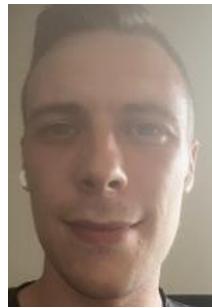


Metalldetektor projekt

Gruppe 1

34621

Electromagnetic sensors and digital signal processing
Danmarks Tekniske Universitet



Sebastian Sørensen,
s233986



Oliver Holm,
s233988



Bilal Alali,
s171678

January 20, 2026

Indholdsfortegnelse

1	Introduktion	2
2	Analyse	3
3	Design	4
3.1	Analogt design	4
3.1.1	Energiberegninger	4
3.1.2	Spoleberegninger	4
3.1.3	Power Amplifier	6
3.1.4	Filtrering af spolesignal	7
3.1.5	Forstærkning af spolesignalet	8
3.2	Digitalt design	9
3.2.1	Moduldiagram og introduktion	9
3.2.2	Brugerinteraktion	10
3.2.3	DFT algoritme og sampling	10
3.2.4	Tilstandsmaskine	10
3.2.5	Digital signal behandling	10
4	Implementering og test	10
5	Ansvarsområder (hvem har lavet hvad)	10
6	Konklusion	11
7	Appendix	11
7.1	Gantt kort	11
7.2	Møderefirater	11

1 Introduktion

Dette projekt omhandler udviklingen af en metaldetektor, herunder skal vi selv designe og implementere hhv. software og hardware. Software delen skrives i embedded C, på en Arduino AtMega2560, hvor data vi sampler fra spolerne skal behandles og outputtes til brugeren.

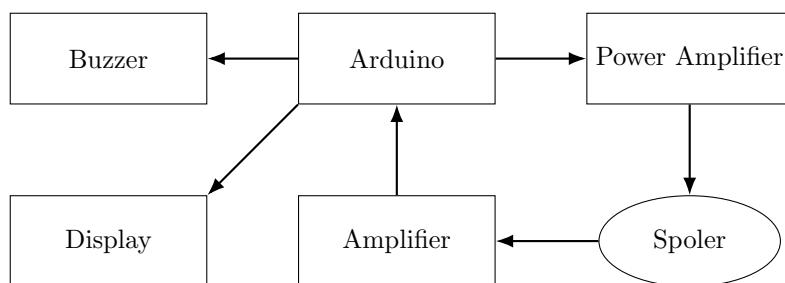
Hardware-delen af projektet omhandler primært design og implementeringen af sensoren til detektion af metaller, samt diverse kredsløb, der har til formål at drive spolen, filtrere samt forstærke signalet. Systemet skal outputte en beregnet amplitude og fase til OLED-display, men vi har valgt også at implementere en buzzer til audio-feedback.

Herunder ses kravspecifikationen metaldetektor projektet har haft til formål at opnå.

Nr:	Navn:	Beskrivelse:	Prioritet
1	Amplitude/Fase detektion	Det skal være muligt at kunne detektere amplitude og fase i det reflekterede signal	1
2	Metal type	Skal kunne skelne jern fra kobber, messing og aluminium	1
3	Distance krav	Skal kunne detektere en jerngenstand (radius på ca. 15mm. Og en længde på 50mm.) i en dybde på 50mm. I fri luft	1
4	Strømforsyning	Hele metaldetektoren skal kunne køre på et 9 volts batteri (batteristørrelse E/6LR61) - kun et batteri.	1
5	Funktionstid	Metaldetektoren skal kunne køre minimum 100 minutter på batteriet og den ubelastede rest-spænding skal være større end 6 volt efter de 100 minutter	1
6a	Processor design	Kredsløbet skal være baseret på en Arduino	1
6b	Hardware design	Kredsløbene til at drive spolerne og forstærke det modtagne signal skal bygges med diskrete komponenter såsom: transistorer, operationsforstærkere, m.m.	1
7	Muligt software design	Brug Arduino'ens ADC og kontroller denne med timer interrupts	2
8a	Bruger interface display	Skal kunne udskrive amplitude og fase i et display	1

