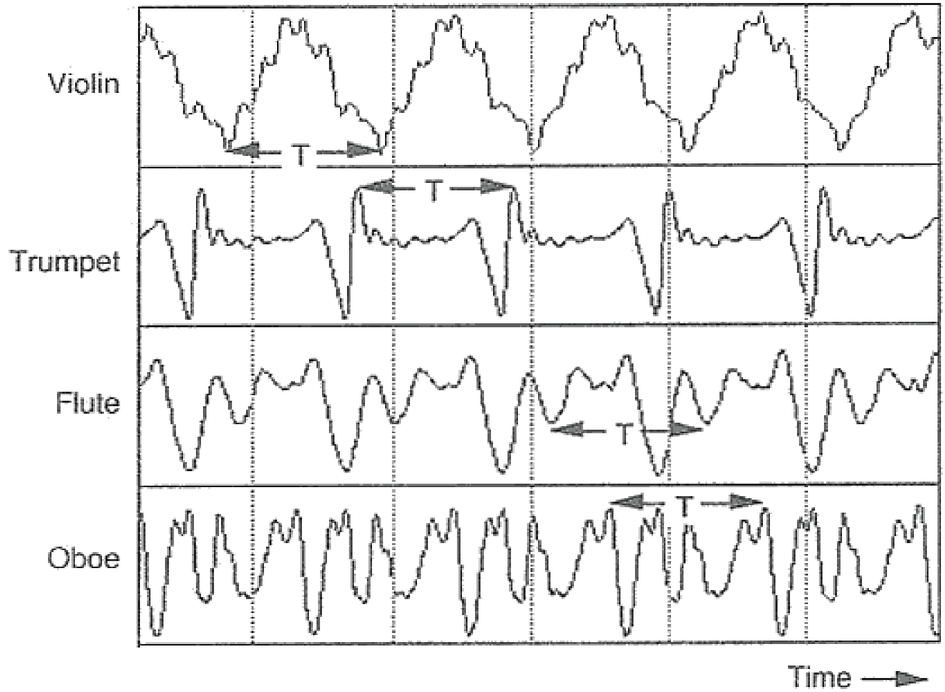

1st Harmonic 200 Hz
2nd Harmonic 400 Hz
3rd Harmonic 600 Hz
4th Harmonic 800 Hz
5th Harmonic 1000 Hz
Composite Waveform

Mathematics of Music

- Octave
  - Note A6 is the next octave of A5
  - It can be obtained by doubling A5's frequency
- Semitone
  - There are 12 semitones in between an octave
  - The notes in between an octave is also not linearly spaced, but logarithmically spaced by $2^{\frac{1}{12}}$

Musical Instruments

- They sound differently even if the musical instruments play the same note (e.g. A4: 440Hz)
- Why? Their waveforms are different



MIDI Messages

- 2 types of MIDI messages
- Channel Messages
  - Carry channel specific information
  - 2 subtypes
    - Voice Messages
    - Mode Messages
- System Messages
  - Carry non-channel specific information
  - 3 subtypes
    - Common Messages
    - Real-time Messages
    - Exclusive Messages

MIDI Messages

- Channel Voice Messages
  - Note On
    - Define which key is pressed, and on which channel (instrument)
  - Note Off
    - Similar to note on command to release a pressed key
  - Other Commands
    - Alter the sound of the currently active note or notes
- Channel Mode Messages
  - Determine how an instrument will process the channel voice messages

MIDI Messages

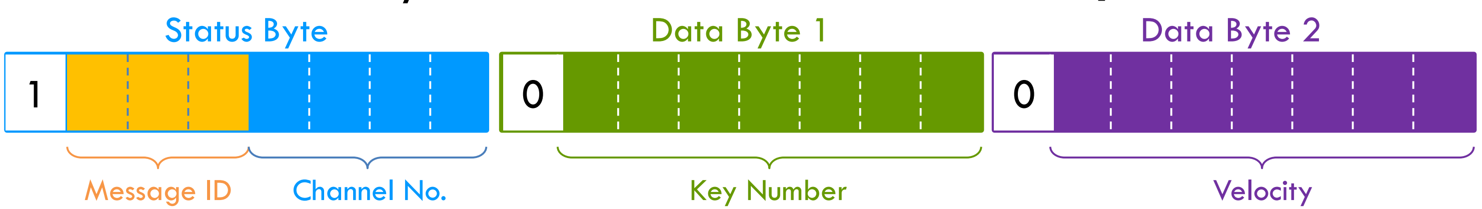
- System Common Messages
  - Used for positioning information in pre-recorded MIDI sequences
  - 1 to 3 bytes long
- System Real-time Messages
  - Used for timing signal for synchronization
  - 1 byte long only
- System Exclusive (Sysex) Messages
  - Messages related to things that cannot be standardized
  - E.g. system dependent creation of sound
  - Bracketed a pair of sysex start byte and sysex end byte
  - The no. of data bytes are system dependent

