

LABOPPGAVER

TTT4280

LABORATORIEOPPGAVER

7. januar 2026

Innhold

I Systemoppsett	3
1 Oppsett av Rpi	4
2 Oppkobling av AD-konverter	4
2.1 Forberedelsesoppgaver	4
2.2 Laboppgaver	5
2.3 Krav for godkjent lab	5
3 Referanser/Datablad	6
 II Akustikk	 7
4 Teori	8
4.1 Krysskorrelasjon	8
4.2 Deteksjon av innfallsretning	10
4.3 Hva om lydkilden ikke er langt vekk (dvs ikke en planbølge)?	14
5 Forberedelsesoppgaver til laben	16
5.1 Forberedelsesoppgave 1	16
5.2 Forberedelsesoppgave 2	16

6 Laboppgave	17
6.1 Krav for godkjent lab	17
7 Appendix - løsning av lign. (II.9) via matriseinvertering	18
7.1 Numerisk eksempel	18
 III Radar	 20
8 Teori	21
9 Oppgaver	29
9.1 Forberedelser	29
9.2 Forberedelse: Test radaren	29
9.3 Laboppgave: Doppler radar	29
9.4 Demonstrasjon	30
 IV Optikk	 31
10 Teori	32
10.1 Introduksjon	32
10.1.1 Måling av puls	32
10.1.2 Generell biooptikk	34
10.2 Kræsjkurs i regning på fotontransport	36
10.2.1 Diffusjonslikningen	36
10.2.2 Penetrasjonsdybde	37
10.2.3 Måling av lys på overflaten	37
10.2.4 Regning på pulsutslag/blodkontrast	39
10.3 SNR i pulssammenheng	40
10.3.1 SNR i frekvensdomenet	41
10.3.2 Avsluttende bemerkninger om SNR	43
10.4 Forskningslitteratur	43
10.4.1 Fjernmåling av puls med kamera	43
10.4.2 Fotontransport	43
11 Forberedelsesoppgaver	44
11.1 Oppgave 1: Fotontransportberegninger	44
11.2 Oppgave 2: SNR	44
12 Laboppgaver	44
12.1 Samle og analysere data	45
12.1.1 Del 1: Basismålinger	45
12.1.2 Del 2: Robusthetstester	46
12.2 Krav for godkjent lab	46

Del I

Systemoppsett

I denne laboppgaven skal dere sette opp målesystemet som er basert på en Raspberry Pi 3B single board computer med eksterne AD-konvertere. Dette målesystemet skal dere senere bruke til å sample analoge signaler fra henholdsvis 3 stk mikrofoner og deretter 2 kanaler med utgangssignal fra radaren.

Ta med Digilent Analog Discovery 2 til hver lab for feilsøking og testing av systemet.

1 Oppsett av Rpi

I denne laboppgaven skal dere koble opp en Raspberry Pi. Denne skal fungere som måleplattformen deres og etterhvert kommunisere med noen AD-konvertere og et kamera.

Følg instruksene i labmanualen for å sette opp Raspberry Pi. Vi anbefaler at dere laster ned Raspberry Pi OS før dere kommer på lab.

Når dere er ferdige med laboppgaven skal dere kunne:

- koble til Raspberry Pi via SSH
- overføre filer til og fra Raspberry Pi med en SFTP-klient
- navigere et filsystem med terminalen

2 Oppkobling av AD-konverter

I denne laboppgaven skal dere koble tre AD-konvertere til Raspberry Pi og gjøre målinger for å verifisere at systemet fungerer som det skal. Følg fremgangsmåten i labmanualen og svar på oppgavene under.

2.1 Forberedelsesoppgaver

1. Tegn et blokkskjema for hvordan en skal kople opp hele systemet med Raspberry Pi og 3 stk AD-konvertere. Blokkskjemaet skal inneholde signalveier, klokkesignal, kommunikasjonsporter, og ledningsopplegg for spenningstilførsel til kretsene. I alle blokkene i skjemaet skal en markere hvilke pinner som brukes. Vi anbefaler at dere bruker draw.io eller tikzmaker.com til å tegne blokkskjema.
2. Bli kjent med databladet for AD-konverteren vi bruker.
 - (a) Hvor mange klokkesykluser er det minste vi må bruke for å ta én punktprøve og overføre dataene fra den tilbake til RPi? Hvilkens samplingsfrekvens vil systemet få?
 - (b) Gitt en driftspenning på 3.3 V (V_{dd}), hva blir AD-konverterens oppløsning? Oppgi svaret i mV.

- (c) Gitt en driftspenning på 3.3 V (V_{dd}) og jord på V_{ss} , hvor mye større spenning enn V_{dd} , evt mindre enn V_{ss} , kan en pinne på AD-konvert dre tåle?
3. I dette laboppsettet benytter vi oss av såkalt Direct Memory Access (DMA). Hva er den store fordelen med dette (evt. hvilket problem relatert til sampling av data med Linux løser dette for oss)?

2.2 Laboppgaver

1. I labmanualen blir dere bedt om å bygge et enkelt lavpassfilter for å redusere støy på 3.3V-linjen for spenningsforsyning til de analoge delene av systemet. Forklar hvordan filteret fungerer, beregn knekkfrekvens og tegn frekvensresponsen (Bode-plott) til filteret. Mål deretter frekvensresponsen (1 Hz - 100 kHz) vha Digilent og sammenlign med teoretisk Bode-plott. Drøft hva som kan gjøres for å dempe eventuelle resonansfenomen.
2. Tegn opp kretsskjema, og kople opp 3 stk AD-konvertere av type MCP3201 slik at de kan sample 3 kanaler samtidig (dvs at alle kanalene skal ha felles klokkesignal og chip-select, men separate data-utganger).
3. Demonstrarer at dere kan sample et signal fra en signalkilde på alle 3 kanalene. Gjør en måling med `adc_samplers` på Pi-en og overfør måledataene til laptopen deres med en SFTP. **NB: husk at signalet må holde seg innenfor 0 – 3.3 V, så dere må legge på en DC-offset.** Dataene skal vises som funksjon av tid. (Tips: Legg på ulike offset på hver kanal i SW når dere plotter tidsserien slik at en kan skille de ulike kanalene fra hverandre.)
4. Skriv kode som lager et frekvensspektrum av dataene ved hjelp av FFT. Plott spekteret av et sinussignal inn på AD-konverteren med kjent frekvens og verifiser at frekvensaksen til plottet er korrekt. Forsøk å prosessere dataene med ulike vindusfunksjoner og zero-padding. Studer spekteret for å se om dere får forventet SNR på alle kanalene ut fra antall bit i AD-konverteren. Dukker det opp noen uønskede (spuriøse) frekvenskomponenter i spekteret? Hva kan i tilfelle disse komme av? Gjenta dette for ulike frekvenser til dere er sikre på at systemet oppfører seg som forventet.

2.3 Krav for godkjent lab

For å bli godkjent må dere gjøre følgende, med stud.ass eller vit.ass nærværende.

1. Ha løst forberedelsesoppgavene.
2. Vis at lavpassfilteret funker som forventet (dvs måle frekvensrespons og sammenligne knekkfrekvens og eventuelle resonansfenomen med beregnet verdi).
3. Vis en ferdig kretstegning i en egnet tegne-app; Paint eller lignende holder *ikke*.
4. Vis at dere kan måle et signal med AD-konverterne og lagre målingene i Python.
5. Vise tidsserie og frekvensspekter for et valgt signal slik det er beskrevet i laboppgave 4.

3 Referanser/Datablad

Referansene finner du også på BlackBoard.

- Rpi kamera bundle (Farnell):
no.farnell.com/raspberry-pi/rpi3-8mp-camera-bundle/raspberry-pi3-8mp-camera/dp/2580632
- Pi Wedge (Sparkfun):
www.sparkfun.com/products/13717
- MCP3201 12 bit ADC (Farnell):
no.farnell.com/microchip/mcp3201-ci-p/ic-adc-12bit-100ksps-pdip-8/dp/9758577

Del II

Akustikk

En mikrofon er en sensor som omformer et lydsignal til en elektrisk spenning: utgangssignalet er proporsjonalt med lydtrykket ved mikrofonens membran over et bredt frekvensområde. Mikrofoner er «*ubiquitous*», «allestedsnærværende», da alt utstyr for talekommunikasjon bruker mikrofoner, og telefoner står derfor for et enormt marked for mikrofoner.

Den absolutt vanligste mikrofontypen er omnidireksjonal, men anvendelsesområdet for mikrofoner øker betraktelig når man kobler sammen mer enn én mikrofon i et «array». Med et array kan man f.eks lage en direktiv mikrofon hvilket kan gi mye bedre signalkvalitet ved at lyd fra uønskede retninger undertrykkes. I tillegg kan man detektere hvor en lydbølge kommer fra, og dette er temaet for denne laboppgaven.

Det finnes ulike måter å detektere hvor en lydbølge kommer fra. En lett fram måte er å bruke krysskorrelasjon mellom par av mikrofonsignaler for å finne ut tidsforsinkelsen. Dette kan da brukes for å finne retning til en lydkilde. Det kan også finnes begrensede muligheter for å finne posisjon, og ikke kun retning for en lydkilde.

I denne oppgaven skal dere lage et array med tre små omnidireksjonale mikrofoner som skal monteres slik at dere kan detektere innfallsretningen i et horisontalt plan.

Se teorikapittelet for gjennomgang av hvordan man bestemmer effektiv tidsforsinkelse ved hjelp av krysskorrelasjon, og hvordan man finner retning ut fra effektiv tidsforsinkelse.

4 Teori

4.1 Krysskorrelasjon

Hvis vi har to diskrete tidssignaler, $x(n)$ og $y(n)$, så er krysskorrelasjonsfunksjonen, r_{xy} , definert som*

$$r_{xy}(m) = \sum_{n=-\infty}^{\infty} x(n)y(n+m), \quad m = 0, \pm 1, \pm 2, \pm 3, \dots \quad (\text{II.1})$$

Denne funksjonen gir altså et tall, r_{xy} , for hver verdi på tidsforskyvingen m (uttrykt i antall sampeler). Hvis krysskorrelasjonsfunksjonen har en høy verdi så betyr det at de to signalene er meget like; en lav verdi betyr at signalet har et stokastisk forhold - for noen n er produktet $x(n+m)y(n)$ positivt; for andre n er det negativt.

