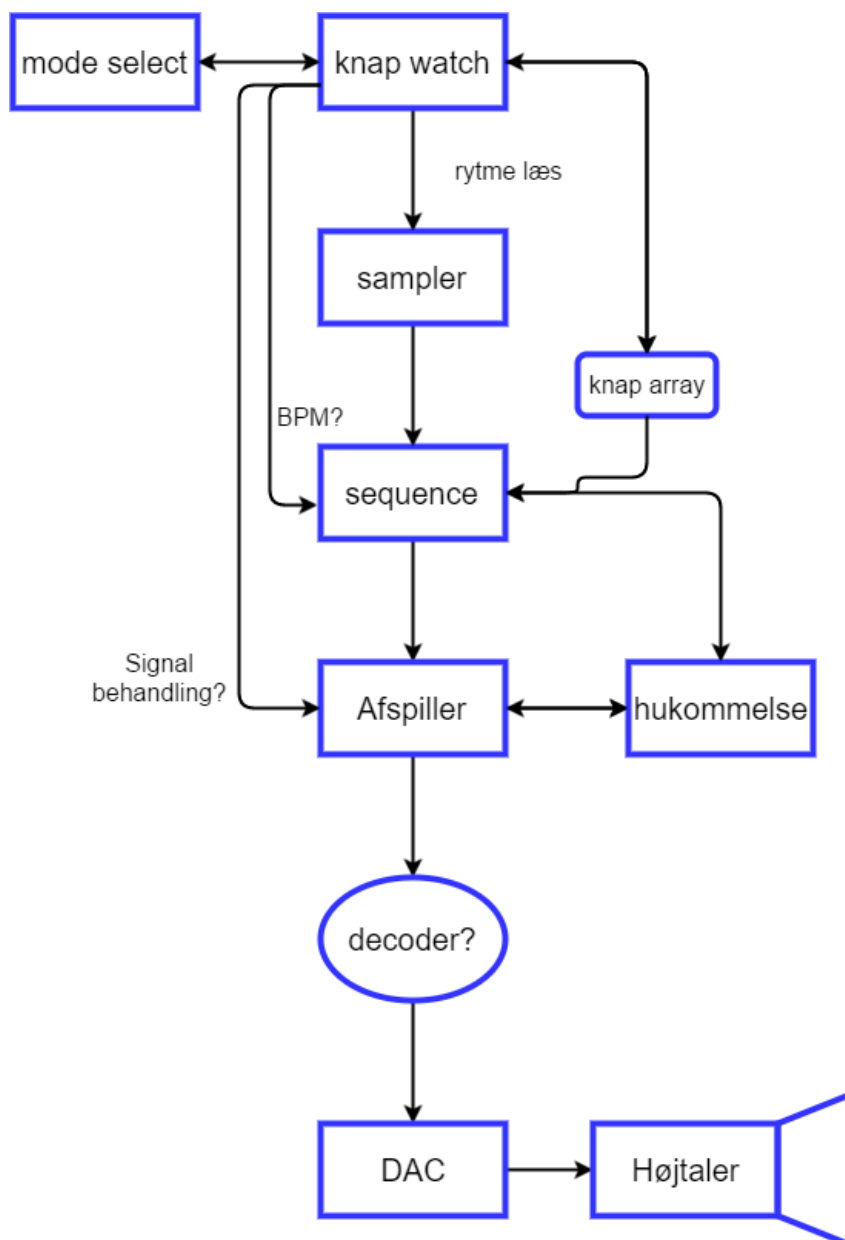


## 0.1 Arbejdsblad

Hej Søren

Teoriafsnittet er så småt ved at være færdigt. Der mangler et par småting, som vi får at vide gennem undervisningen, så det skriver vi færdigt til den tid. Som du bad om har vi lavet en overordnet rapportstruktur, hvilket kan ses i indholdsfortegnelsen. Derudover har vi påbegyndt en modulinddeling som kan ses på figur 0.1.



Figur 0.1: Meget flot moduldiagram(PROTOTYPE). Forklaring af kasse kan vi tage på mødet.

Derudover kunne vi også tænke os nogle små fif til hvordan man smider en .wav-fil på FPGA'en.

# *Bit Boxer 9000*



GRUPPE 412  
ELEKTRONIK & IT  
AALBORG UNIVERSITET  
DATE 21. DECEMBER 2016

# Forord

---

Igennem rapporten anvendes benævnelsen 'sequencer' udelukkende til at beskrive musiksequencere.

