

# **En överblick över förlustbringande komprimering och dess inverkan på ljudåtergivningen av en klassisk gitarr**

Jafet Kackur

Examensarbete  
Informations- och medieteknik  
2012

Jafet Kackur

EXAMENSARBETE	
Arcada	
Utbildningsprogram:	Informations- och medieteknik
Identifikationsnummer:	3328
Författare:	Jafet Kackur
Arbetets namn:	En överblick över förlustbringande komprimering och dess inverkan på ljudåtergivningen av en klassisk gitarr
Handledare (Arcada):	Johnny Biström
Uppdragsgivare:	
<p>Sammandrag:</p> <p>Det finns idag många olika valmöjligheter vad gäller komprimering och lagring av digitalt ljud. Ljudfiler kodas ofta med hjälp av någon kodningsteknik (ett s.k. kodek) som är förlustbringande, dvs. ljudet manipuleras i kodningsprocessen och information som bedöms som onödig eller som inte kan uppfattas av den mänskliga hörseln ”tas bort” för att minska på filstorleken.</p> <p>Syftet med detta arbete är att presentera en överblick över förlustbringande ljudkomprimering och ta reda på om den påverkar ljudåtergivningen av ett enskilt instrument, i det här fallet den klassiska gitarren. I den teoretiska delen presenteras teknikerna bakom digitalt ljud och förlustbringande ljudkomprimering. Den klassiska gitarren och dess ljudegenskaper presenteras också. I den praktiska delen jämförs omkodade ljudfiler med originalfilen och skillnader presenteras.</p> <p>De förlustbringande ljudkodek som används i detta examensarbete är MPEG-1 Layer III (MP3) och MPEG-2 Advanced Audio Coding (AAC).</p>	
Nyckelord:	ljudåtergivning ljudkodning mp3 aac
Sidantal:	55
Språk:	Svenska
Datum för godkännande:	

DEGREETHESIS	
Arcada	
DegreeProgramme:	Information- and mediatechnology
Identificationnumber:	3328
Author:	Jafet Kackur
Title:	An overview over lossy audio compression and its effects on sound reproduction of a classical guitar
Supervisor (Arcada):	Johnny Biström
Commissioned by:	
<p>Abstract:</p> <p>There are nowadays many options for audio compression and storing. Sound files are often coded using some coding algorithm (a codec, short for coder/decoder) that is lossy, meaning that the sound is manipulated in the coding process and information that is deemed unnecessary and/or cannot be heard by the human ear is discarded. This is done to reduce the size of the audio files.</p> <p>The aim of this thesis is to present an overview over lossy audio compression and find out if it affects the audio quality and reproduction of a single instrument, namely the classical guitar. In the theoretical part the technologies behind digital audio and lossy audio compression are presented. The classical guitar and its sound properties are also presented. In the practical part the encoded audio files are compared to the original audio files and differences are presented.</p> <p>The lossy audio codecs used in this thesis are MPEG-1 Layer III (MP3) and MPEG-2 Advanced Audio Coding (AAC).</p>	
Keywords:	sound compression encoding mp3 aac
Numberof pages:	55
Language:	Swedish
Date ofacceptance:	

# INNEHÅLL

Figurer .....	7
<b>1 INLEDNING .....</b>	<b>8</b>
1.1 Bakgrund .....	8
1.2 Syfte och mål .....	9
1.3 Metoder .....	9
1.4 Avgränsning .....	9
1.5 Termer och begrepp .....	10
<b>2 DIGITALT LJUD .....</b>	<b>11</b>
2.1 Analogt till digitalt .....	11
2.2 Sampling .....	11
2.2.1 <i>Samplingteoremet</i> .....	12
2.2.2 <i>Aliasing</i> .....	12
2.3 Kvantisering .....	13
2.3.1 <i>Signal-to-errorratio</i> .....	13
2.3.2 <i>Dithering</i> .....	14
<b>3 LJUDKOMPRIMERING .....</b>	<b>16</b>
3.1 Grunderna i förlustbringande ljudkomprimering .....	16
3.2 Psykoakustik .....	16
3.2.1 <i>Örat</i> .....	17
3.2.2 <i>Kritiska band</i> .....	17
3.2.3 <i>Hörbarhetsgränsen</i> .....	18
3.2.4 <i>Maskering</i> .....	18
3.2.5 <i>Psykoakustiska modeller</i> .....	19
3.3 Filterbanker .....	19
3.4 Subbandskodning .....	20
3.5 Transformkodning .....	20
3.5.1 <i>Fönsterfunktioner (Window functions)</i> .....	21
3.5.2 <i>Transformkodningens funktionsprincip</i> .....	22
3.5.3 <i>Föreko (pre-echo)</i> .....	23
3.6 Huffman-kodning .....	24
<b>4 MPEG-1 LAYER III (MP3) .....</b>	<b>25</b>
4.1 Teknik .....	25
4.1.1 <i>Bithastighet och samplingfrekvens</i> .....	26
4.1.2 <i>Stereolägen</i> .....	26

4.1.3	<i>MP3-kodarens funktionsprincip</i> .....	27
4.1.4	<i>Fönster</i> .....	28
<b>5</b>	<b>MPEG-2 ADVANCED AUDIO CODING (AAC)</b> .....	<b>29</b>
5.1	Teknik .....	29
5.1.1	<i>Bithastighet och samplingfrekvens</i> .....	29
5.1.2	<i>Stereolägen</i> .....	30
5.1.3	<i>Skillnader och förbättringar gentemot MP3</i> .....	30
5.1.4	<i>AAC-kodarens funktionsprincip</i> .....	31
<b>6</b>	<b>DEN KLASSISKA GITARREN</b> .....	<b>33</b>
6.1	Historia .....	33
6.2	Storlek och uppbyggnad .....	33
6.3	Ljudegenskaper .....	35
6.4	Frekvensområde .....	35
<b>7</b>	<b>ANALYS AV ETT INSPELAT STYCKE KLASSISK GITARR</b> .....	<b>36</b>
7.1	Överföring av gitarrstycke från CD-skiva till WAV-Format .....	36
7.2	Kodning av ljudfiler .....	36
7.2.1	<i>MP3</i> .....	37
7.2.2	<i>AAC</i> .....	37
7.2.3	<i>Filstorlekar</i> .....	37
7.3	Analys .....	38
7.3.1	<i>Analys i tidsdomänen</i> .....	38
7.3.2	<i>Frekvensanalys</i> .....	39
7.3.3	<i>Spektral analys</i> .....	40
<b>8</b>	<b>RESULTAT</b> .....	<b>41</b>
8.1	Analys i tidsdomänen .....	41
8.1.1	<i>128 kbps</i> .....	42
8.1.2	<i>256 kbps</i> .....	43
8.2	Frekvensanalys .....	43
8.2.1	<i>128 kbps</i> .....	44
8.2.2	<i>256 kbps</i> .....	45
8.3	Spektral analys .....	46
8.3.1	<i>128 kbps</i> .....	47
8.3.2	<i>256 kbps</i> .....	48
8.3.3	<i>Närmare jämförelser</i> .....	49
<b>9</b>	<b>SLUTSATSER</b> .....	<b>51</b>

9.1	Påverkan på den klassiska gitarrens ljudåtergivning .....	51
9.2	Skillnader mellan kodek .....	52
<b>10</b>	<b>FORTSATT FORSKNING.....</b>	<b>53</b>
	<b>KÄLLOR.....</b>	<b>54</b>

## Figurer

Figur 1. (a) En ljudsignal och dess kvantiserade värden. (b) Kvantiseringsfelet. (Watkinson 2001 s.221).....	14
Figur 2. En 100 Hz sinusvåg i 24- resp. 16-bitars upplösning. (Aldrich 2002 s.2).....	15
Figur 3. En 100 Hz sinusvåg i 24- resp. 16-bitars upplösning, dither tillsatt. (Aldrich 2002 s.7) .....	15
Figur 4. Maskering (Wikipedia 2012a) .....	19
Figur 5. Rektangulärt fönster. (Wikipedia 2012b) .....	21
Figur 6. Två vanliga fönsterformer som används i MDCT.(Talbot-Smith 1999 s.71) .	22
Figur 7. Exempel på föreko. (Karlheinz Brandenburg, s.7). .....	24
Figur 8. Blockdiagram av en MP3-kodare. (Karlheinz Brandenburg, s.4) .....	27
Figur 9. Blockdiagram över AAC-kodaren. (Karlheinz-Brandenburg, s.6).....	31
Figur 10. Exempel på mått på en gitarrkropp. (Eendebak 2011) .....	34
Figur 11. Den akustiska gitarren och dess delar. (Fall 2011) .....	34
Figur 12. WAV-filen av hela "El Colibri" visas i tidsdomänen i iZotope RX.....	38
Figur 13. Jämförelse mellan en WAV-fil och en MP3-fil. WAV grön, MP3 röd.....	39
Figur14. Spektral vy i iZotope RX. Vänster och höger ljudkanal.....	40
Figur 15. Den ursprungliga WAV-filen. Vänster kanal. ....	41
Figur 16. MP3 och AAC 128 kbps.WAV visas som mörkgrå. ....	42
Figur 17. MP3 och AAC 256 kbps. WAV-kurvan syns knappt p.g.a. likheterna. ....	43
Figur 18. Den usprungliga WAV-filen av stycket "El Colibri". ....	44
Figur 19. "El Colibri". MP3 128 kbps (röd), AAC 128 kbps (grön).....	44
Figur 20. "El Colibri". MP3 256 kbps (röd), AAC 256 kbps (grön).....	45
Figur 21. Spektral vy av den ursprungliga WAV-filen. ....	46
Figur 22. MP3 (128 kbps). ....	47
Figur 23. AAC (128 kbps).....	47
Figur 24. MP3 (256 kbps). ....	48
Figur 25. AAC (256 kbps).....	48
Figur 26. WAV till vänster, MP3 (256 kbps) mitten, MP3 (128 kbps) höger.....	49
Figur 27. Wav vänster, AAC (256 kbps) mitten, AAC (128 kbps) höger.....	50
Tabell 1. Ljudfilerna och deras storlek.....	38

# 1 INLEDNING

Det finns idag många olika format att lagra och distribuera ljud med. Att lyssna på musik via internet eller ladda ner ljudfiler till sin dator är en möjlighet som man tar för givet och något som de flesta inte lägger någon närmare eftertanke till. Men vad är det egentligen för skillnad mellan de olika ljudformat som finns tillgängliga?

## 1.1 Bakgrund

När man lyssnar på musik på webben, med Spotify eller i sin MP3-spelare är musiken ofta kodad i ett ljudformat som är förlustbringande. Detta innebär att originalinspelningen av musik- eller ljudstycket i fråga har kodats om för att minska på dess filstorlek. Under denna process har en del av innehållet i originalversionen gått förlorat. Det finns många olika tekniker och ljudformat för detta ändamål och man kan säga att de alla strävar efter samma sak: Att minska på filstorleken så långt det bara går samtidigt som man vill bevara så mycket som möjligt av det ursprungliga innehållet.

Varför gör man då detta? Det handlar om att spara på utrymme och beakta hastigheten på våra internetanslutningar. Okomprimerat ljud är väldigt utrymmeskrävande och en musiksamling kan väldigt snabbt ta upp mycket plats på en hårddisk. Ljudfilernas storlek kunde även orsaka problem om man strömmar musiken via internet. Man skulle behöva en snabb och stabil internetanslutning för att musiken skulle flyta på felfritt utan laddningspauser ifall den vore okomprimerad.

Jag har i mitt slutarbete valt att analysera förlustbringande ljudkodning och vilken inverkan den har på ett särskilt musikinstrument, nämligen den klassiska gitarren. Jag har valt just detta instrument främst för att det är ett instrument jag själv spelar, men även för att ta reda på vilken inverkan förlustbringande ljudkomprimering kan ha på ett enskilt instrument. De förlustbringande ljudkodnings-algoritmer jag kommer att analysera är MPEG-1 Layer III (MP3) och MPEG-2 Advanced Audio Coding (AAC). Dessa kodek är två av de mest använda i dagsläget.



## 1.2 Syfte och mål

Syftet med detta arbete är att presentera en överblick över förlustbringande ljudkomprimering och ta reda på om man kan påvisa skillnader i ljudkvaliteten mellan olika förlustbringande ljudkodningsalgoritmer i en inspelning av ett stycke klassisk gitarr. Målet är sedan att kunna presentera en jämförelse mellan de ljudkodningsalgoritmer jag valt att analysera.

I arbetet framkommer följande frågeställningar:

- På vilket sätt påverkar ljudkomprimering ljudkvaliteten?
- Kan man påvisa skillnader i ljudkvaliteten mellan de olika komprimeringsteknikerna hos ett specifikt instrument, närmare bestämt den klassiska gitarren?

## 1.3 Metoder

Jag har valt att analysera två av de vanligast förekommande förlustbringande ljudkodningstekniker som för närvarande används. Jag kommer att analysera en inspelning av ett stycke klassisk gitarr och sedan omkoda originalinspelningen i de ljudkodek jag valt. Jag gör sedan en analys i tidsdomänen, en frekvensanalys samt en spektral analys på de olika omkodade ljudfilerna och jämför resultaten med varandra samt med originalet.

## 1.4 Avgränsning

Detta arbete begränsar sig till den digitala formen av ljud och ljudbehandling. Jag kommer inte att gå in på analoga komponenter och deras funktion.

Jag kommer enbart att behandla förlustbringande ljudkomprimering. Det finns även så kallad förlustfri ljudkomprimering som strävar efter att komprimera ljudfiler så långt som möjligt utan att någon information går förlorad, men detta kommer inte att behandlas.

## 1.5 Termer och begrepp

### *Strömning*

Strömning (eng. streaming) är uppspelning av ljud- och videofilm på en mottagares dator/mobiltelefon/surfplatta samtidigt som denna information överförs från en annan plats över ett nätverk.

### *Kodek*

Ett kodek (eng. codec, förkortning av coder/decoder) är en komponent i exempelvis ett ljudsystem som kodar/avkodar digitala ljudströmmar.

### *Lågpasfilter*

Ett lågpasfilter (eng. low-pass filter) är en komponent i ett ljudsystem som enbart släpper igenom frekvenser som befinner sig under ett specificerat gränsvärde

### *Högpasfilter*

Ett högpasfilter (eng. high-pass filter) fungerar som ett lågpasfilter fast tvärtom. Enbart frekvenser som befinner sig ovanför ett specificerat värde släpps igenom.

### *Bandpassfilter*

Ett bandpassfilter (eng. band-pass filter) släpper igenom frekvenser som befinner sig mellan två givna värden, t.ex. mellan 3 och 6 kHz.

### *Signal-brusförhållande*

Signal-brusförhållande (eng. signal-to-noise ratio) definierar förhållandet mellan signal och brus i en ljudsignal. Ju lägre brusnivå desto mindre störs man av det.

## 2 DIGITALT LJUD

Innan jag går närmare in på de ljudkodningsformat jag valt att analysera finner jag det nödvändigt att presentera en överblick över digitalt ljud och hur det i grunden fungerar, detta för att underlätta förståelsen för det kommande innehållet i detta examensarbete.

### 2.1 Analogt till digitalt

När man spelar in ljudet av t.ex. en mänsklig röst eller en akustisk gitarr så görs det med hjälp av en mikrofon. När mikrofonen tar emot ljudvågorna från rösten eller gitarren konverteras ljudet till en signal med varierande spänning. Spänningen motsvarar ljudsignalens frekvens och amplitud. Denna analoga signal är kontinuerlig och varierar med ett oändligt antal frekvens- och amplitudvärden.

För att den ska kunna användas av datorer och andra digitala system måste den analoga ljudsignalen i något skede konverteras till digital form. Detta sker med hjälp av en så kallad A/D (Analog/Digital) konverter. När en digital ljudsignal ska konverteras till analog svänger man helt enkelt på steken och resultatet blir då en D/A (Digital/Analog) konverter.