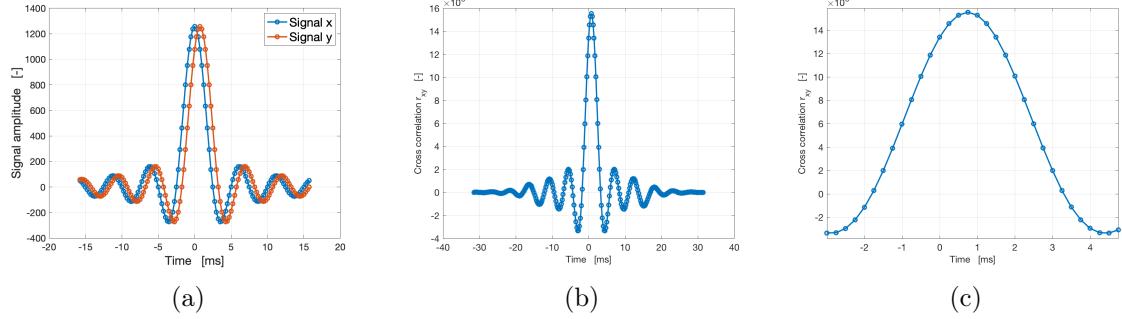
Retningen har betydning: hvis x er meget lik y , men x er forskyvet m sampeler til venstre, så vil $r_{xy}(m)$ gi en høy verdi. Og, hvis x er meget lik y , men x er forskyvet m sampeler til høyre, så vil $r_{xy}(-m)$ å gi en høy verdi.

Her har vi da grunnideen for å finne en tidsforskyving mellom to signaler: beregn $r_{xy}(m)$, og finn for hvilken verdi på m som $|r_{xy}(m)|$ har et maksimum! Tidsforsinkelsen gis da av $\Delta t = m/f_s$, hvor f_s er samplingsfrekvensen.

I figur II.1a ser vi et eksempel på to identiske signaler, hvor x er forskyvet til venstre, med eksakt tre sampeler. Samplingsfrekvensen er 4000 Hz, hvilket tilsvarer at det er 0,25 ms mellom hvert sampel. I

*I noen bøker er denne funksjonen definert motsatt, slik at « $n+m$ » står i argumentet for x . I Python er funksjonen `numpy.correlate` definert på denne alternative måten.

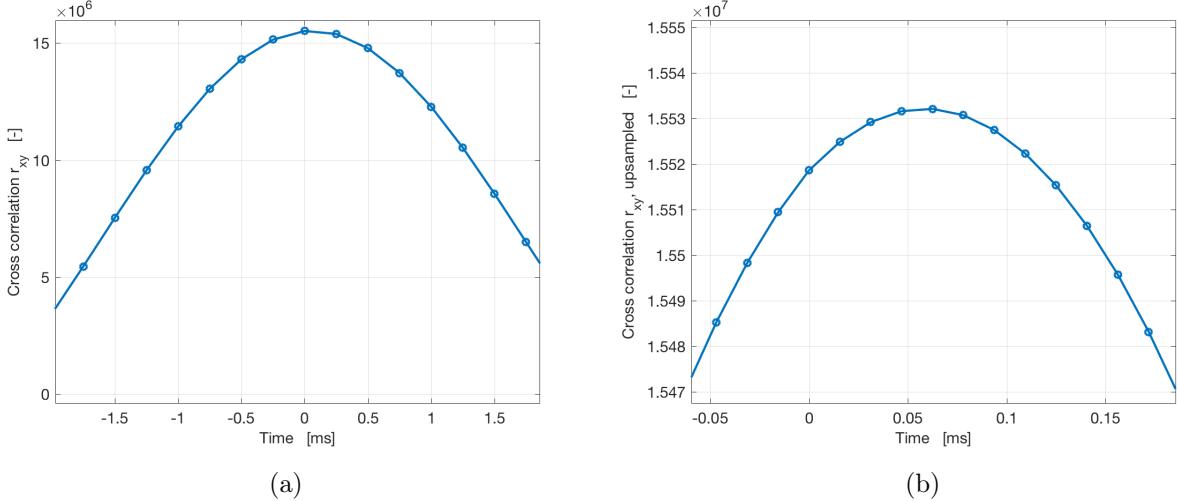
figur II.1b ser vi krysskorrelasjonsfunksjonen, og i figur II.1c en forstørret versjon. I den siste figuren, II.1c, kan vi se en topp ved tidsforskyvingen 0,75 ms. Symmetrien omkring toppen gjør at vi innser at maks av r_{xy} skal være på nøyaktig 0,75 ms.



Figur II.1: (a) To signaler som er identisk, men forskyvet med eksakt tre sampler. (b) Krysskorrelasjonsfunksjonen mellom de to signalene i (a). (c) En forstørret versjon av diagrammet i (b).

Nå ser vi på et eksempel til, hvor vi har laget en forsinkelse som er på 0,25 sampel (dvs $0,25 \cdot 0,25$ ms = $62,5 \mu\text{s}$)! Vi viser kun den forstørrede versjonen i figur II.2a, og innser at når vi søker maks, så kommer vi ikke å kunne avlese toppverdien bedre enn hele sampelverdier. Da får vi altså en feil i deteksjonen av effektiv tidsforsinkelse som er maks. $\pm 0,5$ sampel = $\pm 0,5 \cdot 0,25\text{ms} = 125 \mu\text{s}$.

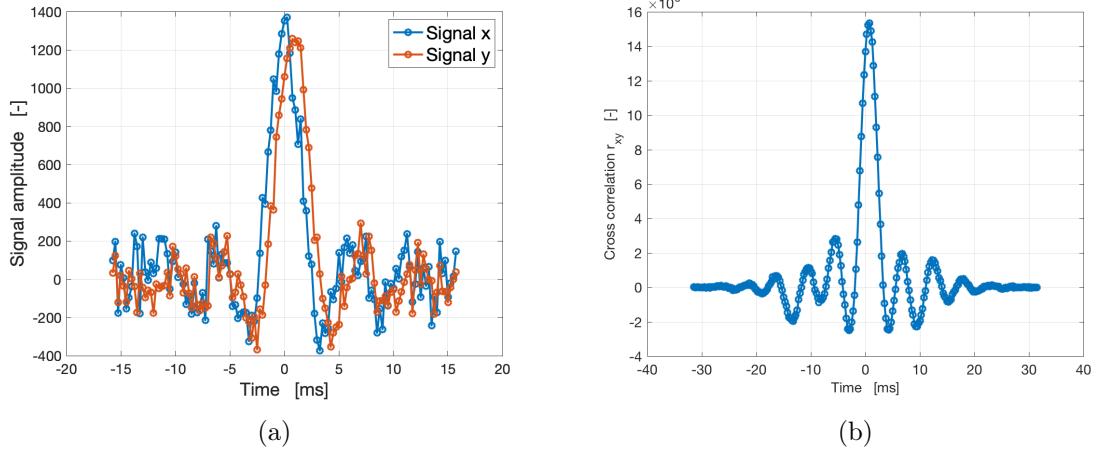
Det er dog mulig å oppsample signalet, og figur II.2b viser at med 16 ganger oppsampling, til $f_s = 64$ kHz, så kan vi avlese toppen med en feil på $\pm 0,5$ sampel = $7,8 \mu\text{s}$.



Figur II.2: (a) Krysskorrelasjonsfunksjonen mellom to signaler som er identisk, foruten en forskyving på 0,25 sampeler, $f_s = 4000$ Hz. (b) En oppsamplet versjon av funksjonen i (a), med Python's `numpy.interp`-funksjon og 16 ganger oppsampling, dvs $f_s=64000$ Hz.

Et siste eksempel: nå går vi tilbake til eksemplet i figur II.1, men legger mye (gaussisk) støy på begge signalene, se figur II.3 (det er uavhengige støysignaler som har blitt lagt på de to signalene). Hvis man zoomer inn rundt toppen i figur II.3b, finner man at toppen fortsatt er på 3 sampeler.

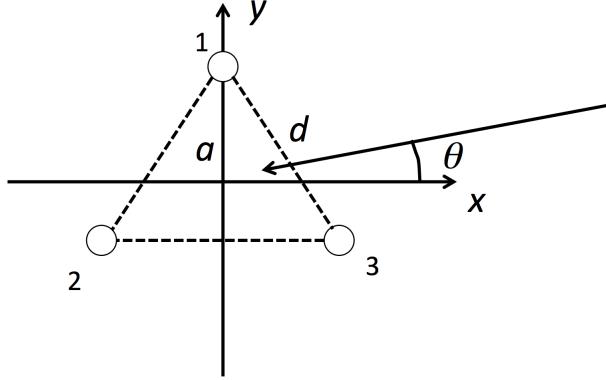
Med litt mer støy, så hadde toppen kunnet vippet over til 2 eller 4 sampler. Det kan hjelpe å båndpassfiltrere signalet (før man gjør krysskorrelasjonen), men kun om nyttesignalet har et mer smalbåndet spektrum enn støysignalet!



Figur II.3: (a) To signaler som er identisk, men forskyvet med eksakt tre sampler, og med gaussisk støy addert. (b) Krysskorrelasjonsfunksjonen mellom de to signalene i (a).

4.2 Deteksjon av innfallsretning

Vi tenker oss tre mikrofoner plassert som i figur II.4.

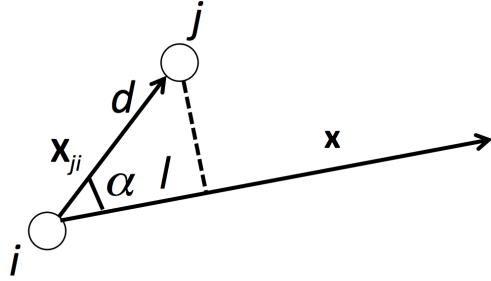


Figur II.4: Et array med tre mikrofoner/sensorer plassert i et likesidet triangel, med sidelengde d .