MIDI Messages

- The structure of MIDI messages
  - MIDI message includes a status byte and up to two data bytes (exception for sysex messages)
  - Status Byte
    - The most significant bit (MSB) of status byte is set (1)
    - The 4 low-order bits identify which channel it belongs to
    - Why 4 bits? At most 16 possible channels
    - The 3 remaining bits identify the message
  - Data Bytes
    - The MSB is cleared (0)

MIDI Messages

- MIDI Note On/Off Command Example



  - To play note 80 with max. velocity 127 on channel 13
  - “Note Off” message id = 000, “Note On” id = 001
  - Channel 13d = 1100b (note: channel is one-indexed)
  - Note 80d = 101 0000b
  - Velocity 127d = 111 1111b
  - The message is 1001 1100 0101 0000 0111 1111
  - (or 9C 50 7F in hex)

Stuck Notes

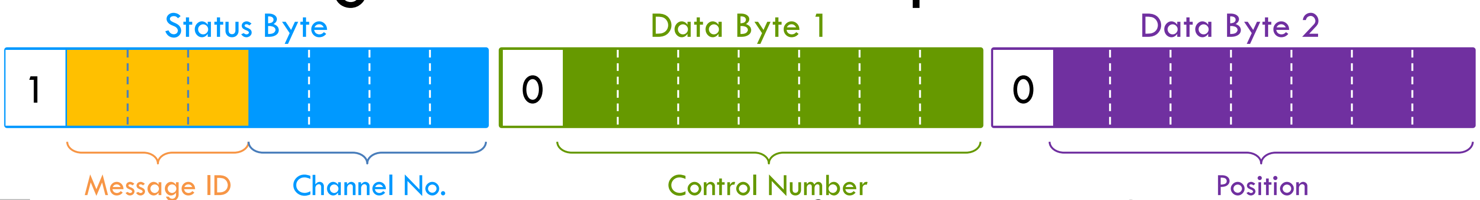
- When a “Note On” message is sent, the note will be played until a corresponding “Note Off” message is received somewhere later
  - What if the corresponding “Note Off” message is missed?
  - The note is still being played even if it sounds as if it has finished
  - This is called stuck note
- When the max. no. of notes being played has been reached, no more notes can be played

Stereo Positioning

- By controlling the relative output level of the stereo speaker, the perceived position of a sound can be controlled
  - If the left speaker outputs the sound louder than the right speaker, the sound will be perceived as coming from the left side
  - If both sound is output at the same volume level, the sound will be perceived to be in the center
- MIDI provides control over the stereo position
  - Called panning

Stereo Positioning

- MIDI Panning Command Example



  - To pan the stereo position of channel 13 to center
  - "Control Change" message id = 011
  - Channel 13d = 1100b (note: channel is one-indexed)
  - Control 10d = panning = 000 1010b
  - Position absolute center = 100 0000b
  - Position absolute left = 000 0000b
  - Position absolute right = 111 1111b
  - The message is 1011 1100 0000 1010 0100 0000

General MIDI (GM)

- MIDI music may not sound the same everywhere
  - Standardize the list of instruments and percussion
  - Forming General MIDI (GM)
- General MIDI
  - On top of MIDI, define an instrument patch map which specifies 16 categories of instruments (in total 128 instruments), and
  - Percussion map which specifies 47 percussion sounds
  - Difference between instrument and percussion: only note on/off, but no pitch information for percussion

