

Ingeniørhøjskolen i Århus
IKT
Dalgas Avenue 2
8000 Århus C

20. december 2004

4. Semesterprojekt
System Arkitektur

MyP3000

I4PRJ4 E2004

Gruppe 4:

Benjamin Sørensen, 02284

Tomas Stæhr Berg, 03539

Nikki Ashton, 01087

Jonas Livbjerg, 02797

Vejleder:

Sten Hansen

1 Versionshistorie

Ver.	Data	Initialer	Beskrivelse
0.1	13.12.2004	BS	Opsætning af dokumentet i LaTeX
0.2	14.12.2004	BS	Review ændringer
1.0	20.12.2004	TB	Afsluttende ændringer

2 Godkendelsesformular

Forfatter(e):	Benjamin Sørensen(BS), Tomas Berg(TB), Nikki Ashton(NA), Jonas Livbjerg(JL)
Projektnummer:	4. semesterprojekt
Antal sider:	15
Godkendes af:	Benjamin Sørensen

Dato og underskrift:

Benjamin Sørensen

Indhold

1	Versionshistorie	1
2	Godkendelsesformular	1
3	Introduktion	3
3.1	Formål og omfang	3
3.2	Definitioner og forkortelser	3
3.3	Læsevejledning	3
4	System oversigt	4
4.1	Systemets funktioner	4
4.1.1	Encoder	4
4.1.2	Decoder	4
4.1.3	Komprimering	5
5	Arkitektursignifikante designpakker	6
5.1	Pakkebeskrivelser	6
5.1.1	Wave IO	6
5.1.2	MyP3 IO	7
5.1.3	Bitallokering og skalering	7
5.1.4	Psykoakustik	7
5.1.5	Kontrolpakker	7
5.2	Samlet funktionsdiagram	8
5.3	Flowcharts	10
6	Implementeringsprog og værktøjer	12
6.1	Fejlhåndtering	12
7	Kvalitet	13
8	Kørsel	14
8.1	Kørselssoftware	14
8.2	Vejledning	14
8.2.1	Encoding	14
8.2.2	Decoding	14
9	Figuroversigt	15

3 Introduktion

3.1 Formål og omfang

Dette dokument fastlægger den overordnede arkitektur for implementeringen af MyP3000. Det indeholder designet for systemet uden at gå i detaljer med den egentlige implementering. Dokumentationen bidrager til, at andre projektgrupper kan overtage udviklingen og opdateringen af MyP3000. Systemarkitekturen kan læses uden kendskab til den øvrige dokumentation.

3.2 Definitioner og forkortelser

MyP3000: Produktets navn.

Psykoakustisk model: Model baseret på den fysiske lydopfattelse i det menneskelige øre.

DFT: Discrete Fourier Transform.

MyP3: Standarden for den komprimerede fil.

IO: Input, output.

Hanningvindue: Et vindue der dæmper i starten og slutningen. Implementeret vha. \cos .

3.3 Læsevejledning

Dokumentet er opbygget, så man først bliver præsenteret kort for systemets funktionalitet, hvorefter der i de senere kapitler vil blive gået mere i detaljer.

Beskrivelsen af projektet starter fra kapitel 4 System oversigt, der giver en kort introduktion til de benyttede principper.

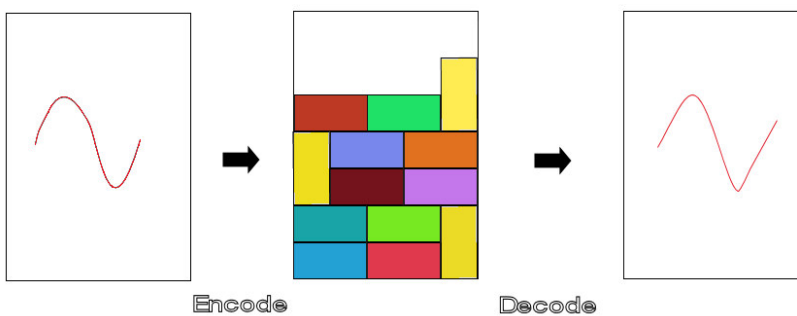
Kapitel 5 viser, hvordan vi har opdelt funktionaliteten i pakker. Der vises funktionsdiagrammer og sekvensdiagrammer.

Kapitel 7 står for en vægtning af kvalitetskrav.

4 System oversigt

Dette afsnit giver et overordnet billede af systemet, samt de omgivelser systemet skal fungere i.