søg og forklar MSB & LSB

# Indholdsfortegnelse

---

0.1 Arbejdsblad . . . . .	1
<b>Kapitel 1 Introduktion</b>	<b>1</b>
1.1 Hvad er en trommemaskine? . . . . .	1
1.2 Initierende problemformulering . . . . .	1
<b>Kapitel 2 Teori</b>	<b>2</b>
2.1 Musik og rytmik . . . . .	2
2.2 Sequencer . . . . .	3
2.3 Digital lyd . . . . .	4
2.4 Digital-til-analog-konvertering . . . . .	7
2.5 Hardware i projektet . . . . .	9
<b>Kapitel 3 Problemformulering</b>	<b>11</b>
3.1 Projektafgrænsning . . . . .	11
3.2 Modulinddeling . . . . .	11
3.3 Problemformulering . . . . .	11
<b>Kapitel 4 Kravspecifikation</b>	<b>12</b>
4.1 Funktionelle krav . . . . .	12
4.2 Krav til delmoduler . . . . .	12
4.3 Beskrivelse af accepttest . . . . .	12
<b>Kapitel 5 Design</b>	<b>13</b>
<b>Kapitel 6 Accepttest</b>	<b>14</b>
<b>Kapitel 7 Diskussion</b>	<b>15</b>
<b>Kapitel 8 Konklusion</b>	<b>16</b>
<b>Kapitel 9 Perspektivering</b>	<b>17</b>
<b>Bibliografi</b>	<b>18</b>
<b>Appendiks A Udledning af DAK-kredløb</b>	<b>19</b>

# Forord

---

Igennem rapporten anvendes benævnelsen 'sequencer' udelukkende til at beskrive musiksequencere.

søg og forklar MSB & LSB

## 1.1 Hvad er en trommemaskine?

En trommemaskine er et redskab der benyttes til let at fremstille rytmer. Dette gøres ved at maskinen kører rundt i et uendeligt loop, hvor brugeren kan indsætte eller fjerne instrumentlyde på de ønskede dele af takter. Et enkelt af disse loops kan kaldes for en sekvens, hvilket svarer til én takt. De fleste trommemaskiner har delt denne takt op i 16 dele, hvor man kan indsætte hver enkelt instrumentlyd på hver 16. del takt. Trommemaskinen består af flere forskellige trommelyde så som stortromme, lilletromme og high-hat. På denne måde kan man fremstille en trommerytme ved at indsætte forskellige lyde på udvalgte steder i cyklussen.

Trommemaskinens lyde og forskellige alt efter trommemaskinens producent. Nogle producenter anvender digitale lydsamples, hvor andre benytter sig af analoge kredsløb til at imitere trommelyde. Begge metoder har sine fordele og ulemper. For eksempel kræver digitale trommemaskiner konvertering mellem analoge- og digitale signaler, men de har til gengæld større mulighed indspille personlige lydklip. Fælles for begge typer af trommemaskiner, er at de kan afspille flere instrumentlyde på samme brøkdels af takten, hvilket betyder at de begge har en metode hvorpå de sammenlægger lyde.

Trommemaskinen var med til at skabe den kendte lyd fra 80'erne. Blandt de mest kendte trommemaskiner finder man Roland TR-808, som blev introduceret i 1980, og dens efterfølger, Roland TR-909 fra 1984. TR-808 imiterede diverse trommelyde ved hjælp af analoge kredsløb, hvilket betød at den havde den velkendte "robotagtige" lyd i form af kliklyde fra hardwaren. Opfindelsen af trommemaskinen gjorde det muligt for enhver at fremstille rytmer uden at kunne spille trommer eller have anden musikalsk kendskab, hvilket også grunder den popularitet.

## 1.2 Initierende problemformulering

Da dette er et 4. semesters projekt, som omhandler digital design, vil der tages udgangspunkt i en digital trommemaskine. Derfor opstilles en initierende problemformulering således at teorien bag trommemaskinens opbygning og elementer kan undersøges. Den initierende problemformulering lyder som følgende:

*Hvilke elementer indgår i en trommemaskine, der skal fremstilles digitalt?*

I dette kapitel beskrives teorien bag elementer i en digital trommemaskine og hvordan disse fungerer.

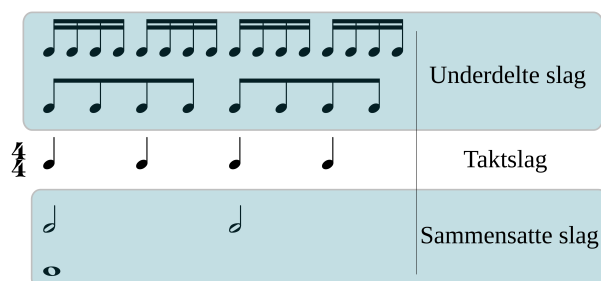
## 2.1 Musik og rytmik

En trommemaskines primære funktion er at kunne indspille rytmer og afspille disse. Derfor er det nødvendigt at redegøre for hvordan en rytme kan beskrives.

Musik inddeles i takter som udgør grundrytmen. Disse takter opdeles i taktslag. En taktart noteres som en brøk f.eks.  $\frac{4}{4}$ . Tallet over brøkstregen bestemmer hvor mange taktslag der går på en takt, og tallet i nævneren bestemmer nodeværdien af taktslagende.

