

Perceptual Coding: Agenda

- Motivation für psychoakustische Kodierung
- Übersicht der wichtigsten Verfahren
- Literatur



MPEG1 - Audio:

- Konzept
- Encoder / Decoder
- Layer 1 / 2 / 3
- Einige Details der Algorithmen
- Beispiele zur Klangqualität
- Ausblick auf MPEG2 und MPEG4
- Ausblick auf MS-Audio (Windows Media) und RealAudio

Perceptual Coding: Motivation

perceptual coding = psychoakustische Kodierung :=

- verlustbehaftete Audiokodierung
- minimale Dateigröße bei möglichst hoher Klangqualität
- Kompression bis ca. 10:1 bei CD-Qualität: 2 bits / sample (!)
- durch Ausnutzen der Gehörwahrnehmung:
 - Ruhehearschwelle
 - Maskierungseffekte
- erfordert psychoakustische Modelle des Gehörs
- erfordert (Frequenz-) Analyse der Eingangsdaten:
 - "subband coders"
 - "transform coders"

Perceptual Coding: Verfahren

Verfahren	Methode	kHz	Kanäle	kb/s	Qualität
MPEG-1 layer 1	hybrid	32, 44.1, 48	1, 2	32 .. 448	AM / FM / (CD)
MPEG-1 layer 2	hybrid	32, 44.1, 48	1, 2	32 .. 384	AM / FM / (CD)
MPEG-1 layer 3	hybrid	32, 44.1, 48	1, 2	32 .. 320	AM / FM / CD
MPEG-2 AAC	hybrid	16 .. 48	1 .. 5.1	16 ..	AM / FM / CD
ATRAC	subband	44.1	2	256/ch	CD
Dolby AC-2	trans.	44.1	2	256/ch	CD
Dolby AC-3	trans.	44.1	1 .. 5.1	32 .. 384	CD
AT&T PAC	subband	44.1	1 .. 5.1	128/stereo	FM / CD
MS Audio ASF	?	44.1	2	10 ..	AM / FM / CD

- viele weitere Verfahren vorgeschlagen / erforscht
- detaillierte Übersicht siehe Spanias Review

Perceptual Coding: Literatur

- T. Painter, A. Spanias: Perceptual Coding of Digital Audio, Proc. IEEE 88-4, p.451ff (2000)
- T. Painter, A. Spanias: Review of Algorithms for Perceptual Coding of Audio Signals
www.eas.asu.edu/~spanias/papers.html
- D. Pan, An Overview of the MPEG/Audio Compression Algorithm, Proc. SPIE 260 (1997)
- S. Shlien, Guide to MPEG-1 Audio Standard, IEEE Trans. Broadcasting 40-4, p.206 (1994)
- Microsoft / RealAudio: msdn.microsoft.com/windowsmedia/, www.realaudio.com
- Fraunhofer IIS Dokumentation, www.iis.fhg.de
- LAME MP3 Encoder, www.sulaco.org/mp3/
- ISO Reference Software, ftp.tnt.uni-hannover.de/mpeg2/software/
- ISO/IEC Standards:
- MPEG-1, part 3: audio, ISO/IEC 11172-3 (1993)
 - MPEG-2, part 3: audio, ISO/IEC 13818-3 (1994) (Mehrkanal)
 - MPEG-4, part 3: audio
- c't, MP3-Feature (Algorithmus, Tools, Klangtest), c't 03/2000 p.122-151
- c't, MP3 vs MSAudio vs. RealAudio (mit Klangtest), c't 23/2000 p.152-161

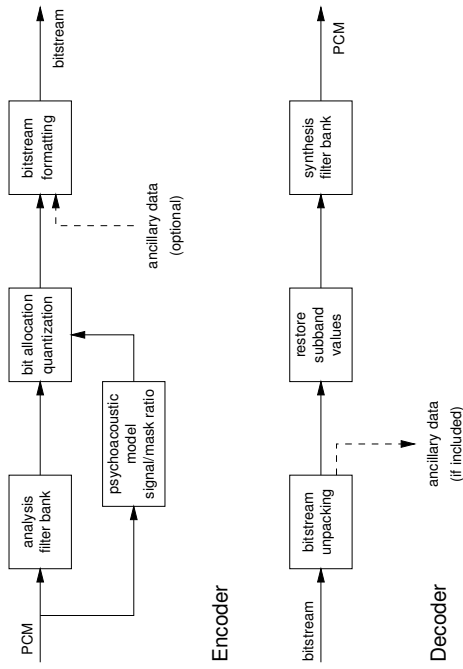
MPEG-1 Audio

- erster internationaler Standard für (HiFi-) Audio-Kompression
- entwickelt im Rahmen von MPEG-1
- bei Bitraten bis 1.5 Mb/s für Video mit Audio (Video-CD)
- verabschiedet Ende 1992 als ISO/IEC 11172-3 (1993)
- definiert das Format des Audio-Bitstroms
- und die Funktion des Decoders
- aber nicht den Encoder: Raum für spätere Verbesserungen
- Format geeignet für Sprache und Musik
- keine Quellenkodierung / keine Annahmen über Herkunft des Signals
- statt dessen psychoakustisches Modell
- Ausnutzen von Maskierungseffekten
- drei Varianten (layers), jeweils mono / stereo / joint stereo
- Samplerraten 32, 44.1, 48 KHz, Bitraten 32 .. 224 kb/s/channel

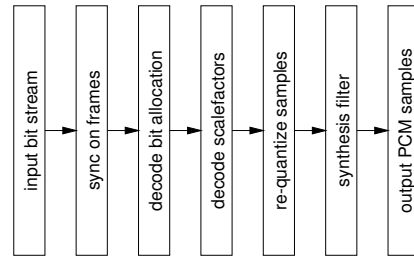
MPEG-1 Audio: Varianten

- drei Varianten, Bandbreite gesamt bis 1.5 Mb/s für Video+Audio
 - zwei psychoakustische Modelle
 - Encoder dürfen eigene psychoakustische Modelle verwenden
- Layer 1:
- einfachstes Verfahren
 - für Bitraten höher als 128 kb/s pro Kanal
 - eingesetzt für Philips DCC: 384 kb/s stereo
- Layer 2:
- mittlere Komplexität
 - für Bitraten von 128 kb/s pro Kanal
 - eingesetzt für CD-I und Video-CD
- Layer 3:
- bessere Qualität, aber deutlich komplexer
 - ab 64 kb/s pro Kanal, gute Qualität ab 128 kb/s
 - "MP3", ISDN-Übertragungen