Systemets navn er MyP3000. MyP3000 opfylder to formål. Det primære mål er komprimering af WAVE-filer ved brug af MyP3 standarden. Det sekundære mål er at dekomprimere MyP3 filer tilbage til WAVE filer.



Figur 1: Sammenhæng mellem det primære og sekundære mål.

Det primære mål varetages af en encoder med indbygget komprimering. Det sekundære mål varetages af en decoder. Vi skal lave både encoder og decoder.

4.1 Systemets funktioner

Herunder beskrives henholdsvis encoder, decoder og komprimering.

4.1.1 Encoder

Encoderen skal indlæse lydsignalet fra en WAVE fil og opdele det i blokke. Disse blokke bliver behandlet med 50 % overlap. Der ganges et Hanningvindue på og efterfølgende udføres en DFT, så vi transformerer signalet over i frekvensområdet. Herefter foretages den egentlige komprimering af data, dette gøres vha. en psykoakustisk model, se under "Komprimering".

4.1.2 Decoder

Decoderen skal udføre en invers DFT på blokkene for at transformere signalet over i tidsområdet. Til sidst samles blokkene, og de gemmes i WAVE lydformatet.

4.1.3 Komprimering

Komprimeringen finder sted efter at lydsignalet er transformeret over i frekvensområdet.

Komprimeringen er destruktiv. Det betyder, at vi smider dele af dataene i lydsignalet væk for at gøre komprimeringen bedre. Dette kan vi tillade os, da det menneskelige øres opbygning gør, at det ikke er i stand til at opfatte visse dele i et lydsignal. Der benyttes 2 principper:

1. Alt efter det menneskelige øres følsomhed i forskellige frekvensområder, kan vi allokere et større antal bits til dataene i et følsomt frekvensområde, og et mindre antal bits i et mindre følsomt frekvens område.
2. Ørets opbygning gør, at hvis det præsenteres for en kraftig puls i et snævert område, vil det være mindre følsomt overfor fejl i nærliggende områder. Derved kan vi udfra en frekvensanalyse afgøre, hvor stor en kvantiseringsfejl vi kan introducere, og stadig have et transparent signal.

Ovenstående principper udgør vores psykoakustiske model. Derudover kan vi også skalere de enkelte frekvenser, og derved opnå bedst mulig udnyttelse af den valgte præcision.

5 Arkitektursignifikante designpakker

Her angives de pakker, der har betydning for udformningen af arkitekturen.

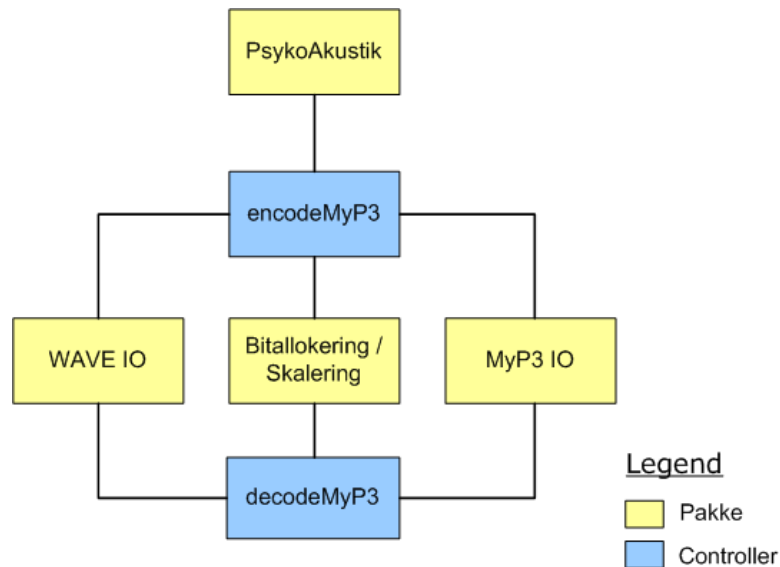
5.1 Beskriver de enkelte pakker.

5.2 Viser funktionsdiagrammer.

5.3 Udførelse angivet på flowcharts.

5.1 Pakkebeskrivelser

I det efterfølgende beskrives de enkelte pakker. Samspillet mellem pakkerne kan illustreres på følgende vis. På billedet er de to kontrolpakker med blå.



Figur 2: Pakkediagram

5.1.1 Wave IO

- **Ansvar:**
Håndtering af WAVE filer. Både indlæsning og udlæsning af WAVE fra Matlab.
- **Funktioner:**
readWAVE - Indlæsning af en WAVE fil. Returnerer samples, samplingsfrekvens samt bit opløsning.

writeWAVE - Udlæsning af samples til WAVE fil. Er hardcoded til en samplingsfrekvens på 44100 [Hz] og en bitopløsning på 16.

hanning - Indbygget Matlab funktion.