Den mest anvendte taktart i vestlig musik er  $\frac{4}{4}$  der dermed består af fire taktslag af fjerdedelsnoder. [5]

Taktslagene i en takt kan inddeles på mange måder. Eksempelvis kan en fjerdedelsnode erstattes med to ottendedelsnoder, som derefter kan opdeles i to sekstendedelsnoder, o.s.v. På modsatvis kan to fjerdedelsnoder erstattes af en halvnode, hvormed to halvnoder kan erstattes med en helnode, der fylder en hel takt. Alle disse nodeværdier kan også erstattes med en pause af samme længde.



Figur 2.1:  $\frac{4}{4}$  inddelt i serier af forskellige nodeværdier **how do i poot dis?**. Illustration fra [7]

I musik bliver tempoet bestemt af antallet af taktslag pr. minut også kaldet BPM (beats per minute). Normalt musik har et tempo mellem 80 til 160 bpm.[4]. Dette vil sige at en sang med et tempo på 100 bpm i  $\frac{4}{4}$  vil have 25 takter i minuttet.

For at kunne holde en konstant stabil rytme bruger musikkere ofte et apparat kaldet en metronom. En metronom er oftest konstrueret på en sådan måde at brugeren vælger et givent tempo i bpm, hvorefter metronomen spiller en puls i det givne tempo, ofte i en given taktart med markering på første slag.

Metronomer findes i et utal af størrelser lige fra mekaniske metronomer bestående af et pendul, til apps til smartphones.

Det viser sig at selv trænede musikkere ikke ramme hver enkelt takslag helt perfekt. Musik spillet eller indspillet af mennesker har typisk en gennemsnitlig afvigelse fra taktslagene på 10-20 ms.

Dette er dog ikke noget problem, faktisk tvært imod. Computergenereret musik der falder præcis på slagene lyder i menneskelige ører som meget unaturligt og følelsesløst, hvilket gør at man ofte tilføjer små afvigelser i computergenereret musik for at gøre det mere "menneskeligt"[3].

## 2.2 Sequencer

En sequencer har til formål at styre hvornår bestemte lyde bliver afspillet i trommemaskinen. Dette gøres ved at afspille en sekvens, som er opdelt i mindre dele. Alt efter hvilke funktionaliteter der søges af sequenceren kan antallet af dele i en sekvens ændres. Det samme gælder for tiden det tager at afspille en sekvens. Eksempelvis kan der være 16 dele for en trommemaskine, da dette repræsenterer en takt. Disse giver mulighed for at afspille en bestemt sekvens, som indeholder række lyde. Denne sekvens kan så gentages eller en anden sekvens kan efterfølgende afspilles.

Sequenceren har den fordel at tiden det tager at afspille en bestemt sekvens er forudbestemt. Det er derved muligt at få sequenceren til at holde en bestemt rytme. Derudover er det muligt at ændre tiden det tager af afspille en sekvens, så man derved kan hæve eller sænke tempoet der afspilles i.

De første sequencere der blev opfundet var byggede af analoge komponenter. Disse bestod af en "clock", som gennemgår en række steps og derefter starter forfra. Disse sequencere var så sat til en synthesizer styret af flere række potentiometre, oftes med otte per række. Hver række blev så brugt til at styre en kanal for synthesizer. Et eksempel på dette kan ses på figur 2.2, hvor sequenceren, Moog 960, ses midterst til højre. Derved kunne man skrue på potentiometrene for at få en bestemt lyd ud og vælge om den skulle afspilles eller ej. Nogle af de første musikere, som lavede elektronisk musik, ville indstille disse potentiometre live på scenen når de optrådte.



Figur 2.2: På billedet ses en Moog 960, som er en analog sequencer og synthesizer

Moderne sequencere er nemme at styre ved hjælp af software. Ud over at gøre det nemmere at sætte op, har det også givet mulighed for at tilføje en række funktioner. Eksempelvis kan de ønskes at sequenceren kan afspille i reel tid. Her vil mængden af steps per sekvens blive øget betydeligt, så det er nemmere at indpasse hvornår et bestemt step skal afspilles i forhold til



reeltid.

Der findes mange måde at programmere en sequencer til at afspille en sekvens. En meget anvendt metode er med en række knapper der kan rykkes ned. 16 knapper kan f.eks. svare til en takt inddelt i 16 dele som nævnt tidligere.

En måde er at lade en bruger slå på nogle 'trommer' i en bestemt rytme, der derefter gemmes og afspilles. Til dette formål kan man anvende en såkaldt sampler.

### 2.2.1 Sampler

En sampler er en form for musikinstrument der har til formål at afspille nogle bestemte lyder eller "samples", når der gives et input. Inputtet kan for eksempel være store knapper, der kan trommes på, eller tangenter på et keyboard.

Sampleren bruges som en erstatning til et virkeligt instrument f.eks. til et trommesæt ved at lade den kunne spille samples fra forskellige trommer på et trommesæt, eller et sample fra et klaver der kan afspilles i forskellige toner.

Ofte er knapperne trykfølsomme sådan at der kan kendes forskel på om en knap berøres let eller den trykkes langt ned, så man både kan spille 'blødt' og 'hårdt'.

