



Designnotat

Tittel: Støyfiltrering

Forfattere: Eirik Mathias Silnes

Versjon: 1.0

Dato: 8. mai 2023

Innhold

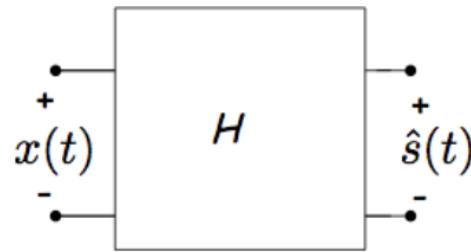
1 Problembeskrivelse	1
2 Prinsipiell løsning	2
3 Realisering og test	3
4 Konklusjon	8
5 Takk	8
Referanser	9

1 Problembeskrivelse

2489Hz

2 Prinsipiell løsning

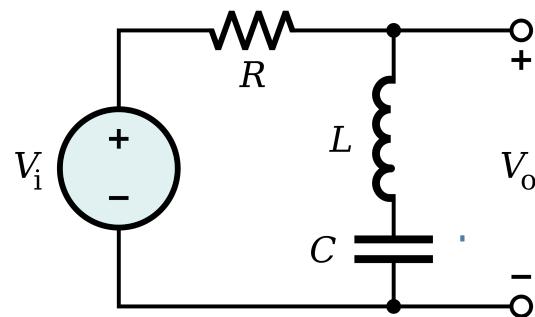
Et signal inneholder ofte flere frekvenser, dette gjelder også for lyd. Ettersom lydsignalet som er gitt ut er en musikkspilling så kan man forvente å finne frekvenser fra 20Hz til 20kHz [3] i lydsignalet. Det er derfor viktig å kunne filtrere ut de frekvensene som ikke er ønsket. Dette kan gjøres ved hjelp av et system som vist i Figur 1. Dette systemet kan moddeleres med likningen 1. [2]



Figur 1: Overordnet system

$$x(t) = s(t) + \omega(t). \quad (1)$$

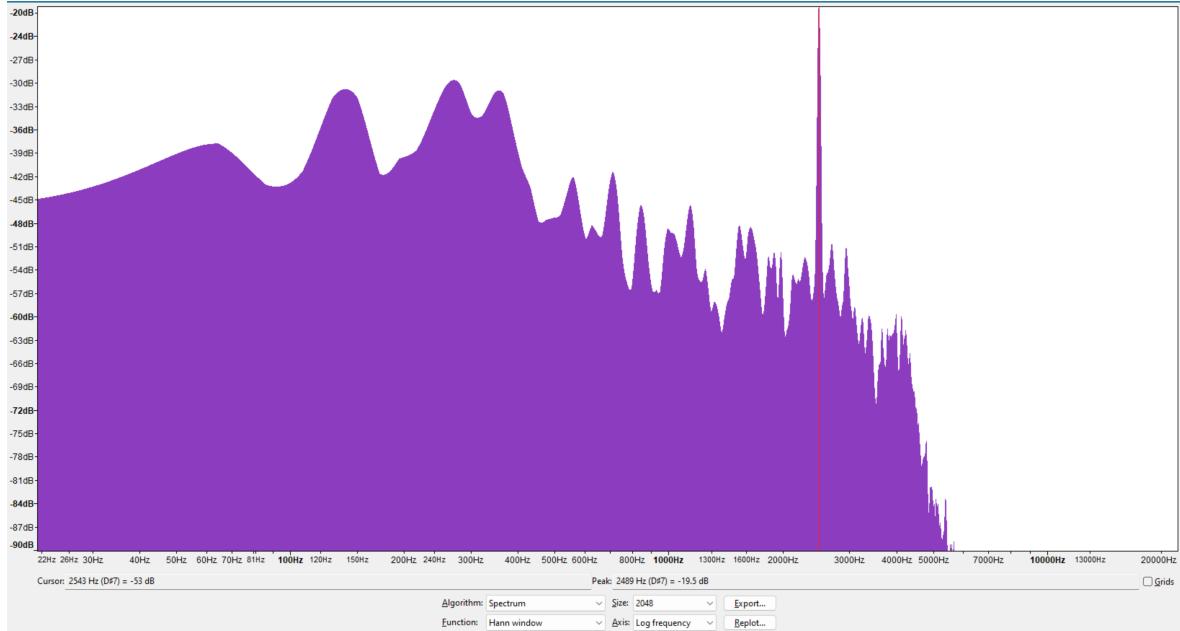
For å få til ønsket effekt så burde man benytte seg av et båndstopp filter som fjerner støy på en bestemt frekvens. Siden vi skal blokere ut samme frekvens hele tiden så kan vi lage et tidsinvariant filter. Det finnes mange forskjellige båndstopp filterdesign, både aktive og passive, noen populære varianter er: Butterworth-, Chebyshev- og Elliptic båndstoppfiltere, disse er relativt kompleks og krever flere komponenter enn det som er tilgjengelig. I tillegg finnes det et veldig enkelt RLC filter som kan brukes til å filtrere ut støy på en bestemt frekvens. Dette filteret er veldig enkelt å lage og krever kun 3 komponenter. Dette filteret er et 2. ordens filter som består av en kondensator, en spole og en motstand som vist i figur 2