5.1.2 MyP3 IO

- **Ansvar:**
Håndtering af MyP3 filer. Både indlæsning og udlæsning af MyP3 fra Matlab.
- **Funktioner:**
myp3writer - Udlæsning af MyP3 fil. Skriver bearbejdet data til disken samt header, allokering og skalering.

myp3reader - Indlæser 1 frame af en MyP3 fil fra et givet offset. Offset er i bytes og leveres som parameter.

5.1.3 Bitallokering og skalering

- **Ansvar:**
Håndtering af allokeringen og skalering af samples.
- **Funktioner:**
scaling - Foretager maksimal skalering indenfor hvert bånd.

alloc - Foretager kvantisering iht. den givne allokering.

5.1.4 Psykoakustik

- **Ansvar:**
Håndtering af psykoakustikken der danner baggrund for bitallokeringen.
- **Funktioner:**
PsykoMasking - Foretager udvælgelsen af bitopløsligheden for hvert bånd.

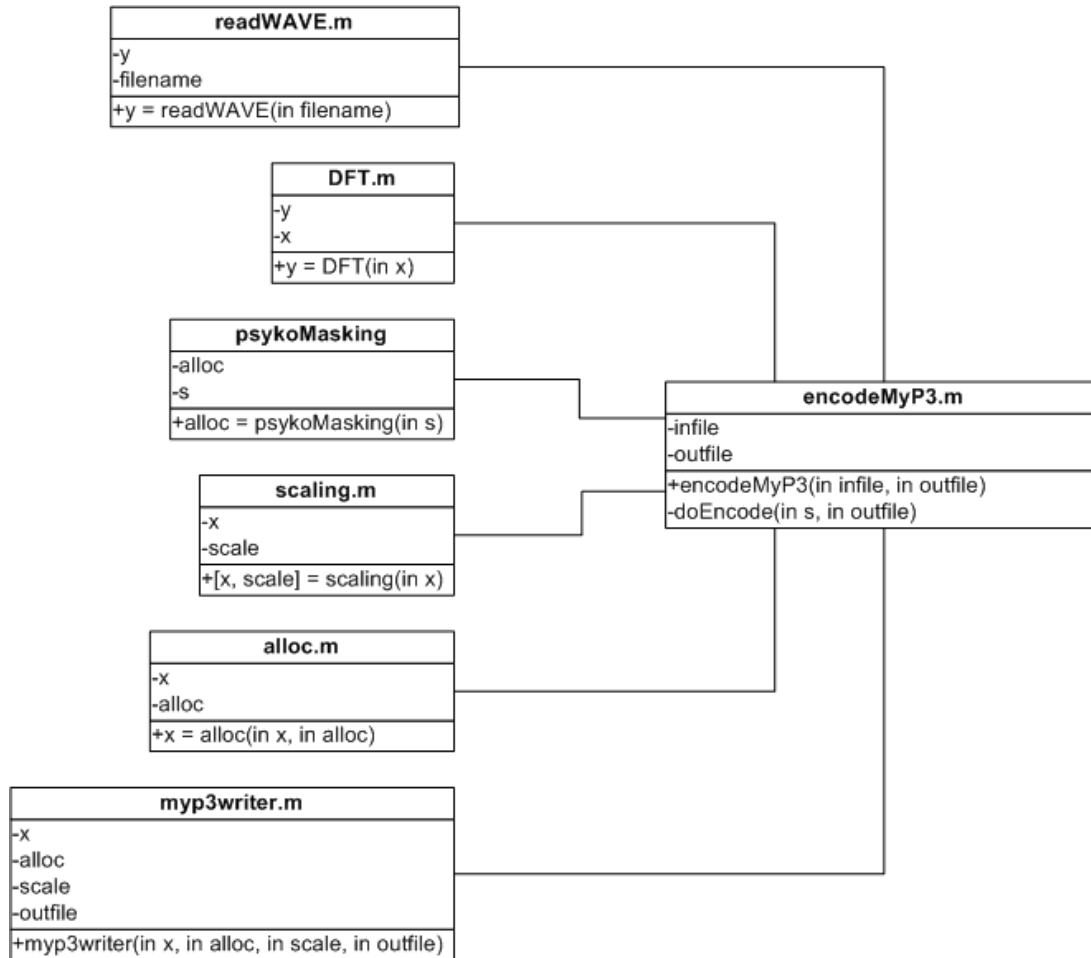
5.1.5 Kontrolpakker

- **Ansvar:**
Indeholder kontrolsekvensen.
- **Funktioner:**
EncodeMyP3 - Styring af encoderen.