En innfallende plan lydbølge med vinkelen θ mot x -aksen vil gi forskjellig ankomsttid til de tre sensorene. Vi er spesielt interessert i *tidsdifferansen* mellom ankomsttidene til hvert av de tre parene av sensorer. Derfor ser vi på et eneste par, med sensor i og j , som i figur II.5.

Der innfører vi to vektorer:

- \mathbf{x} = den ukjente vektoren for innfallende bølge. NB: propageringsretningen for bølgen = $-\mathbf{x}$.
- \mathbf{x}_{ji} = vektoren fra sensor i til sensor j ($\mathbf{x}_{ji} = \mathbf{x}_{\text{sensor},j} - \mathbf{x}_{\text{sensor},i}$); denne vektoren er kjent, og har lengden d .



Figur II.5: Et par av sensorer, og en innfallende bølge som propagerer i retning $-x$ (som er ukjent). Vektoren \mathbf{x}_{ji} går fra sensor i til sensor j .

Vi antar at vi har gjort en måling og funnet ut forsinkelsen fra sensor i til sensor j , τ_{ji} (med krysskorrelasjon). Den forsinkelsen tilsvarer en gangveisforskjell, l : $\tau_{ij} = -l/c$ hvor lengden l vises i figur II.5. Det blir et minustegn fordi vi har en bølge som utbreder seg i retningen $-x$, og da må τ_{ji} (tiden fra sensor i til sensor j) være negativ.

Lengden l er altså kjent, og henger sammen med den ukjente vinkelen α som

$$l = d \cos \alpha = |\mathbf{x}_{ji}| \cos \alpha \quad (\text{II.2})$$

dvs vi kan uttrykke den ukjente vinkelen som

$$\cos \alpha = \frac{l}{d} = \frac{l}{|\mathbf{x}_{ji}|} = \frac{-c\tau_{ji}}{|\mathbf{x}_{ji}|} \quad (\text{II.3})$$

Med to sensorer kan vi tydeligvis finne vinkelen α - men den vinkelen er tvetydig fordi det er to innfallsvinkler som oppfyller ligning (II.3)*. I figur II.5 kan vi «speile» vektoren \mathbf{x} rundt vektoren \mathbf{x}_{ji} (slik at \mathbf{x} peker nesten rett opp), og vinkelen mellom vektorene \mathbf{x} og \mathbf{x}_{ji} vil fortsatt være α . For å finne hvilken av de to mulige løsningene som er riktig så trenger vi en sensor til. Da viser det seg å være praktisk å finne vektoren \mathbf{x} heller enn vinkelen α (fordi \mathbf{x} vil være felles for alle tre mikrofonparene, men vinkelen α vil være forskjellig), og vi kan finne forholdet mellom dem fra prikkproduktet for de to vektorene:

$$\mathbf{x}_{ji} \cdot \mathbf{x} = |\mathbf{x}_{ji}| |\mathbf{x}| \cos \alpha \quad (\text{II.4})$$

Setter vi nå inn (II.3) i (II.4) så får vi

$$\mathbf{x}_{ji} \cdot \mathbf{x} = |\mathbf{x}_{ji}| |\mathbf{x}| \frac{(-c\tau_{ji})}{|\mathbf{x}_{ji}|} \quad (\text{II.5})$$

$$\Rightarrow c\tau_{ji} = -\frac{\mathbf{x}_{ji} \cdot \mathbf{x}}{|\mathbf{x}|} \quad (\text{II.6})$$

Nå kan vi velge å definere \mathbf{x} slik at $|\mathbf{x}| = 1$: $\mathbf{x} = [\cos \theta, \sin \theta]$ (se figur II.4) og da er

$$\mathbf{x}_{ji} \cdot \mathbf{x} + c\tau_{ji} = 0 \quad (\text{II.7})$$

*Hvis vi har kun to sensorer så vil vinkelen α i 3D tilsvare ikke bare to vinkler men en *kon* av innfallsvinkler.

Denne likningen kunne vi skrevet ut som

$$x_{ji} \cos \theta + y_{ji} \sin \theta + c\tau_{ji} = 0 \quad (\text{II.8})$$

og her har vi jo én ukjent og burde kunne løse likningen - men den gir, som kommentert over, ikke en entydig løsning. To innfallsretninger, speilet rundt vektoren \mathbf{x}_{ji} gir samme tidsforsinkelse τ_{ji} !

Derfor setter vi opp tre sensorer som i figur II.4, og innfører tre vektorer fra sensor til sensor: \mathbf{x}_{21} , \mathbf{x}_{31} , \mathbf{x}_{32} og tre tilsvarende forsinkelser: τ_{21} , τ_{31} , τ_{32} . Hvis vi nå setter opp tre likninger som i (II.7), får vi i stedet et overbestemt ligningssystem (tre likninger og én ukjent, θ):

$$\mathbf{x}_{ji} \cdot \mathbf{x} + c\tau_{ji} = 0, \quad ji = 21, 31, 32 \quad (\text{II.9})$$

Overbestemte systemer er faktisk bra fordi da kan vi håndtere at:

- våre målinger av τ_{ji} ikke er perfekte (støy, andre målefeil, mer enn en bølge,...)
- innfallende bølge kanskje har en innfallsretning som ikke er i samme plan som sensorene!

Men hvordan skal vi løse ligningssettet (II.9)? Jo, med minste kvadraters metode, «*Least mean squares*». På samme måte som når vi utleder linjetilpassing til et sett med datapunkter så innfører vi et feilmål, ε , som forteller hvor dårlig vårt estimat av x, y passer til måleverdiene $\tau_{21}, \tau_{31}, \tau_{32}$. Det målet definerer vi slik (idealt skal ε være 0):

$$\varepsilon^2 = \sum_{ji} (\mathbf{x}_{ji} \cdot \mathbf{x} + c\tau_{ji})^2 = \sum_{ji} (x_{ji}x + y_{ji}y + c\tau_{ji})^2 \quad (\text{II.10})$$

Målet er å finne de to verdiene x og y slik at ε^2 minimeres! Her er da den søkte vinkelen θ skjult i x og y slik at på slutten, når vi har funnet x og y så får vi θ som:

$$\tan \theta = \frac{y}{x}, \quad \theta = \text{atan} \frac{y}{x} \quad (\text{II.11})$$

For å minimere ε^2 så skal vi da derivere ε^2 med hensyn på x , og med hensyn på y , og sette de deriverte til 0:

$$\begin{cases} \frac{\partial \varepsilon^2}{\partial x} = \frac{\partial}{\partial x} \sum_{ji} (x_{ji}x + y_{ji}y + c\tau_{ji})^2 = \sum_{ji} 2(x_{ji}x + y_{ji}y + c\tau_{ji})x_{ji} = 0 \\ \frac{\partial \varepsilon^2}{\partial y} = \frac{\partial}{\partial y} \sum_{ji} (x_{ji}x + y_{ji}y + c\tau_{ji})^2 = \sum_{ji} 2(x_{ji}x + y_{ji}y + c\tau_{ji})y_{ji} = 0 \end{cases} \implies \quad (\text{II.12})$$

$$\begin{cases} x \sum_{ji} 2(x_{ji}^2) + y \sum_{ji} 2(x_{ji}y_{ji}) + \sum_{ji} 2(c\tau_{ji}x_{ji}) = 0 \\ x \sum_{ji} 2(x_{ji}y_{ji}) + y \sum_{ji} 2(y_{ji}^2) + \sum_{ji} 2(c\tau_{ji}y_{ji}) = 0 \end{cases} \implies \begin{cases} xA + yB + C = 0 \\ xB + yD + E = 0 \end{cases} \quad (\text{II.13})$$

Dette kan man løse:

$$x = \frac{CD - BE}{AD - B^2}, \quad y = \frac{-BC + AE}{AD - B^2} \quad (\text{II.14})$$

Nå kan vi sette inn verdier på A, B, C, D og E fordi vi kan se i figur II.4 hvor sensorene plasseres:

$$\mathbf{x}_{\text{sensor},1} = [0, 1]a \quad (\text{II.15})$$

$$\mathbf{x}_{\text{sensor},2} = [-\sqrt{3}/2, -0.5]a \quad (\text{II.16})$$

$$\mathbf{x}_{\text{sensor},3} = [\sqrt{3}/2, -0.5]a \quad (\text{II.17})$$

slik at vektorene \mathbf{x}_{ji} blir

$$\mathbf{x}_{21} = \mathbf{x}_{\text{sensor},2} - \mathbf{x}_{\text{sensor},1} = \left[-\frac{\sqrt{3}}{2}, -\frac{3}{2} \right] a \quad (\text{II.18})$$

$$\mathbf{x}_{31} = \mathbf{x}_{\text{sensor},3} - \mathbf{x}_{\text{sensor},1} = \left[\frac{\sqrt{3}}{2}, -\frac{3}{2} \right] a \quad (\text{II.19})$$

$$\mathbf{x}_{32} = \mathbf{x}_{\text{sensor},3} - \mathbf{x}_{\text{sensor},2} = \left[\sqrt{3}, 0 \right] a \quad (\text{II.20})$$

Da får vi

$$A = 2 \sum (x_{ji})^2 = 9a^2 \quad (\text{II.21})$$

$$B = 2 \sum (x_{ji}y_{ji}) = 0 \quad (\text{II.22})$$

$$D = 2 \sum (y_{ji})^2 = 9a^2 \quad (\text{II.23})$$

Symmetrien gjorde tydeligvis at $B = 0$, så vi får enklere uttrykk enn i (II.14):

$$x = \frac{C}{A}, \quad y = -\frac{E}{D} \quad (\text{II.24})$$

og når vi setter inn uttrykkene for A, C, D og E :

$$x = c \frac{\sum x_{ji}\tau_{ji}}{9a^2/2} = \frac{2c}{9a^2} \sum x_{ji}\tau_{ji}, \quad y = -c \frac{\sum y_{ji}\tau_{ji}}{9a^2/2} = -\frac{2c}{9a^2} \sum y_{ji}\tau_{ji} \quad (\text{II.25})$$