MPEG-1: Blockschaltbild (grob)



MPEG-1: Decoder für Layer 1 und 2



- Layer 3 Decoder deutlich komplexer (siehe ISO 11172-3 S.40)

MPEG-1: Bitstrom-Header

```

audio_sequence()
{
    while (true)
    {
        frame()
    }
}

header()
{
    syncword      12 bits
    ID            1 bit
    layer         2 bits
    protection_bit 1 bit
    bitrate_index 4 bits
    sampling_frequency 2 bits
    padding_bit   1 bit
    private_bit   1 bit
    mode          2 bits
    mode_extension 2 bits
    copyright     1 bit
    original/home 1 bit
    emphasis      2 bits
}
    
```

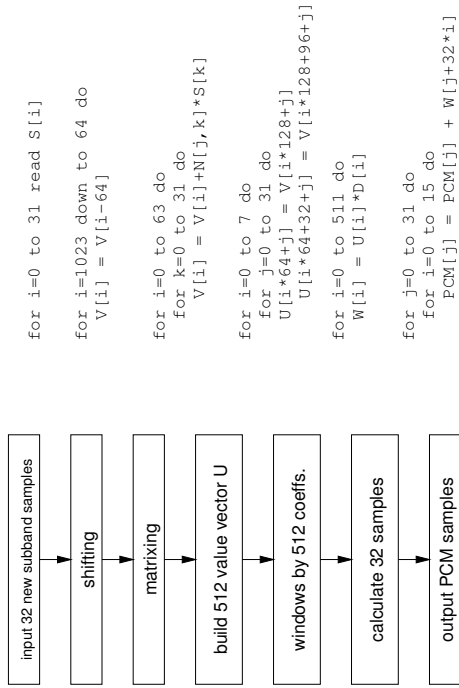
- bsbf: bit string, left bit first
- Synchronisation über Bitmuster '1111 1111 1111'

MPEG-1: Bitstrom-Daten, Layer 1

```

audio_data()
{
    if (mode==single_channel)
    {
        ...
    }
    if (mode==stereo) || (mode==dual_channel)
    {
        for (sb=0; sb<32; sb++)
        for (ch=0; ch<2; ch++)
            allocation[ch][sb] 4 bits bsmsbf
        for (sb=0; sb<32; sb++)
        for (ch=0; ch<2; ch++)
            if (allocation[ch][sb]!=0)
                scalefactor[ch][sb] 6 bits uimsbf
        for (s=0; s<12; s++)
        for (sb=0; sb<32; sb++)
        for (ch=0; ch<2; ch++)
            if (allocation[ch][sb]!=0)
                sample[ch][sb][s] 2..15 bits uimsbf
    }
    if (mode==intensity_stereo)
    {
        ...
    }
}
    
```

MPEG-1: Synthesis Subband Filter



MPEG-1: Zahldarstellung

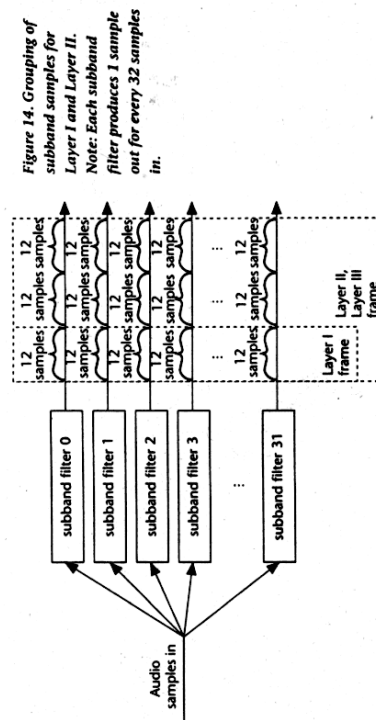
- möglichst kompakte Zahldarstellung gesucht
 - also mit datenabhängiger Genauigkeit
 - aber mit hohem Dynamikbereich
- => MPEG-1 Bitstrom verwendet spezielles Gleitkommaformat:
- gemeinsamer Skalenfaktor (Exponent) für eine Anzahl von Samples
 - variable Anzahl nb von Nachkommastellen
 - Auswahl der jeweiligen Stellenanzahl über Codetabellen
 - Mantisse hat nb Bits, Scalefactor aus Tabelle
 - Layer 3 zusätzlich Huffman-Codierung
- wert = scalefactor * (2**nb / (2**(nb-1)) * (mantisse + 2**(-nb+1)))

MPEG-1: Subband-Kodierung, Filterbank

"subband-Kodierung":

- Filterbank zerlegt das Eingangssignal
 - in mehrere (gleich breite) Frequenzbänder
- Idee:
- Gesamtenergie des Signals verteilt sich auf die Bänder
 - und zwar meistens ungleichmäßig
 - wenige Bänder mit viel Energie: präzise kodieren
 - viele Bänder mit wenig Energie: mit wenig Bits kodieren
 - Maskierung und Hörschwelle: kleine Koeffizienten auf Null setzen
 - inverse Filterbank setzt das Signal (fast) perfekt wieder zusammen

MPEG-1: Filterbank, Layer 1 und 2



- 32 Filterbänder, gleiche Frequenzbreite (je 20/32 KHz)
- Layer 1: 384 samples/frame input, 32x12 samples output
- Layer 2: 1152 samples/frame, 32x 3groups x 12 samples

MPEG-1: Filterbank, Layer 1 und 2

$$s_t(i) = \sum_{k=0}^7 \sum_{j=0}^7 M(i)[k] \cdot (C[k+64j] \cdot x[k+64j]) \quad (1)$$

where:

i is the subband index and ranges from 0 to 31,
 $s_t(i)$ is the filter output sample for subband i at time t , where t is an integer multiple of 32 audio sample intervals,
 $C(n)$ is one of 512 coefficients of the analysis window defined in the standard,
 $x(n)$ is an audio input sample read from a 512 sample buffer, and
 $M(i)[k] = \cos\left(\frac{(2^k+1)\pi(k-16)}{64}\right)$ are the analysis matrix coefficients.