Figur 2: RLC filter

3 Realisering og test

Når lydfilen blir spilt av er det en veldig tydelig høy tone som ligger oppå musikken, og ved hjelp av lydredigeringsprogrammet Audacity så kan man lett finne hvilken frekvens denne pipelyden er på. I spektrumsanalysen vist i figur Figur 3 ser man tydelig hvor pipelyden ligger. Under grafen er det et tekstfelt med Peakfrekvens rundt pekeren. 2489Hz er frekvensen til pipelyden.



Figur 3: Frekvensanalyse med Audacity

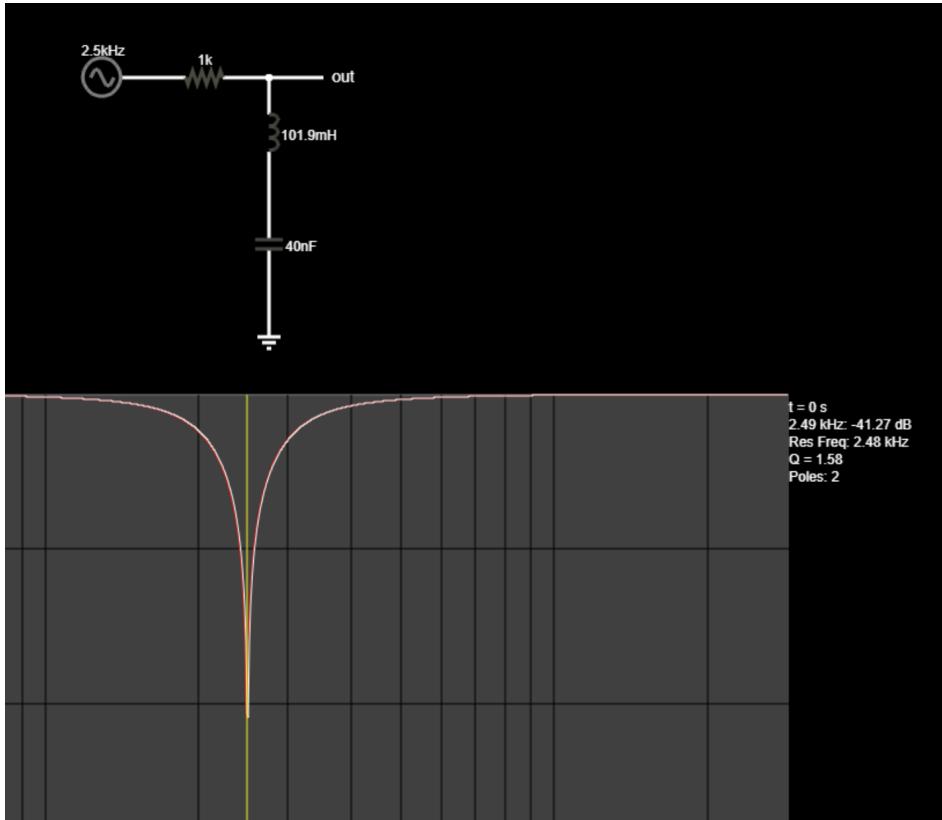
Med utgangspunkt i hvilke komponenter som er tilgjengelig så er store deler av filteret allerede bestemt. Spolen som ble brukt var på $101.9mH$ som ble målt og verifisert i tidligere arbeid. Motstanden settes til $1k\Omega$ i begynnelsen av testingen ettersom den kun skalerer spenningen ut av filteret og er defor ikke med på å bestemme frekvensen til filteret. For å beregne verdien til kondensatoren så tar man utgangspunkt i formelen for resonansfrekvensen til et RLC filter Ligning 2 og løser det med hensyn på kondensatoren Ligning 3. Kondensatoren blir da $C = 40nF$ siden den kan lett lages ved å parallellkoble 4 kondensatorer på $10nF$ hver.