8b	Display udlæsningsstabilitet	Værdierne i displayet skal være læsbar for et flertal af brugere (spørg om hjælp)	1
8c	Forbedret display udlæsningsstabilitet	Der anvendes et FIR eller IIR filter til at lavpas-filtrere (midle) udlæsningen.	2
9a	Bruger interface primære knapper	Der skal være en start/stop knap	1
9b	Bruger interlace sekundære knapper	Der skal være en nulstillingsknap (fratrækker værdierne når detektoren ikke er i nærheden af metal)	1
10	Detektions princip	Very Low Frequency (VLF)	1
11	Samplingsfrekvens	8 kHz +/- 100Hz	1
12	Detektionsfrekvens	2 kHz +/- 100Hz	1
13	Metaldetektorudformning	3D-printet/trækkonstruktion/plast	2
14	Spolernes bærenhed	3D-printet spoleform /læserskåret plastik	2
15	Spoleudformning	Spolerne anvender en centreret udformning med sende-spolen yderst og feedback/modtager spolen inderst.	2
16	Modtager spolens selvinduktans	Modtager spolen skal have en selvinduktans på mindst 10 mHy.	1

2 Analyse



Figur 1: Overordnet moduldiagram af systemet

3 Design

3.1 Analogt design

3.1.1 Energiberegninger

Vi undersøgte os frem til at batteriet har en kapacitet på 550mAh. Da batteriet aflades fra 9V til 6V, har vi lavet udregningen:

$$E_{tot} = V \cdot I \cdot 3600 \iff E_{tot} = \frac{9V + 6V}{2} \cdot 0.55A \cdot 3600s = 14850J$$

Ved måling af strømforbruget finder vi at OLED-display og Arduinoen forbuger $\approx 70.5\text{mA}$. Da vi skal kunne drive systemet i 100 minutter, kan vi derfor beregne A

$$E_{MCU} = 5V \cdot 70.5 \cdot 10^{-3}\text{A} \cdot (100 \cdot 60)\text{s} = 2115\text{J}$$

Vi reserverer 10% af batteriet som buffer, for at være sikker på at overholde kravene. Resten af energien kan bruges til TX-spolen.

$$E_{TX} = E_{tot} \cdot (1 - 0.15) - E_{MCU} = 11250\text{J}$$

3.1.2 Spoleberegninger

TX-spole

Ved resonans kan spændingen gennem spolen approksimeres ved grundtonen i fourierrækken

$$a_n \approx 2 \left(\frac{A}{n\pi} \right) \sin \left(\frac{n\pi}{2} \right) = 2 \left(\frac{9V}{\pi} \right) \sin \left(\frac{\pi}{2} \right) = 5.73V$$

Hvorved RMS-spændingen kan bestemmes

$$V_{RMS} = \frac{a_n}{\sqrt{2}} = 4.05V$$

Derved kan vi nu beregne strømmen vi kan sende igennem spolen

$$I_{TX} = \frac{E_{TX}}{t \cdot V_{RMS}} = \frac{11250\text{J}}{(60 \cdot 100)\text{s} \cdot 4.05\text{V}} = 0.4628\text{A}$$

Radiusen på TX-spolen vælges til $r_{TX} = 10\text{cm}$, med $D_{TX} = \emptyset 0.4\text{mm}$. Derved kan antallet af viklinger beregnes ved

$$R_{TX} = \frac{V_{RMS}}{I_{TX}} = 8.754\Omega$$

$$N_{TX} = \frac{R_{TX} \cdot \left(\frac{D_{TX}}{2} \right)^2}{2 \cdot r_{TX} \cdot \rho_{cu}} = \frac{8.754\Omega \cdot \left(\frac{0.4\text{mm}}{2} \right)^2}{2 \cdot 10\text{cm} \cdot 1.77 \cdot 10^{-8}\Omega \cdot m} = 98.9$$

hvilket afrundes til $N_{TX} = 99$.