I daglig tale benyttes sampler ogse som en fællesbetegnelse for et apparat der både indeholder sampling og en sequencer, som for eksempel sampleren vist på figur 2.3



Figur 2.3: Sampleren Akai MPC2000 fra 1997. Billede fra [6]

Sampleren kan eventuel også bruges til at programmere en sequencer i realtid. Dette gør at den indspillede input kan blive gentaget og eksporteret.

skal der stå mere?

## 2.3 Digital lyd

Dette afsnit er skrevet af !Mathias Hvorfor er der behov for at gemme lyd digitalt?

Når to personer snakker sammen, foregår det ved at taleren producerer lydbølger, som lytteren hører. Lydstyrken lytteren oplever afhænger af hvor højt taleren snakker, samt afstanden mellem dem. Fastholdes lydstyrken mens afstanden mellem dem øges, vil lytteren opleve en lavere og lavere lydstyrke og efter en vis afstand kan lytteren ikke længere høre taleren. Der er derfor et behov for en bedre måde at transportere lyden fra taleren til lytteren.

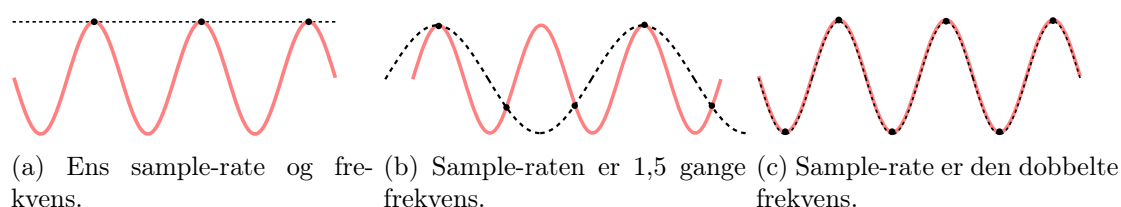
Ved at placere en transducer (f.eks. en mikrofon) ved lydkilden, kan lydbølgerne omdannes til elektriske signaler som kan videregives via ledninger. Placeres endnu en transducer (f.eks. en højttaler eller hovedtelefoner) ved lytteren, kan det elektroniske signal igen omdannes til hørbare lydbølger. På denne måde kan afstanden mellem lydkilden og lytteren blive væsentligt længere. Det elektroniske signal kan være enten analogt eller digitalt. Den store fordel ved gøre systemet digitalt er, det er mere hårdfør overfor støj. Derudover kan det gemmes til senere brug.

### Analog til digital

Mikrofonen ved taleren omdanner lydbølgerne til et analogt elektrisk signal. Når dette analoge signal skal konverteres til et digitalt signal tages en analog-til-digital-konverter i brug. Denne kaldes fremover for en ADC. En ADC konverterer det analoge signal til en sekvens af digitale bits, som repræsenterer spændingen. Denne kaldes en sample. Modtager den en DC-spænding på indputtet vil den altid give det samme på outputtet. Sendes en AC-spænding ind, vil udgangssignalet blive omdannet til en digital værdi for spændingen, på det tidspunkt samplen er taget.

#### 2.3.1 Sampleratens indflydelse på genskabelsen af det originale signal

For at kunne genskabe AC-signalet er der behov for et antal samples og når der snakkes om AC-signaler snakkes der også om en frekvens derfor angives antallet af samples, som en sample-frekvens. Ifølge Nyquist, skal samplefrekvensen være over dobbelt så høj som den højeste frekvens man vil kunne genskabe.



Figur 2.4: Nyquist læresætning for samplerate. Den røde er signalet og den sorte er den bedste genskabelse af signalet.

Som det kan ses på figur 2.4 skal samplingfrekvensen være minimum den dobbelte frekvens for at opfange signalet ordenligt. Da hørbar lyd ligger i frekvensområdet mellem 20 Hz og 20 kHz skal samplingfrekvensen være over 40 kHz. Samplefrekvensen for CD'er er valgt til 44,1 kHz, jævnfør IEC 60908 KILDE, da denne ligger over de tidligere nævnte 40 kHz. Aliasing er et udtryk for, når to eller flere signaler ikke kan kendes fra hinanden og vil i praksis betyde at CD-systemet ikke kan kende forskel på det ønskede signal og et forkert signal. KILDER! For at højere frekvenser ikke kan ødelægge signalet grundet aliasing, er det valgt at filtrere signaler over 20 kHz fra. Dette filter er et lavpas filter, der er implementeret således at frekvenser, som kan skabe problemer for samplern, bliver filtreret fra. Dette er også kendt som "anti-aliasing". Når sampleraten er valgt til 44,1 kHz er det muligt at genskabe frekvenser under 22,05 kHz, og

alle frekvenser herover skal derfor filtreres fra. Dette betyder også at alle de hørbare frekvenser kan opfanges.

Da nogle af problematikkerne ved genskabelsen af det originale signal, i forhold til sampleraten af perfekte samples, er undersøgt, er det interessant at undersøge hvilken indflydelse præcisionen af disse samples har.