När ljudet blivit digitaliserat kan man kort och gott säga att det hela handlar om siffror, närmare bestämt binära värden. I princip alla digitala system använder sig av det binära talsystemet. Systemet består av två värden; 1 och 0 (på och av). Enbart dessa två värden behövs för att representera maskinens grundläggande elektriska funktion för om spänningen är på eller av. (Pohlmann 2011 s.6)

Det finns ett flertal olika tekniker för behandling av digitalt ljud, men det allra vanligaste systemet kallas *Pulse Code Modulation* (PCM). Systemet går ut på att en inkommande ljudsignal mäts i ett givet intervall. (Watkinson 2001 s.3)

### 2.2 Sampling

Grundstenarna i digitalt ljud (och i det här fallet PCM) är *sampling* och *kvantisering* (Pohlmann 2011 s.19). Sampling går ut på att den analoga ljudsignalen mäts (samplas)

med ett givet tidsintervall. Med en samplingfrekvens på exempelvis 48 kHz samplas den analoga signalen 48 000 gånger i sekunden.

### 2.2.1 Samplingteoremet

När en analog ljudsignal samplas med ett givet intervall dyker kanske en fråga upp: Vad händer ”mellan samplen”? Tappar man inte en del av informationen i den ursprungliga ljudsignalen? Det enkla svaret är ”nej” om ljudsignalen behandlas på korrekt sätt innan samplingen äger rum. (Pohlmann 2011 s.19) För att sampling ska ske utan förluster kör man den ursprungliga ljudsignalen genom ett så kallat *lågpasfilter*, det vill säga man begränsar bandbredden på ljudsignalen, exempelvis till intervallet 0 – 24 kHz. När man gör detta så kan man sedan helt och hållet (i teorin) återskapa den ursprungliga ljudsignalen från den samplade signalen.

*Samplingteoremet* säger oss att en kontinuerlig bandbegränsad signal helt och hållet kan återskapas utan förluster, men för att detta ska vara möjligt måste samplingfrekvensen vara dubbelt högre än den högsta förekommande frekvensen i ljudsignalen. (Pohlmann 2011 s.21) Detta innebär alltså att en ljudsignal med frekvensområdet 0 – 24 kHz måste samplas med en frekvens på 48 kHz för att inga förluster ska förekomma.

Med andra ord kan ljudsignaler bestående av frekvenser mellan 0 och  $S/2$  Hz helt återskapas om man använder sig av samplingsfrekvensen  $S$  Hz. Frekvensen  $S/2$  Hz kallas även *Nyquist-frekvensen* (uppkallad efter ingenjören Harry Nyquist som lade grunden för det samplingsteorem som idag används inom digitalt ljud). (Pohlmann 2011 s.20)

### 2.2.2 Aliasing

Aliasing kan sägas vara orsaken till att man bör bandbegränsa en ljudsignal före den samplas. Aliasing innebär att en frekvens som befinner sig över frekvensen  $S/2$  Hz ”reflekteras tillbaka” och dyker upp på ett ställe i ljudsignalen där den inte hör hemma. Ta exempelvis ett system med en samplingsfrekvens på 44 kHz. Enligt samplingsteoremet borde den högsta förekommande frekvensen då vara 22 kHz. Om vi då föreställer oss att en signal med frekvensen 26 kHz kommer in i systemet kommer

denna frekvens att reflekteras tillbaka som frekvensen  $S - F = F_j$ , eller  $44 - 26 = 18$  kHz. (Pohlmann 2011 s.25)

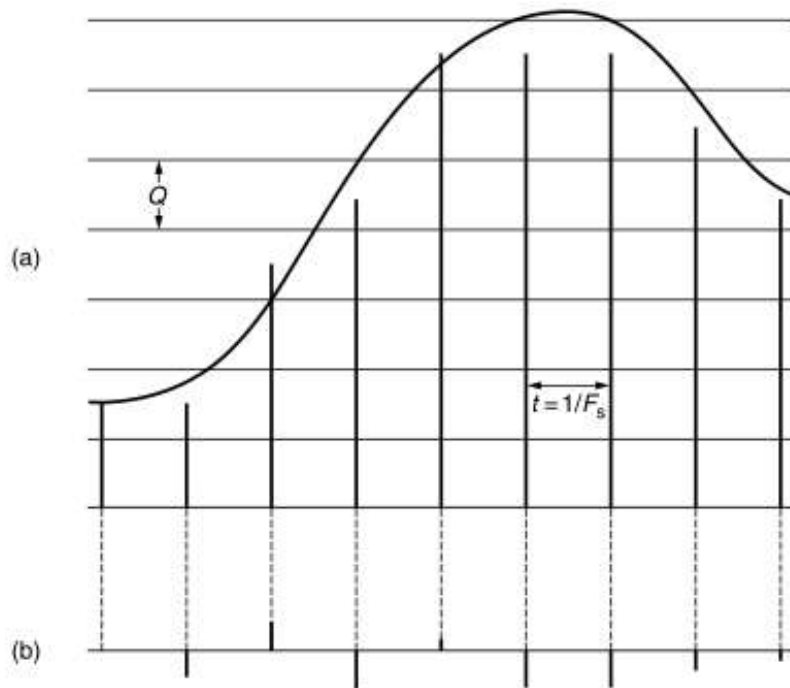
## 2.3 Kvantisering

När man mäter en ljudsignal räcker det inte med att mäta den enbart i tidsdomänen (sampling), utan man behöver även mäta själva *värdet* av signalen. När det handlar om en ljudsignal är detta värde signalens *amplitud*. (Pohlmann 2011 s.28) Eftersom en analog ljudsignal består av oändligt många amplitudnivåer handlar kvantisering i grund och botten om att återskapa en oändligt varierande kvantitet med hjälp av ett bestämt antal diskreta värden (Watkinson 2001 s.217).

### 2.3.1 Signal-to-errorratio

När den analoga signalen kvantiseras kommer den inkommande signalen att ”avrundas” till det närmaste kvantiseringsintervallet. I ett binärt system bestäms intervallet av  $2^n$  kvantiseringsintervall där  $n =$  bitantalet. Om bitantalet är 16 bitar så kommer antalet kvantiseringsintervall att vara  $2^{16} = 65536$ .

Eftersom en analog ljudsignal består av ett oändligt antal intervall kommer givetvis fel att uppstå eftersom analoga värden som förekommer ”mellan” två kvantiseringsvärden måste avrundas till det närmaste kvantiserbara värdet. Ett högre bitantal bidrar givetvis till att kvantiseringsfel blir mindre och i något skede kommer fel att bli ohörbara, men problemet med kvantiseringsfel går inte att kringgå. (Pohlmann 2011 s.29) Kvantiseringsfel kallas ofta kvantiseringsbrus eller kvantiseringsdistorsion och det framträder i en kvantiserad ljudsignal som sådant om man inte gör något åt saken.



Figur 1. (a) En ljudsignal och dess kvantiserade värden. (b) Kvantiseringsfelet. (Watkinson 2001 s.221)

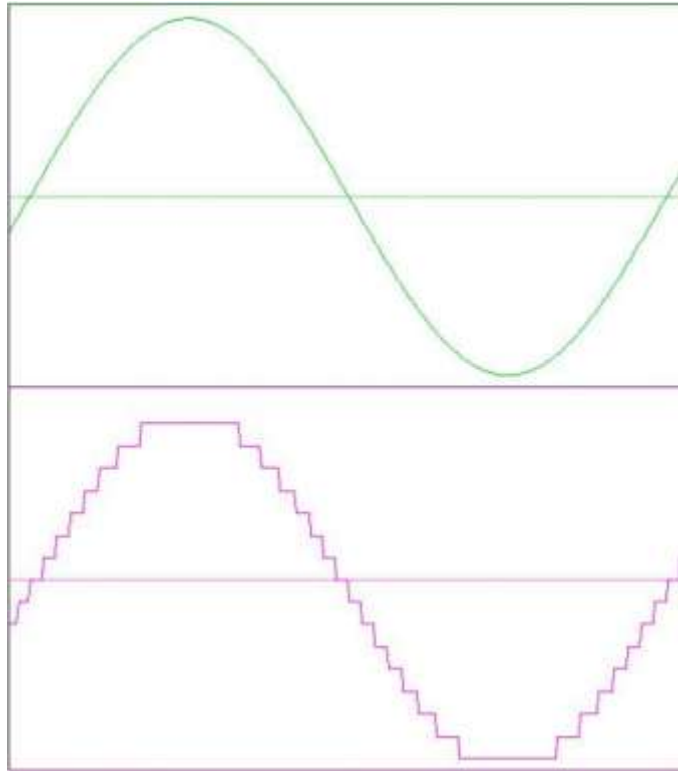
### 2.3.2 Dithering

För att avhjälpa problemet med kvantiseringsfel används en process som kallas *dithering*. Processen går ut på att man lägger till brus i ljudsignalen. (Fries 2000 s.125) Att lägga till brus i en ljudsignal för att *reducera* brus/distorsion låter kanske lite bakvänt, men bruset motverkar att en kvantiserad ljudsignal blir alltför linjär då kvantiserade värden kommer att variera mer tack vare det tillagda bruset.

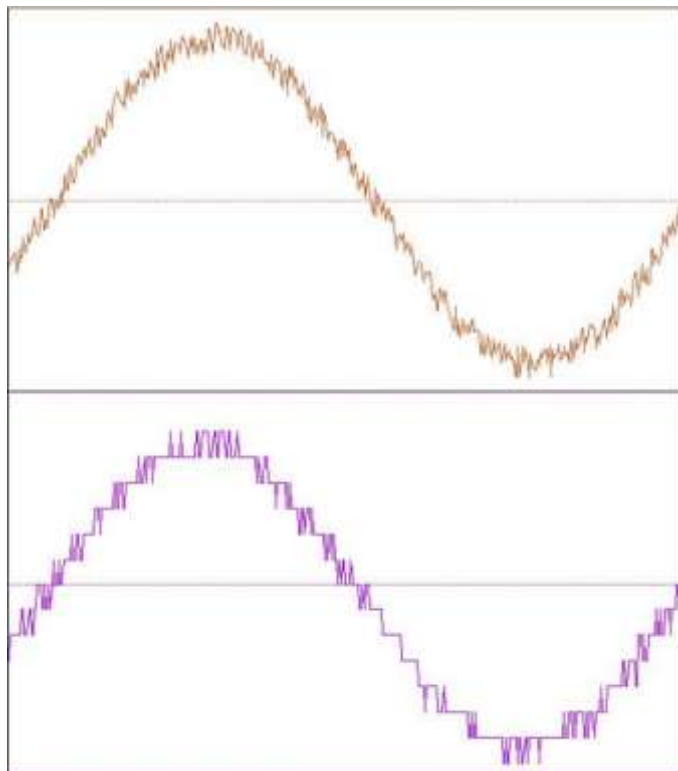
Tack vare dither reduceras mängden distorsion i ljudsignalen avsevärt, men det lägger istället till en liten mängd brus i ljudsignalen (Pohlmann 2011 s.36). Detta brus innebär att ljudsignalens signal-brusförhållande minskar. Detta är dock en försumbar förlust jämfört med vad man vinner på att använda dithering. (Watkinson 2001 s.225)

Figur 2 visar upptill en 100 Hz sinusvåg i 24-bitars upplösning, nertill samma sinusvåg i 16-bitars upplösning.

Figur 3 visar upptill en 100 Hz sinusvåg, denna gång med tillagt brus. Nertill visas samma sinusvåg i 16-bitars upplösning. Kvantiseringsfelet är nu inte lika framkommande eftersom det är mer utspritt.



Figur 2. En 100 Hz sinusvåg i 24- resp. 16-bitars upplösning. (Aldrich 2002 s.2)



Figur 3. En 100 Hz sinusvåg i 24- resp. 16-bitars upplösning, dither tillsatt. (Aldrich 2002 s.7)

### 3 LJUDKOMPRIMERING

I en värld där hårddiskar är oändligt stora och internetanslutningar oändligt snabba skulle det inte finnas något behov av komprimerat ljud (eller något behov av komprimering över huvudtaget), men med den teknik som finns tillgänglig i nuläget är behovet av komprimeringen verklighet.

Det finns två huvudsakliga former av ljudkomprimering; förlustbringande och förlustfri (Fries 2000 s.132). Förlustbringande ljudkodning komprimerar ljudfiler, ”kastar bort” all onödig information samt sådant som inte kan höras av det mänskliga örat. Förlustfri ljudkodning innebär att ljudfiler komprimeras utan att någon information går förlorad. Detta är i princip samma form av komprimering som görs av program som WinRAR och WinZip. Ingen information går förlorad när man komprimerar exempelvis ett Word-dokument med något av dessa program. Jag kommer dock inte att gå in på förlustfri ljudkomprimering i detta arbete.

#### 3.1 Grunderna i förlustbringande ljudkomprimering

Förlustbringande ljudkomprimering går ut på att komprimera ljud, lämna bort onödig information i ljudsignalen samt lämna bort all information som inte kan uppfattas av det mänskliga örat. En av de viktigaste delarna i förlustbringande ljudkomprimering är användandet av *psykoakustik* och olika *psykoakustiska modeller*. En annan benämning på detta när det handlar om ljudkomprimering är *perceptuell kodning* (Pohlmann 2011 s.335).

Förutom psykoakustik presenteras även övriga vanliga tekniker som används i perceptuell ljudkodning, närmare bestämt filterbanker, subbandskodning, transformkodning och Huffman-kodning.

#### 3.2 Psykoakustik

Psykoakustik handlar om örats uppfattning av ljud och på vilka sätt örat ”tolkar” olika ljud (Pohlmann 2011 s.336). Tillämpandet av psykoakustiska modeller av den



mänskliga hörseln är grunden i alla ljudkodnings-algoritmer med effektiv komprimering (Talbot-Smith 1999 s.1·49).

### 3.2.1 Örat

Det mänskliga örat använder sig av en komplex kombination av neurologiska och mekaniska processer för upptagning (och tolkning) av ljud. Örat överför energin från ljudvågor till mekanisk energi som sedan överförs som elektriska impulser till hjärnan. (Pohlmann 2011 s.339) Den mänskliga hörseln täcker ett väldigt brett frekvensområde och kan urskilja kring 10 oktaver (en oktav är en fördubbling i frekvens). (Watkinson 2001 s.38) Skillnaden i tonhöjd mellan 100 Hz och 200 Hz kallas en *oktav*. Skillnaden mellan 1000 och 2000 Hz uppfattas som lika stor. Det totala frekvensområdet för den mänskliga hörseln anses i allmänhet ligga mellan 20 Hz – 20 kHz. (Watkinson 2001 s. 45)

Örat och dess nervsystem är som sagt väldigt komplext, men det har sina begränsningar. Örat kan vara väldigt effektivt på att upptäcka nyanser i en ljudsignal samtidigt som det kan vara ganska oprecist med vissa aspekter av signalen. (Pohlmann 2011 s.336)

Örats känslighet för amplitud varierar mycket beroende på frekvensområde. En knappt hörbar ton med frekvensen 30 Hz borde vara 60 dB högre i amplitud än en knappt hörbar ton vid 1 kHz. (Pohlmann 2011 s.338) Örat är som känsligast i området 1 kHz – 5 kHz. Vid låga frekvenser är örat relativt oprecist. (Pohlmann 2011 s.336)

### 3.2.2 Kritiska band

Det centrala temat i en modell av den mänskliga hörseln är konceptet med *kritiska band* som framtoogs första gången av Harvey Fletcher 1940. (Talbot-Smith 1999 s.1·50) Konceptet baserar sig på experiment som visade att när brus maskerar (täcker) en ren ton har enbart frekvensmässigt närliggande komponenter i bruset betydelse i maskering av själva tonen. Detta närliggande område kring tonen kallas ett *kritiskt band*. Örat upptar mer information vid låga frekvenser och mindre information vid höga frekvenser, vilket innebär att kritiska band är smala vid låga frekvenser och från 500 Hz uppåt växer de kritiska banden i storlek relativt proportionellt i förhållande till ljudsignalen.

(Pohlmann 2011 s.340) Om ett band med brus spelas med en konstant ljudnivå så kommer ljudnivån att bibehållas när bandbredden ökas, men så fort bandbredden överstiger bredden av ett kritiskt band kommer volymen att öka (Pohlmann 2011 s.342).