- Berechnung der Subband-Samples $S_t(i)$
- $M(i)[k]$ ist um k bzw. j verschoben: "polyphase" Filterbank
- effizientere Berechnung via FFT möglich
- Layer 3 verwendet aufwendigere Filterbank

MPEG-1: Filterbank

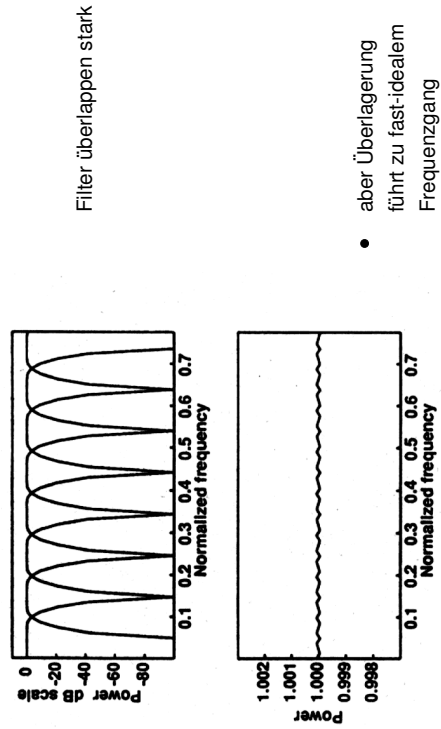


Fig. 1. Top: frequency response of the first 8 subband filters. Bottom: frequency response of the analysis filter bank. The frequency has been normalized to the Nyquist frequency 2π .

MPEG-1: Filterbank Aliasing

- einzelne Filterbänder überlappen
- Signale nahe der Übergänge werden in zwei Bänder kodiert
- Alias-Effekte möglich
- Layer-3 verwendet nachgeschaltete MDCT-Transformation
- zur Verbesserung der Frequenzauflösung
- mit Versuch der Alias-Reduzierung

Frequenzgruppen, Bark-Skala

Tabelle 9.1: Frequenzgruppen nach Zwicker 1982

z /Bark	f_z /Hz	f_c /Hz	Δf_c /Hz	f_m /Hz
0	0	100	100	150
1	100	200	100	250
2	200	300	100	350
3	300	400	100	450
4	400	510	110	550
5	510	630	120	670
6	630	770	140	840
7	770	920	150	1000
8	920	1080	160	1170
9	1080	1270	190	1370
10	1270	1480	210	1600
11	1480	1720	240	1850
12	1720	2000	280	2150
13	2000	2320	320	2500
14	2320	2700	380	2900
15	2700	3150	450	3400
16	3150	3700	550	4000
17	3700	4400	700	4800
18	4400	5300	900	5800
19	5300	6400	1100	7000
20	6400	7700	1300	8500
21	7700	9500	1800	10500
22	9500	12000	2500	13500
23	12000	15500	3500	
24	15500			

[Zwölzer]

Tonhöhenwahrnehmung?

- Messungen
- vgl. Tuningkurven
- tiefe Frequenzen: ca. 100 Hz Bandbreite
- hohe Frequenzen: log. Breite der Gruppen => "Bark" Frequenzskala
- Ausgangspunkt für psychoakustische Modelle

MPEG-1: Maskierung

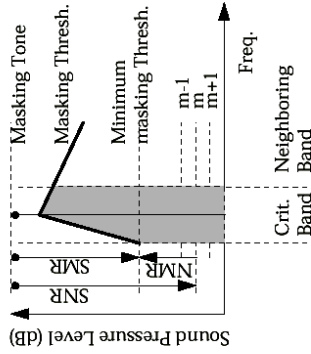
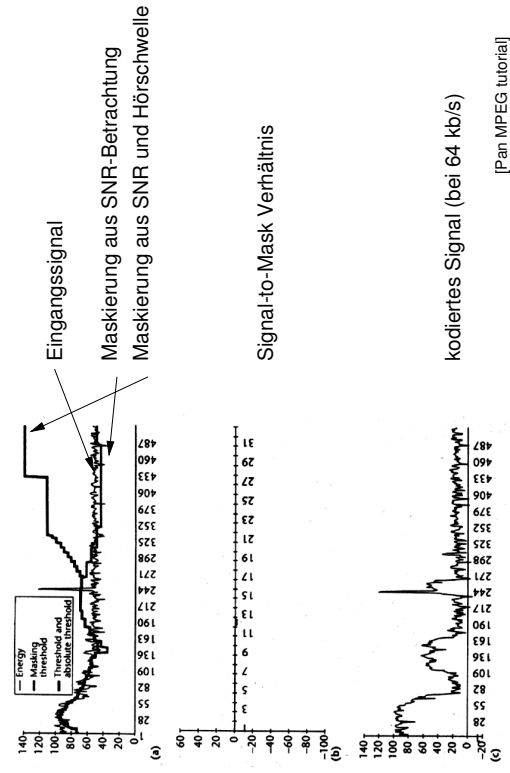


Fig. 6. Schematic Representation of Simultaneous Masking (after [30])

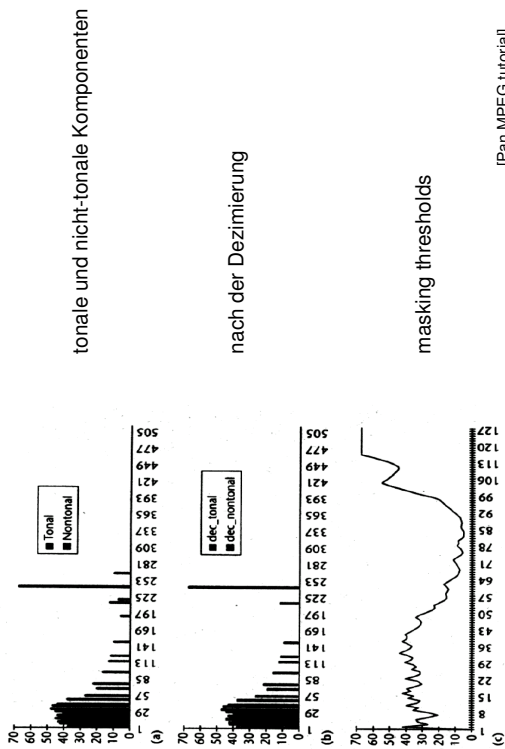
- lautester Ton im Frequenzband maskiert benachbarte
- Maskierung linear darstellbar, wenn Frequenz in Bark gemessen wird
- Töne leiser als der Masking-Threshold nicht wahrnehmbar
- SNR / SMR / NMR Verhältnisse