DecodeMyP3 - Styring af decoderen.

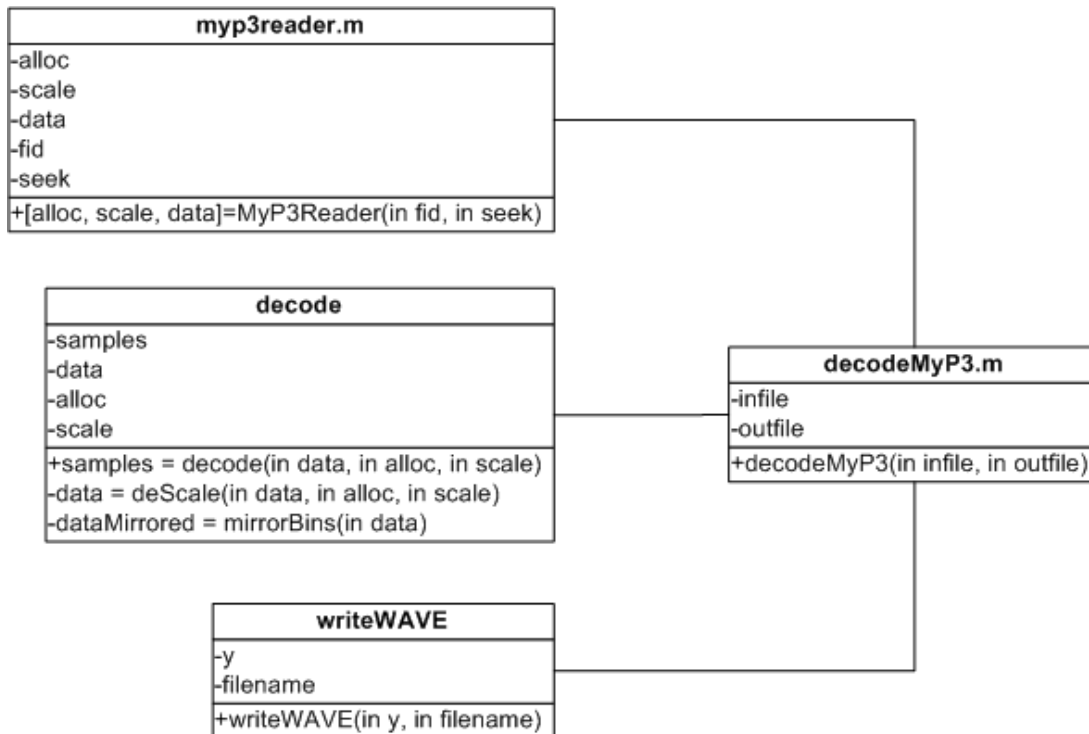
5.2 Samlet funktionsdiagram

Sammenhængen mellem funktionerne ses i disse funktionsdiagrammer.



Figur 3: Funktioner i encoderen

Figuren viser funktioner som bruges af encoderen.

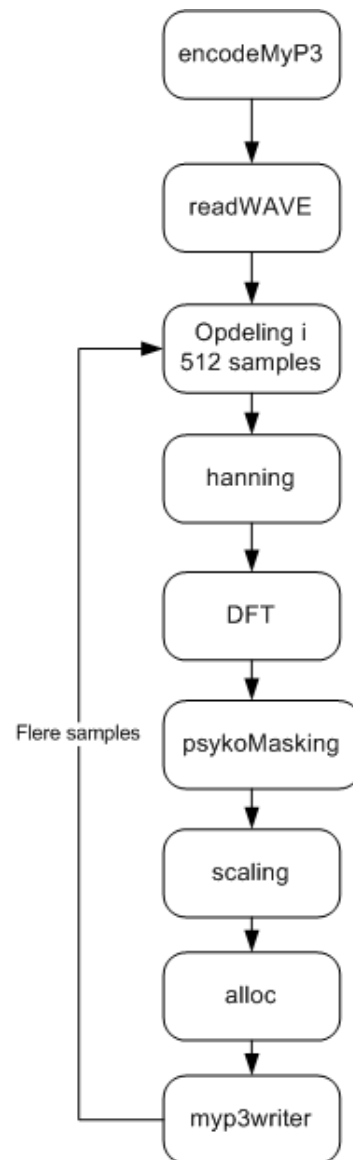


Figur 4: Funktioner i decoderen

Figuren viser funktioner som bruges af decoderen.