Storleken (bandbredden) på ett kritiskt band kan uppskattas med formeln

$$\text{Kritisk Bandbredd} = 25 + 75 [1 + 1,4 (f / 1000)^2]^{0,69} \text{Hz.}$$

där  $f$  är den mittersta frekvensen inom det kritiska bandet. (Pohlmann 2011 s.341)

### 3.2.3 Hörbarhetsgränsen

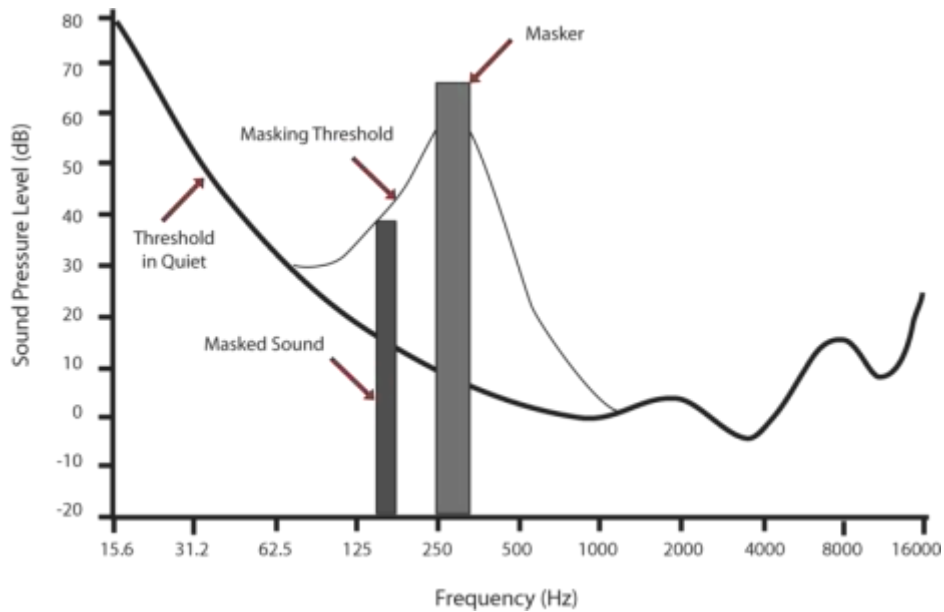
En av de viktigaste aspekterna av den mänskliga hörseln är *hörbarhetsgränsen*. Hörbarhetsgränsen specificeras som 0 dB vid 1 kHz. Örats känslighet för ljud varierar beroende på vilket frekvensområde ljudet befinner sig i. Två ljud med lika amplitud och olika frekvens uppfattas alltså inte som lika starka. Ett förlustbringande ljudkodek använder sig av detta för att jämföra signalen med hörbarhetsgränsen och ”kastar bort” ljud som faller under denna gräns. Detta innebär även att ett kodek utan problem kan tillåta kvantiseringsbrus som befinner sig under hörbarhetsgränsen. (Pohlmann 2011 s. 344)

### 3.2.4 Maskering

Maskering är ett fenomen som sker inom kritiska band. Maskering innebär att örat är mindre känsligt för ljud som frekvensmässigt ligger nära ett annat ljud men är förhållandevis lägre i ljudnivå. Detta resulterar i att det svagare ljudet maskeras (förblir ohört). De flesta ljudkodningsalgoritmer använder sig av detta fenomen och det går ofta under benämningen *perceptuell maskering* (Watkinson 2001 s.278).

Maskeringseffekten varierar mycket beroende på vilken frekvens det rör sig om och huruvida signalen är av tonal art eller om den är mera lik brus. (Talbot-Smith 1999 s.1-52)

Figur 4 visar hur maskering går till. Amplituden på det maskerande ljudet (Masker) gör att det maskerade ljudet (Masked Sound) hamnar under dess maskeringsnivå (Masking Treshold) och förblir ohörbart.



Figur 4. Maskering (Wikipedia 2012a)

### 3.2.5 Psykoakustiska modeller

Psykoakustiska modeller är utformade att efterlikna det mänskliga hörselsystemet och används för att ”emulera” den mänskliga hörseln vid kodningen av en ljudfil. (Pohlmann 2011 s.349) Man kan enkelt säga att den psykoakustiska modellen avgör vilka delar av ljudsignalen som ”kastas bort” och vilka delar som sparas (MultimediaWiki 2011).

Ett förlustbringande ljudkodek använder sig av en psykoakustisk modell för att identifiera irrelevanta (ohörbara) delar i ljudsignalen. Ljudsignalen kodas sedan med hjälp av denna information. Detta minskar effektivt på den mängd information som behövs för att representera ljudströmmen, men det ökar samtidigt på mängden kvantiseringsbrus. Kvantiseringsbruset kan dock formas i enlighet med ljudströmmen och ”gömmas” bakom andra delar av signalen. (Pohlmann 2011 s.336)

### 3.3 Filterbanker

Förlustbringande ljudkodningsalgoritmer använder ofta så kallade *filterbanker* för att dela in ljudsignalen i flera mindre delar eller *subband* (Pohlmann 2011 s.364). En filterbank är till sin funktion ett bandpassfilter som delar in ljudsignalen i flera komponenter (Wikipedia 2011a).

### 3.4 Subbandskodning

Subbandskodning (eng. sub-band coding) är en teknik som utvecklades av Bell Labs i början av 1980-talet (Pohlmann 2011 s.356). Enkelt uttryckt går subbandskodning ut på att man delar in ljudsignalen i ett antal frekvensband (Watkinson 2001 s.293). Kodaren analyserar energin i varje frekvensband och räknar sedan ut medelvärdet av amplituden inom varje band. Detta medelvärde används sedan för att räkna ut maskeringsnivån för varje band. Maskeringsnivån från närliggande band beaktas även i processen. Amplitudtoppar mäts och jämförs med maskeringsnivån och om närliggande ljud ligger under denna nivå kodas de inte. Band som inte innehåller någon information lämnas bort i kodningsprocessen och ljud som ligger under tröskeln för hörbarhet lämnas också bort. (Pohlmann 2011 s.358)

### 3.5 Transformkodning

Transformkodning är allmänt inom ljud- och bildkompression. I transformkodning samplas information från tidsdomänen till frekvensdomänen med hjälp av *Fouriertransform*.

I digitala system ses en vågform som ett antal diskreta amplitudvärden i tidsdomänen (sampel). Fouriertransform omvandlar dessa amplitudvärden till ett likvärdigt antal diskreta *frekvenser*. Detta kallas *Diskret Fouriertransform* (eng. Discrete Fourier Transform, DFT). (Watkinson 2001 s.175) Man får då en uppsättning koefficienter vilka grupperas i ungefär 32 band (vilket ”emulerar” analys av kritiska band). Detta representerar blocket av sampel i tidsdomänen. Koefficienterna kvantiseras sedan enligt en psykoakustisk modell. (Pohlmann 2011 s.361)

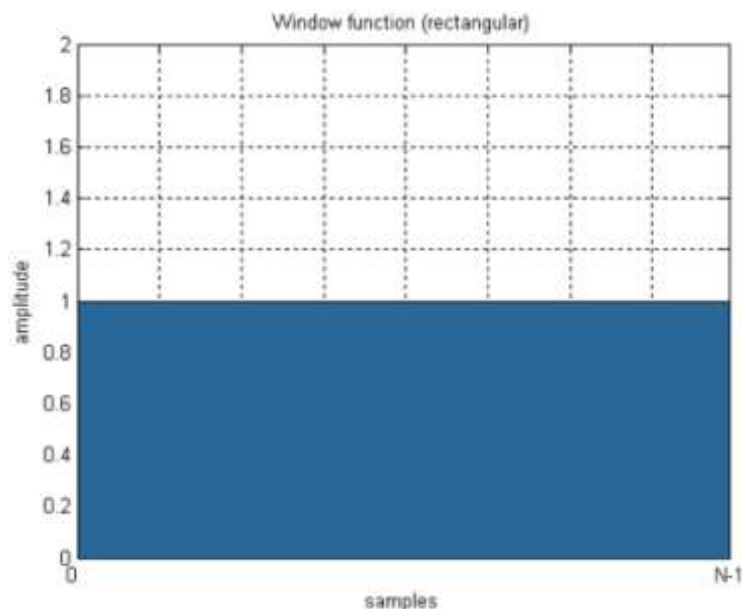
Kodek som utnyttjar transformkodning använder ofta en form av Fourieranalys vid namn *diskret cosinustransform* (eng. Discrete Cosine Transform, DCT) eller *modifierad diskret cosinustransform* (eng. Modified Discrete Cosine Transform, MDCT). Man använder dessa algoritmer eftersom de är mindre komplexa än vanlig Fouriertransform och därmed snabbare att beräkna. (Pohlmann 2011 s.361)

### 3.5.1 Fönsterfunktioner (Window functions)

När man gör en Fouriertransform på en signal görs det bara på en liten del av signalen åt gången. Man vet ingenting om hur signalen ser ut utanför det område som transformeras. Vi är dock tvungna att göra en *uppskattning* av hur signalen ser ut, och Fouriertransformen antar att signalen inom det transformerade blocket upprepar sig oändligt. Detta leder i de flesta fall till problem när man analyserar en ljudsignal eftersom den väldigt sällan upprepar sig på ett sådant sätt. Det uppstår således diskontinuiteter i ”skarvarna” mellan de analyserade blocken. (Bores, s.1.) Dessa diskontinuiteter ger upphov till oönskade artefakter i ljudsignalen. Artefakter uppstår dock inte om värdet vid början och slutet av det analyserade blocket råkar vara 0. Att detta är fallet är väldigt osannolikt, men det går att fabricera med hjälp av s.k. *fönsterfunktioner*.

Att göra en transformering av en signal utan att använda sig av ett fönster är i princip samma sak som att använda sig av ett *rektangulärt fönster* vid transformeringen. Värdena vid början och slutet av fönstret möjliggör uppkomsten av artefakter.

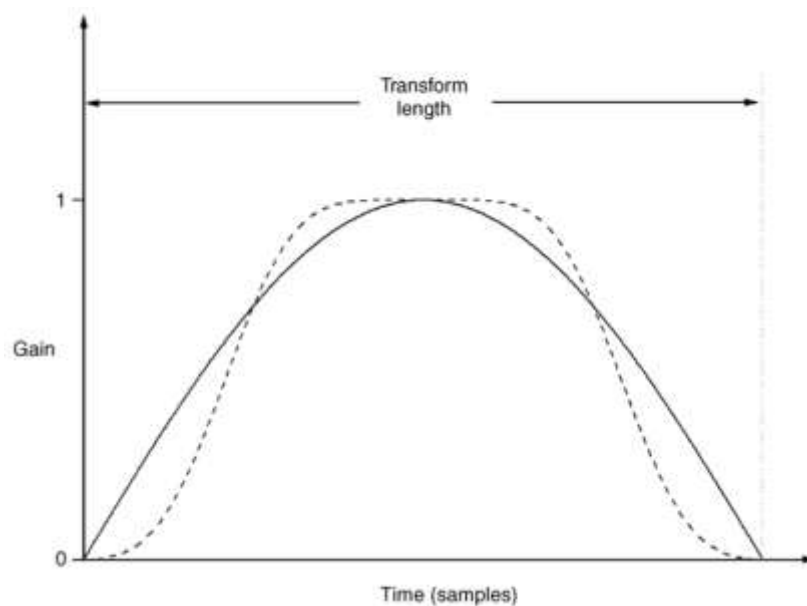
Figur 5 visar ett rektangulärt fönster. Fönstrets form ger upphov till artefakter i signalen eftersom värdet i början och slutet av fönstret inte är 0.



Figur 5. Rektangulärt fönster. (Wikipedia 2012b)

För att avhjälpa detta problem använder man sig alltså av s.k. fönster i en process som kallas *fönstring* (eng. windowing). Denna process äger rum före transformeringen. Processen går ut på att man multiplicerar den inkommande signalen med en funktion som går mot 0 i början och slutet av blocket som ska transformeras (se Figur 6). (Bores, s.2)

Användandet av fönster måste dock kompenseras vid avkodningsprocessen. Detta görs genom att två fönstrade block överlappas och resultatet adderas (fönstren är skapade för att möjliggöra detta) (Pohlmann 2011 s.368).



Figur 6. Två vanliga fönsterformer som används i MDCT.(Talbot-Smith 1999 s.71)

### 3.5.2 Transformkodningens funktionsprincip

Transformkodningen inleds med att filterbanken delar upp den inkommande ljudsignalen i block (i tidsdomänen). Blocken överlappar alltid varandra med ungefär 50%, vilket innebär att varje sampel förekommer i två block. (Talbot-Smith 1999 s.1-58) Detta görs för att förbättra upplösningen i tidsdomänen och för att jämna ut skillnader i den spektrala upplösningen mellan blocken. Blocken transformeras sedan, oftast med DCT eller MDCT. Vid uppspelning/avkodning av ljudströmmen invers-transformeras koefficienterna tillbaka till tidsdomänen. (Pohlmann 2011 s.360)

Man kan enkelt säga att Transformkodning går ut på att omvandla den inkommande ljudsignalen med någon typ av Fouriertransform. Den omvandlade signalen bearbetas och konverteras sedan vid avkodning tillbaka med hjälp av invers Fouriertransform.

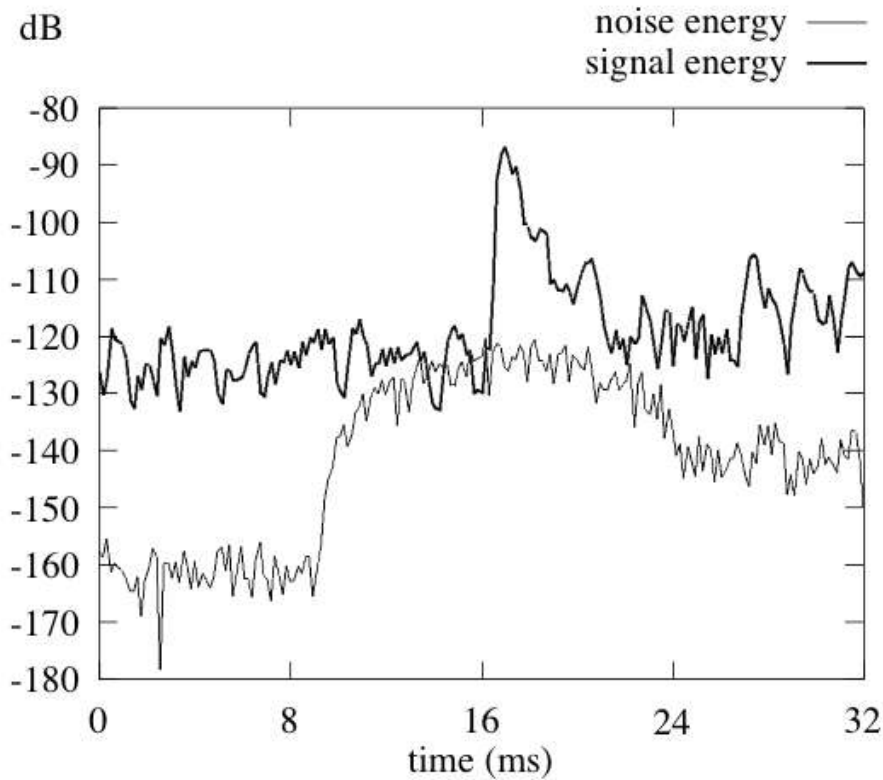
### 3.5.3 Föreko (pre-echo)

Begreppet *föreko* (eng. pre-echo) beskriver en ljudartefakt som kan uppstå vid digital ljudkomprimering om man inte använder en lämplig fönsterfunktion vid transformeringen (se kap. 3.5.1). Föreko tar formen av brus som förekommer i ljudsignalen *innan* själva ljudet som orsakar bruset hörs.