$$f_0 = \frac{1}{2\pi\sqrt{LC}} \quad (2)$$

$$C = \frac{1}{(2\pi f_0)^2 L} \quad (3)$$

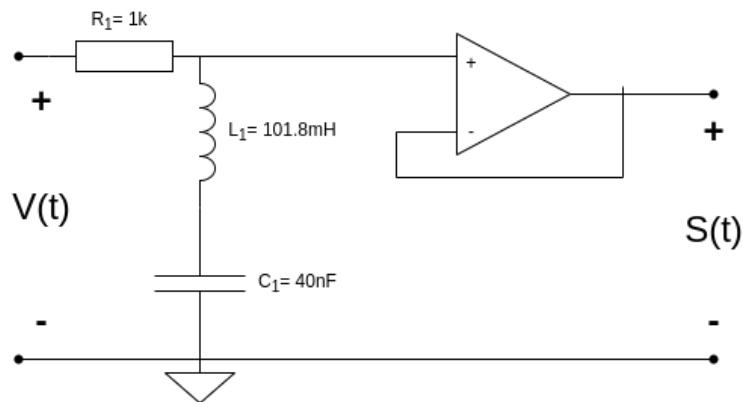
$$C = \frac{1}{(2\pi \cdot 2489Hz)^2 \cdot 101.9mH} = 4.01645 \cdot 10^{-8} \approx 40nF \quad (4)$$

For å verifisere at filteret fungerer som det skal så ble det simulert i Falstad og det ble laget et bilde av simuleringen som er vist i figur Figur 4.

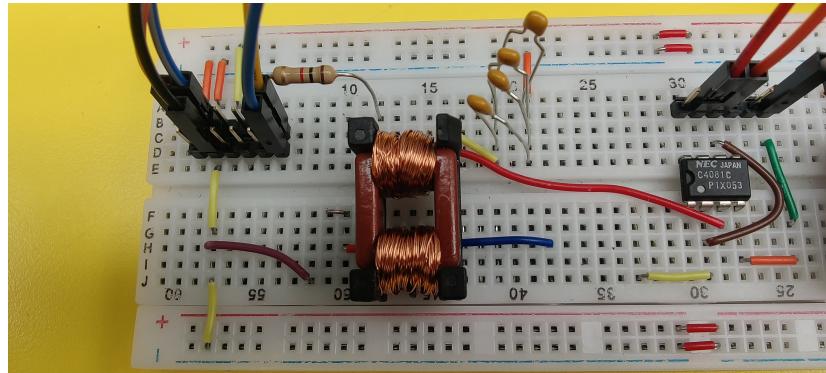


Figur 4: Simulert filter i Falstad

For å teste filteret ble Analog Discovery 2 brukt for å generere lydsignalet og måle spenningen på utgangen av filteret. For å sørge for at eventuelle hodetelefoner eller andre systemer utenfor kretsen skal påvirke systemet blir det lagt til en op-amp for å buffre signalet. I figur Figur 5 er det vist et bilde av hvordan filteret ble koblet opp og i figur Figur 6 vises realiseringen av filteret.