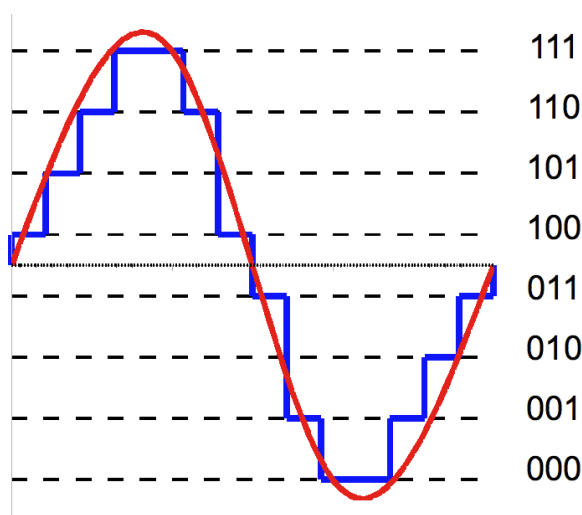
### 2.3.2 Nødvendig præcision for analog-digital konvertering

#### indsæt sejt billede med ADC Opløsning

Når et analogt signal skal lagres digitalt, skal det først konverteres til et digitalt. I de tidligere afsnit er behovet for konverteringen beskrevet og sampleraten er undersøgt. I dette afsnit undersøges den nødvendige bit-dybde til optagelse af analoge signaler.

Når det analoge signal skal konverteres, tages en Analog til Digital Konverter i brug (ADC). Denne virker ved at måle spændingen på et given tidspunkt, bestemt af sampleraten. Spændingen repræsenteres så med en digital værdi af et antal bits bestemt af opløsningen af ADC'en. Dette er illustreret på figur 2.5 herunder.

#### SQNR - Wiki



Figur 2.5: Sammenligning mellem en 3-bit repræsentation (blå) af et analogt signal (rød)

På figur 2.5 ses en periode af et sinusoidalt signal (den røde) og en 3-bit ADC (den blå). Som det kan ses er den digitale repræsentation ikke helt ens. Dette skyldes ADC'ens opløsning, da denne definerer de mindste værdier signalet kan inddeles i. En 3-bit ADC har maksimalt  $2^3 = 8$  værdier hvorimod en f.eks. 16-bit ADC har  $2^{16} = 65.536$ , hvilket medfører flere bit giver en bedre repræsentation, da afrundingsfejlen bliver mindsket. Størrelsen af afrundingsfejlen kaldes "Signal-to-Quantization-Noise Ratio" (SQNR). Antages at ADC'en er en ideal komponent hvor fejlmarginen for afrundingsfejl er  $\pm 1/2$  LSB og afrundingsfejlen ikke afhænger af området der måles i, kan SQNR udregnes som:

$$\text{SQNR} = 20 \cdot \log_{10}(2^Q) \quad [\text{dB}] \quad (2.3.1)$$

hvor:

$$Q = \text{ADC bit-dybde} \quad [.]$$

Ud fra ligning (2.3.1) kan ses en sammenhæng mellem antallet af bits i ADC'en og fejlmargenen. Eksempelvis er SQNR for en 6 bit ADC  $\approx 36,1$  dB. Sammenlignes denne med en 8 bit ADC ( $\approx 48,1$  dB) er forskellen  $\approx 12$  dB.

I praksis anvender lydstudier ofte 24 bit, eller derover, når der samples lyd, da dette giver en mere præcis AD-konvertering. Typisk er den software der arbejdes med, udviklet til at kunne håndtere endnu flere bit når der mixes, som derefter nedskaleres til eksempelvis 16-bit CD kvalitet.

### 2.3.3 Lagring af digital lyd

Efter konverteringen til digital lyd, skal dataene kunne gemmes, således at den kan tilgås efter ønske. Der er tre overordnede måder at gøre dette på. Det første metode er at gemme dataet **Til SKO - hvordan bøjtes data** i rå format. Det vil sige at der ikke sker nogen komprimering af data, hvilket betyder at denne metode kræver meget lagerplads. Til gengæld er lydkvalitet på sit højeste, da man på ingen måde behandler dataen **og at lyden kan genskabes hvis ellers nyquist er k**. En anden fordel ved denne datatype er at der ikke behøves nogen speciel form for læsning af dataen, hvilket vil gøre afspilning lettere. Filtyper der gemmes som rå data kan for eksempel være .wav-formatet (Waveform Audio File Format). Denne filtype inkluderer dog en header der indeholder information om filen, og derudover er den rå data delt op i bider i stedet for en lang bitstreng.

En anden måde at gemme sin data på er med lossless komprimering, hvor det er muligt at genskabe den originale lyd, uden at tabe data. Dette er filtyper som .flac (Free Lossless Audio Codec) eller .alac (Apple Lossless Audio Codec). Fordelen ved disse filtyper er at komprimeringen kan reducere filens størrelse med op til 50 % uden tap af data. Dette fungerer ved at beskrive et antal bit med et mindre antal bit.[2] Kravet for at afspille en sådan fil, er at afspilningsenheden skal vide hvordan den skal håndtere filen, hvilket også kaldes audiocodecs.