Transformkodek är särskilt drabbade av fenomenet eftersom de ofta använder sig av långa block i kodningsprocessen för att ge bättre frekvensnoggrannhet. Lång blocklängd lämpar sig väl för långsamt varierande och/eller tonala signaler där upplösningen i frekvensdomänen är viktig, men signaler som innehåller transienter, t.ex. en slaget av en kastanjett, behöver kort blocklängd där upplösningen i tidsdomänen är bättre. (Pohlmann 2011 s.362)

För att kringgå problemet och undvika föreko använder sig de flesta transformkodek av dynamiskt anpassande blocklängd. En annan teknik som används i vissa kodek är *temporal brusformning* (eng. temporal noise shaping, TNS). När en transient upptäcks använder sig TNS av en prediktiv kodningsmetod för att forma kvantiseringsbruset därefter.

Figur 7 visar ett exempel på föreko. Den undre kurvan (brus) överstiger nivån av ljudsignalen före en transient.



Figur 7. Exempel på föreko. (Karlheinz Brandenburg, s.7).

### 3.6 Huffman-kodning

Huffman-kodning är en form av förlustfri kodning som används inom både ljud och video (Pohlmann 2011 s.385). Kodningen går ut på att en signal analyseras för att upptäcka delar som upprepas. Dessa delar kodas om och ersätts med symboler (för att spara på utrymme). Symbolerna ersätts sedan med den ursprungliga signalen vid avkodning. (Fries 2000 s.149)

Huffman-kodningen är även konstruerad så att den använder kortare/enklare symboler för partier som upprepas oftare än andra. Man kan jämföra Huffman-kodning med att ersätta ord med siffror i en text. De ord som förekommer oftast ersätts med de lägsta siffrorna. (Fries 2000 s.149)



## 4 MPEG-1 LAYER III (MP3)

MPEG är en förkortning av *Moving Picture Experts Group*. Arbetsgruppen grundades av International Standards Organization (ISO) och International Electrotechnical Commission (IEC) år 1988 med målet att skapa internationella standarder för kodning av digitala media såsom ljud och video. (Fries 2000 s.141) MPEG har bl.a. hand om specifikationen av MP3-formatet. Totalt sett har MPEG hittills publicerat tre standarder: MPEG-1, MPEG-2 och MPEG-4.

MPEG-1 Layer III (MP3) hör - som namnet antyder - till MPEG-1 standarden och handhas således av MPEG. Formatet skapades ursprungligen av Fraunhofer Institut i Tyskland, Multimedia SA i Frankrike m.fl. och standarden publicerades 1993. (Hacker 2000 s.15)

MPEG-1 ljudstandarden baserar sig på tidigare algoritmer från datareduceringsalgoritmerna MUSICAM och ASPEC. Båda dessa kodek hade sina fördelar gentemot varandra och därför kom tekniker från båda två att utgöra basen från vilken MPEG-1 ljudstandard sedan utvecklades. (Pohlmann 2011 s.395)

MPEG-1 ljudstandard delas in i tre olika lager: *Layer I*, *Layer II* och *Layer III*. De olika lagren fungerar i grunden på samma sätt, men de blir gradvis mer komplexa. Layer I är det minst sofistikerade. Layer II är en vidareutveckling av Layer I och lite mer komplext. Layer III skiljer sig konceptuellt en aning från Layer I och II och är det mest sofistikerade av de tre. (Pohlmann 2011 s.395) Alla lager är bakåtkompatibla med de föregående.

Eftersom detta kapitel handlar om MPEG-1 Layer III kommer jag inte att gå närmare in på de två övriga lagren i MPEG-1 ljudstandard.

### 4.1 Teknik

I likhet med de flesta förlustbringande ljudkodek är MP3 ett *perceptuellt ljudkodek*. Det använder sig med andra ord av en psykoakustisk modell för omkodning av ljudsignalen. (Waugh, s.8)

Till skillnad från MPEG-1 Layer I och II är MPEG-1 Layer III (MP3) även ett transformkodek.

#### 4.1.1 Bithastighet och samplingfrekvens

Stöd för flera olika bithastigheter finns specificerade i MP3-standarden: 32, 40, 48, 56, 64, 80, 96, 112, 128, 160, 192, 224, 256 och 320 kbps. (Wikipedia 2012c) Man kan även välja om bithastigheten ska vara konstant (Constant Bit Rate, CBR) eller varierande (Variable Bit Rate, VBR). CBR betyder att bithastigheten är konstant genom hela den kodade filen. VBR innebär att bithastigheten kan variera enligt låtens komplexitet. Med andra ord kodas t.ex. partier innehållande många instrument och/eller effekter med hög bithastighet och tysta partier kodas med låg bithastighet. (Raissi 2002 s.8) Detta gör komprimeringen effektivare eftersom lägre bithastighet upptar mindre utrymme.

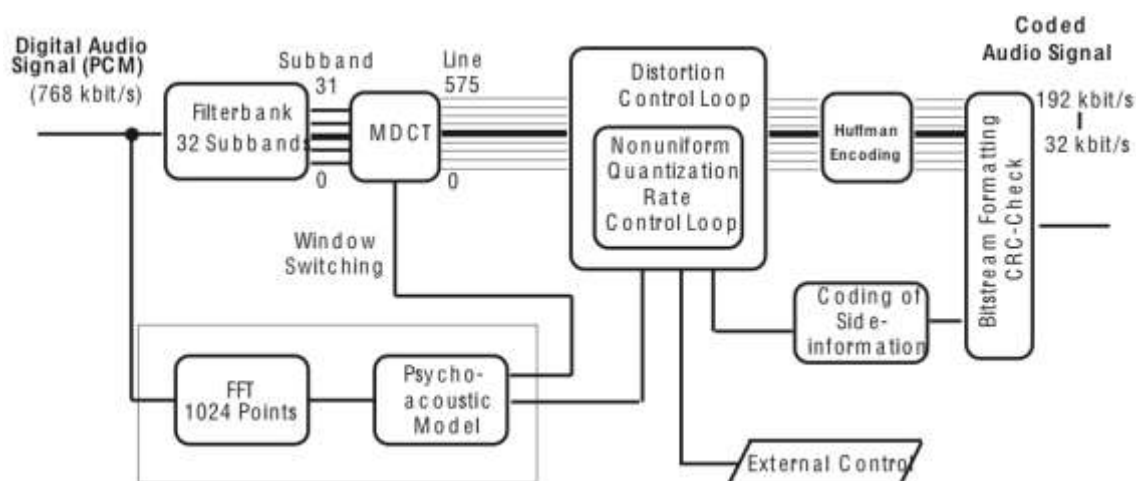
Stöd för olika samplingsfrekvenser finns även definierat i MP3-standarden, närmare bestämt samplingsfrekvenserna 32, 44,1 och 48 kHz

#### 4.1.2 Stereolägen

MP3 stöder ett flertal olika stereolägen:

- **Simple Stereo.** Detta läge använder sig av två separata kanaler för kodning av stereoljud.
- **Joint Stereo.** Detta läge använder sig av s.k. middle/side stereo, vilket innebär att en kanal innehåller information som är lika för båda kanalerna och den andra kanalen innehåller skillnaden.
- **Intensity Stereo.** I detta läge kodas enbart information som anses viktig för stereobilden. Intensity Stereo ger högst komprimering, men stereobilden blir lidande vid låga bithastigheter. (Fries 2000 s.149)

### 4.1.3 MP3-kodarens funktionsprincip



Figur 8. Blockdiagram av en MP3-kodare. (Karlheinz Brandenburg, s.4)

Till att börja med läses ett block bestående av 1152 sampel från den inkommande ljudsignalen. Detta block kallas en *frame* och hela MP3-filen delas upp på detta sätt. Varje frame kan avkodas separat och är således en ”fristående del” i en MP3-fil. (Pohlmann 2011 s.396) Denna frame delas sedan in i 32 subband med hjälp av en filterbank. Innehållet i varje subband transformeras till frekvensdomänen med MDCT som sedan ger vidare maximalt  $32 \times 18 = 576$  sampel. (Pohlmann 2011 s.404) Eftersom MDCT inte tillämpas direkt på signalen utan först går genom denna filterbank säger man att MP3 använder sig av en *hybrid* MDCT (Wikipedia 2012d).

Samtidigt analyseras ljudsignalen av en 1024-punkts FFT som transformerar information från tidsdomänen till frekvensdomänen. Detta går sedan igenom en psykoakustisk modell som emulerar den mänskliga hörseln. Information som skapas av den psykoakustiska modellen används av MDCT för att optimera kodningsprocessen. (Raissi 2002 s.21)

Värdena från MDCT skickas till en *icke-likformig kvantiserare* (eng. non-uniform quantizer). Här körs informationen genom två slingor: en slinga för *distorsionsreglering* (eng. distortion control loop) och en slinga för *flödeskontroll* (eng. rate control loop). (Raissi 2002 s.19)

Flödeskontrollen har hand om kvantisering av signalen och i MP3 används *dynamisk kvantisering* vilket innebär att kvantiseringsnivån kan justeras efter behov. Samplen

kvantiseras med stegvis lägre upplösning ända tills värdena kan kodas med en lämplig Huffman kodningstabell. (Raissi 2002 s.22)

För att hålla kvantiseringsbruset på en acceptabel nivå kontrolleras kvantiseringsnivån av distorsionsregleringen som strävar efter att hålla kvantiseringsbruset under maskeringströskeln. (Pohlmann 2011 s.407)

#### **4.1.4 Fönster**

I MP3 överlappar fönstren varandra med 50%. Två olika fönsterstorlekar används för att motverka föreko vid transienter. (Watkinson 2001 s.313)

## 5 MPEG-2 ADVANCED AUDIO CODING (AAC)

Advanced Audio Coding (AAC) hör till MPEG-2 standarden och är ämnat att vara efterföljaren till MP3. MPEG-2 ljudstandarden är på många sätt lik MPEG-1 och använder i flera fall samma kodnings- och avkodningsalgoritmer. (Pohlmann 2011 s.417) Diverse test som gjordes åt 1994 visade dock att man genom att skippa full bakåtkompatibilitet med MPEG-1 och istället introducera nya kodningsalgoritmer kunde göra ljudkodningen mycket mera effektiv. Detta gav upphov till att AAC skapades och lades till i MPEG-2 standarden. (Karlheinz-Brandenburg, s.3) AAC är därför inte bakåtkompatibelt med MP3. En MPEG-1-avkodare kan inte avkoda en AAC-kodad ljudfil.

### 5.1 Teknik

I likhet med MP3 är AAC ett perceptuellt ljudkodek som använder en psykoakustisk modell för att emulera den mänskliga hörseln. Den psykoakustiska modell som används i AAC är baserad på den som används i MP3 (Pohlmann 2011 s.421). Vid tiden när AAC lanserades var det ämnat som en ljudstandard som gav den högsta möjliga ljudkvaliteten med hjälp av nyutvecklade tekniker. (Watkinson 2001 s.316)

AAC består av tre profiler: *Main*, *Low Complexity (LC)* och *Scalable Sampling Rate (SSR)*. Main Profile är den mest sofistikerade profilen av de tre. LC-profilen kringgår vissa delar av ljudkodeket och begränsar andra delar för att det ska fungera mer resurssnålt (Watkinson 2001 s.316). SSR-profilen delar in ljudsignalen i fyra likvärdiga frekvensband som alla har en egen, separat bitström och avkodare vilket gör det möjligt att variera bandbredden på utsignalen (Pohlmann 2011 s.420). LC är den vanligaste profilen och används t.ex. när man kodar AAC-filer med iTunes.

#### 5.1.1 Bithastighet och samplingfrekvens

MPEG-2 AAC stöder bithastigheter från 48 kbps till 576 kbps. Samplingsfrekvenser som stöds som standard är 32, 44,1 och 48 kHz, men stöd för andra samplingsfrekvenser mellan 8 – 96 kHz finns. (Pohlmann 2011 s.420)

### 5.1.2 Stereolägen

AAC stöder mono (1/0), stereo (1/1) och ett flertal möjliga kombinationer för flerkanaligt ljud (upp till 3/2 + 1). Totalt finns det stöd för upp till 48 ljudkanaler. (Pohlmann 2011 s.420)

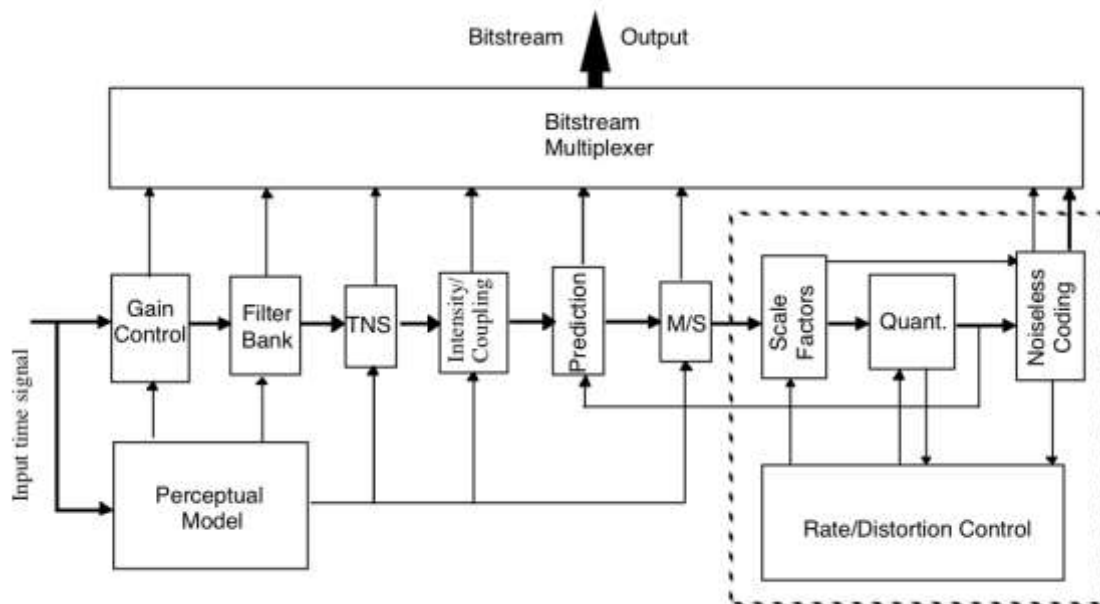
### 5.1.3 Skillnader och förbättringar gentemot MP3

AAC skiljer sig på flera punkter från MP3 och innehåller även ett flertal förbättringar. Alla dessa små förbättringar torde ge AAC samma ljudkvalitet som MP3 vid 70% av bithastigheten. (Karlheinz-Brandenburg, s.6)

AAC innehåller bl.a. dessa förbättringar och nyheter:

- **Högre frekvensupplösning.** Antalet frekvensband från MDCT har ökats till 1024 från MP3:s 576.
- **Förbättrad joint-stereo kodning.** Mid/side- och Intensity-läget för stereokodning har förbättrats och är betydligt mer flexibelt.
- **Förbättrad Huffman-kodning.**
- **Temporal brusformning** (eng. temporal noise shaping, TNS) är en teknik som gör det möjligt att forma kvantiseringsbruset i varje fönster (Pohlmann 2011 s.423). Tekniken används främst vid kodning av transienter för att forma kvantiseringsbruset enligt deras vågform och därmed undvika uppkomsten av föreko. (Watkinson 2001 s.319)
- **Förbättrad teknik för byte av blockstorlek.** AAC använder sig av en ren MDCT filterbank istället för den hybrid som används i MP3. För korta block ligger impulssvaret hos AAC på 5,3 ms jämfört med MP3:s 18,6 ms. Detta förebygger uppkomsten av föreko. (Karlheinz-Brandenburg, s.7)

### 5.1.4 AAC-kodarens funktionsprincip



Figur 9. Blockdiagram över AAC-kodaren. (Karlheinz-Brandenburg, s.6)

Som tidigare nämnts använder sig AAC av en ren MDCT filterbank till skillnad från den hybrida filterbank som används i MP3. Fönstren överlappar varandra med 50%. Filterbanken använder sig av en blocklängd på 1024 för stadiga signaler och 128 för transienter. När kort blocklängd väljs bibehålls den i 8 iterationer för att undvika fasfel ( $128 \times 8 = 1024$ ). (Pohlmann 2011 s.420)

I AAC finns två olika fönstertyper tillgängliga för MDCT: Sinusformat fönster eller Kaiser-Bessel-derived (KBD) fönster. Båda fönstertyperna har sina för- och nackdelar och är ämnade för olika typer av signaler. Kodaren kan välja optimal fönstertyp baserat på signalen. (Watkinson 2001 s.318)

Efter Filterbanken följer *temporal brusformning* som presenterats ovan. Tack vare denna teknik kan kodaren oftare använda sig av lång blocklängd och behöver inte lika ofta byta till kort blocklängd för att motverka föreko. Temporal brusformning står för en stor del av den förbättrade effektiviteten hos AAC (Watkinson 2001 s.319).