MPEG-1: psychoakustisches Modell 2



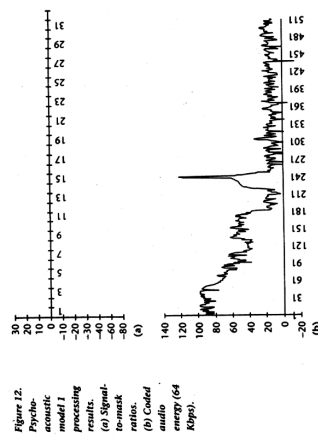
[Pan MPEG tutorial]

MPEG-1: psychoakustisches Modell 1



Digitale Audioverarbeitung | WS 2000 | 18.205

MPEG-1: psychoakustisches Modell 1



- Ergebnis der Kodierung nach Modell 1
- teilweise deutlich schlechter als Modell 2

Digitale Audioverarbeitung | WS 2000 | 18.205

[Pan MPEG tutorial]

MPEG-1: bit allocation

- psychoakustisches Modell liefert "signal-to-mask" Verhältnis
 - für jedes einzelne Subband
- Bit-Allocation Algorithmus:
- Berechnung von MNR und SNR (mask-noise, signal-noise-ratio)
 - wieviele Bits stehen zur Verfügung? Bitrate minus Sync. etc.
 - Aufgabe: Bitzuweisung, so daß das minimale MNR maximal
 - für die Skalierungsfaktoren
 - und die Koeffizienten aller einzelnen Subbänder
 - Details für die Layer 1,2,3 leicht unterschiedlich
 - iterativ: berechne MNR und SNR für jedes Subband
 - weise dem Subband mit der schlechtesten MNR Bits zu
 - solange noch Bits zur Verfügung stehen

Digitale Audioverarbeitung | WS 2000 | 18.205

MPEG-1: bit allocation

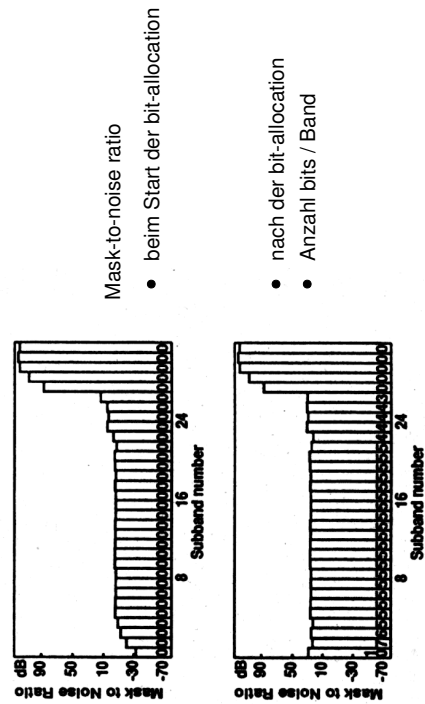
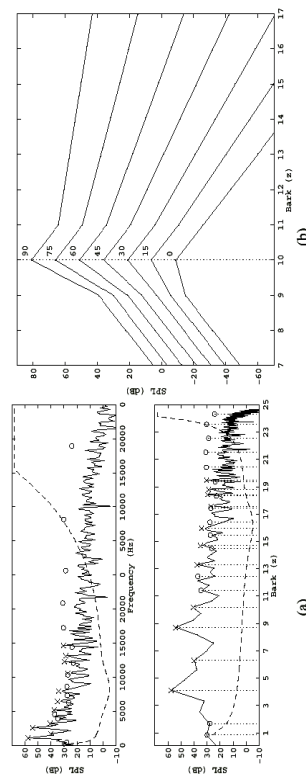


Fig. 8. Mask to noise ratio at the beginning (top) and end (bottom) of the bit allocation algorithm. The number of bits assigned to each subband sample appears at the base of each bar.

[Shlien MPEG guide]

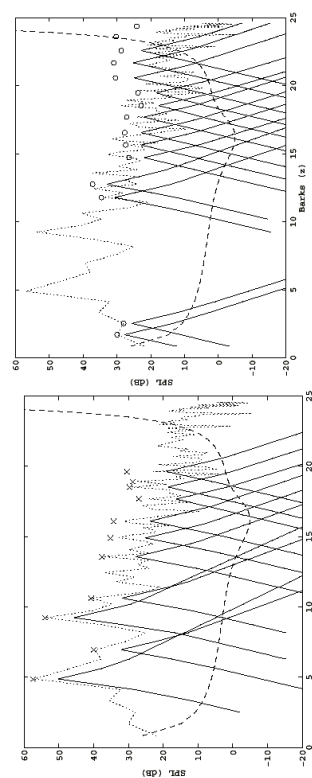
Digitale Audioverarbeitung | WS 2000 | 18.205

MPEG-1: model 1, tonal/noise maskers



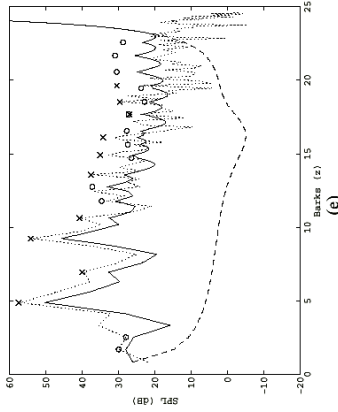
(a) Step 1: Obtain PSD, express in dB SPL. Top panel gives linear frequency scale, bottom panel gives Bark frequency scale. Absolute threshold superimposed. Step 2: Tonal maskers identified and denoted by 'X' symbol; Noise maskers identified and denoted by 'O' symbol. (b) Collection of prototype spreading functions (Eq. 28) shown with level as the parameter. These illustrate the incorporation of excitation pattern level-dependence into the model. Note that the prototype functions are defined to be piecewise linear on the Bark scale. These will be associated with maskers in steps 3, 4.