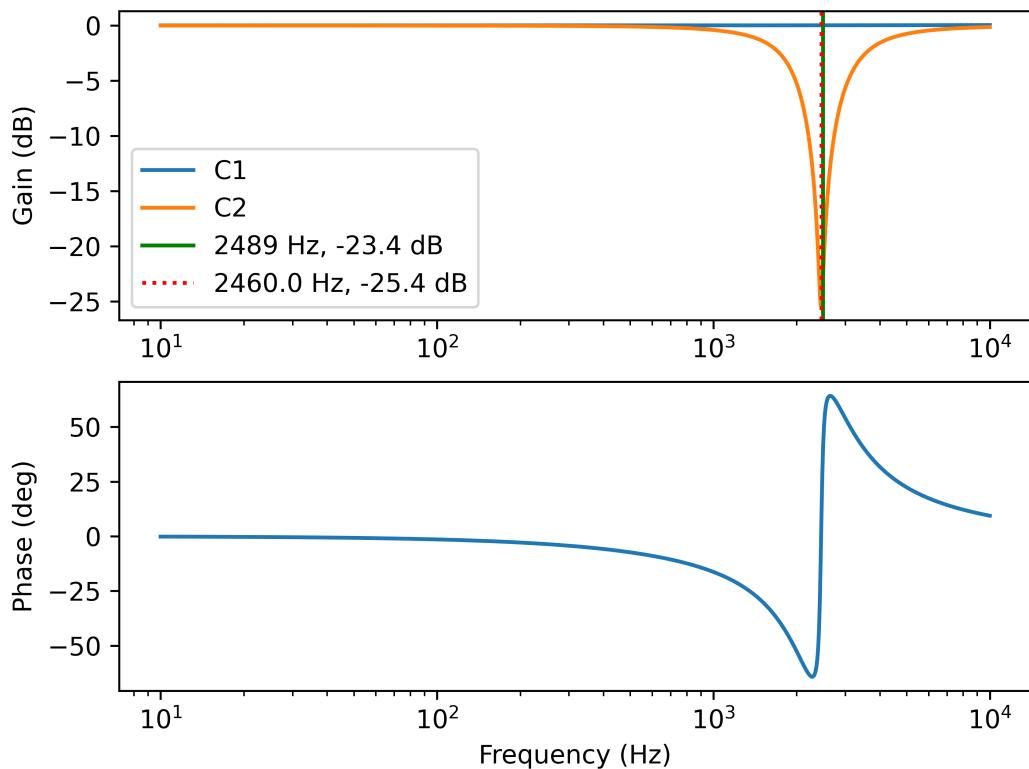


Figur 5: Skjematikk av filteret



Figur 6: Kobling av filteret

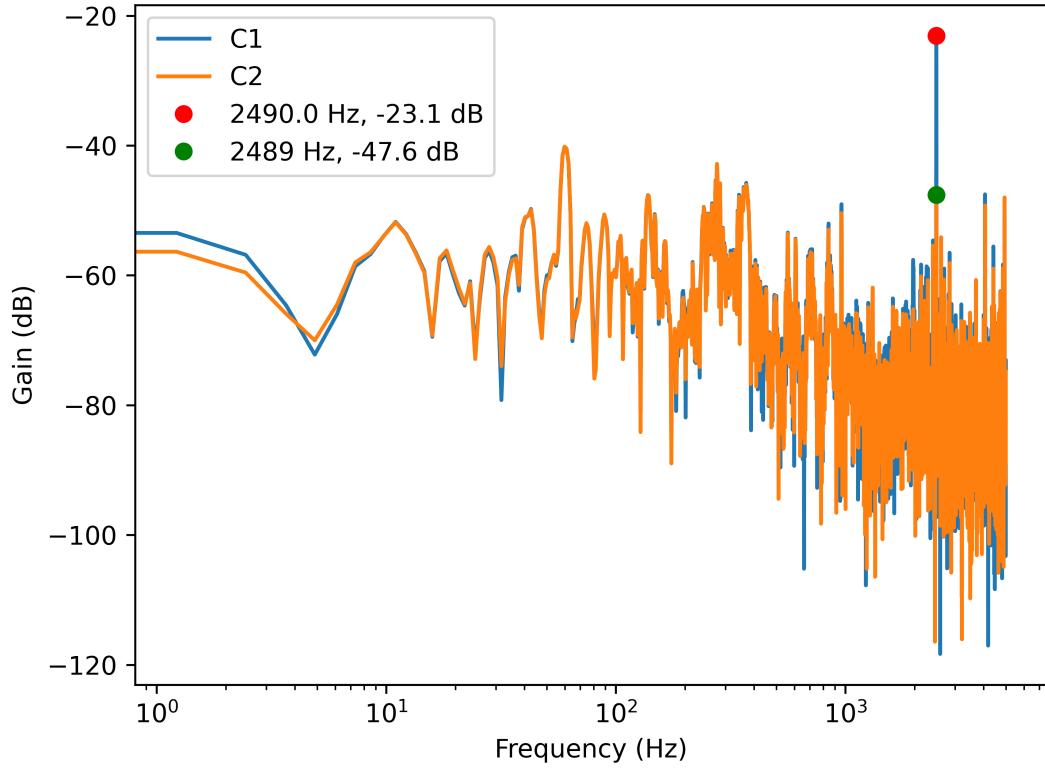
Først ble filteret testet ved å kjøre en netverksanalyse på det og generere et bodeplot av dataen i python. I figur Figur 7 er det vist et bodeplot av filteret. Det er tydelig at filteret har mest demping på 2460Hz men det er også 23 dB demping på 2489Hz.