Efter temporal brusformning följer en modul med namnet Intensity/Coupling. Denna modul är valbar och används främst vid låga bithastigheter. (Watkinson 2001 s.320)

En prediktor tillämpas för att rensa bort överflödigt information i stationära signaler förekommande i långa block. Prediktorn beräknar värden från två tidigare block för att få fram ett förutsagt värde för det aktuella blocket. Det förutsagda värdet subtraheras från det verkliga värdet för att ta fram en rest som sedan kvantiseras. Alla ”rester” grupperas i skalfaktorband som emulerar hörselns kritiska band. En algoritm arbetar sedan med att avgöra om prediktion ska aktiveras inom vissa band eller om den ska användas över huvudtaget, beroende på huruvida det optimerar kodningen eller inte. (Pohlmann 2011 s.421)

Efter prediktorn följer två allokeringslingor. En inre slinga kvantiserar signalen och använder sig av Huffman-kodning för att optimera kvantiseringsprocessen. En yttre slinga använder sig av skalfaktorer för att vid behov förstärka skalfaktorband med målet att motverka hörbarheten av kvantiseringsbrus. Om distorsionen är för hög inom ett skalfaktorband förstärks det (vilket ökar bithastigheten). Den yttre slingan analyserar även signalen och beräknar distorsionen, vilken sedan jämförs med den mängd distorsion som tillåts av den psykoakustiska modellen. Det bästa resultatet sparas. (Pohlmann 2011 s.422)

Till slut utförs Huffman-kodning. Ett flertal Huffman kodböcker finns tillgängliga för att optimera processen. (Pohlmann 2011 s.423)



## 6 DEN KLASSISKA GITARREN

Alla känner förmodligen till den klassiska gitarren som ett 6-strängat, akustiskt instrument och många har förmodligen stiftat bekantskap med den antingen genom skolans musikundervisning eller av eget intresse. Gitarren är ett av världens populäraste instrument (Pinksterboer 2001 s.2).

Den klassiska gitarren brukar även kallas *spansk gitarr* (eftersom den fick sin nuvarande form i Spanien) eller *nylonsträngad gitarr* (eftersom flera av gitarrens strängar är gjorda av nylon). Nylonsträngarna ger gitarren ett mjukt och behagligt ljud. (Pinksterboer 2001 s.2)

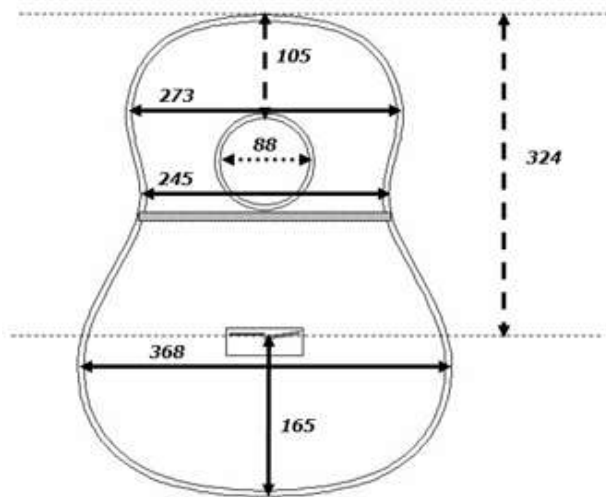
### 6.1 Historia

Den klassiska gitarren har en lång historia och otaliga förfäder, men den kanske mest kända förfadern är *lutan*. (Pinksterboer 2001 s.94) Den moderna klassiska gitarren fick sin nuvarande form någon gång mellan åren 1850 – 1860 av spanjoren Antonio de Torres Jurado. Han gav gitarren en större kropp och förbättrade det ribbmönster som sitter under gitarrens lock. Ribbmönstret stöder gitarrens konstruktion och påverkar även till viss grad dess ljud. (Pinksterboer 2001 s.95)

### 6.2 Storlek och uppbyggnad

Gitarren består av tre huvudsakliga delar: ett huvud, en hals och en kropp. Kroppen är ihålig och fungerar som en resonanslåda som förstärker ljudet.

Alla klassiska gitarrer har i stort sett samma storlek, men den kan variera lite grann mellan olika märken och modeller. Det finns även mindre modeller som är särskilt anpassade för barn.



Figur 10. Exempel på mått på en gitarrkropp. (Eendebak 2011)

Man kan säga att gitarren består av tre huvudsakliga delar, men om man går in på djupet finns det givetvis mer än så. De viktigaste delarna finns illustrerade på bilden nedan. Längst till vänster syns gitarrens kropp. Sedan följer halsen och längst till höger huvudet.



Figur 11. Den akustiska gitarren och dess delar. (Fall 2011)

De flesta klassiska gitarrer har en mensur på 65 cm. Mensuren är avståndet mellan gitarrens sadel och stall, inom vilket strängarna vibrerar.

### 6.3 Ljudegenskaper

En av gitarrens största nackdelar (beroende hur man ser på saken) är dess ljudstyrka. Den är förhållandevis lågmäld och i t.ex. ensemblespel ”försvinner” den lätt bland andra instrument. En cello, fiol eller flöjt överröstar gitarren ganska enkelt. (Hallgren 2011).

Gitarrens kropp fungerar som en slags högtalare. När strängarna vibrerar överförs vibrationen till kroppen vilket leder till att ljudet förstärks. Kroppen ger även ett antal *resonanser* som ger varje gitarr en unik klang. Dessa resonanser är i praktiken frekvenser som svänger mer än andra frekvenser inuti gitarrkroppen. (Hallgren 2011)

Den klassiska gitarren spelas i allmänhet med fingrarna. Man använder sig inte av plektrum. Detta bidrar även till den låga ljudnivån. Det ger också ett varmare och mjukare ljud än hos en stålsträngad gitarr som ofta spelas med hjälp av ett plektrum. (Huber et al. 2005 s. 154)

### 6.4 Frekvensområde

Som tidigare nämnts har gitarren sex strängar. I allmänhet är strängarna stämde i tonerna **E<sub>2</sub>, A<sub>2</sub>, D<sub>3</sub>, G<sub>3</sub>, B<sub>3</sub> och E<sub>4</sub>** (från lägsta till högsta), vilket motsvarar frekvenserna **82, 100, 147, 196, 247 och 330 Hz**. Dessa frekvenser är *grundtonen* för varje sträng. Det som dock skapar gitarrens unika ljud är de *övertoner* som ljuder tillsammans med grundtonen. Dessa övertoner är det som särskiljer olika instrument från varandra. Varje överton förändras också beroende på hur och var man spelar på gitarren, t.ex. om man knäpper försiktigt eller hårt på en sträng. Det finns många faktorer som påverkar dessa övertoner, t.ex. typen av strängar, instrumentets ålder, typen av trä osv. (Billington 1999)

## 7 ANALYS AV ETT INSPELAT STYCKE KLASSISK GITARR

För att genomföra min analys kommer jag att använda några förinspelade stycken med klassisk gitarr tagna från en kommersiellt tillgänglig CD-skiva. Jag överför de valda styckena till datorn och sparar dem i Wave-format, vilket är helt och hållet förlustfritt. Jag omkodar därefter denna originalfil med hjälp av de kodek jag valt att analysera.

Jag har försökt välja ett gitarrstycke som representerar den klassiska gitarrens egenskaper så bra som möjligt. Jag har valt stycket ”El Colibri” komponerat av Julio S. Sagreras. I den version av stycket jag använt framförs det av den svenska gitarristen Per Skareng. Stycket är taget från skivan ”El Colibri” (Caprice Records 1989).

### 7.1 Överföring av gitarrstycke från CD-skiva till WAV-Format

När jag skapar en Wave-kopia av ljudfilen från CD-skivan vill jag givetvis att den skall vara helt identisk med originalet. För att säkerställa detta använder jag mig av programmet *Exact Audio Copy* (hemsida: <http://www.exactaudiocopy.de>), vilket är ett kraftfullt verktyg för kopiering och kodning av CD-skivor och ljudfiler. Programmet gör en noggrann överföring från CD-skivan och skapar därefter en logg av processen. I denna logg kan man se om det uppstått eventuella fel vid överföringen.

I *Exact Audio Copy* kan man ställa in en hel del parametrar för den kopia man skapar. Jag väljer att skapa en Wave-fil med 16-bitars upplösning och 44,1 kHz samplingfrekvens, vilket är samma som på en CD-skiva.

### 7.2 Kodning av ljudfiler

När jag nu har en originalfil att utgå ifrån är det dags att skapa de övriga ljudfilerna. Jag väljer ut ett populärt kodek för vardera format och skapar därefter omkodade versioner av ljudfilen med hjälp av dessa.

Många kodek ger användaren möjlighet att definiera olika parametrar som inverkar på kodningsprocessen och därmed den slutgiltiga kvaliteten på den omkodade ljudfilen. För att göra testet konsekvent valde jag att enbart definiera de mest grundläggande

parametrarna för varje kodek, vilket är samplingfrekvens och bithastighet. Jag valde att skapa två versioner av varje ljudfil med följande parametrar:

- Version 1: 44,1 kHz, 128 kbps konstant bithastighet.
- Version 2: 44,1 kHz, 256 kbps konstant bithastighet.

Att koda en ljudfil med konstant bithastighet är relativt ineffektivt jämfört med kodning i variabel bithastighet, men för att hålla testet konsekvent och underlätta jämförelserna använder jag konstant bithastighet.

### **7.2.1 MP3**

För att koda om ljudfilen till MP3-format använder jag mig av MP3-kodeket *LAME*. LAME är ett öppenkällkod-projekt och har kommit att bli ett av de allra populäraste MP3-kodeken. LAME använder sig av en psykoakustisk modell vid namn *GPSYCHO* som sägs ha en hel del förbättringar jämfört med den psykoakustiska modell som hör till MPEG-standarden. LAME-projektets hemsida: <http://lame.sourceforge.net/>.

### **7.2.2 AAC**

För kodning i AAC använder jag mig av programmet *iTunes*. iTunes är ett program utvecklat av Apple för hantering och uppspelning av musik-, ljud- och videofiler. iTunes ger även användaren möjlighet att köpa musik- och videofiler via nätet, importera musik från CD-skivor, katalogisera musik mm. För importering och kodning av musik tillhandahåller iTunes ett flertal olika kodek, däribland AAC.

### **7.2.3 Filstorlekar**

Efter import- och kodningsprocessen har jag följande filer som presenteras i tabell 1 nedan. Filstorleken för de kodade ljudfilerna är ganska lika i förhållande till bithastigheten de är kodade i. AAC är en aning större i storlek än MP3, men skillnaden är i båda fallen under 80 kB.

Format	Storlek (MB)	Storlek (byte)
Wave	14,5 MB	14 528 348 byte
MP3 (128 kbps)	1,3 MB	1 318 660 byte
MP3 (256 kbps)	2,6 MB	2 637 321 byte
AAC (128 kbps)	1,4 MB	1 378 496 byte
AAC (256 kbps)	2,7 MB	2 717 236 byte

Tabell 1. Ljudfilerna och deras storlek.

## 7.3 Analys

För att analysera de olika ljudfilerna använder jag mig av programmet *Adobe Audition* för frekvensanalys och programmet *iZotope RX* för analys i tidsdomänen och den spektrala analysen. Adobe Audition är ett program för inspelning, redigering och korrigering av ljud och innehåller många kraftfulla verktyg för ändamålet. Programmet gick tidigare under namnet *Cool Edit Pro* (Adobe 2012). *iZotope RX* är främst ämnat för redigering, korrigering och analys av ljud.

### 7.3.1 Analys i tidsdomänen

När en ljudfil visas i ett ljudinspelnings och -redigeringsprogram visas vågformen oftast representerad i tidsdomänen, dvs. ljudet visas i tid (x-led) och amplitud (y-led). I denna analys väljer jag helt enkelt en kort del i musikstycket och jämför sedan denna del mellan de olika kodade filerna samt originalfilen. För att göra denna analys använder jag mig av programmet *iZotope RX*.



Figur 12. WAV-filen av hela "El Colibri" visas i tidsdomänen i *iZotope RX*.

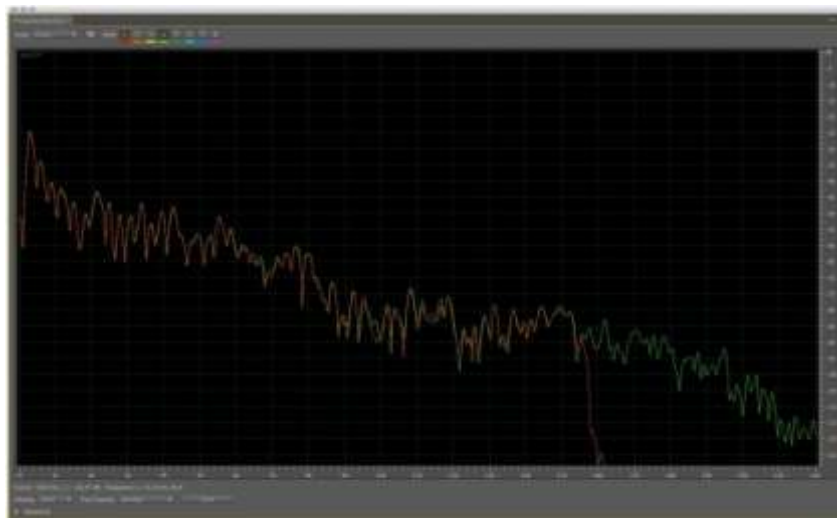
### 7.3.2 Frekvensanalys

Vid frekvensanalysen jämför jag de olika ljudfilerna med varandra vid exakt samma punkt i tidsdomänen. Jag ställer in tidslinjen i Adobe Audition att visa tiden i *sampel* istället för i minuter, sekunder och millisekunder. Sedan väljer jag motsvarande sampel i alla ljudfiler. I fallet med MP3-filen är jag tvungen att addera 2256 sampel i tidsdomänen för att få fram motsvarande sampel som i de övriga filerna, detta på grund av MP3-kodekets funktion. MP3-kodeket – i detta fall LAME-kodeket – lägger till en s.k. *header frame* i början som innehåller information om filen (en frame är som tidigare nämnts 1152 sampel lång). Vid kodningen av MP3-filen tillkommer en *kodningsfördröjning* (eng. encoding delay) som är 576 sampel lång. Vid avkodningen tillkommer ännu en *avkodningsfördröjning* (eng. decoding delay) som är 528 sampel lång. Totalt  $1152 + 576 + 528 = 2256$  sampel.

Värt att notera är att informationen förändrar sig ytterst lite mellan sampel, så ett kast på några sampel mellan jämförelsepunkterna i filerna är försumbar.

Vid frekvensanalysen använder jag mig av en FFT-analys med FFT-storleken 1024. Fönstret som används är av typen Blackmann-Harris. Dessa är standardinställningar i Adobe Audition.