Her kan vi sette inn verdier for x_{ji} og y_{ji} for vår spesifikke arraygeometri, fra lign. (II.18)-(II.20):

$$\begin{aligned} x_{21} &= -\frac{\sqrt{3}}{2}a, \quad x_{31} = \frac{\sqrt{3}}{2}a, \quad x_{32} = \sqrt{3}a \\ y_{21} &= -\frac{3}{2}a, \quad y_{31} = -\frac{3}{2}a, \quad y_{32} = 0 \end{aligned}$$

og da blir

$$x = \frac{2c}{9a} \left(-\frac{\sqrt{3}}{2}\tau_{21} + \frac{\sqrt{3}}{2}\tau_{31} + \sqrt{3}\tau_{32} \right) = \frac{c}{3\sqrt{3}a} (-\tau_{21} + \tau_{31} + 2\tau_{32}) \quad (\text{II.26})$$

$$y = -\frac{2c}{9a} \left(-\frac{3}{2}\tau_{21} - \frac{3}{2}\tau_{31} \right) = \frac{c}{3a} (\tau_{21} + \tau_{31}) \quad (\text{II.27})$$

Så setter vi inn disse uttrykkene for x og y i (II.11):

$$\theta = \text{atan} \frac{\frac{c}{3a} (\tau_{21} + \tau_{31})}{\frac{c}{3\sqrt{3}a} (-\tau_{21} + \tau_{31} + 2\tau_{32})} = -\text{atan} \left(\sqrt{3} \frac{\tau_{31} + \tau_{21}}{\tau_{31} - \tau_{21} + 2\tau_{32}} \right) \quad (\text{II.28})$$

Vi må være litt forsiktig med atan-funksjonen fordi $\text{atan}(1/1) = \text{atan}(-1/-1)$, hvilket leder til at atan kun gir oss vinkler i intervallet $\theta \in [-90^\circ, 90^\circ]$. Hvis vi ønsker å detektere vinkler fra alle retninger må vi gå tilbake til uttrykkene for x og y i lign. (II.26) og (II.27): hvis $x < 0$, skal

vi legge til π til θ -estimatet fra lign. (II.28). Vi kan notere at fortegnet på x gis av fortegnet på $(-\tau_{21} + \tau_{31} + 2\tau_{32})$.

I en praktisk implementering vil vi bruke samplede signaler, og tidsforsinkelsene, τ_{ji} , vil beregnes fra topper i korrelasjonsfunksjoner ved hele antall sampler, n_{ji} slik at $\tau_{ji} = n_{ji}/f_s$. Da kan vi like gjerne bruke forsinkelser uttrykt direkte i sampler:

$$\theta = \text{atan} \left(\sqrt{3} \frac{n_{31} + n_{21}}{n_{31} - n_{21} + 2n_{32}} \right) \quad (\text{II.29})$$

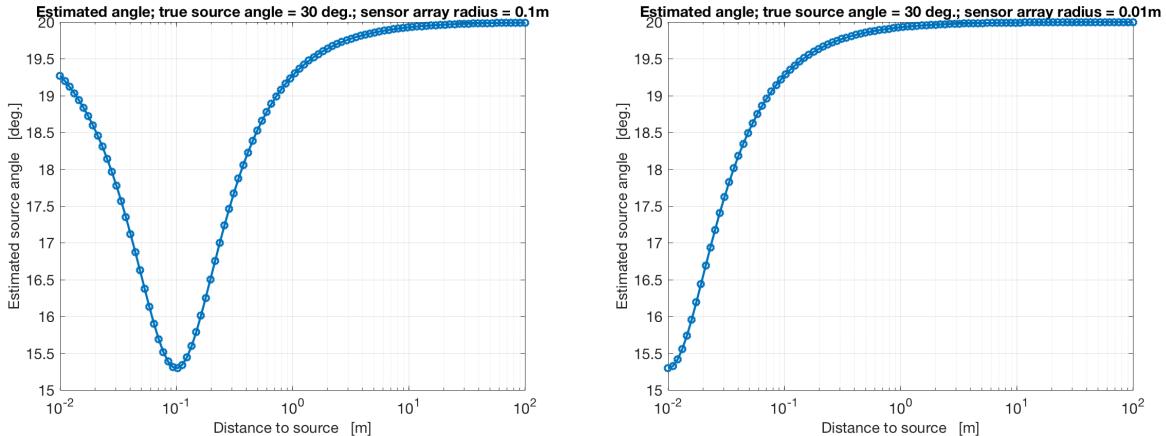
Igjen må vi legge til π til θ -estimatet hvis $x < 0$, dvs hvis $(-n_{21} + n_{31} + 2n_{32}) < 0$. Vi kan også innse at heltallene n_{ji} kun kan ha et begrenset antall forskjellige verdier: $[\pm n_{max}, \pm(n_{max} - 1), \dots 0]$. Det betyr at kun et begrenset antall vinkler kan estimeres av lign. (II.29)! Vi kan få en høyere verdi på n_{max} ved å øke størrelsen på arrayet, d , og/eller ved å øke samplingsfrekvensen, f_s .

4.3 Hva om lydkilden ikke er langt vekk (dvs ikke en planbølge)?

La oss nå se på hvordan uttrykket i lign. (II.28) fungerer hvis kilden ikke er langt vekk. Vi kan gjøre en enkel evaluering ved å plassere en kilde i forskjellig avstand, r_{source} , men altid med vinkel θ , dvs

$$\mathbf{x}_{\text{source}} = r_{\text{source}} [\cos \theta, \sin \theta] \quad (\text{II.30})$$

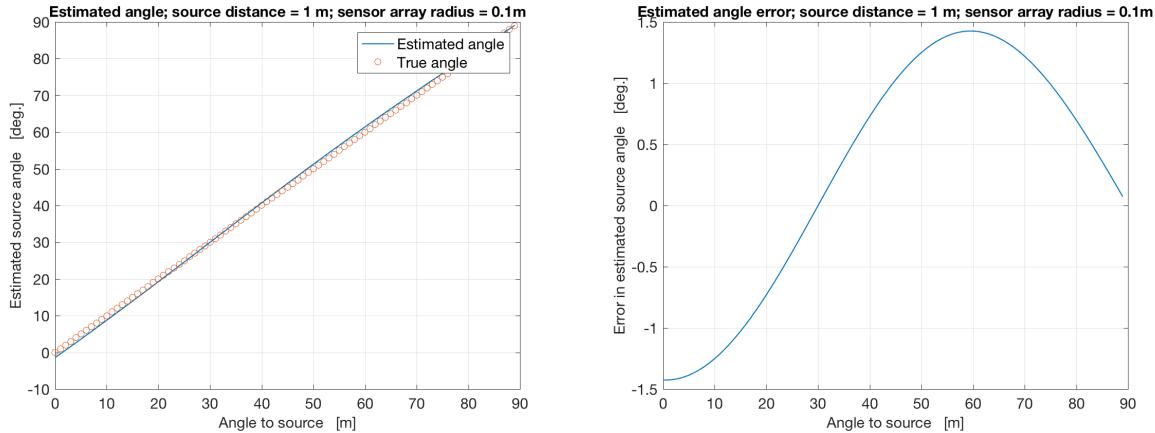
Da er det rett frem å beregne τ_1 , τ_2 og τ_3 som funksjon av r_{source} . Vi har gjort det for en kilderetning $\theta = 20$ grader, og to forskjellige sensorarraystørrelser, og de vises i figur II.6.



Figur II.6: Estimert vinkel til en lydkilde i virkelig retning 20 grader (tittelen på grafen er feil: 30 grader skal altså være 20 grader).

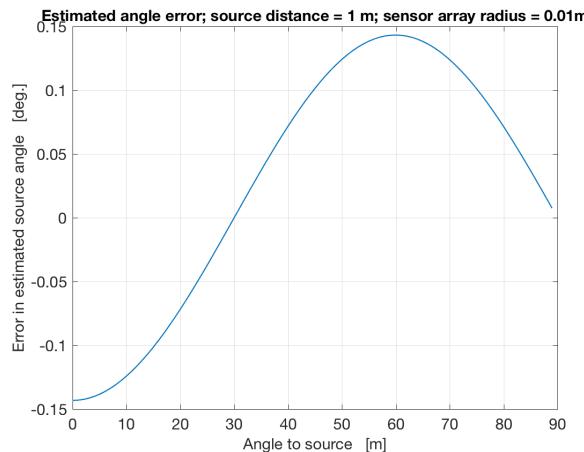
Så, hvis kilden er minst 1 meter fra sentrum av sensorarrayet så er feilen i kildevinkelestimatet mindre enn 0.8 grader for 10cm-arrayet, og mindre enn 0.1 grader for 1cm-arrayet.

Vi kan også se på feilen på en annen måte; hvor stor er feilen for forskjellige retninger til den virkelige kilden? Vi velger å studere avstanden fra kilden til 1 m, og med arrayet som har $a = 10$ cm. Resultatene vises i figur II.7.



Figur II.7: Estimert vinkel til en lydkilde for forskjellige innfallsretninger (venstre), og tilsvarende feil i estimatet (høyre).

Feilen i vinkelestimatet er tydelig avhengig av hvilken vinkel det er til kilden. For vinkelen 30 grader så viser det seg at feilen er 0, uavhengig hva avstanden er til kilden! Så kan vi igjen se hva som skjer når vi har et ti ganger så lite array, med $a = 1\text{cm}$, se figur II.8.



Figur II.8: Feil i estimert vinkel til en lydkilde for forskjellige innfallsretninger for et lite array.

Feilen i vinkelestimat ser ut å være nesten presis 10 ganger mindre hvis arrayet er 10 ganger mindre.

Det ser ut som om det er mye bedre med et lite array! Er det noen ulempe med et mindre array? Ja, en gitt tidsdifferanse tilsvarer mye mindre fasedifferanse med et mindre array! Og, sensorene har ikke perfekt fasemålingsnøyaktighet. I tillegg, vi må jo plassere sensorene perfekt symmetrisk, og det blir mer vanskelig jo mindre arrayet er.