Figur 7: Bodeplot av filteret

For å teste hvorvit realiseringen faktisk fungerer så ble lydfilen spilt gjennom filteret og en spektrumsanalyse ble gjort på utgangssignalet til filteret med WaveForms. Pipelyden er tydelig redusert i lydklippen og med spektrumsanalysen i figur Figur 8 er det tydelig at

pipelyden er redusert med 24.5dB.

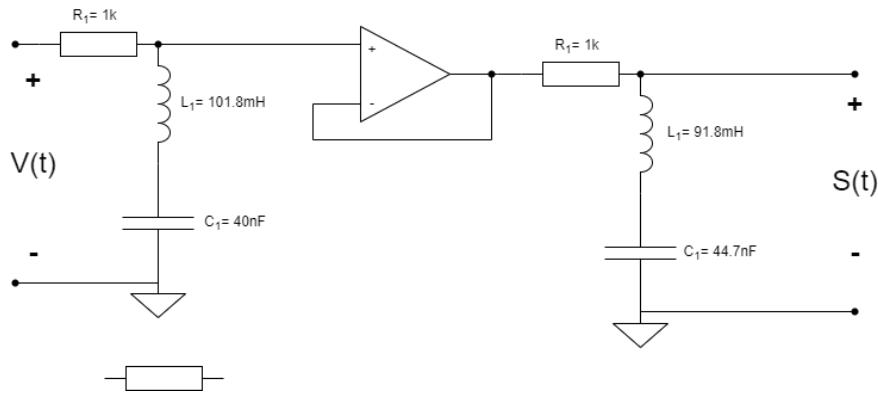


Figur 8: Spektrumsanalyse av filteret

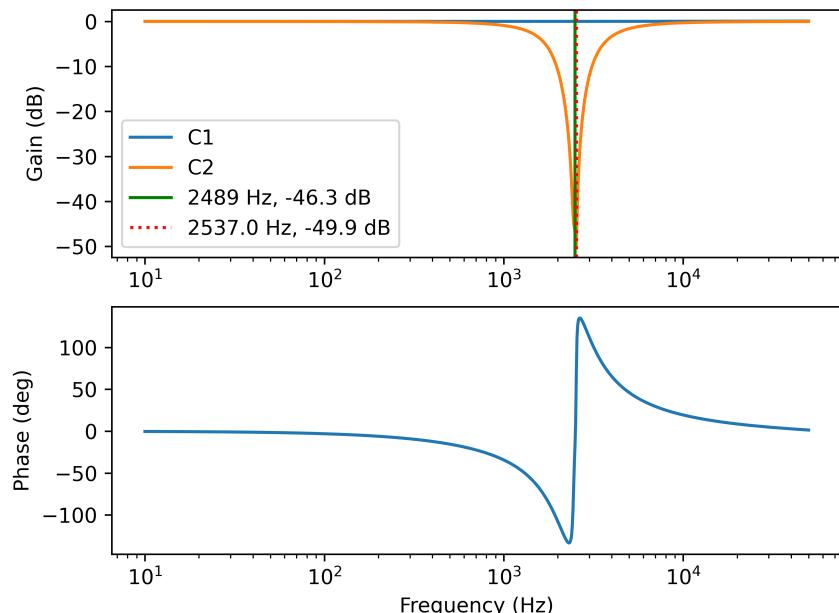
Problemstillingen var å forbedre signalkvaliteten til en lydfil, men siden pipetonen fremdeles kan høres så anses det derfor som en delvis løsning. For å få en fulstendig løsning så blir det konstruert et identisk filter i serie med det første filteret. Dette filteret vil ha samme resonansfrekvens, men grunnet unøyaktige komponenter så må det utregnes en ny kondensatorverdi med formelen i Ligning 3. Den nye spolen har en verdi på $91.8mH$. Den optimale kondensatorverdien blir da $44.5nF$ som sett i utregning Ligning 5, men kondensatoren blir realisert med $C = 44.7nF$ som er vist i figur Figur 9.

$$C = \frac{1}{(2\pi \cdot 2489\text{Hz})^2 \cdot 91.8mH} = 4.45398 \cdot 10^{-8} \approx 44.5nF \quad (5)$$