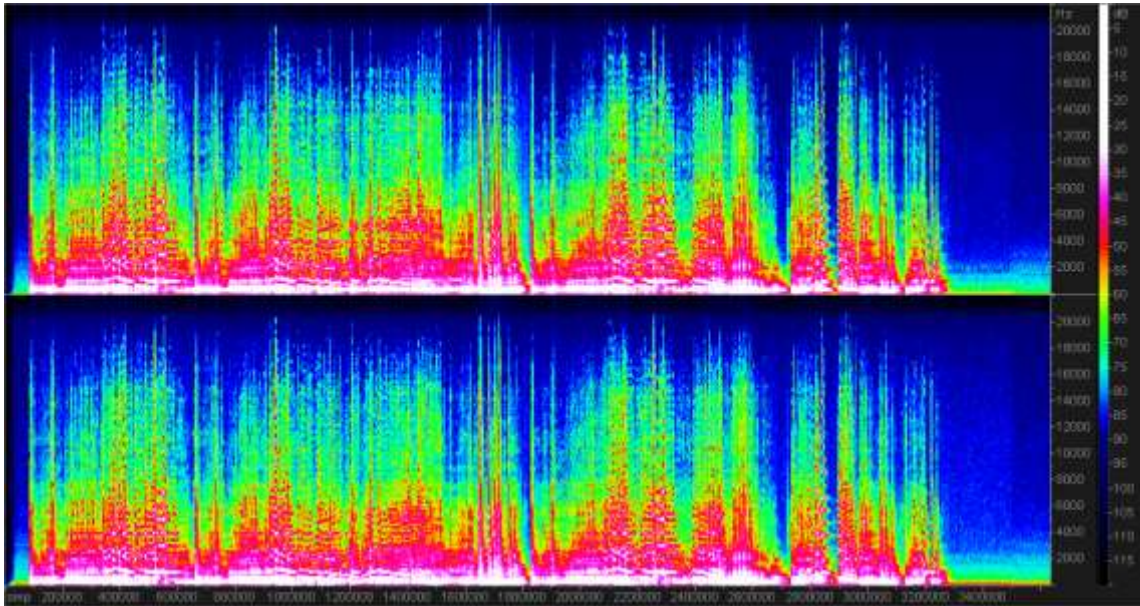
När jag fått fram jämförelsepunkterna kan jag sedan jämföra de olika vågformerna med varandra och se om det finns skillnader. På figuren nedan ses en jämförelse av en WAV-fil och en MP3-fil kodad med 128 kbps upplösning.



Figur 13. Jämförelse mellan en WAV-fil och en MP3-fil. WAV grön, MP3 röd.

### 7.3.3 Spektral analys

Med hjälp av den spektrala vyn i programmet iZotope RX har man möjlighet att inspektera vågformen visuellt. I den spektrala vyn visas musikstyckets längd i x-led och frekvensen i y-led. Ju starkare färg desto mera energi finns det i frekvensområdet. Med hjälp av denna vy kan man jämföra de olika filerna och se hur energinivåerna skiljer sig i olika frekvensområden eller om det finns andra synbara artefakter.



Figur14. Spektral vy i iZotope RX. Vänster och höger ljudkanal.

I Figur 14 ovan ses ett exempel av en spektral vy av en ljudfil. I detta exempel ser man att det förekommer mest energi i området 50 Hz – 1 kHz. Därefter avtar energimängden för att sedan avta helt vid ca 22 kHz.

I iZotope RX kan man enkelt zooma in på olika delar av den spektrala representationen av ljudfilen, vilket underlättar sökandet efter eventuella skillnader mellan filerna.

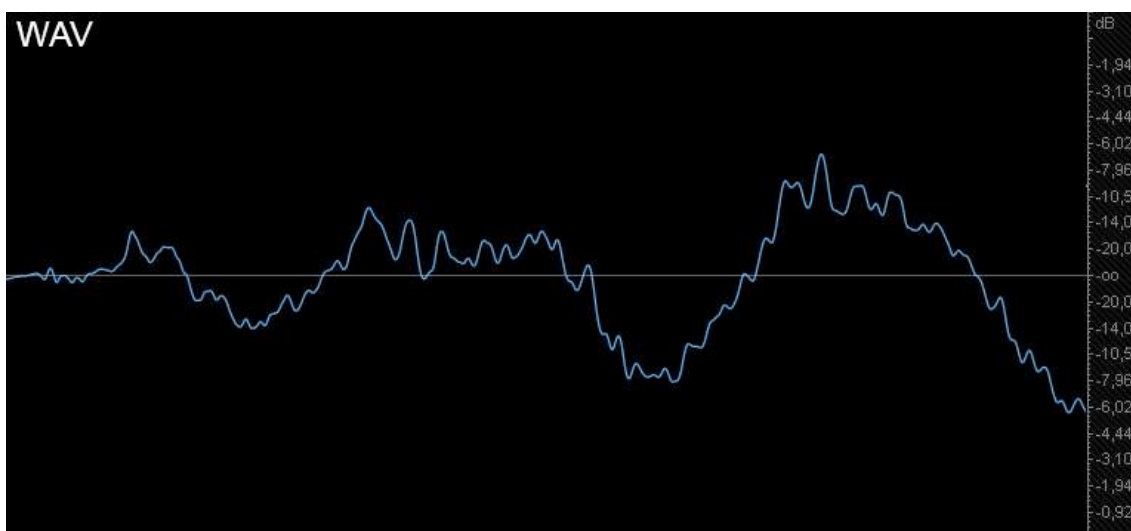


## 8 RESULTAT

I detta kapitel presenteras resultaten av de jämförelser jag gjort mellan originalet och de omkodade filerna. Först presenteras resultaten av analys i tidsdomänen, sedan frekvensanalysen och till sist resultaten av den spektrala analysen.

### 8.1 Analys i tidsdomänen

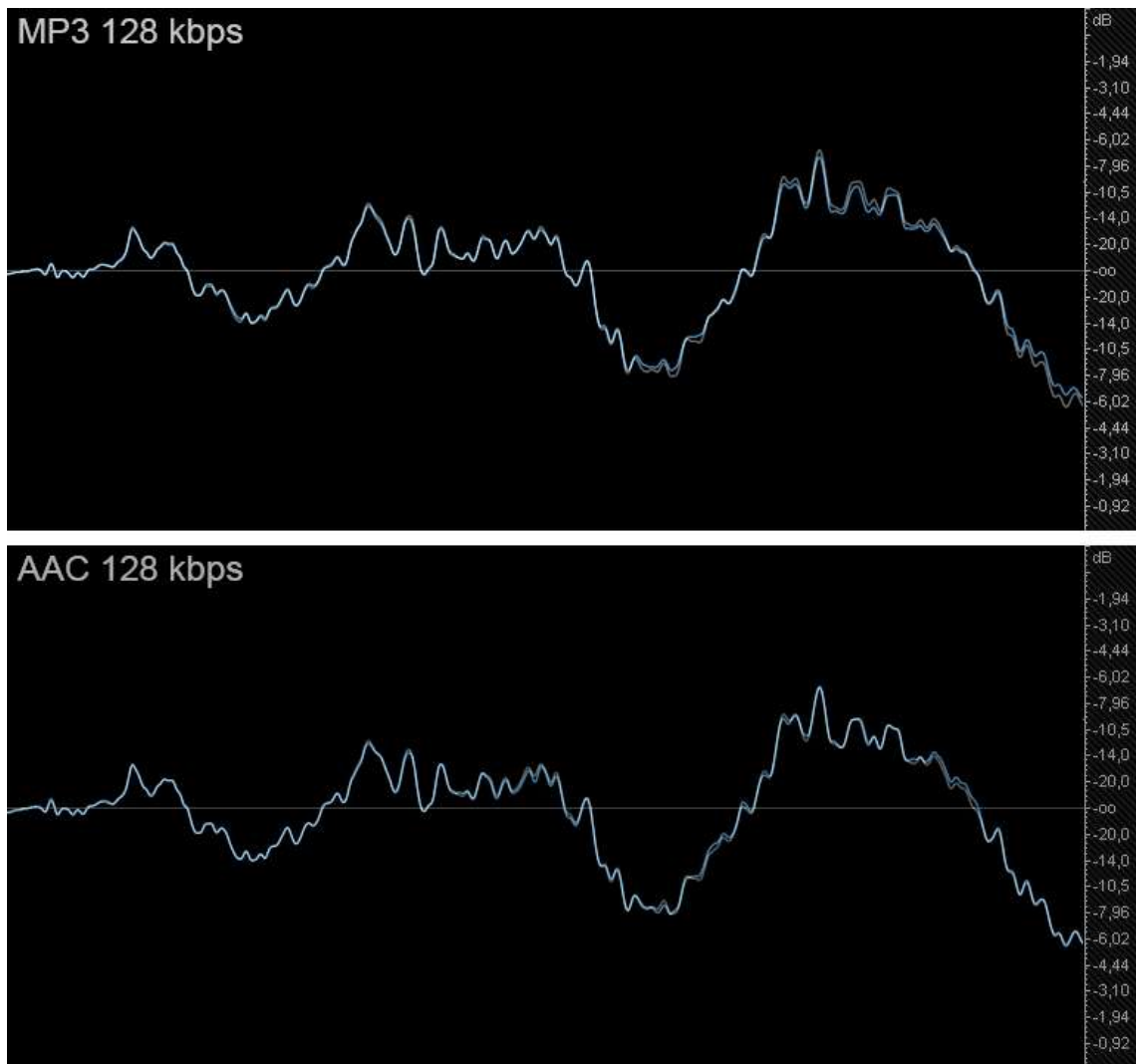
Vid analys i tidsdomänen kan en del skillnader uskiljas. I det stora hela är dock dessa skillnader små. De största skillnaderna förekommer i filerna som är kodade i 128 kbps upplösning, vilket kanske är föga förvånande.



Figur 15. Den ursprungliga WAV-filen. Vänster kanal.

I Figur 15 ovan ses vänster kanal av en utvald del i musikstycket. Delen i fråga är 490 sampel lång, vilket vid en samplingshastighet på 44100 sampel per sekund motsvarar ca 11,1 ms ljud. En väldigt kort ljudsnutt alltså. Själva ljudet är i huvudsak ett anslag på tonen A#<sub>3</sub>.

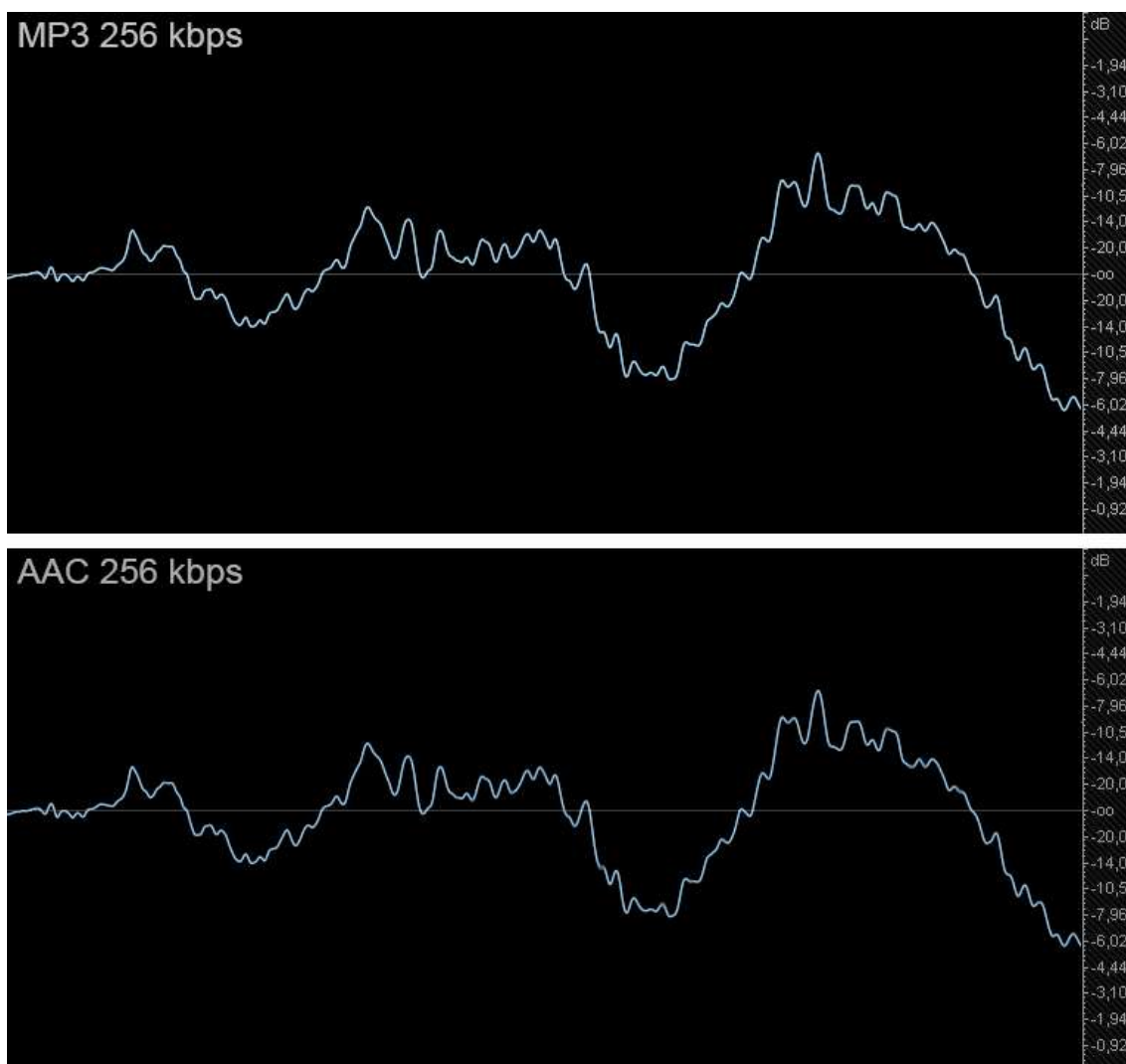
### 8.1.1 128 kbps



Figur 16. MP3 och AAC 128 kbps.WAV visas som mörkgrå.

I Figur 16 ovan ses en jämförelse mellan MP3- och AAC-filen kodade i 128 kbps. Den ursprungliga WAV-filens vågform visas som mörkgrå. Amplituden skiljer sig mer från originalet i MP3-filen medan AAC-filen är ganska lik originalet. Totalt sett är dock skillnaderna små.

### 8.1.2 256 kbps

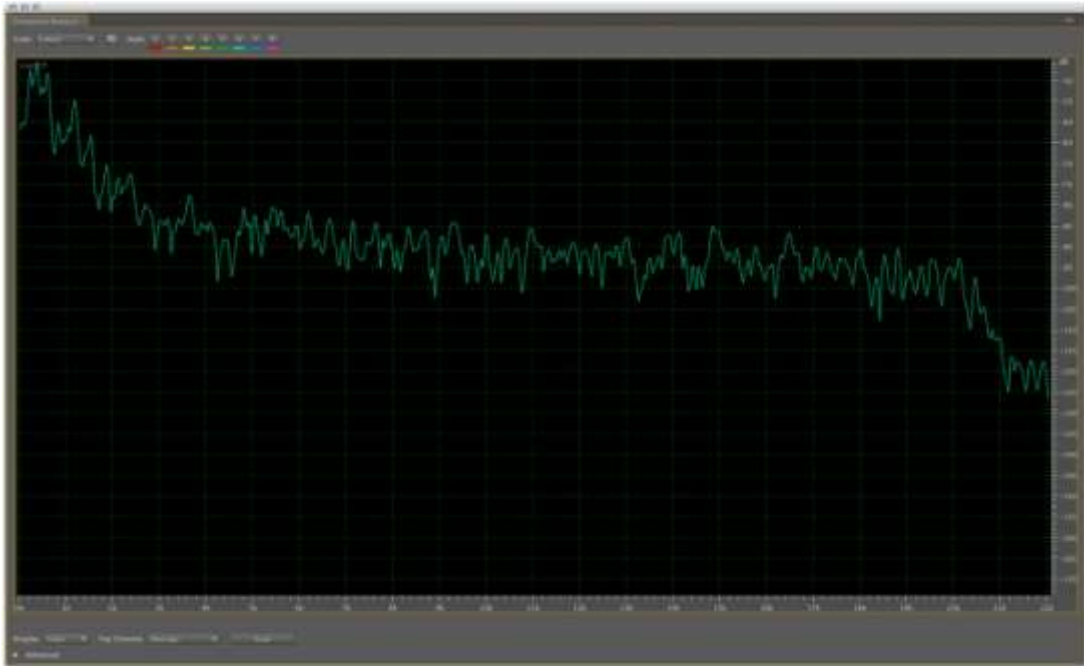


Figur 17. MP3 och AAC 256 kbps. WAV-kurvan syns knappt p.g.a. likheterna.