Til sist, som ble påpekt etter lign. (II.29), jo mindre arrayet er, jo mindre er n_{max} , og jo færre antall vinkler kan estimeres.

5 Forberedelsesoppgaver til laben

Gjøres før laboppgaven.

5.1 Forberedelsesoppgave 1

Beskriv med Python-kode hvordan krysskorrelasjon kan brukes for å finne effektiv forsinkelse mellom to lydsignaler som er tatt opp med samplingsfrekvensen f_s .

Hint: du skal finne for hvilken forsinkelse som `numpy.abs(krysskorrelasjonsfunksjonen)` har maksimum.

5.2 Forberedelsesoppgave 2

- (a) Hvis du har et triangelformet array som i figur II.4, med sidelengde d , og samplingsfrekvensen er f_s ; hva er maksimalt antall sampeler med forsinkelse som kan oppstå?
- (b) Hvis du har et array med kun to mikrofoner, kan du detektere flere unike innfallsvinkler om maksimal forsinkelse mellom dem er 2 sampel eller 4 sampel? *Hint:* Et lite vinkelområde vil gi samme forsinkelse, n_{21} , grunnet avrunding til helt antall sampel!

6 Laboppgave

1. Lag et mikrofonarray med geometri som i figur II.4. Velg d slik at alle mikrofonene får plass på koblingsbrettet.
2. Gjør et opptak av et lydsignal med de tre mikrofonene, og vis at de tre signalene ser ut som forventet, dvs amplituden skal variere rundt 0*, og avhengig av hvilket signal du brukte skal alle tre se ut som sinustoner, impulslyder, støysignaler, eller et slikt signal som du brukte.
3. Implementer en krysskorrelasjonsberegnning for de tre mikrofonsignalene. Finn tidsforsinkelsen fra toppen av krysskorrelasjonen, uttrykket i antall sampler og vis de tre krysskorrelasjonssignalene. Du må også vise et eksempel på et autokorrelasjonssignal[†] - sjekk at du får en topp for eksakt 0 sampels «lag».[‡]

Er toppen for disse krysskorrelasjonssignalene innenfor et intervall som kan forventes, dvs innenfor $\pm n_{max}$, hvor n_{max} gis av avstand mellom mikrofonene, samplingfrekvensen og lydhastigheten?

4. Beregn innfallsvinkelen (i planet som de tre mikrofonene er plassert) fra de tre tidsforsinkelsene. Vis at du kan detektere innfallsvinkel fra -180 til $+180$ grader.
5. Gjør en systematisk variasjon av estimert innfallsvinkel for flere innfallsvinkler. Gjør flere, minimum 5, gjentatte målinger for samme innfallsvinkel for å få et mål på usikkerheten. Standardavviket og variansen er mål som godkjennes. Gjør dette for minimum 3 forskjellige vinkler.

Tips: Best resultat fra krysskorrelasjon oppnås ved å bruke et signal med mange frekvenskomponenter (feks støysignal, eller et impulsregnende signal). Husk også å fjerne DC-komponenten fra rådataene (etter digitaliseringen).

Tips: Et båndpassfilter (i Python) før krysskorreleringen kan forbedre signal-til-støy-forholdet - men da må du vite i hvilket frekvensområde nyttesignalet du bruker ligger i.

6.1 Krav for godkjent lab

For å få godkjent laben må dere, i tillegg til å få godkjent på forberedelsesoppgavene, vise at dere har gjort laboppgaver 3 til 5.

*dvs du må fjerne DC-komponenten fra signalene!

[†]autokorrelasjonsfunksjon = krysskorrelasjon mellom to identiske signaler

[‡]Ved en implementering av krysskorrelasjon så kan det være lett å få feil «lag» med ± 1 sampel. Testen med autokorrelasjon er meget bra fordi vi vet at den *altid* skal ha en topp ved 0 sampels «lag».

7 Appendix - løsning av lign. (II.9) via matriseinvertering

Ligning (II.9) representerer 3 ligninger:

$$\mathbf{x}_{21} \cdot \mathbf{x} + c\tau_{21} = 0 \quad (\text{II.31})$$

$$\mathbf{x}_{31} \cdot \mathbf{x} + c\tau_{31} = 0 \quad (\text{II.32})$$

$$\mathbf{x}_{32} \cdot \mathbf{x} + c\tau_{32} = 0 \quad (\text{II.33})$$

hvor den ukjente er $\mathbf{x} = [x, y]$ (som altså er en vektor av størrelse (1,2)), og vektorene $\mathbf{x}_{ji} = [x_{ji}, y_{ji}]$ (også vektorer av størrelse (1,2)) beskriver hvor sensorene er i forhold til hverandre. Disse tre ligningene kan sammenfattes i en matriseligning

$$\mathbf{X}_{ij}\mathbf{x}^T = -c\boldsymbol{\tau} \quad (\text{II.34})$$

hvor

$$\boldsymbol{\tau} = [\tau_{21}, \tau_{31}, \tau_{32}]^T, \text{ hvilket er en kolonnevektor av størrelse (3, 1)} \quad (\text{II.35})$$

$$\mathbf{X}_{ji} = [\mathbf{x}_{21}^T, \mathbf{x}_{31}^T, \mathbf{x}_{32}^T]^T, \text{ hvilket er en matrise av størrelse (3, 2)} \quad (\text{II.36})$$

Denne måten å beskrive en matrise kan se tungvint ut; i Python ville vi skrive, etter `import numpy as np`,

```
>>> Xmatrix = np.array([[x21, y21], [x31, y31], [x32, y32]])
```

Som nevnt i hovedteksten så er (II.34) et overbestemt ligningssystem, hvilket kan løses med minste kvadraters metode. Dette kan gjøres elegant og effektivt ved å bruke pseudo-inversen til matrisen \mathbf{X}_{ji} (Python-funksjonen `numpy.linalg.pinv` implementerer den såkalte Moore-Penrose pseudoinversen til en ikke-kvadratisk matrise):

$$\mathbf{x}^T = -c\mathbf{X}_{ji}^\dagger \boldsymbol{\tau} \quad (\text{II.37})$$

hvor \mathbf{X}^\dagger oppfyller

$$\mathbf{X}_{ji}\mathbf{X}_{ji}^\dagger \mathbf{X}_{ji} = \mathbf{X}_{ji} \quad (\text{II.38})$$

7.1 Numerisk eksempel

Vi har et mikrofonarray som i figur II.4, med $a = 0.1\text{m}$. En plan lydbølge har infallsvinkelen $\theta = 20$ grader. Da får vi følgende tidsforsinkelser

```
>>> t_matrix = 1e-3*np.array([0.386367, -0.087589, -0.473957])
```

og sensorene er plassert i

```
>>> Xmatrix = np.array([[-0.086603, -0.15], [0.086603, -0.15], [0.173205, 0]])
```

Med metoden i hovedteksten får vi ut θ som:

$$\theta = \arctan \left(\sqrt{3} \frac{\tau_1 + \tau_2}{\tau_1 - \tau_2 - 2\tau_3} \right) = \arctan \left(\sqrt{3} \frac{0.386367 - 0.087589}{0.386367 + 0.087589 + 2 \cdot 0.473957} \right) \quad (\text{II.39})$$

$$= 0.349054 \text{ rad} \approx 20.00 \text{ grader} \quad (\text{II.40})$$

Med matriseinvertering:

```
>>> xvector = -343.4 * np.linalg.pinv(Xmatrix)@t_matrix
```

så får vi løsningen

```
>>> xvector  
array([0.93967499, 0.34200122])
```

og

```
>>> theta = np.arctan(xvector[1]/xvector[0])  
>>> print(f"{theta:.4f}")  
0.3491
```

Dvs, de to metodene gir samme resultat. Vi kan også notere at i matriseligning (II.34) så er det enkelt å innføre en tredje dimensjon! Vi trenger kun å la

$$\mathbf{x} = [x, y, z] \quad (\text{II.41})$$

og så spesifisere differansen mellom z -koordinatene for de tre sensorene i

$$\mathbf{x}_{ji} = [x_{ji}, y_{ji}, z_{ji}] \quad (\text{II.42})$$

Dermed blir matriseligningen fortsatt lik - men vi oppdager at selv om kilden ikke er i planet, så kommer vi alltid å få løsningen $z = 0$. Da virker det som om en slik tre-sensorarray ikke kan oppdage en «elevasjonsvinkel»? Det kan den, men vi må gå via en sjekk av lengden på vektoren \mathbf{x} . Lengden skal være 1, men med innfall-elevasjonsvinkel, så kommer lengden $\|\mathbf{x}\|$ ikke å være 1. Da kan vi få frem elevasjonsvinkelen, ϕ (phi) via

```
>>> phi = np.arcsin(np.linalg.norm(xvector))
```

Vi kan dog konstatere at et «flatt» mikrofonarray ikke kan detektere om bølgen kommer ovenfra eller underfra.

Del III

Radar

En radar er en sensor som sender ut radiobølger (mikrobølger) og mottar ekko fra objekter i nærheten og fra omgivelsene generelt. Radar brukes i mange sammenhenger innenfor alt fra luftovervåkning, navigasjon på sjøen, fartsmåling, og helt ned til kortholdssystemer for medisinske anvendelser.

8 Teori

En radar kan måle avstand til et objekt ved å måle tiden det tar fra utsendt puls til en mottar ekko fra et objekt. Ved å multiplisere denne tiden med lyshastigheten, kan en beregne avstanden til et objekt ut fra ligningen

$$R = \frac{c\tau}{2}, \quad (\text{III.1})$$

der c er lysets hastighet, og τ er tidsforsinkelsen.

Ved å bruke en antenn med smal strålebredde kan en også beregne retningen til objektet ved å måle hvilken vinkel radarantennen peker. Det mest vanlige er å la radarantennen rotere mens den hele tiden sender ut pulser som reflekteres fra objekter ved forskjellige vinkler.