MPEG-1: model 1, masking and spreading



(c) Steps 3.4: Spreading functions are associated with each of the individual tonal maskers satisfying the rules outlined in the text. Note that the Signal-to-Mask Ratio (SMR) at the peak is close to the widely accepted tonal value of 14.5 dB. (d) Spreading functions are associated with each of the individual noise maskers that were extracted after the tonal maskers had been eliminated from consideration, as described in the text. Note that the peak SMR is close to the widely accepted noise-masker value of 5 dB.

MPEG-1: model 1, masking and spreading



(e) Step 5: A global masking threshold is obtained by combining the individual thresholds as described in the text. The maximum of the global threshold and the absolute threshold are taken at each point in frequency to be the final global threshold. The figure clearly shows that some portions of the input spectrum require SNRs of better than 20 dB to prevent audible distortion, while other spectral regions require less than 3 dB SNR. In fact, some high-frequency portions of the signal spectrum are masked and therefore perceptually irrelevant, ultimately requiring no bits for quantization without the introduction of artifacts.

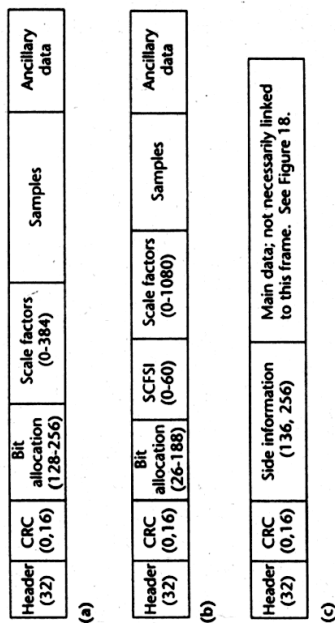
MPEG-1: Bitraten für Layer 1 .. 3

MPEG-1 Bitraten (mono/stereo/joint stereo):

index	Layer I		Layer II		Layer III	
	free	format	free	format	free	format
'0000'	32	kbit/s	32	kbit/s	32	kbit/s
'0001'	32	kbit/s	48	kbit/s	40	kbit/s
'0010'	64	kbit/s	56	kbit/s	48	kbit/s
'0011'	96	kbit/s	64	kbit/s	56	kbit/s
'0100'	128	kbit/s	80	kbit/s	64	kbit/s
'0101'	160	kbit/s	96	kbit/s	80	kbit/s
'0110'	192	kbit/s	112	kbit/s	96	kbit/s
'0111'	224	kbit/s	128	kbit/s	112	kbit/s
'1000'	256	kbit/s	160	kbit/s	128	kbit/s
'1001'	288	kbit/s	192	kbit/s	160	kbit/s
'1010'	320	kbit/s	224	kbit/s	192	kbit/s
'1011'	352	kbit/s	256	kbit/s	224	kbit/s
'1100'	384	kbit/s	320	kbit/s	256	kbit/s
'1101'	416	kbit/s	384	kbit/s	320	kbit/s
'1110'	448	kbit/s			384	kbit/s

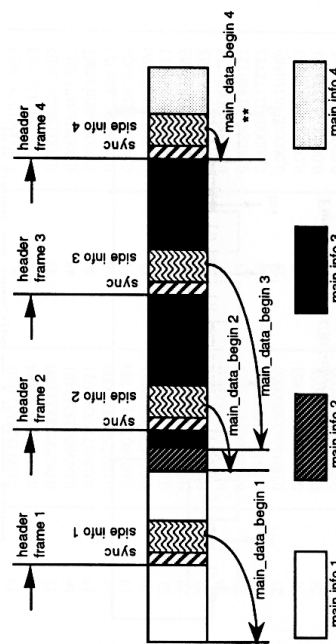
- 32 .. 448 kb/s

MPEG-1: bitstream format



- Header mit speziellem Sync-Muster, dann CRC
- Layer 3 erlaubt "borrowing" zwischen Frames

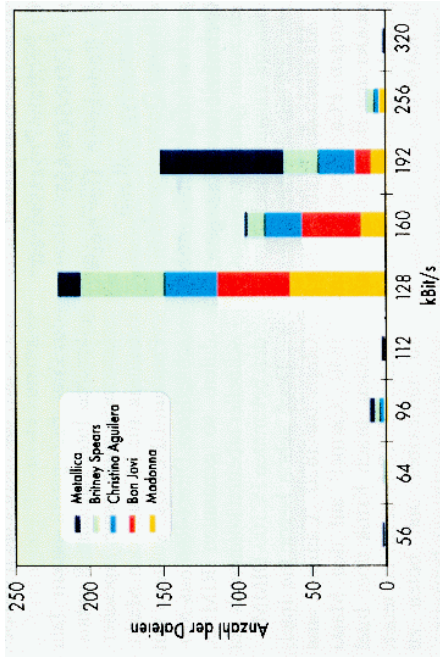
MPEG-1: bitstream format



**) main_data_begin 4 == 0: This signifies that main data starts directly after the side information for frame 4. This is the lower limit for main_data_begin; main_data cannot start later than this point. Note that data bytes used by "sync" and "side info" are not counted by the main_data_begin pointer.
 Note: "info" means information

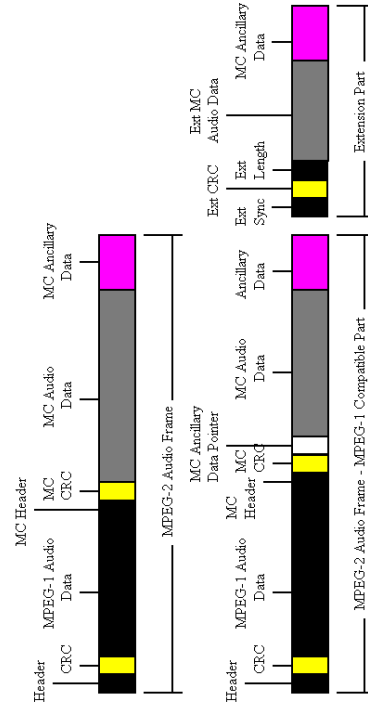
Figure A.7.b -- Layer III bitstream organization with peak demand at main info 3 and small demand at main info 2.