Den sidste metode er at gemme sin lyd i et lossy format. Ved denne metode vil der være tab af data, som ikke kan genskabes under afspilning af lyden. Til gengæld er filens størrelse meget lille i forhold til den rå data. I stedet for at gemme den nøjagtige data, gemmes der i stedet for data der minder om den originale.[2] Der er mange forskellige måder at komprimere sin data på, og de har alle deres fordele og ulemper. To af disse metoder kan være ved brug af A-law og  $\mu$ -law, hvor komprimeringen tager udgangspunkt i den menneskelige hørelse. Da den menneskelige hørelse er logaritmisk er den også mere sensitiv ved små signaler frem for store. Der afsættes derfor flere bits til små signaler hvilket giver dem en højere opløsning.[8]

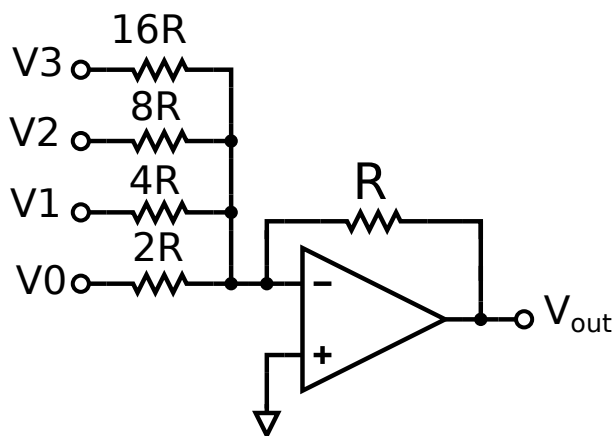
Ligesom filer med lossless komprimering kræver filer med lossy komprimering også kendskab til hvordan filen skal afspilles.

## 2.4 Digital-til-analog-konvertering

For at en trommemaskine skal kunne afspille lyd eller give et audiosignal som output, er det nødvendigt at kunne konvertere digitale lydsignaler til analoge. Dette kan gøres på forskellige måder. En række metoder til digital-til-analog-konvertering vil blive beskrevet i dette afsnit.

### 2.4.1 Binær vægtet D/A-konverter

En simpel måde at konvertere fra et tal repræsenteret som binære cifre eller bits, er ved en såkaldt binær vægtet D/A-konverter.



Figur 2.6: 4 bit D/A-konverter med binært vægtede modstande

Kredsløbet som kan ses på figur 2.6 er baseret på en summerende operationsforstærker. For dette kredsløb kan opstilles sammenhængen:

$$V_{out} = V_0 \cdot \frac{R}{2 \cdot R} + V_1 \cdot \frac{R}{4 \cdot R} + V_2 \cdot \frac{R}{8 \cdot R} + V_3 \cdot \frac{R}{16 \cdot R} = V_0 \cdot \frac{1}{2} + V_1 \cdot \frac{1}{4} + V_2 \cdot \frac{1}{8} + V_3 \cdot \frac{1}{16} \quad [\text{V}] \quad (2.4.1)$$

Hver inputspænding skal have værdien 1 eller 0 svarende til en spænding på  $V_{IO}$  og 0 V. Det kan ses i formel 2.4.1 at inputspændingernes forstærkning halveres, med MSB ved  $V_0$ . På denne måde bliver spændingen  $V_{out}$  en sum af spændingerne på inputtet, hvilket resulterer i den digitale til analoge konvertering med inkremerter af  $\frac{1}{16}$  i intervallet  $0 \text{ V} - \frac{15}{16} \cdot V_{IO}$ . Det kan ligeledes ses at jo flere parallelle inputs, jo højere opløsning kan der opnås.

fordele

simpel

kan laves selv

ulemper

krav til mange præcise modstandsværdier

?????

P.g.a. mange komponentværdier...

### 2.4.2 R-2R-netværk

En R-2R-D/A-konverter virker på mange måder som den binære vægtede operationsforstærker som beskrevet i afsnit 2.4.1. fordelene ved en R-2R-D/A-konverter er at den udelukkende anvender to komponentværdier. Modstande med størrelsen  $R$  og modstande med størrelsen  $2 \cdot R$ .

I appendiks A vises det at R-2R-konverteren er ækvivalent med den binært vægtede D/A-konverter.

For at tilføje yderligere inputspændinger skal der blot tilføjes to modstande med størrelsen  $R$  og  $2R$ .

bruger flere komponenter men har færre komponentværdier

### 2.4.3 Pulsbredde-modulation

En yderligere simpel måde at omdanne fra et digital signal til et analogt er ved at omdanne det digitale signal til et pulsbreddemoduleret signal<sup>1</sup>. mere om pwm kommer senere.... Dette signal kan derefter omdannes til et analogt signal, ved at lade det passere et analogt lavpasfilter med betydeligt lavere knæfrekvens end frekvensen på PWM-signalet. Dette vil medføre at outputtet vil fremkomme som et gennemsnit af inputsignalet, hvilket vil give en analog spænding.