En radar kan måle radiell hastighet til objektet ved å måle dopplerskiftet på mottatt signal. Dette gjøres ofte ved å sende en kontinuerlig bærebølge med konstant frekvens, eller pulser med konstant frekvens (puls-doppler radar).

Dopplerskiftet f_D til et radarekko er gitt ved

$$f_D = \frac{2f_0v_r}{c}, \quad (\text{III.2})$$

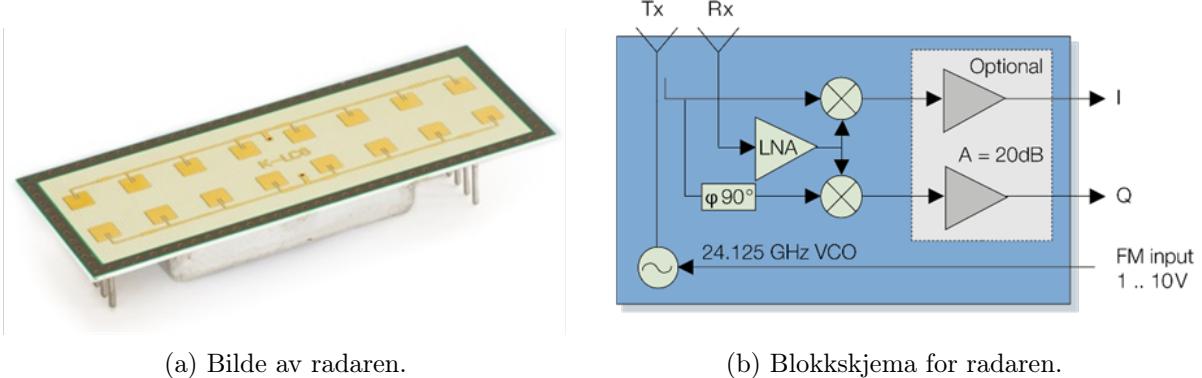
der f_0 er radarens senterfrekvens og v_r er den radielle hastigheten som objektet beveger seg imot eller vekk fra radaren.

I denne oppgaven skal vi bruke en 24 GHz radar fra RfBeam GmbH til å måle hastigheten til et objekt. Radaren er en såkalt Continuous Wave (CW) radar som sender en kontinuerlig bærebølge*. Figur III.1b inneholder blokkskjemaet for RFbeam K-LC6_v2 radaren sammen med et bilde av enheten i Figur III.1a der selve mikrobølgdedelen er innkapslet i en skjermboks mens antennene sitter på den motsatte flaten av kretskortet.

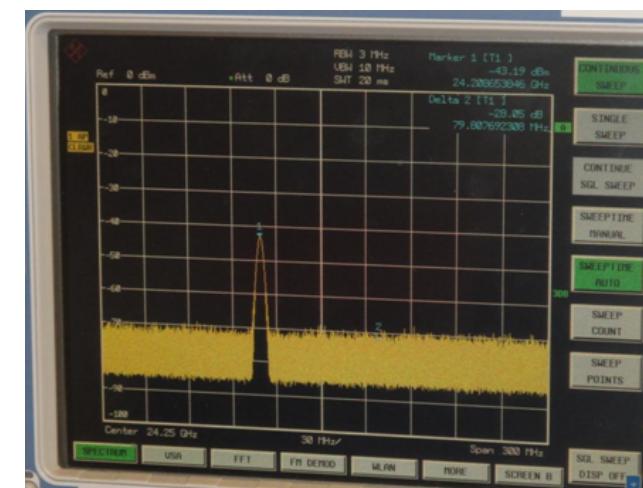
Kjernen i radaren er en såkalt VCO (Voltage Controlled Oscillator) som kan justeres innenfor frekvensområdet 24.00 – 24.25 GHz. Frekvensen justeres ved hjelp av en spenning på inngangen FM inputsom kan varieres mellom 1 og 10V. Hvis en ikke setter på noen spenning på FM input, vil oscillatoren svinge på ca 24.13 GHz (kan variere noe fra enhet til enhet). Figur III.2 viser utsendt signal fra radaen på skjermbildet til en spektrumsanalysator.

Hoveddelen av signalet sendes ut på senderantennen, mens en liten del av signalet koples over til mottakeren og brukes som referanse (dette mottakersignalet kalles ofte for Local-Oscillator, LO). I

*Radaren kan også opereres i en såkalt FMCW modus (Frequency Modulated Continuous Wave), der en bruker en lineær frekvensvariasjon til å beregne avstanden til målet.



Figur III.1: RfBeam K-LC6_v2 24 GHz radar.



Figur III.2: Spektrum av RF-signal fra radaren (FM input = 0V).

K-LC6_v2 er LO-signalet delt i to deler der den ene delen er 90-grader faseforskjøvet i forhold til det andre. De to resulterene utgangene kalles kvadratur-utganger (I og Q kanal).

Mottakersignalet fra antennen blir forsterket i en lavstøyforsterker (LNA) og fordeles inn på to mikserne (I og Q kanal). Mikserne er ulineære komponenter som multipliserer signalet inn på LO-porten med signalet inn på RF-porten. Resultatet av denne ulineære prosessen er at det på utgangsporten (IF, Intermediate Frequency) dukker opp to nye signal. Det ene signalet er summen av de frekvensene f_1 og f_2 som påtrykkes hhv RF- og LO-porten, mens det andre signalet består av differansen mellom frekvensene som påtrykkes hhv. RF- og LO-porten. Dette kan uttrykkes matematisk gjennom den velkjente matematiske identiteten

$$\cos \omega_1 \cos \omega_2 = \frac{1}{2} [\cos(\omega_1 - \omega_2) + \cos(\omega_1 + \omega_2)], \quad (\text{III.3})$$

der $\omega = 2\pi f$.

I vårt tilfelle vil sum-frekvensen havne på $2 \times 24 \text{ GHz} = 48 \text{ GHz}$ som er langt over båndbredden til utgangen fra mikserne. Dette signalet vil dermed bli kraftig dempet slik at vi kan se bort fra det. Hvis det mottatte signalet ikke har dopplerskift, vil differanse-frekvensen bli lik 0 – dvs en DC-spennin siden en mikser to helt like frekvenser. Denne mikseprosessen kalles *homodyning*, og vi sier at utgangssignalet havner i basisband (dvs omkring 0 Hz).

Hvis målet beveger seg, vil en på grunn av dopplerskiftet til det mottatte signalet få en liten frekvensforskjell ut av mikserne. Utgangene fra de to mikserne kalles hhv I (In-phase) og Q (quadrature) siden LO-signalene til de to mikserne har en faseforskjell på 90 grader. Disse to signalene kan settes sammen til et komplekst signal

$$x(t) = I(t) + jQ(t). \quad (\text{III.4})$$

Likning (III.4) vil utgjøre en roterende viser (phasor) som vil rotere mot klokken ved positivt dopplerskift og med klokken ved negativt dopplerskift. Dette gjør det mulig å måle både positivt og negativt dopplerskift avhengig av om målet beveger seg imot eller vekk fra radaren.

Utgangene fra hver av mikserne skal forsterkes 20 dB for å få mer utsving på svake signaler. Ved å ta Fouriertransform av I og Q signalene kan en få ut frekvensspektret og dermed lese av dopplerskiftet som vist i Figur III.3. Fouriertransformen gjøres ved hjelp av FFT i Python etter at basisbandsignalet er digitalisert vha A/D-konverterne.

Ved måling av dopplerskift med en CW radar vil måleoppløsningen til dopplerskiftet være gitt ved

$$\delta f_D = \frac{1}{T}, \quad (\text{III.5})$$

der T er observasjonstiden (dvs tiden radaren observerer målet mens det er i bevegelse). Dette vil i praksis være måletiden (lengden på måledatane dere samler inn). Hvis hastigheten er konstant under observasjonstiden, vil δf_D tilsvare 3dB-bredden på frekvenslinjen i dopplerspekteret som kommer fram etter Fouriertransformen.

NB: Husk at *måleoppløsningen* i dette tilfellet er evnen radaren har til å skille to mål med ulike hastighet (dvs å skille mellom frekvenslinjer i dopplerspekteret). *Målenøyaktigheten* til hastighetsmålingen vil være gitt av hvor nøyaktig en kan estimere selve frekvensen til frekvenslinjen. En enkel

måte å estimere dopplerskiftet er å finne frekvensen til peaken i frekvenslinja. I faglitteraturen for spektralestimering fins et antall mer avanserte metoder for å estimere frekvenser i et frekvensspektrum uten at vi skal gå inn på det her.

Antennesystemet til radaren består av 2 rekker med 8 såkalte patch-antennene i hver rekke som vist i figur III.5. Den ene rekken brukes som senderantenne mens den andre rekken er koplet til mottakeren. På den måten kan en sende og motta samtidig. Ved å studere kretskortet ser en at det går transmisjonslinjer som fordeler effekten ut til antennene. Linjelengdene er avstemt slik at alle elementene får samme fase på utsendt signal. Det resulterende elektromagnetiske feltet fra denne gruppeantennen (antenna array) blir dermed en kombinasjon av feltbidragene fra hvert enkelt element. Siden alle elementene svinger i fase, vil en få konstruktiv interferens i retningen som er normal på den flaten som antennene ligger langs. Arrayet kalles derfor et «broadside array» siden det gir maksimalt signal i retning av «breddesiden» til arrayet. For andre vinkler vil en få svakere signal på grunn av destruktiv interferens for disse retningene. Dette kommer av at det blir ulik gangvei (og dermed også forskjellig faseskift) for feltene fra de ulike elementene når en ser i en vinkel som er forskjellig fra normalen til kretskortet. Resultatet av dette blir at en får en smal stråle i horisontalplanet (azimuth). Dette planet er definert av normalvektoren til kretskortet og den aksen som de 8 elementene er distribuert langs. Jo lengre rekke en har, jo smalere vil strålen kunne bli i azimuth-planet.