MPEG-1: gängige Bitraten



- Histogramm der Bitraten (MP3)
- Suche per Napster, aus [c't 2/2000]

MPEG-2: multichannel bitstream



- MPEG-1 compatibles Stereosignal, plus Mehrkanal-Erweiterung

Figure A.2 Structure of the ISO 13818-3 Layer II multichannel extension, using the ISO/IEC 11172-3 compatible bit stream as well as the extension bit stream

MPEG-1: pre-echos

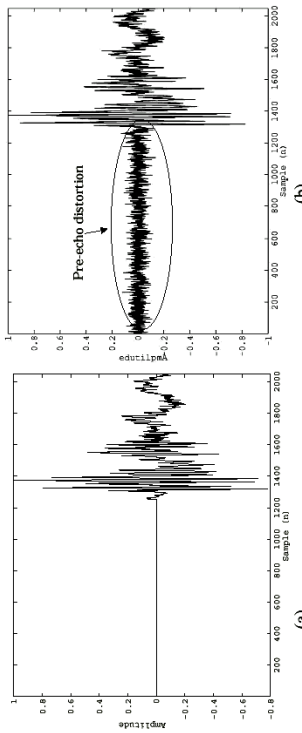
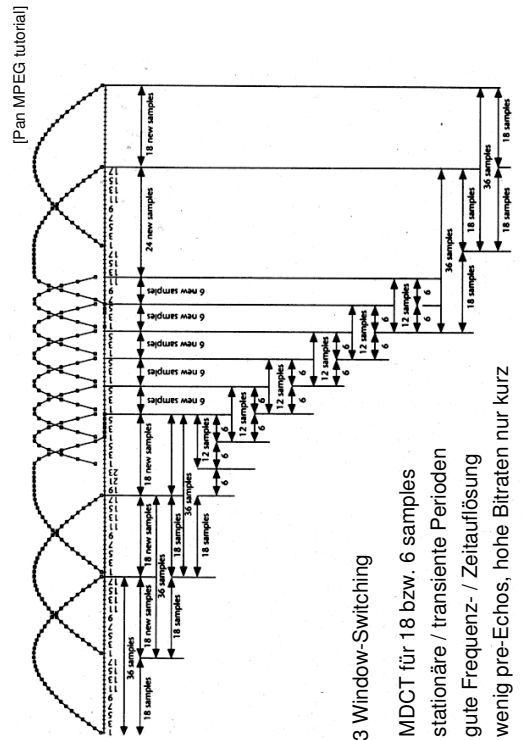


Fig. 14. Pre-Echo Example: (a) Unencoded Castanets. (b) Transform Coded Castanets, 2048-Point Block Size [Painter Spanias]

- Problem mit plötzlichen lauten Signalen nach einer Blockgrenze
 - vorheriger Block hat sehr wenig Bits kodiert
 - Rauschen des lauten Signals wird nicht mehr maskiert
- => Situation erkennen und behandeln (kürzere MDCT Blöcke)

MPEG-1: window-switching



Layer 3 Window-Switching

- MDCT für 18 bzw. 6 samples
- stationäre / transiente Perioden
- gute Frequenz- / Zeitauflösung
- wenig pre-Echos, hohe Bitraten nur kurz

MPEG-1: temporal noise-shaping

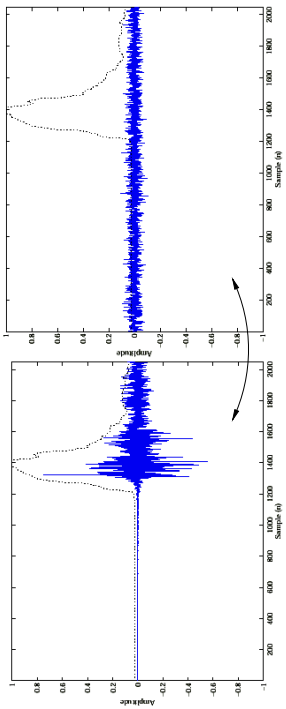


Fig. 17. Temporal Noise Shaping Example Showing Quantization Noise and the Input Signal Energy Envelope for Castanets: (a) Without TNS, and (b) With TNS [Painter Spanias, a) b) vertauscht]

weitere Verbesserung durch TNS:

- Verlagerung des Rauschens in den Bereich mit höchster Maskierung
- wird für MPEG-2 AAC eingesetzt

MPEG-2: BC und NBC (AAC)

"backwards compatible":

- ähnliches Format wie MPEG-1 layer 3
- aber mit 3/2-Mehrkanal-Erweiterung
- abwärtskompatibel (für die 2/0 Stereosignale)
- ungeeignet für 3/2 mit Bitraten unter 640 kb/s

"non backwards compatible" / "advanced audio coding"

- gleiches Konzept wie MPEG-1 layer 3
- aber viele Verbesserungen, mit Ziel geringerer Bitrate
- viele neue Details, bessere Filterbank, bessere Zeitauflösung
- temporal noise shaping, . . .
- zusätzliche Bitraten bis herunter zu 16 kb/s/channel

MPEG-2: AAC Eigenschaften

MPEG-2 AAC (ISO 13818-7, 1997):

- Mehrkanal-Audio (3/2) bei Bitraten bis 384 kb/s
- transparente Kodierung (Hörtests) ab 320 kb/s
- mehrere "Profile": main / low / scalable sample rate profile
- bessere Filterbank als MP3:
- Frequenzauflösung 23 Hz (75Hz), Zeitauflösung 2.6 ms
- keine Hybrid-Filterbank, sondern nur MDCT
- MDCT window shape switching (stationary / transient signals)
- time prediction of transform coefficients
- more efficient Huffman code tables
- temporal noise shaping
- scalability and complexity management tools
- eingesetzt u.a. für DVD und Liquid Audio

MPEG-2: AAC-Decoder Performance

"revised report on complexity of MPEG-2 AAC tools2"
 "ISO/IEC JTC1/SC229/WD11/N2957 MPEG99"

welche Hardware-Anforderungen für AAC-Dekodierung?!