Filerna som är kodade i 256 kbps är ytterst lika originalet. Endast små skillnader förekommer på vissa ställen, men de är så små att kurvorna knappt går att särskilja ifrån varandra. Högre bithastighet ger alltså en klart bättre representation av originalet i denna analys.

## 8.2 Frekvensanalys

Skillnader i vågformen mellan de olika kodeken kan definitivt påvisas. Skillnaderna jämfört med originalfilen är störst hos de ljudfiler som kodats i 128 kbps upplösning och vid 256 kbps upplösning är skillnaderna betydligt mindre.



Figur 18. Den ursprungliga WAV-filen av stycket "El Colibri".

I Figur 18 ovan ses den ursprungliga WAV-filen. Filen innehåller information ända upp till ca 20 kHz där den sedan avtar i amplitud. Som tidigare nämnts sträcker sig bandbredden för den mänskliga hörseln i regel från 20 Hz till 20 kHz.

### 8.2.1 128 kbps



Figur 19. "El Colibri". MP3 128 kbps (röd), AAC 128 kbps (grön).

I Figur 19 ses en jämförelse mellan 128 kbps MP3 (röd) och 128 kbps AAC (grön). Frekvenskurvan för MP3-filen har en del utstickande delar och skiljer sig ganska markant från AAC-filen.

AAC-filen har en väldigt precis avklippning strax över 16,5 kHz medan avklippningen hos MP3-filen är betydligt mer diffus. MP3-filen innehåller även en hel del rippel efter avklippningen, men detta ligger under tröskeln för hörbarhet (under -155 dB). Motsvarande rippel i AAC-filen är så gott som obefintligt.

Vidare jämförelser visar att AAC-filen är mer lik originalet än MP3-versionen vid 128 kbps bithastighet.

### 8.2.2 256 kbps



Figur 20. "El Colibri". MP3 256 kbps (röd), AAC 256 kbps (grön).

I Figur 20 ovan ses en jämförelse mellan en 256 kbps MP3-fil (röd) och en 256 kbps AAC-fil grön. Mätningen har gjorts på exakt samma ställe som hos 128 kbps filerna. Frekvenskurvorna är betydligt mer lika varandra och avklippningen hos båda filerna ligger högre upp i frekvensområdet. Avklippningen hos AAC-filen ligger strax över 20

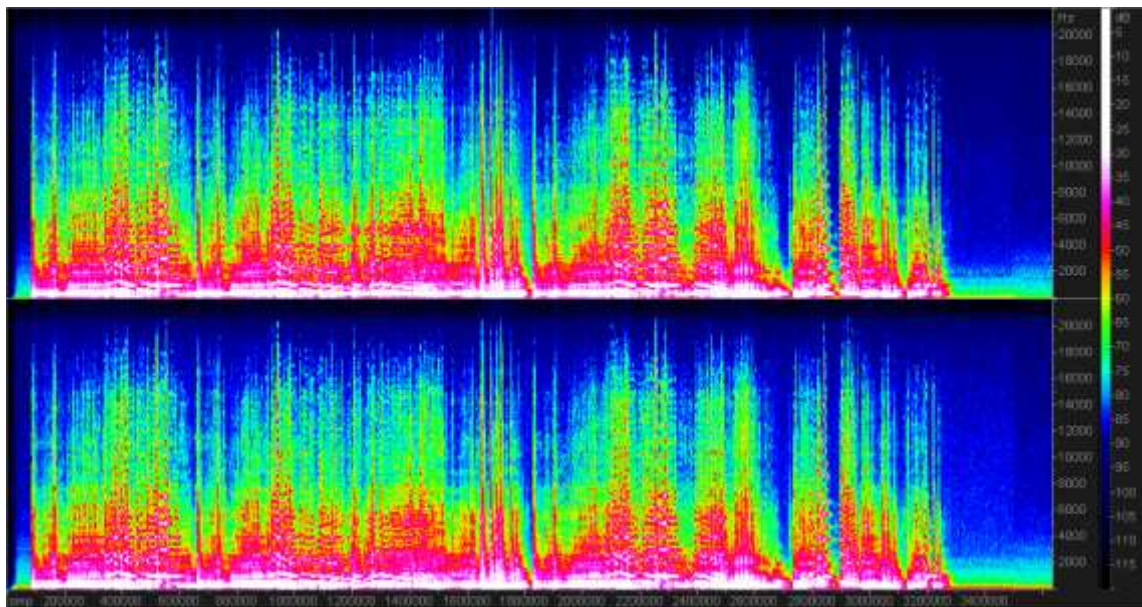
kHz och hos MP3-filen ligger den mellan 19,5 – 19,7 kHz. MP3-filen innehåller samma rippel efter avklippningen som i 128 kbps filen.

Jämförelser visar att AAC-filen även här är mer lik originalet än MP3-filen. I det stora hela är dock skillnaderna dem sinsemellan betydligt mindre än i filerna kodade med 128 kbps. Detta är föga förvånande eftersom en högre bithastighet i regel innebär högre ljudkvalitet.

Faktum är dock att man även vid högre bithastigheter kan se skillnader mellan MP3- och AAC-filen, vilket är vad jag eftersträvade att upptäcka.

### 8.3 Spektral analys

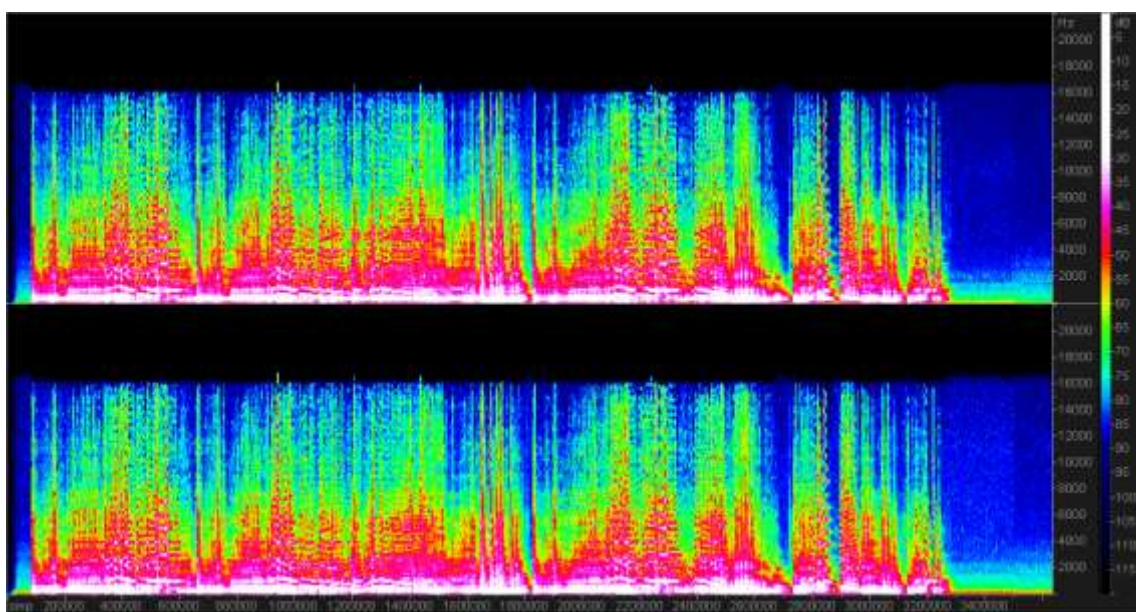
Med hjälp av denna analys kan man visuellt se på de olika ljudfilerna för att upptäcka eventuella skillnader. Om färgintensiteten skiljer sig mellan filerna tyder det på förändringar i ljudbilden.



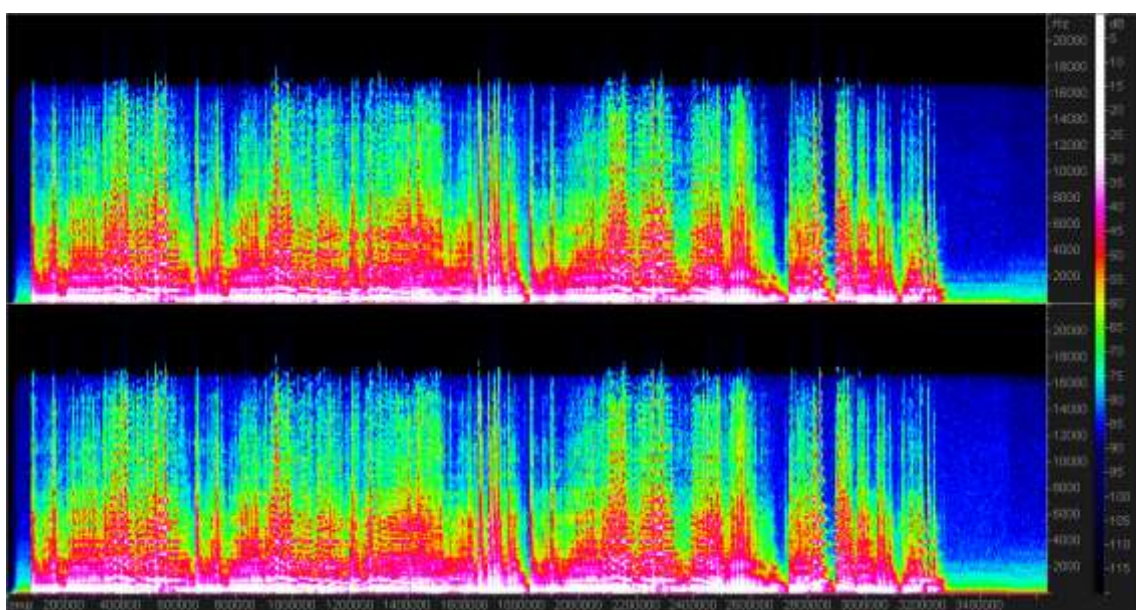
*Figur 21. Spektral vy av den ursprungliga WAV-filen.*

I Figur 21 ses den ursprungliga WAV-filen. Man kan se att det finns ljudinformation ända upp till ca 22 kHz. Amplituden är som högst från ca 40 Hz – 2 kHz varefter den sedan avtar i amplitud.

### 8.3.1 128 kbps



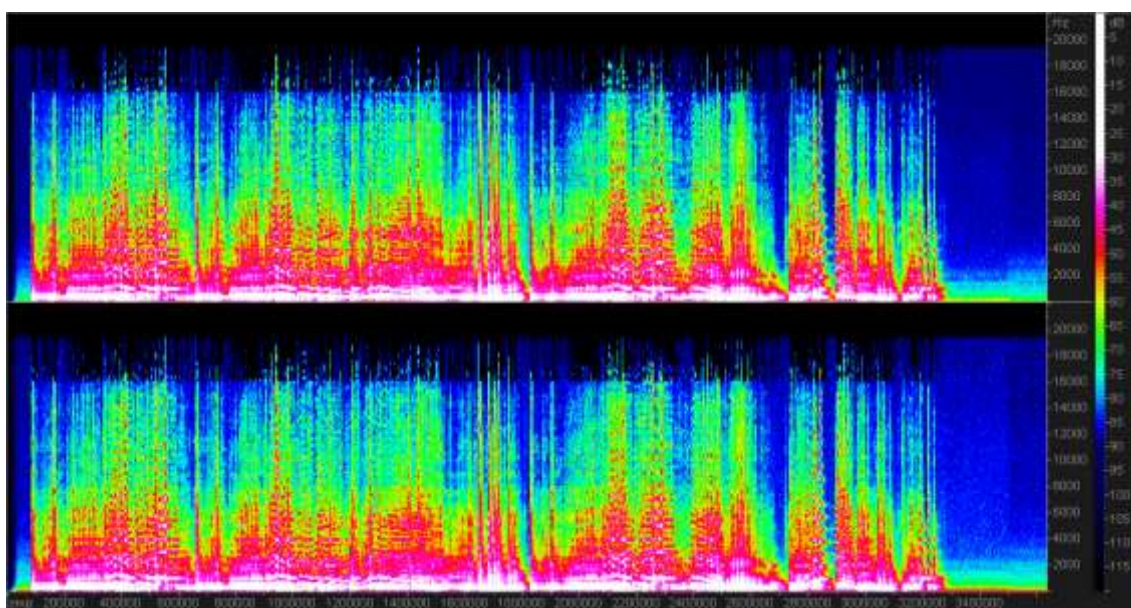
Figur 22. MP3 (128 kbps).



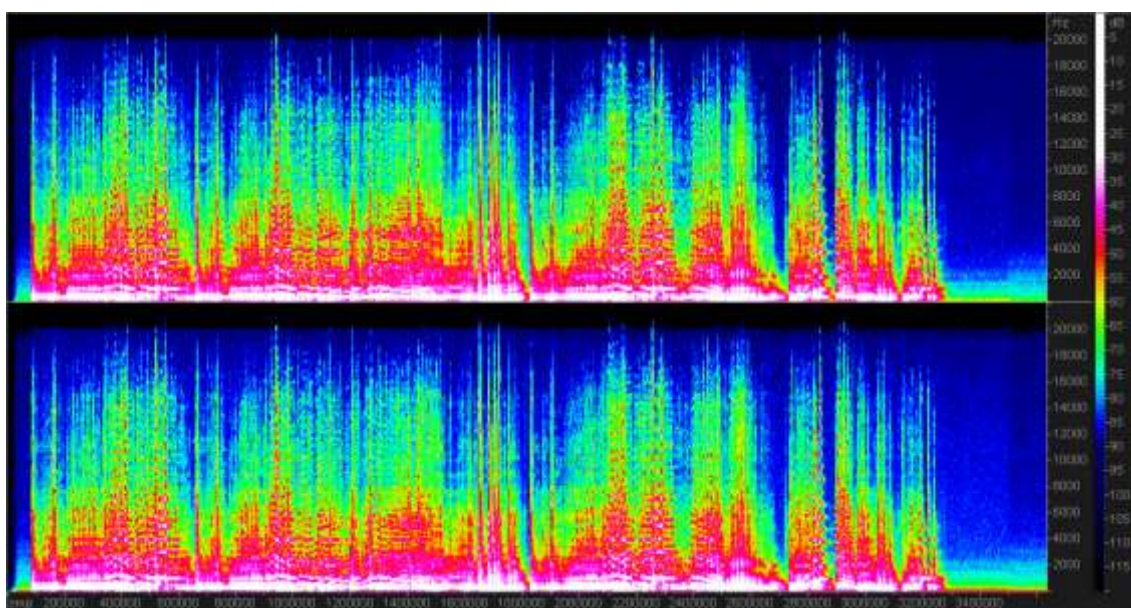
Figur 23. AAC (128 kbps).

I Figur 22 och 23 visas spektrala bilder av MP3-filen respektive AAC-filen vid 128 kbps upplösning. Man kan urskilja vissa skillnader mellan de olika filerna, men de är relativt små. Den största skillnaden är avklippningen vid ca 16 kHz som förekommer både i MP3- och AAC-filen. Avklippningen i AAC-filen är en aning högre i frekvens än i MP3-filen, vilket överensstämmer med frekvensanalysen. I AAC-filen förekommer även den del frekvenstoppar ovanför avklippningsfrekvensen.

### 8.3.2 256 kbps



Figur 24. MP3 (256 kbps).



Figur 25. AAC (256 kbps).