I vertikalplanet (elevation) vil strålebredden bli bestemt av strålingssegenskapene til en enkelt patch-antenne. I faget TTT4215 Antenneteknikk behandles både patch-antennene og gruppeantennene utførlig, og illustrasjonen av patch-antennen i figur III.4 er hentet fra læreboka til Balanis [**balanis2015antenna**].

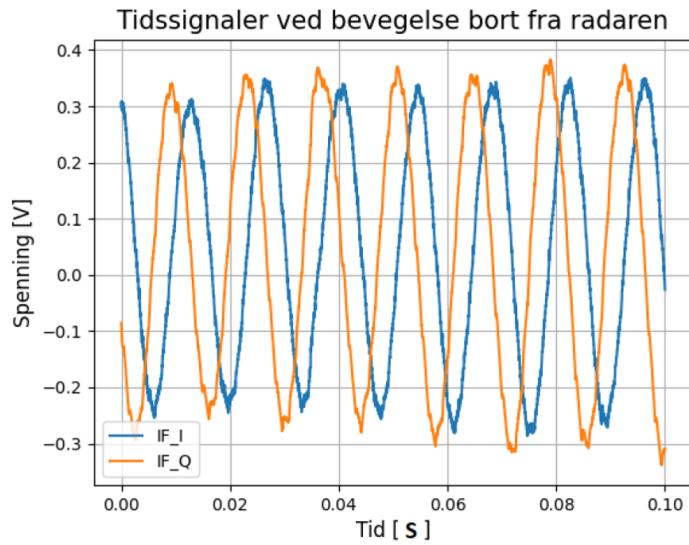
En patch-antenne er en antenn som utnytter feltene som oppstår langs kantene på en rektangulær metallstruktur i topplaget på et 2-lags kretskort med jordplan under. Lengden L på patchen er ca en halv bølgelengde for mikrobølgene som går langs substratet, og dermed får en et felt som reflekteres fram og tilbake fra begge endene slik at en får en stående bølge på og under patchen. Den delen av feltet som brer seg ut fra kantene langs sidene W (såkalte fringing fields) gir opphav til utstråling fra antennen. En enkel patchantenne stråler i en nokså bred lobe i det halvplanet som patchen vender ut mot. Figur III.5 viser strålingsdiagrammet i henholdsvis azimuth- og elevasjon for hele antennen-arrayet.

Strålebredden i elevasjon for antennen-arrayet er ca 80 grader, og det er i samsvar med det en kan forvente fra en enkel patch antennen. I azimuth er strålebredden fra antennen-arrayet ca 12 grader, og dette stemmer også bra med tommelfingerregelen for strålebredden θ til en antennen der den fysiske størrelsen D er stor i forhold til bølgelengden λ .

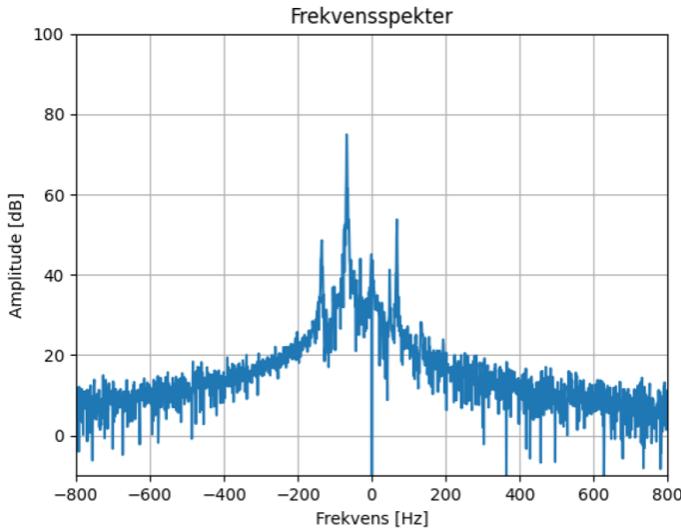
$$\theta \approx \arctan \left(\frac{\lambda}{D} \right), \quad (\text{III.6})$$

der θ gis i radianer og D er lengden på arrayet.

En antenn karakteriseres ofte med hvor stor vinning (Gain) den har. Dette er et forholdstall som forteller hvor mye sterkere det utsendte feltet i en gitt retning fra antennen er i forhold til om samme sendereffekt hadde blitt sendt ut på en såkalt *isotropisk* antennen. Sistnevnte antennen sender all effekt likt i alle retninger over et kuleskall. For en retningsbestemt antennen vil maksimal vinning opptre midt i hovedloben der feltet er sterkest. Jo smalere hovedloben er, jo sterkere vil feltet være

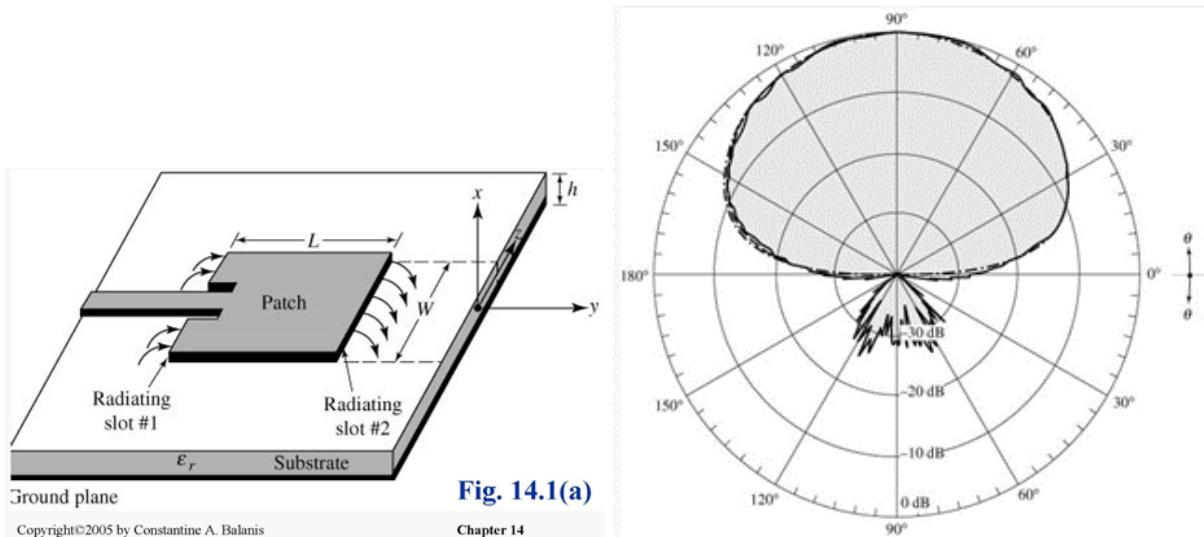


(a) Rådata fra I - og Q -mikser.



(b) Dopplerspektrum.

Figur III.3: Dopplerskift til et objekt som beveger seg vekk fra radaren. Den sterkeste frekvenslinja ved ca -80 Hz representerer objektet. Legg merke til speilfrekvensen som opptrer ved ca +80 Hz. Denne kommer av at IQ mikseren ikke er helt ideell. Årsaken kan være at faseforskjellen mellom I og Q ikke er nøyaktig lik 90 grader og/eller at amplitudene på I og Q signalene er forskjellige. Legg også merke til at vi har en 2. harmonisk av dopplerfrekvensen ved -160 Hz.



Figur III.4: Enkel patch-antenne med strålingsdiagram [balanis2015antenna].

i hovedloben, og antennen får dermed høyere vinning. Vinning måles ofte i dB_i (desibel relativt isotropisk antenne), og det er viktig å huske at antennevinning ikke er aktiv forsterkning av signalet, men et uttrykk for at utsendt effekt samles i én retning i stedet for å bli spredt isotropisk.

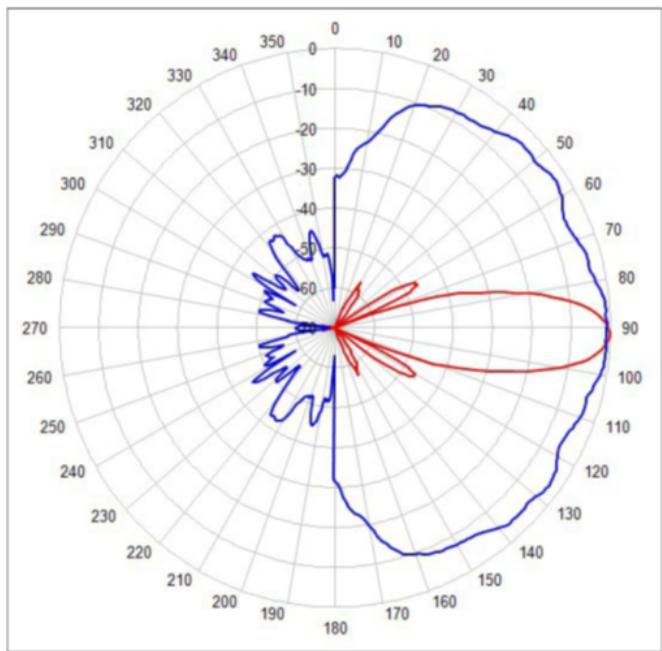
En førsteordens tilnærming til å beregne antennevinning G [dB_i] finnes i Balanis [balanis2015antenna]

$$G \approx 10 \log_{10} \left(\frac{30000}{\theta_e \theta_a} \right), \quad (\text{III.7})$$

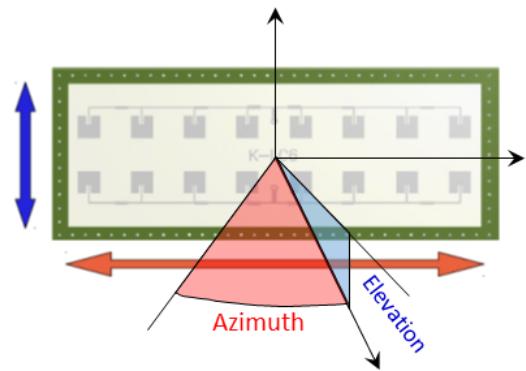
der vinklene θ_e og θ_a er 3 dB strålebredden i hhv elevasjon og asimut oppgitt i grader.