Decoder	Complexity
2-channel MP software decoder	40% Pentium/133
2-channel low complexity sw decoder	25% Pentium/133
5-channel MP hardware decoder	90 mm² 0.5 µm CMOS
Huffman decode	13.3 %
inv. quantization, scaling	1.7 %
temporal noise shaping	6.6 %
prediction	43.2 %
IMDCT	19.4 %
coupling channel	15.5 %

MPEG-2: und Nachfolger

- entwickelt für geringe Bitraten
- Hörtests bescheinigen (sehr) gute Qualität
- auch für kritisches Audio-Material
- aber nicht unbedingt besser als MP3
- siehe MPEG-2 quality report
- Hörtests / Vergleich mehrerer Algorithmen
- MPEG-4 (ISO/IEC 14496, 1998) enthält MPEG-2
- zusätzlich "objekt-orientierte" Algorithmen
- für Bitraten von 64 kb/s bis hinunter zu 200 b/s (l)
- für Sprachkodierung / -synthese
- für Musikkodierung / -synthese (a la MIDI)

MPEG-1: LAME

"Lame ain't an MPEG encoder"

- MP3-Encoder unter GPL Lizenz, im Quellcode verfügbar
- ursprünglich als Patch gegen den ISO Referenzcode
- derzeit als völlig neuentwickeltes Programm
- Decoder basiert auf mpeglib (www.mpg123.de)
- mit eigenem psychoakustischem Modell
- Qualität vergleichbar mit dem FhG-Encoder
- ideal als Grundlage für Eigenentwicklungen
- unterstützt MPEG-1 layer 1 bis 3
- recht schnelle Kodierung
- www.sulaco.com/mp3 (aktuell ist Lame 3.87)

LAME: psychoakustisches Modell

```

/* PsyModel.c (c) 1999 Mark Taylor
 * Copyright (c) 1999 Mark Taylor
 * Compute FFTs mid and side channel is derived from L & R
 * Compute energies
 * Compute unpredictability of first six spectral lines
 * Compute unpredictability of next 200 spectral lines
 * Calculate the energy and the unpredictability in the threshold
 * calculation partitions
 * convolve the partitioned energy and unpredictability
 * with the spreading function, sp_[B][k]
 * calculate the tonality of each threshold calculation partition
 * calculate the energy in each threshold calculation partition
 * tonality = 0; use NMT (lots of masking)
 * tonality = 1; use TMN (little masking)
 * at this point, tbb represents the amount the spreading function
 * will be reduced. The smaller the value, the less masking.
 * long block pre-echo control for both short and long blocks
 * threshold calculation for short blocks
 * compute M/S thresholds from Johnston & Ferreira 1992 ICASSP paper
 * determine final block type
 * compute side_energy / (side+mid)_energy
 * 0 = no energy in side channel
 * .5 = half of total energy in side channel
 * Read long block data for converting threshold calculation
 * Read short block data for converting threshold calculation
 * compute which partition bands are in which scalefactor bands
 * compute bark value and ATH of each critical band
 * Now compute the spreading function, s[j][i], the value of the spread-
 * ing function, centered at band j, for band i, store for later use
 * / setup stereo demasking thresholds
  
```

LAME: Status, Performance

- Echtzeit-Kodierung ist möglich
 - auch bei hohen Bitraten / guter Qualität
- ```

vs. Echtzeit
lame --preset studio 256 kb/s 3X
lame --preset cd 192 kb/s 3X
lame --preset fm 112 kb/s 3X
auf Celeron/466

```
- psychoakustisches Modell liegt im Quelltext vor
  - weitere Verbesserungen vorgeschlagen und geplant
  - aber massive rechtliche Probleme:
  - fast alle MPEG-1/2 Verfahren patentiert und lizenzpflichtig
  - siehe MPEG-2 AAC Lizenzvertrag (PDF)

## LAME: Presets

Presets are some shortcuts for common settings.  
They can be combined with -v if you want VBR MP3s.

```

=====
phone phont+ lw sw fm voice radio tape hifi cd studio
=====
--resample 8 11 11 11 32 24
--highpass 0.125 0.1 4 4 15 12 15 18 18
--lowpass 3.4 4 4 0 0 0 0 0 0.9 0.9
--lowpass-width yes yes no no yes no no no no no
--noshort -mm -mm -mm -mm -mm -mm -mm -mm -h -h
-b 16 24 24 24 112 56 128 128 160 192 256
-- PLUS WITH -v -----
-v 6 4 3 3 3 4 3 3 2 1 0
-b 8 16 16 16 80 40 96 112 128 160
-B 24 32 56 56 256 112 256 256 320 320 320
=====
EXAMPLES:
a) --preset fm
equal to: -mj -b112 --resample 32 --lowpass 15 --lowpass-width 0
b) -v --preset studio
equals to: -h -ms -v0 -b160 -B320

```

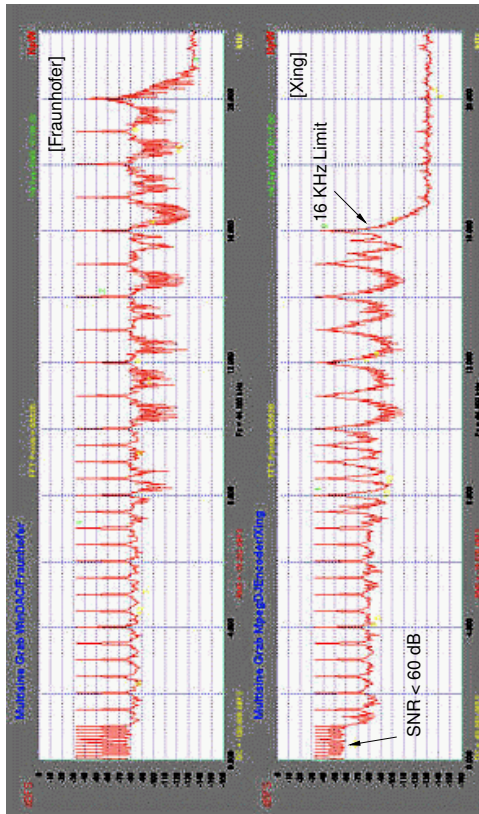
- feste Bitraten von 16 kb/s (mono) bis 256 kb/s (stereo)
- -B setzt maximal zulässigen Wert bei variabler Bitrate
- -V 0 für beste Qualität, -V 9 für minimale Dateigröße