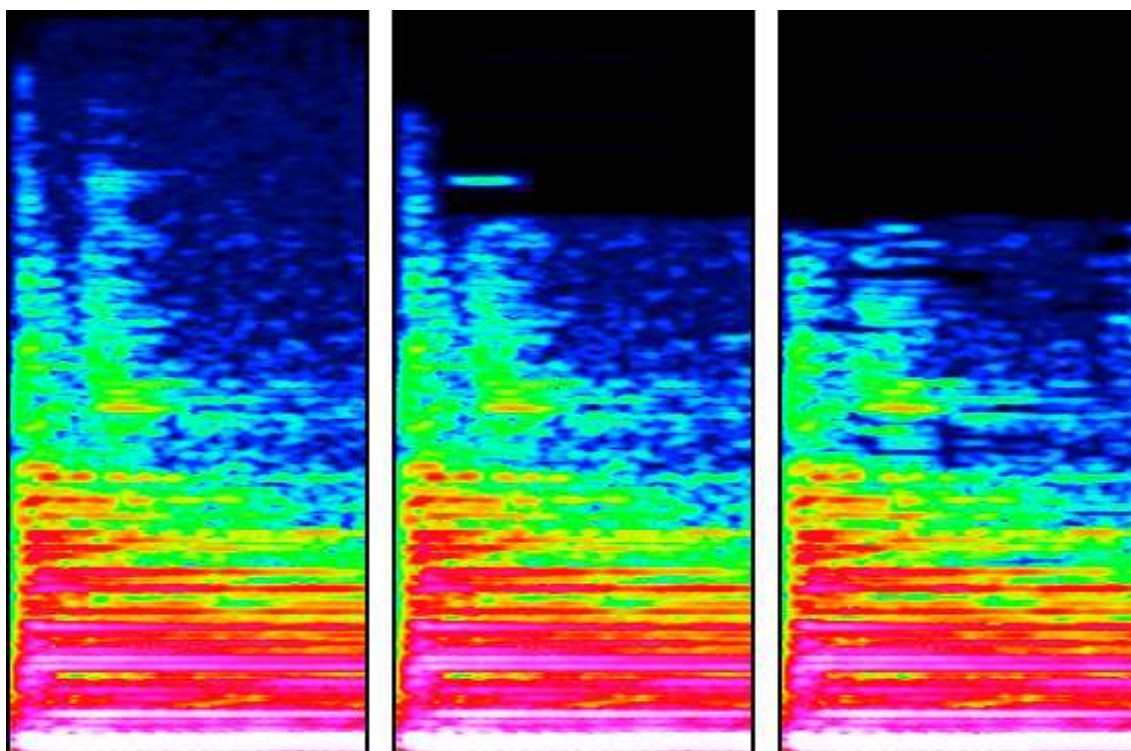
I Figur 24 och 25 visas MP3-filen respektive AAC-filen vid 256 kbps upplösning. Avklippningen som förekom i 128 kbps filerna förekommer också här men den befinner sig aningen högre upp i frekvensområdet. Även här överensstämmer detta med frekvensanalysen.



Avklippningen i MP3-filen ter sig ganska diffus och verkar släppa igenom enbart frekvenstoppar i ljudsignalen. I AAC-filen är avklippningen mycket mer jämn vid strax under 20 kHz, men vissa frekvenstoppar ser ut att släppas igenom även här.

Baserat på den spektrala vyn ser AAC ut att vara effektivare än MP3 vid högre bithastigheter, eftersom det enbart framkommer minimala skillnader jämfört med originalet, både vad gäller ljudinnehåll och avklippning.

### 8.3.3 Närmare jämförelser

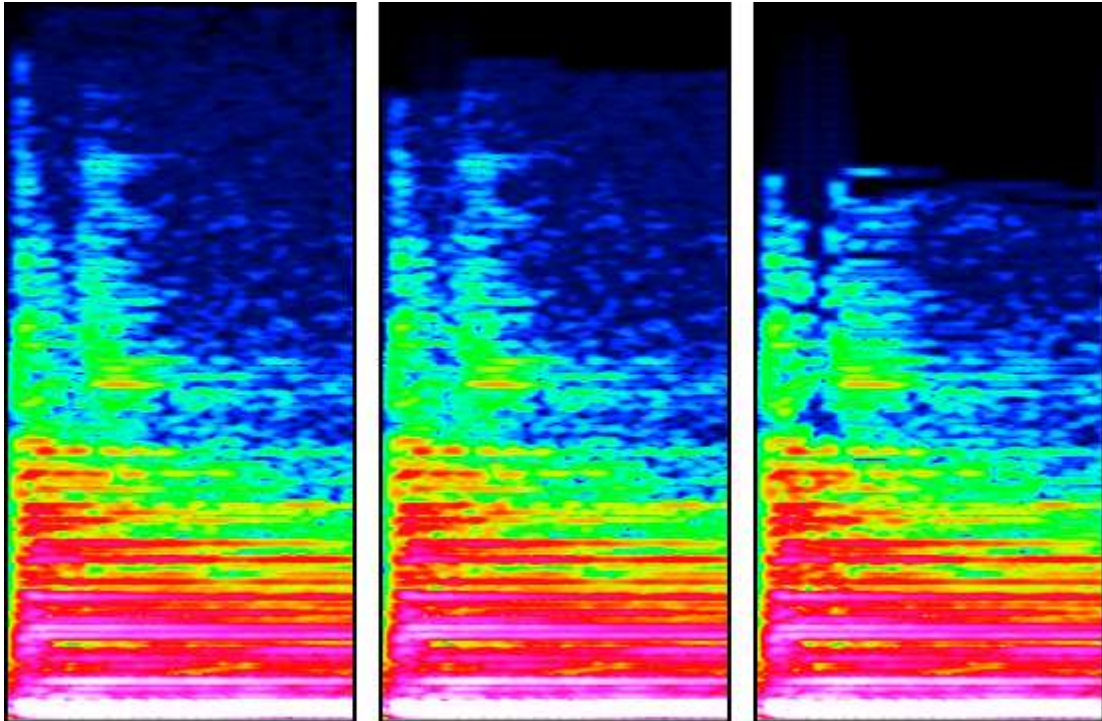


Figur 26. WAV till vänster, MP3 (256 kbps) mitten, MP3 (128 kbps) höger.

I Figur 26 ses en jämförelse av en inzoomad del av den spektrala bilden i tre versioner av ljudfilen: Den ursprungliga WAV-filen, MP3-filen kodad i 256 kbps samt MP3-filen kodad i 128 kbps. Ljudet i fråga är ett anslag på tonen  $A\#_3$ . De horisontella röda ”streck” som syns i bilden är de toner (grundton samt övertoner) som återges av gitarren. Det övriga innehållet kan vara t.ex. omgivningsljud som funnits vid inspelningen.

Den största skillnaden är avklippningen i båda MP3-versionerna (ses som svart upptill).

Man kan även urskilja vissa skillnader i den övriga informationen, speciellt i MP3-filen kodad i 128 kbps. Man kan urskilja vissa "hål" i den spektrala bilden där ljudinformation lämnats bort p.g.a. den relativt låga bithastigheten.



*Figur 27. Wav vänster, AAC (256 kbps) mitten, AAC (128 kbps) höger.*

I Figur 27 ses en jämförelse mellan den ursprungliga Wave-filen, AAC-filen kodad i 256 kbps samt AAC-filen kodad i 128 kbps. Avklippningen är även här den största skillnaden mellan filerna.

AAC-filen kodad i 256 kbps är väldigt lik originalet och det är svårt att urskilja några större skillnader över huvudtaget. I AAC-filen kodad i 128 kbps är det dock lättare att upptäcka en hel del skillnader. Det förekommer en del "hål" i ljudinformationen på samma sätt som i MP3-filen kodad i motsvarande bithastighet.

## 9 SLUTSATSER

När jag började skriva detta examensarbete hade jag ingen aning om vilket djupt ämne jag egentligen gav mig in på. Hade jag vetat bättre skulle jag nog specificerat min frågeställning lite noggrannare och fördjupat mig mer på utvalda delar. Med den frågeställning jag har använt mig av blev innehållet i detta examensarbete ganska brett med en kort genomgång av många olika delar inom digitalt ljud och ljudkomprimering. På ett sätt anser jag dock att detta representerar vårt utbildningsprogram i Informations- och medieteknik ganska bra. Allmän kunskap inom många olika områden.

De slutsatser jag kommit fram till i detta examensarbete är till vissa delar säkert föga förvånande för många. Det är ingen myt att högre bithastighet ger bättre ljudkvalitet och därmed en noggrannare representation av originalet. Jag strävade dock främst efter att påvisa om förlustbringande ljudkomprimering påverkar ljudåtergivningen i en inspelning av ett enskilt instrument, vilket jag anser att jag lyckats göra.

### 9.1 Påverkan på den klassiska gitarrens ljudåtergivning

Förlustbringande ljudkomprimering påverkar definitivt ljudåtergivningen av den klassiska gitarren. Man bör dock ha i åtanke att de flesta perceptuella kodek använder sig av psykoakustiska modeller för att emulera den mänskliga hörseln. De skillnader som framkommer i denna analys kan därmed antas vara formade för att vara så diskreta och ohörbara som möjligt. Vid högre bithastigheter är skillnaderna mellan originalfilen och de omkodade filerna även väldigt små.

Den största förlusten som förekommer är den avklippning som de flesta kodek tillämpar antingen som sådant eller med hjälp av den psykoakustiska modellen. Avklippningen kan ske från 16 kHz uppåt beroende på vilka kvalitetsinställningar man använder sig av i kodningsprocessen. Avklippningen väljs i vanliga fall med hjälp av den kvalitetsinställning som valts vid själva kodningsprocessen, men vissa ljudkodek ger även användaren möjlighet att själva ändra denna inställning om det så behövs.

## 9.2 Skillnader mellan kodek

Man kan definitivt påvisa skillnader i vågformen mellan de kodek jag använt i denna analys, men det är väldigt svårt att avgöra vilken inverkan skillnaderna har på den uppfattade ljudkvaliteten. För att avgöra detta borde man tillämpa ett lyssningstest där de olika ljudfilerna spelas upp för ett antal personer. Dessa personer får sedan i uppgift att avgöra om det finns hörbara skillnader i ljudkvaliteten eller inte. Lyssningstestet borde givetvis vara ett blindtest där försökspersonerna inte har någon kännedom om ljudfilerna som spelas upp.

## 10 FORTSATT FORSKNING

Den analys som gjorts i detta examensarbete baserar sig på ett förinspelat musikstycke taget från en kommersiellt tillgänglig CD-skiva. Denna inspelning kan ha genomgått en mastringsprocess som förändrat den ursprungliga ljudbilden. För att helt utesluta denna möjliga förvrängning vore det skäligt att göra en helt ny inspelning som inte skulle genomgå någon postprocessering över huvudtaget.

Man skulle kunna fördjupa forskningen ytterligare genom att använda sig av lyssningstest för att noggrannare avgöra om det finns förluster i ljudkvaliteten. Lyssningstestet bör vara ett blindtest där ett antal personer med erfarenhet inom musikbranschen eller från andra lyssningstest får lyssna på de olika filerna (originalet och alla omkodade ljudfiler) och avgöra om det förekommer hörbara skillnader.

Man skulle även kunna fortsätta forskningen med att göra jämförelser med flera instrument utöver den klassiska gitarren. Man kunde t.ex. analysera ett rytminstrument där transienter utgör en stor del av ljudet vilket skulle skapa en annan typ av belastning på ljud-kodeket.

För att enklare kunna utvärdera kodekens funktion helt separat kunde man skapa ett test med ljudfiler bestående av enkla signaler (t.ex. olika kombinationer av sinusvågor). I ett sådant test skulle man tydligare kunna se de olika kodekens funktion och hur deras algoritmer och psykoakustiska modeller arbetar.

## KÄLLOR

Adobe. 2012. *Cool Edit Pro is now Adobe Audition*. Tillgänglig:  
<http://www.adobe.com/special/products/audition/syntrillium.html>  
Hämtad 25.4.2012.

Aldrich, Nika. 2002. *Dither Explained*. Tillgänglig:  
<http://www.users.qwest.net/~volt42/cadenzarecording/DitherExplained.pdf>  
Hämtad 28.1.2012.

Billington, Ian. 1999. *The Physics of the Acoustic Guitar*. Tillgänglig:  
<http://ffden-2.phys.uaf.edu/211.web.stuff/billington/main.htm>  
Hämtad 26.4.2012.

Eendebak, Bert. 2011. *Design of a classical guitar*. Tillgänglig:  
<http://www.designofaclassicalguitar.com/dimensions.php>  
Hämtad 18.3.2012.

Fall, Michael. 2011. [www]. Tillgänglig:  
[http://michael-fall.se/gitarr/main\\_1.php](http://michael-fall.se/gitarr/main_1.php)  
Hämtad 18.3.2012.

Fries, Bruce. 2000, *The MP3 And Internet*. TeamCom, 302s.

Hacker, Scot. 2000, *MP3: The Definitive Guide*. O'reilly, 377s.

Hallgren, Per. 2011. *Klassisk gitarr, klassiska problem*. Tillgänglig:  
[http://www.hallgrenguitars.com/Per\\_Hallgren,\\_Gitarrbyggare/Blogg/Poster/2011/7/22\\_Klassisk\\_gitarr%2C\\_klassiska\\_problem.html](http://www.hallgrenguitars.com/Per_Hallgren,_Gitarrbyggare/Blogg/Poster/2011/7/22_Klassisk_gitarr%2C_klassiska_problem.html)  
Hämtad 23.3.2012.

Huber, David Miles. Runstein, Robert E. 2005. *Modern Recording Techniques*. Elsevier, 652s.

Hydrogenaudio. 2012. *MP3*. Tillgänglig:  
<http://wiki.hydrogenaudio.org/index.php?title=MP3>  
Hämtad 22.3.2012.

Karlheinz-Brandenburg. 1999. *MP3 and AAC Explained*. Tillgänglig:  
[http://graphics.ethz.ch/teaching/mmcom10/slides/MP3\\_and\\_aac\\_brandenburg.pdf](http://graphics.ethz.ch/teaching/mmcom10/slides/MP3_and_aac_brandenburg.pdf)  
Hämtad 13.2.2012.

Pinksterboer, Hugo. 2001, *Tipbook Akustisk Gitarr*. The Tipbook Company, 125s.

MP3-tech.org. 2012. *MP3 Info Tag rev 1 specifications - draft 0*. Tillgänglig:  
<http://gabriel.MP3-tech.org/MP3infotag.html>  
Hämtad 25.4.2012.

MultimediaWiki. *Psychoacoustic Model*. Tillgänglig:  
[http://wiki.multimedia.cx/index.php?title=Psychoacoustic\\_Model](http://wiki.multimedia.cx/index.php?title=Psychoacoustic_Model)  
Hämtad 18.11.2011.

Pohlmann, Ken C. 2011. *Principles of Digital Audio*, Sixth Edition. McGraw Hill, 800s.

Raissi, Rassol. 2002. *The Theory Behind MP3*. Tillgänglig:  
[http://www.MP3-tech.org/programmer/docs/MP3\\_theory.pdf](http://www.MP3-tech.org/programmer/docs/MP3_theory.pdf)  
Hämtad 13.2.2012.

Talbot-Smith, Michael. 1999. *Audio Engineer's Reference Book*. Focal Press, 672s.

Watkinson, John. 2001. *The Art of Digital Audio*, Elsevier, 770s.

Wikipedia. 2011a. *Filter Bank*. Tillgänglig: [http://en.wikipedia.org/wiki/Filter\\_bank](http://en.wikipedia.org/wiki/Filter_bank)  
Hämtad: 21.1.2012.

Wikipedia. 2012a. [www]. Tillgänglig:  
[http://en.wikipedia.org/wiki/File:Audio\\_Mask\\_Graph.png](http://en.wikipedia.org/wiki/File:Audio_Mask_Graph.png)  
Hämtad 13.2.2012.

Wikipedia. 2012b. [www]. Tillgänglig:  
[http://en.wikipedia.org/wiki/File:Window\\_function\\_\(rectangular\).png](http://en.wikipedia.org/wiki/File:Window_function_(rectangular).png)  
Hämtad 13.2.2012.

Wikipedia. 2012c. *MP3*. Tillgänglig: <http://en.wikipedia.org/wiki/MP3>  
Hämtad 22.3.2012.

Wikipedia. 2012d. *Modified discrete cosine transform*. Tillgänglig:  
[http://en.wikipedia.org/wiki/Modified\\_discrete\\_cosine\\_transform](http://en.wikipedia.org/wiki/Modified_discrete_cosine_transform)  
Hämtad 22.3.2012.