GM Instrument Patch Map

- Piano 鋼琴
  - 001 Acoustic Grand 平臺鋼琴
  - 002 Bright Acoustic 亮音鋼琴
  - 003 Electric Grand 平臺電鋼琴
  - 004 Honky-Tonk 叮噹琴
  - 005 Rhodes Piano 電鋼琴一
  - 006 Chorused Piano 電鋼琴二
  - 007 Harpsichord 大鍵琴
  - 008 Clavinet 古鋼琴
- Chrom Percussion 半音階打擊樂器
  - 009 Celesta 鋼片琴
  - 010 Glockenspiel 鐵琴
  - 011 Music box 音樂盒
  - 012 Vibraphone 抖音琴
  - 013 Marimba 立奏木琴
  - 014 Xylophone 柔音木琴
  - 015 Tubular Bells 管鐘
  - 016 Dulcimer 揚琴
- Organ 風琴
  - 017 Hammond Organ 爵士風琴
  - 018 Percussive Organ 敲擊風琴
  - 019 Rock Organ 搖滾風琴
  - 020 Church Organ 教堂風琴
  - 021 Reed Organ 簧風琴
  - 022 Accordion 手風琴
  - 023 Harmonica 口琴
  - 024 Tango Accordion 探戈手風琴
- Guitar 結他
  - 025 Acoustic Guitar (nylon)古典結他
  - 026 Acoustic Guitar (steel) 民謠結他
  - 027 Electric Guitar (jazz) 爵士電結他
  - 028 Electric Guitar (clean) 電結他
  - 029 Electric Guitar (muted) 悶音電結他
  - 030 Overdriven Guitar 濁音電結他
  - 031 Distortion Guitar 變音電結他
  - 032 Guitar Harmonics 合音結他

Note: similar to channel numbers, program numbers are also one-indexed

GM Instrument Patch Map

- Bass 貝司
  - 033 Acoustic Bass 原音貝司
  - 034 Electric Bass (finger) 手彈貝司
  - 035 Electric Bass (pick) 匹克貝司
  - 036 Fretless Bass 無格貝司
  - 037 Slap Bass1 重貝司一
  - 038 Slap Bass2 重貝司二
  - 039 Synth Bass1 合成貝司一
  - 040 Synth Bass2 合成貝司二
- Strings 弦樂器
  - 041 Violin 小提琴
  - 042 Viola 中提琴
  - 043 Cello 大提琴
  - 044 Contrabass 低音提琴
  - 045 Tremelo Strings 顫弓弦樂
  - 046 Pizzicato Strings 彈撥弦樂
  - 047 Orchestral Harp 豎琴
  - 048 Timpani 定音鼓
- Ensemble 合奏
  - 049 String Ensemble1 合奏弦樂一
  - 050 String Ensemble2 合奏弦樂二
  - 051 Synth Strings1 合成弦樂一
  - 052 Synth Strings2 合成弦樂二
  - 053 Choir Aahs 唱詩樂 (啊)
  - 054 Voice Oohs 唱詩樂 (喔)
  - 055 Synth Voice 合成人聲
  - 056 Orchestra Hit 交響打擊樂
- Brass 銅管樂器
  - 057 Trumpet 小號
  - 058 Trombone 伸縮小號
  - 059 Tuba 低音小號
  - 060 Muted Trumpet 悶音小號
  - 061 French Horn 法國號
  - 062 Brass Section 銅管樂
  - 063 Synth Brass1 合成銅管一
  - 064 Synth Brass2 合成銅管二