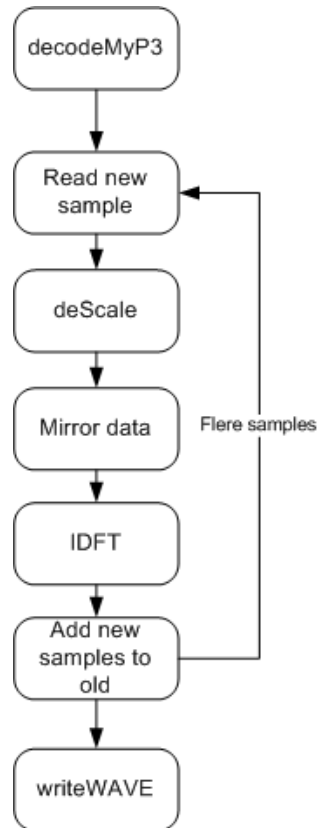
5.3 Flowcharts

Her vises rækkefølgen i funktionskald for encoderen.



Figur 5: Flowchart for encoderen

Her vises rækkefølgen i funktionskald for decoderen.



Figur 6: Flowchart for encoderen

6 Implementeringssprog og værktøjer

Vi har bevidst valgt Matlab som implementeringsmiljø. Vi har tidligere kun arbejdet overfladisk med matlab. Derfor var det et delmål for projektet, at få et indgående kendskab til Matlab.

Matlab er en funktionel fortolker, så derfor skal designet af MyP3000 være funktionelt orienteret.

6.1 Fejlhåndtering

Der er ikke lagt vægt på optimal fejlhåndtering, men vi sikrer, at indlæsingen foregår korrekt. Dette gøres ved at tjekke om headeren i hver frame er som forventet. Hvis der opstår en fejl, afsluttes programmet med besked om, at headeren er ude af synkronisering.

7 Kvalitet

Dette afsnit giver et kort overblik over kvalitetskravene og beskriver hvad der er gjort for at overholde dem bedst muligt.

- **Pålidelighed: Særdeles vigtigt.**
Det er vigtigt at programmet er konsistent, og ikke skaber tilfældige fejl i MyP3 filer.
- **Udvidelsesvenligt: Vigtigt.**
Designet er opbygget i pakker, så funktionaliteten uden de store problemer kan forbedres.
Specielt vil man kunne forbedre den psykoakustiske funktion, uden det vil påvirke resten af såvel encoder som decoder.
- **Brugervenlighed: Ikke vigtigt.**
Da dette produkt er udviklet med henblik på udforskning af lydkomprimering, og ikke skal bruges som et endeligt produkt, anses brugervenlighed for ikke vigtigt. Det må formodes at brugere er på samme niveau som udviklere.
- **Genbrugbarhed: Ikke vigtigt.**
Det anses ikke som vigtigt, da dette kun skal bruges til udforskning. Det er dog vigtigt, at kunne udskifte den psykoakustisk model.
- **Integritet: Vigtigt.**
Det er vigtigt at skrivningen til disken foregår korrekt, så det ikke introducerer unødvendig støj, eller i værste fald korrupte MyP3 filer. Dette undersøges ikke løbende, men er testet under udviklingen.
- **Effektivitet: Nødvendigt.**
Det er nødvendigt med et effektivt produkt, for at mindske encoding og decodingtider. Vi har anvendt Matlabs profiler rutine til at afsløre tidskrævende funktioner. Disse er efterfølgende optimeret.

8 Kørsel

Dette afsnit beskriver hvordan en WAVE fil encodes til MyP3, samt hvordan MyP3 filen decodes tilbage til WAVE format.

8.1 Kørselssoftware

Matlab 7.0.0.19920 (R14)

8.2 Vejledning

8.2.1 Encoding

Start Matlab

Ændre mappen til mappen hvor MyP3000's encoder (eg. c:\myp3000\encode)

Skriv `encodeMyP3('wavefilnavn','myp3filnavn');`

Encodingen foretages. MyP3 filen placeres i encoder mappen.

8.2.2 Decoding

Start Matlab

Ændre mappen til mappen hvor MyP3000's decoder (eg. c:\myp3000\decode)

Skriv `decodeMyP3('myp3filnavn','wavefilnavn');`

Decodingen foretages. WAVE filen placeres i decoder mappen.

9 Figuroversigt

Figurer

1	Sammenhæng mellem det primære og sekundære mål.	4
2	Pakkediagram	6
3	Funktioner i encoderen	8
4	Funktioner i decoderen	9
5	Flowchart for encoderen	10
6	Flowchart for decoderen	11