Selvinduktansen i spolen findes ved

$$L = \frac{0.394 \cdot r^2 \cdot N^2}{9 \cdot r + 10A} = \frac{0.394 \cdot 10^2 \cdot 99^2}{9 \cdot 10 + 10} = 3.855\text{mH}$$

For at kunne lave Arduino'ens firkantsignal til en sinuskurve anvendes et resonanskredsløb, som har resonansfrekvens på 2kHz. Størrelsen af kondensatoren findes ved at løse ligningen

$$f = \frac{1}{2\pi\sqrt{LC}} \iff 2\text{kHz} = \frac{1}{2\pi\sqrt{3.855\text{mH} \cdot C}} \implies C = 1.642\mu\text{F} \quad (1)$$

Magnetfeltet fra TX-spolen beregnes, da bucking-spolen skal udligne B-feltet fra TX-spolen, for på den måde at tilsikre at vi kun mäter på magnetfeltet fra metallet i detektoren.

$$|B| = \frac{\mu_0 I_{TX} N_{TX}}{2r_{TX}} = \frac{4\pi \cdot 10^{-7} \frac{\text{H}}{\text{m}} \cdot 0.4628\text{A} \cdot 99}{2 \cdot 10\text{cm}} = 287.6\mu\text{T}$$

Bucking-spole

Derved kan Bucking-spolen beregnes. Vi vælger en radius på $r_{Buck} = 5\text{cm}$, og antallet af viklinger beregnes ved

$$|B| = \frac{\mu_0 \cdot I_{TX} \cdot N_{Buck}}{2 \cdot r_{Buck}} \iff 287.6\mu\text{T} = \frac{4\pi \cdot 10^{-7} \cdot 0.4628\text{A} \cdot N_{Buck}}{2 \cdot 5\text{cm}} \implies N_{Buck} = 49.458$$

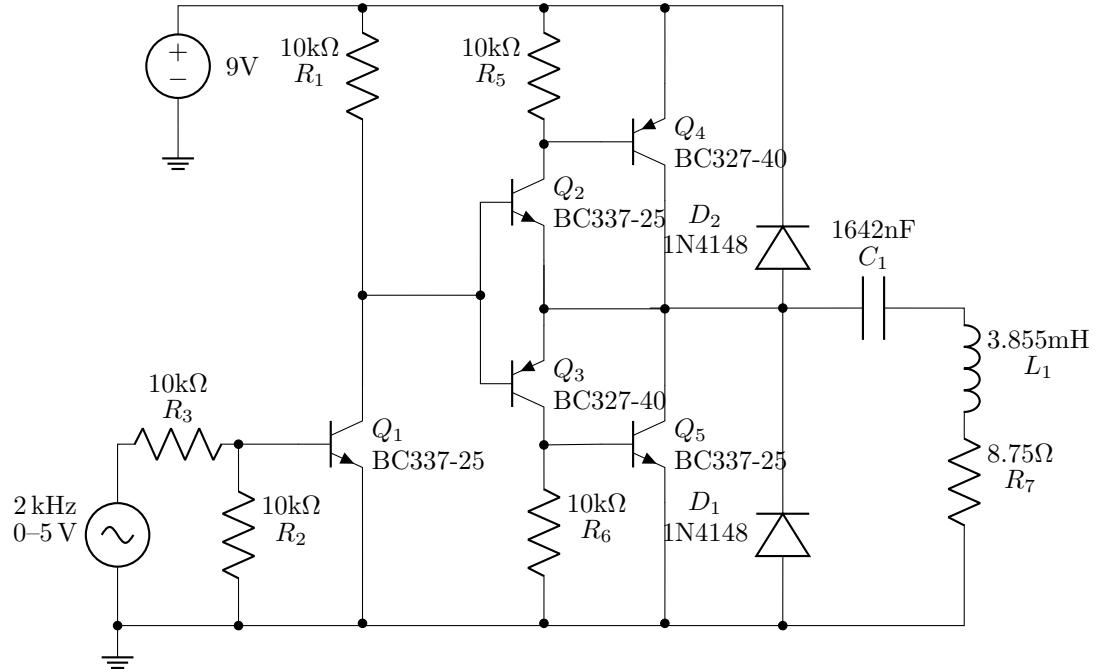
hvilket afrundes til $N_{Buck} = 50$.