GM Instrument Patch Map

- Reed 簧樂器
  - 065 Soprano Sax 高音薩克管
  - 066 Alto Sax 中音薩克管
  - 067 Tenor Sax 次中音薩克管
  - 068 Baritone Sax 上低音薩克管
  - 069 Oboe 雙簧管
  - 070 English Horn 英國管
  - 071 Bassoon 低音管
  - 072 Clarinet 單簧管
- Pipe 吹管樂器
  - 073 Piccolo 短笛
  - 074 Flute 長笛
  - 075 Recorder 直笛
  - 076 Pan Flute 排笛
  - 077 Bottle Blow 吹瓶聲
  - 078 Shakuhachi 尺八簫
  - 079 Whistle 笛哨聲
  - 080 Ocarina 陶笛
- Synth Lead 合成音一
  - 081 Lead 1 (square) 合成方波
  - 082 Lead 2 (sawtooth) 合成鋸齒波
  - 083 Lead 3 (calliope lead) 合成詩歌
  - 084 Lead 4 (chiff lead) 合成吹管
  - 085 Lead 5 (charang) 合成電結他
  - 086 Lead 6 (voice) 合成人聲鍵盤
  - 087 Lead 7 (fifths) 合成五度音
  - 088 Lead 8 (bass+lead) 貝司結他合奏
- Synth Pad 合成音二
  - 089 Pad 1 (new age) 合成新歲月
  - 090 Pad 2 (warm) 合成溫暖
  - 091 Pad 3 (polysynth) 多重合音
  - 092 Pad 4 (choir) 合成人聲合唱
  - 093 Pad 5 (bowed) 合成玻璃
  - 094 Pad 6 (metallic) 合成金屬
  - 095 Pad 7 (halo) 合成光華
  - 096 Pad 8 (sweep) 合成掃掠

GM Instrument Patch Map

- Synth Effects 合成音效
  - 097 FX 1 (rain) 合成音效1(雨聲)
  - 098 FX 2 (soundtrack) 合成音效2(聲帶)
  - 099 FX 3 (crystal) 合成音效3(水晶)
  - 100 FX 4 (atmosphere) 合成音效4(大氣)
  - 101 FX 5 (brightness) 合成音效5(明亮)
  - 102 FX 6 (goblins) 合成音效6(魅影)
  - 103 FX 7 (echoes) 合成音效7(迴音)
  - 104 FX 8 (sci-fi) 合成音效8(科幻)
- Ethnic 民族樂器
  - 105 Sitar 西塔琴
  - 106 Banjo 五絃琴
  - 107 Shamisen 三味線
  - 108 Koto 十三絃琴
  - 109 Kalimba 卡利瑪鐘琴
  - 110 Bagpipe 蘇格蘭風笛
  - 111 Fiddle 古提琴
  - 112 Shanai 鎖吶
- Percussive 敲擊樂器
  - 113 Tinkle Bell 叮噹鈴
  - 114 Agogo 阿哥哥鼓
  - 115 Stell Drums 鋼鼓
  - 116 Woodblock 木塊
  - 117 Taiko Drums 日本太鼓
  - 118 Melodic Tom 古式高音鼓
  - 119 Synth Drum 合成鼓
  - 120 Reverse Cymbal 鈸
- Sound Effects 特殊音效
  - 121 Guitar Fret Noise 磨弦聲
  - 122 Breath Noise 呼吸聲
  - 123 Seashore 海浪聲
  - 124 Bird Tweet 鳥叫聲
  - 125 Telephone Ring 電話鈴聲
  - 126 Helicopter 直昇機聲
  - 127 Applause 拍手聲
  - 128 Gunshot 槍聲

GM Percussion Key Map

- 35 Acoustic Bass Drum
- 36 Bass Drum 1
- 37 Side Stick
- 38 Acoustic Snare
- 39 Hand Clap
- 40 Electric Snare
- 41 Low Floor Tom
- 42 Closed Hi-Hat
- 43 High Floor Tom
- 44 Pedal Hi-Hat
- 45 Low Tom
- 46 Open Hi-Hat
- 47 Low-Mid Tom
- 48 Hi-Mid Tom
- 49 Crash Cymbal 1
- 50 High Tom
- 51 Ride Cymbal 1
- 52 Chinese Cymbal
- 53 Ride Bell
- 54 Tambourine
- 55 Splash Cymbal
- 56 Cowbell
- 57 Crash Cymbal 2
- 58 Vibraslap

GM Percussion Key Map

- 59 Ride Cymbal 2
- 60 Hi Bongo
- 61 Low Bongo
- 62 Mute Hi Conga
- 63 Open Hi Conga
- 64 Low Conga
- 65 High Timbale
- 66 Low Timbale
- 67 High Agogo
- 68 Low Agogo
- 69 Cabasa
- 70 Maracas
- 71 Short Whistle
- 72 Long Whistle
- 73 Short Guiro
- 74 Long Guiro
- 75 Claves
- 76 Hi Wood Block
- 77 Low Wood Block
- 78 Mute Cuica
- 79 Open Cuica
- 80 Mute Triangle
- 81 Open Triangle