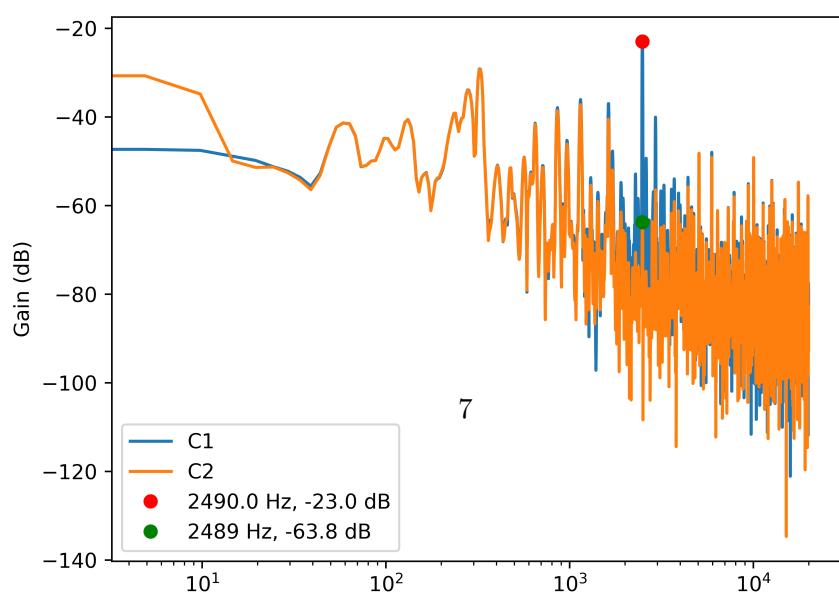
Når det blir gjort identiske tester kan man se tydelig forbedring i lydklippet, og det pipelyden er ikke lenger hørbar. I bodeplottet, Figur 10, og i spektrumsanalysen, Figur 11 kan det leses ut en ny dempning på $40.8dB$.



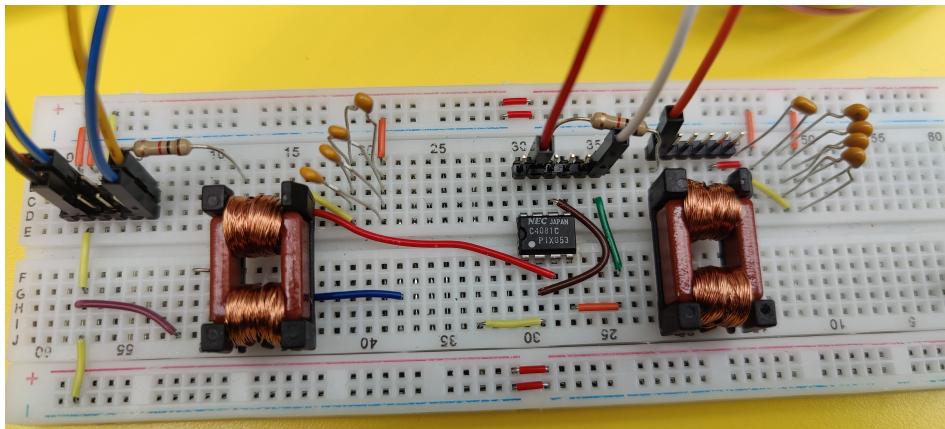
Figur 9: Design nummer to



Figur 10: Bodeplot av filteret



Et bilde av den realiserte kretsen er vist i figur Figur 12.



Figur 12: Design nummer to

4 Konklusjon

I dette designnotatet har det blitt designet et analogt 2.ordens båndstopp filter for å fjerne støy på et lydsignal. Det viste seg å ha for lite demping og ulyden i filen var fermdeles hørbar, så signalet ble buffret og et identisk filter med indentisk knekkfrekvens ble lagt til. Dette ga en dempning på ca. $40dB$, og oppnår kravene til oppgaven.

5 Takk

Takk til Reidar Nerheim for veiledning i dette designnotatet

Referanser

- [1] L. Lundheim, *Designprosjekt 3*, Institutt for elektronisk systemdesign, NTNU, 2023.
- [2] L. Lundheim, *Enkle prinsipper for støyfjerning*, Teknisk Notat, Elsys-2016-LL-2, NTNU, 2016.
- [3] Jeff Smoot, *Understanding Audio Frequency Range in Audio Design*, CuiDevices, www.cuidevices.com, 2023