RX-spole

For at opnå en god følsomhed på sensoren, er vi opmærksomme på at induktansen i RX-spolen skal være minimum $L_{RX} = 10\text{mH}$. Derved beregnes antallet af viklinger ved

$$10 \cdot 10^3 = \frac{0.394 \cdot r_{RX}^2 \cdot N_{RX}^2}{9r_{RX} + 10A_{RX}} = \frac{0.394 \cdot 5^2 \cdot N_{RX}^2}{9 \cdot 5 + 10} \implies N_{RX} = 236$$

3.1.3 Power Amplifier

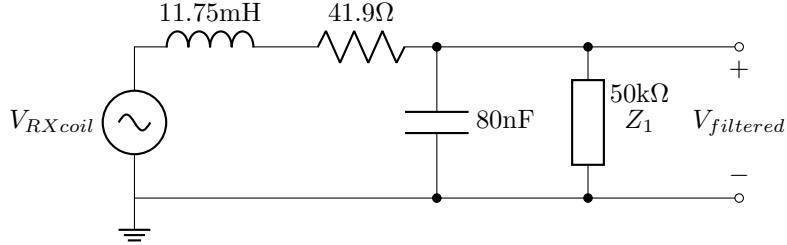


Figur 2: Power amplifier kredsløb til at øge strømmen gennem TX-spolen.

Ovenfor ses vores design af effektforstærkeren, som bruges til at forstærke signalet fra Arduino'en til at kunne drive TX-spolen. Kredsløbet virker ved:

- Firkantsignalet sendes ind i forstærkeren gennem en modstand som beskytter Arduino'en og styrer strømmen ind i transistoren Q_1 .
- Transistor-konfigurationen fungerer som en forstærker som via. en styrestrøm kontrollerer en større strøm fra batteriforsyningen (9V batteriet), som driver TX-spolen.
- TX-spolen (L_1) genererer et magnetfelt ved resonansfrekvensen
- Kondensatoren (C_1) er beregnet (Ligning 1) således at der opstår resonans ved detektionsfrekvensen på 2kHz
- Modstanden R_7 repræsenterer den ohmske modstand i TX-spolen.
- Dioderne D_1 og D_2 beskytter kredsløbet fra den induserede spænding i spolen.

3.1.4 Filtrering af spolesignal



Figur 3: Kredsløb til at filtrere støj væk ($\approx \frac{F_s}{2} = 4\text{kHz}$) inden signalet bliver forstærket (som vist i Figur 5). Modstanden på $50\text{k}\Omega$ repræsenterer indgangsimpedansen i operationsforstærkerkredsløbet.

Som det ses i figuren, er $V_{filtered} = V_{Z_1}$. Filterets overføringsfunktion kan opstilles ved almindelig spændingsdeling, dvs.

$$H(\omega) = \frac{Z_1 \cdot \frac{1}{j\omega C}}{j\omega L + R_1 + Z_1 \cdot \frac{1}{j\omega C}}$$