Kvaliteten uddyb, din scrublord af D/A-konverteren kommer i høj grad til at afhænge af det analoge filter man vælger at anvende [1].

1

### 2.4.4 Oversamlende D/A-konverter

En meget anvendt måde at lave D/A-konvertere er ved at....

bruger en samplingfrekvens en faktor større end nødvendigt, til at opnå en større opløsning.

dette vil uddybes når vi engang fatter hvordan det virker.

## 2.5 Hardware i projektet

I dette projekt er det blevet valgt, at bruge en FPGA til at lave trommemaskinen. måske lidt tidligt at tage valg i projektet i teori delen... jeg kan ikke se noget godt alternativ men overvej at omformulere det. måske asic vs fpga vinklen? Dette er blevet valgt, da der på forhånd er blevet givet projekt gruppen en FPGA i form af en Papilio Duo, samtidig med at undervisningen tager udgangspunkt i denne.

Fordelen ved denne er, at den giver indsigt i hvordan bl.a. logic gates fungere sammen med andre digitale komponenter. Samtidige med at FPGA'en er et ekstremt ageilt værktøj, i den forstand at det ikke er nødvendigt at lave nyt print, finde nye komponenter og lignede hvis man har lavet fejl eller har brug for at implementere nye funktioner, FPGA'en kan blot omprogrammeres, selvfølgelig inden for FPGA'ens egen begrænsninger. Alternativet ville være, at man selv skulle sætte passende logic gates sammen. Men eftersom der utvivlsomt ville komme ændringer, som følger af ny viden eller fejl, er FPGA'en igen et godt valg. Brugen af FPGA'en vil også betyde, at man langt hurtigere kan starte designfasen, da eventuelle fejl ikke vil give et tilageslag, eller stort spildtid, men blot en omprogrammering af FPGA'en kan løse fejlen.

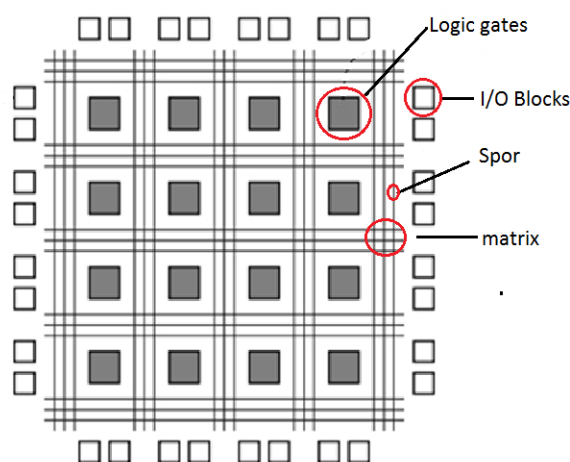
ER ASIC ÉN CHIP ELLER ER DET OGSÅ EN PASSENDE BESKRIVELSE AF PRINT SOM VI SELV LAVER? Når en færdig og fuldt funktionelt trommemaskine er lavet, kan man

<sup>1</sup>Ofte kaldet PWM

eventuelt lave den som en ASIC (application-specific integrated circuit), altså et anvendelses specifik integreret kredsløbs. ASIC er modparten til FPGA'en, som navnet antyder bliver en ASIC brugt til én specifik opgave. Det kan ikke omprogrammeres, men har en bedre performance og strøm forbrug.

### 2.5.1 Teori om FPGA

Felt Programmerbar Port Table som på engelsk hedder Field-Programmable Gate Array (FPGA), er i alt sin enkelthed et gate array **som er?** som er programmertbart, som det også fremgår af navnet. FPGA'en består af configurable logic blocks (CLB). disse CLB'er består af loop-up table(s) (LUT), Full adders (FA) og Flip Flops (FF). **vi skal lige finde ud af om vi bruger engelske eller danske udtryk**



Figur 2.7: General arkitektur i en FPGA

Disse CLB er forbundet med hinanden vha. forbindende spor, som gennem en forbindelsesmatrix giver mulighed for at forbinde CLB'sne på den måde man ønsker sig. Yderligere er dette også forbundet til programmerbar Input Output blokke (IOB) giver mulighed for at skabe forskellige in- og outputs fra eksterne kilder.

### 2.5.2 Papilio duo

Papilio duoen består af en Spartan 6 (XC6SLX9) FPGA fra Xilinx, som har 9.152 logic cells, 11.440 flip-flops og maksimalt 200 bruger I/O. Yderligere har den en AVR ATmega32U4 Microcontroller. Dette er en 8 bit microcontroller med en maksimal clock frekvens på 8/16 MHz (alt efter forsyningspænding), denne microcontroller er bl.a. brug på arduino leonardo.