MIDI

- Limitations
  - Limited no. of channels (16) and programs (128)
  - Limited resolution in data values (most are 8-bit)
  - Solution: some MIDI manufacturer utilities two midi data values to allow for large range of values (e.g. 16 bit range)
- MIDI 2.0 Standard (introduced in Jan 2020)
  - Backward compatible with MIDI 1.0
  - Support more channels and controllers
  - Support greater data range and resolution
  - Simplified messages and support more events

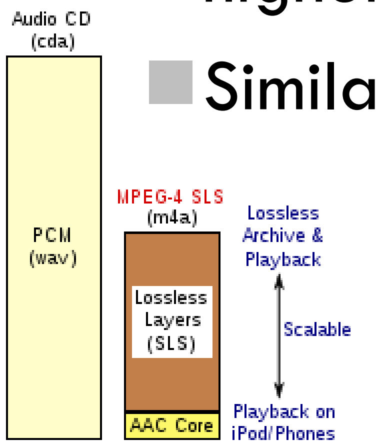
MPEG-4 Audio

MPEG-4 Audio

- MPEG-4 Part 3 was first published in 1999
  - ISO/IEC 14496-3 (known as MPEG-4 Audio)
  - Supports audio synthesis, structured audio, text-to-speech interface, lossy audio coding, lossless audio coding
  - Improve MPEG-2 AAC's compression efficiency to High-Efficiency Advanced Audio Coding (HE-AAC) (in 2003)
  - HE-AAC version 2 (HE-AAC v2) enhances compression efficiency of stereo audio (in 2004)
- MPEG-4 Audio Object Types
  - AAC, General MIDI, MPEG-1/2 Layer I/II/III, MPEG Surround, SAOC, USAC, etc

MPEG-4 Audio

- Subpart 12 of MPEG-4 Part 3: Scalable Lossless Coding (SLS)
  - Allow lossless audio compression scalable to lossy MPEG-4 General Audio coding (e.g. AAC)
  - SLS has a lower bandwidth lossy layer and a higher bandwidth lossless correction layer
  - Similar to the idea of hierarchical mode of JPEG



MPEG-4 Audio: Structured Audio

- MPEG-4 comprises of 6 structured audio tools
  - SAOL, SASL, SASBF, MIDI Semantics, Scheduler, AudioBIFS
- #1 SAOL: Structured Audio Orchestra Language
  - The central part of the structured audio toolset
  - A software-synthesis language for describing synthesizers (instruments)
  - Open support any known underlying synthesis methods
- #2 SASL: Structured Audio Score Language
  - A language to control the synthesizers specified by SAOL instruments
  - A score contains instructions that tell SAOL
    - Which notes to play, how loud to play them
    - What tempo to play them at, how long do they last, etc

MPEG-4 Audio: Structured Audio

- #3 MIDI Semantics:
  - Describe how to control SASL with a subset of MIDI
  - Reason to use MIDI to control SASL
    - MIDI is today's most commonly used representation for music score data
    - Many sophisticated authoring tools work with MIDI
- #4 SASBF: Structured Audio Sample Bank Format
  - A format for transmitting banks of sound samples
  - Used in wavetable, or sample-based synthesis
  - Partly compatible with MIDI Downloaded Sounds (DLS) format
- #5 Scheduler
  - The main body of the structured audio definition
  - Specify how SAOL is used to create sound when it is drivel by SASL or MIDI

MPEG-4 Audio: Structured Audio

- #6 AudioBIFS: Audio BInary Format for Scene description
  - Describes how the different objects (e.g. video clips, animations, sounds, other pieces of multimedia) in a structured media scene fit together
  - Specify the mixing and post-production of audio scenes when they are played back
    - e.g. mix voice track with the background music
    - Fade out voice track after 10 seconds, then another music fade in and has a reverb on it
- MPEG-4 Structured Audio vs. MIDI
  - MPEG-4 has more sophisticated controller structure
  - MPEG-4 does not suffer from MIDI's restriction on data range and resolution