hvorefter fasen kan analyseres ved

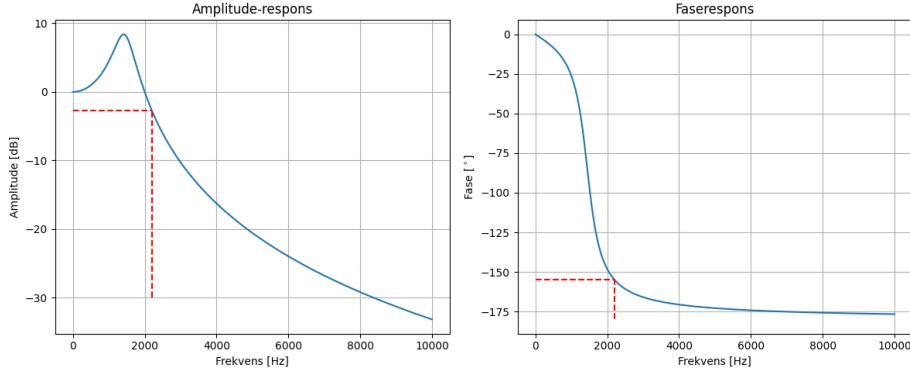
$$\phi(\omega) = \tan^{-1} \left(\frac{\Im\{H\}}{\Re\{H\}} \right)$$

og amplitude responset ved

$$H_{dB} = 20 \log 10(H)$$

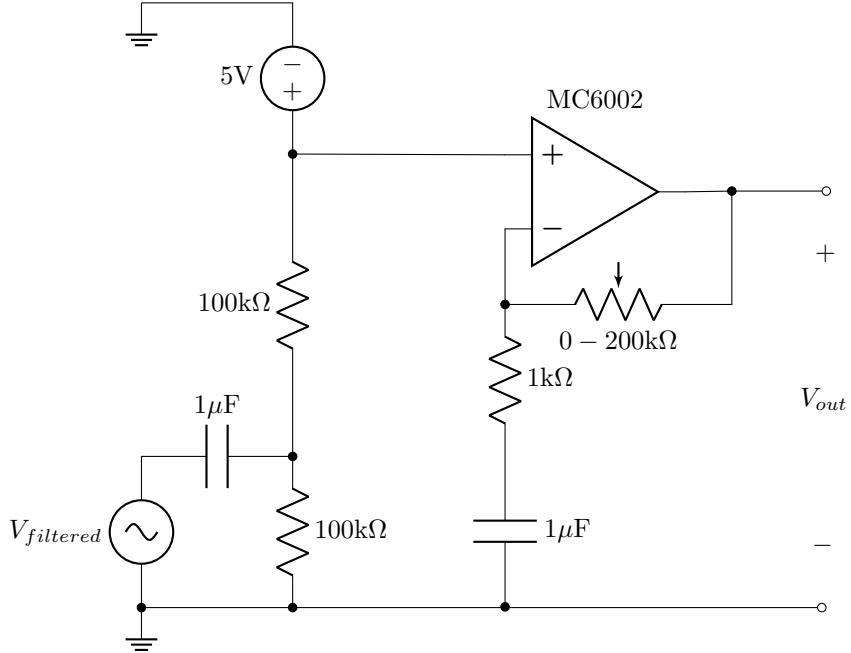
Ifølge Shannon's sampling teorem ved vi at $F_{max} < \frac{F_s}{2}$. Da vi ikke har noget signal som ligger over $\frac{F_s}{2} = 4\text{kHz}$ anvender vi dette filter, for at undgå problemer med aliaseringer. Filteret er designet således at fasen bliver påvirket mindst muligt, da det netop er fasen vi bruger til at skelne mellem materialer.

Figur 4: RX-filterets fase- og magnitude respons, med markering af 2.2kHz



Da metaldetektorens følsomhed overfor frekvenser afhænger af bl.a temperatur, har vi valgt at designe filteret efter et udgangspunkt på 4.2kHz. Som det ses i plottet får vi dæmpt det værste af eventuelt højfrekvent støj, samtidigt med at vi aflæser fasen nogenlunde præcist med udgangspunkt i 4.2kHz.