**Skriv løbende når ny information opdages**

Nu er det muligt at afgrænse/inddele i moduler/lave problemformulering kom frisk

# Problemformulering 3

---

## 3.1 Projektafgrænsning

En trommemaskine er et redskab hvor der kan tilføjes mange elementer, og endnu flere fordybelsesområder. Der foretages derfor en projektafgrænsning for at mindske projektet omfang, således at et realistisk mål for projektet kan opnåes.

I projektet vil det være nødvendigt at omdanne digitale signaler til analoge hvilket kræver en DAC, men da projektets fokus ligger på **groove og lækre rytmer** vælges der at benytte en allerede eksisterende løsning.

Derudover vælges der at bruge allerede eksisterende lydsamples frem for at fremstille dem selv, grundet at de anvendte lyde ikke har stor indflydelse på resten af projektet. Der kunne i princippet benyttes lyde af hunde der gør, uden at hardwaren skulle ændres.

## 3.2 Modulinddeling

## 3.3 Problemformulering



# Kravspekifikation 4

---

## 4.1 Funktionelle krav

1. Skal kunne sekvensere
2. Skal kunne mikse lydsamples
3. Skal have knapper til valg af lydsample
4. Skal have knapper til valg af sample på ønskede takt

## 4.2 Krav til delmoduler

### 4.2.1 Sequencer

1. Indstille tempo mellem 50 BPM og 200 BPM.
2. if/else-afspil 16 gange.
3. Trommeslag fra brugeren skal maksimalt ligge 20 ms fra sekvensielle taktslag.

### 4.2.2 DAC

1. Skal findes hos Kenneth

## 4.3 Beskrivelse af accepttest

# Design 5

---

# Accepttest 6

---

# Diskussion 7

---

# Konklusion 8

---

# Perspektivering 9

---

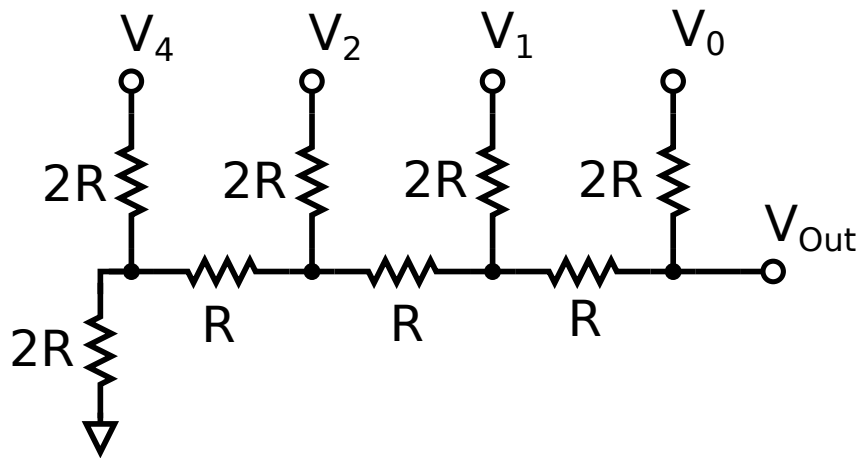
# Bibliografi

---

- [1] David M. Alter. *sing PWM Output as a Digital-to-Analog Converter on a TMS320F280x Digital Signal Controller*. Tek. rap. url: <http://www.ti.com/lit/an/spraa88a/spraa88a.pdf>. Texas Instruments, 2008.
- [2] Date-compression. *Theory*. Sidst set: 22/02/17. URL: <http://www.data-compression.com/theory.html>.
- [3] Ragnar Fleischmann Holger Hennig og Theo Geisel. „Musical rhythms: The science of being slightly off“. I: *Physics Today* (2012). URL: <http://physicstoday.scitation.org/doi/full/10.1063/PT.3.1650>.
- [4] Musikpedia. *Puls og tempo*. Sidst set: 17/02/17. URL: <http://www.musikpedia.dk/puls-og-tempo>.
- [5] Musikpedia. *Takt og underdeling*. Sidst set: 17/02/17. URL: <http://www.musikpedia.dk/takt-og-underdeling>.
- [6] Wikipedia. *Akai MPC2000*. Sidst set: 01/03/17. URL: [https://commons.wikimedia.org/wiki/File:Akai\\_MPC2000.jpg](https://commons.wikimedia.org/wiki/File:Akai_MPC2000.jpg).
- [7] Wikipedia. *Metre (music)*. Sidst set: 25/02/17. URL: [https://en.wikipedia.org/wiki/Metre\\_\(music\)](https://en.wikipedia.org/wiki/Metre_(music)).
- [8] Young-Engineering. *A-law and  $\mu$ -law companding*. Sidst set: 22/02/17. URL: [http://www.young-engineering.com/docs/YoungEngineering\\_ALaw\\_and\\_MuLaw\\_Companding.pdf](http://www.young-engineering.com/docs/YoungEngineering_ALaw_and_MuLaw_Companding.pdf).

# Udledning af DAK-kredsløb A

---



Figur A.1: et kredsløb

Her vises noget magi med thevenin ækvivalenter