## Klangvergleich

|            | MP3 | WMA | RA8 | ePAC | AAC | MP+ |
|------------|-----|-----|-----|------|-----|-----|
| Modem      | 0   | ++  | +   | ++   | -   | X   |
| Rock       | +   | +   | +   | +    | 0   | ++  |
|            | 160 | 160 | 185 | 160  | 256 | 184 |
| Pop        | +   | ++  | +   | ++   | +   | ++  |
|            | 160 | 128 | 185 | 128  | 192 | 171 |
| Klassik    | +   | +   | ++  | +    | +   | ++  |
|            | 160 | 160 | 136 | 160  | 190 | 158 |
| Stresstest | +   | --  | 0   | --   | ++  | ++  |
|            | 320 | 192 | 360 | 160  | 320 | 296 |

- Vergleichstest in c't 2/3/2000 p.152
- Klangqualität bei 128/138 kb/s / Bitrate für CD-nahe Qualität
- MP+ ist Eigenentwicklung der Autoren, "audiophil"

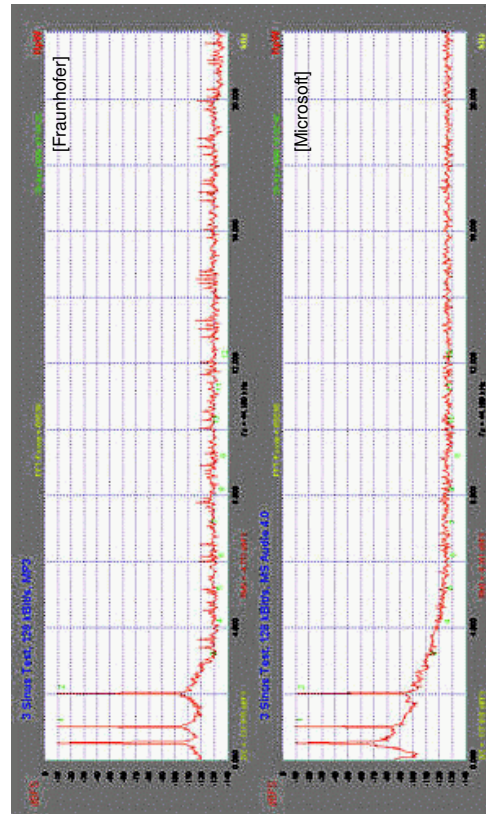
**MP3: zwei Encoder ...**



- Eingangssignal mit 44 Sinustönen ...

[ct 21/98 242]

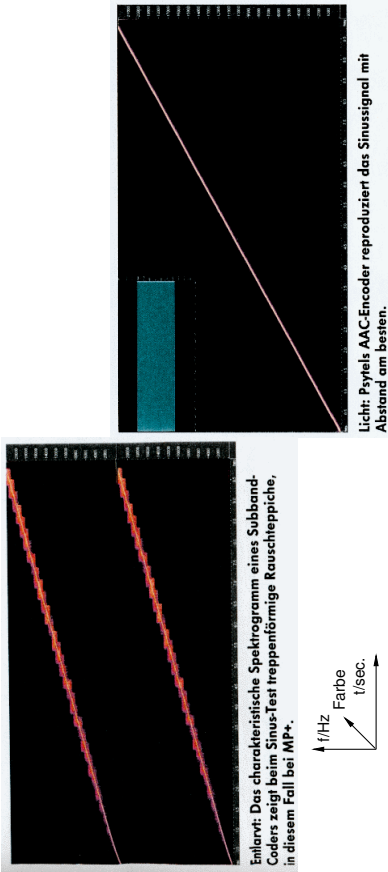
**MP3: vs. MS Audio 4.0**



- neuere Versionen von WMA sind besser

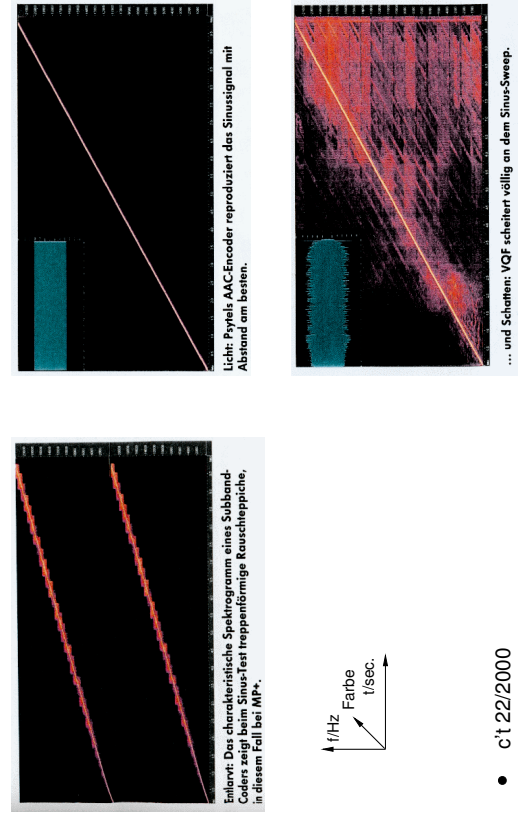
[ct 21/98 242]

**MPEG-I: Sinus-Sweep**



- Subband-Kodierung zerlegt das Signal in Frequenzbereiche
- Frequenz-Sweep besonders schwer zu kodieren
- Test für MP3, AAC, MP+, VQF in ct 22/2000, 152

**Sinus-Sweep: MP+, AAC, VCQ**



- ct 22/2000