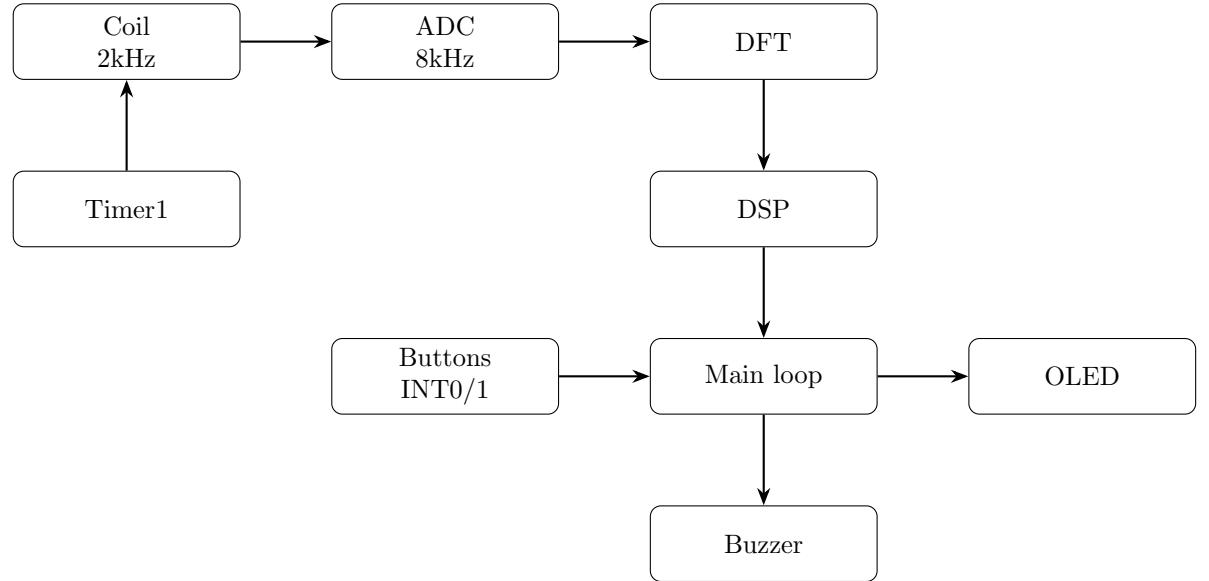
3.1.5 Forstærkning af spolesignalet



Figur 5: Kredsløb til at forstærke den filtrerede spænding fra RX-spolen op, inden det læses ind i MCU ADC'en.

3.2 Digitalt design

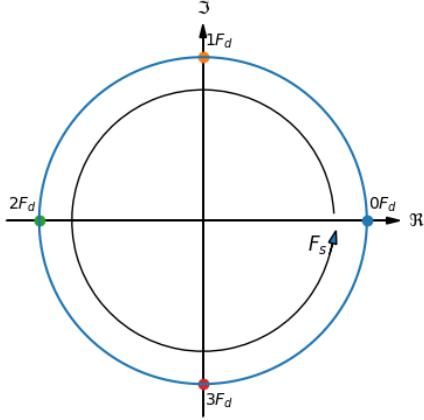
3.2.1 Moduldiagram og introduktion



Figur 6: Overordnet moduldiagram af systemet

3.2.2 Brugerinteraktion

3.2.3 DFT algoritme og sampling



Figur 7: Grafisk illustration over princippet bag DFT-algoritmen. Ved at vælge $F_d = \frac{F_s}{4}$, undgår vi at lave komplikerede komplekse beregninger.

3.2.4 Tilstandsmaskine

3.2.5 Digital signal behandling

4 Implementering og test

5 Ansvarsområder (hvem har lavet hvad)

Fælles

- Vikling af spoler

Bilal

- 3D-print
- Power Amplifier
- Spoleberegninger
- Energiberegninger

Sebastian

- C-programmering

- Test og simulation af C-program
- RX-filter
- Grafik til rapport (plots, kredsløbstegninger, indskrivning af rapport)
-

Oliver

- C-programmering
- Test og simulation af C-program
- Implementering af kredsløb

6 Konklusion

7 Appendix

7.1 Gantt kort

7.2 Møderefater