Når utsendt radarpuls treffer et objekt, vil innfallende effekt reflekteres og spres i ulike retninger. Noe av effekten blir spredt tilbake til radaren og kan fanges opp av mottakeren. Styrken på det tilbakespredte signalet vil avhenge av objektets form og størrelse, bølgelengden på radarsignalet, og de elektriske materialegenskapene til objektet (elektrisk permittivitet og konduktivitet). For å få et kvantitativt mål på spredningsegenskapene, har en innført størrelsen radartverrsnitt σ [m²]. Dette er arealet til tverrsnittet av en tenkt kule som sprer innfallende radarbølgje likt i alle retninger (dvs isotropisk) - og som gir tilsvarende signalstyrke i radarmottakeren som den faktiske refleksjonen fra det virkelige objektet. Et stort metallobjekt som f.eks et stålskip kan ha et radartverrsnitt på flere hundre kvadratmeter, mens en liten fugl kan ha radartverrsnitt på 0.1 m².

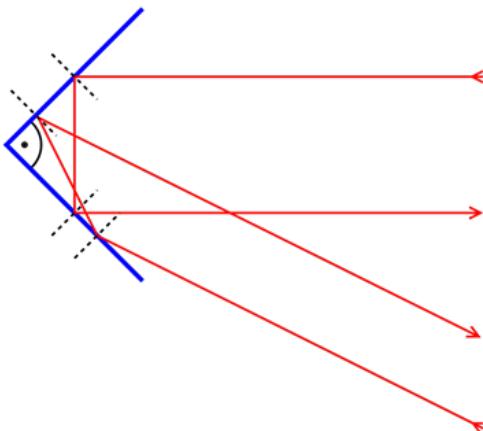
En hjørnereflektor består av 2 eller 3 metallflater som står 90 grader på hverandre som vist i Figur III.6. En slik reflektor har den unike egenskapen at innkommende bølger reflekteres tilbake i nøyaktig den samme retningen som de kom fra. Dette skjer fordi bølgene reflekteres to ganger på to plane metallflater som står 90 grader på hverandre. Radartverrsnittet for en trihedral hjørnereflektor med triangulære flater er tilnærmet gitt ved



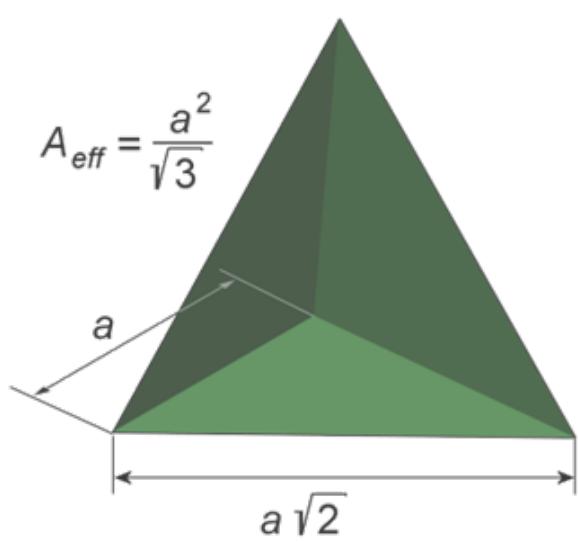
Azimuth 12° , Elevation 80°
At IF output voltage -6dB ,
corresponds to -3dB Tx power



Figur III.5: Strålingsdiagram for antennearrayet i henholdsvis azimuth- (rød kurve) og elevasjonsplanet (blå kurve).



(a) Dihedral hjørnereflektor.



(b) Trihedral hjørnereflektor.

Figur III.6: To ulike hjørnereflektorer.

$$\sigma = \frac{4\pi a^4}{3\lambda^2} \quad (\text{III.8})$$

Når en kjener antennevinningen og radartverrsnittet, kan en beregne hvor mye effekt som kommer tilbake til radarmottakeren fra et objekt ved å bruke radarligningen [skolnik2001introduction]

$$\frac{P_r}{P_t} = \frac{G_t G_r \sigma \lambda^2}{(4\pi)^3 R^4}, \quad (\text{III.9})$$

der P_r er mottatt effekt, P_t er sendereffekt, G_t og G_r er hhv antennevinning til sender- og mottakerantennene, og R er avstanden til målet. Legg merke til at mottatt effekt er proporsjonal med R^{-4} . Hvis en kan beregne støyeffekten i mottakeren, kan en også finne ut hva minimum mottakersignal må være for at et mål skal detekteres. Som regel kreves det et gitt signal-til-støyforhold for å få en ønsket deteksjonssannsynlighet. Vi kan for enkelhet skyld anta SNR = 10 dB som et krav.

Støyeffekten til mottakeren beregnes ved hjelp av ligningen

$$N = kT_{rx}B, \quad (\text{III.10})$$

der $k = 1.38 \cdot 10^{-23}$ J/K er Boltzmanns konstant, T_{rx} er mottakerens støytemperatur i Kelvin, og B er båndbredden i Hz. Hvis en bruker en FFT til å prosessere hele tidsserien (observasjonstiden T), vil en få en teoretisk båndbredde etter prosessering på $B = \frac{1}{T}$.

9 Oppgaver

9.1 Forberedelser

1. Utled et uttrykk for teoretisk dopplerskift som funksjon av radiell hastighet ved 24.13 GHz senterfrekvens.
2. Anta at dere kan måle hastigheten til et objekt med observasjonstiden T på ett sekund. Hva blir teoretisk måleplassning for måling av hastigheten?
3. Beregn antennevinning ut fra ligning (III.7) og sammenlign med data som fins i databladet for radaren.
4. Beregn radartverrsnittet ved 24 GHz til en hjørnerreflektor som har sidekant $a = 15.5$ cm.
5. Hvor mye må reflektoren beveges radielt for at I - Q -phasoren skal foreta et 360-graders faseomløp?

9.2 Forberedelse: Test radaren

- Koble I - og Q -kanalene på radaren til hver sin oscilloskopkanal på Digilenten.
- Bruk spenningsforsyningen til Digilenten til å gi +5V til radaren. Les databladet **nøyne** så dere ikke ødelegger radaren. Dere trenger ikke å kople til det aktive filtert for dette eksperimentet.
- Åpne oscilloskopet i Waveforms. Se på de to kanalene mens dere beveger en hånd foran radaren. Hva ser dere?
- Gå til menybaren i oscilloskop-vinduet og trykk på "+XY". Dette plotter kanalene i et XY-plott. Hva er faseforskjellen mellom kanalene?
- Frivillig: Gå til menybaren i oscilloskop-vinduet og åpne audio-menyen. Prøv å spille av signalet på den ene kanalen som audio. Bruk 48 kHz som utgangspunkt i playback rate, men det er mulig dere må prøve frem med instillingene. Kan dere høre dopplerskiftet?

9.3 Laboppgave: Doppler radar

1. Kople opp radaren og bygg de aktive filtrene med forsterker med 20 dB gain til hver av kanalene ved hjelp av operasjonsforsterker (se labmanual). Forsterkeren skal også inneholde et aktivt bandpassfilter med passband fra 3.5 Hz — 2.8 kHz. Hvorfor er disse knekkfrekvensene fornuftige å bruke bl.a ut fra kjennskap til forventet dopplerskift og AD-konverterne som brukes? Finn ut hvilke komponenter i oppkoplingen til op-amp/filter som bestemmer henholdsvis øvre- og nedre knekkfrekvens og beregn disse komponentverdiene ut fra de spesifiserte knekkfrekvensene. Mål frekvensresponsen til forsterken med filteret. Lagre data slik at dere kan generere Bode-plott til labrapport.
2. Bruk radaren til å måle hastigheten til et objekt som beveger seg radielt i mot eller vekk fra radaren. Finn en egnet alternativ metode for å måle hastigheten (f.eks vha stoppeklokke over en kjent avstand). Bruk forskjellige hastigheter og lag plott av målt hastighet versus teoretisk hastighet. Bruk en kompleks Fouriertransform av I - og Q -signalene til å finne spekteret slik at en kan finne både negativt og positivt dopplerskift.

- Gjenta dette punktet 4 ganger for hver hastighet. Mål minimum 2 forskjellige hastigheter når bilen kjører mot radaren og 1 hastighet når bilen rygger vekk fra radaren. I alt blir dette minimum 12 målinger.
3. Plott eksempler på tidsserier av rådata (I - og Q -signalene). Sammenlign amplitudene til disse signalene. Forsøk å skalere amplitudene slik at de blir mest mulig like før dere prosesserer dataene videre vha FFT. Plott dopplerspektrum i dB, og beregn SNR ut fra dopplerspekteret. (Her må dere vurdere hvilke deler av støyen i spekteret som er termisk støygulv og hva som er interferens).
 4. Undersøk om oppløsningen i dopplerspekteret samsvarer med teoretisk verdi for doppleropp-løsning. Tips: Forsøk å estimere 3-dB bredden på det signalet som dere mener representerer dopplerskiftet.
 5. Gjør en analyse av målenøyaktigheten til hastighetsmålingen og estimer standardavviket ved forskjellige hastigheter.
 6. Drøft feilkilder og hvilke faktorer som påvirker målenøyaktigheten.

Hvis tiden tillater det, kan en også forsøke å lage et spektrogram av dopplerskiftet. Et spektrogram er et intensitetsplot som viser dopplerskift som funksjon av tiden målingen pågår.

9.4 Demonstrasjon

For å få godkjent oppgaven skal dere gjennomføre en enkel demonstrasjon av systemet. Denne demonstrasjonen skal som minimum inneholde følgende, i tillegg til å få godkjent på forberedelsesoppgavene:

1. Bode-plott som viser frekvensresponsen til begge båndpassfiltrene. NB: Ta vare på måledata og plott data i Python; det holder ikke å bare ta skjermdump av Bode-plottet fra Digilent.
2. Vis at dere har gjort alle hastighetsmålinger som ble etterspurt i laboppgaven.
3. Plott tidsserie og spektrum av minst én hastighetsmåling, og vis hvordan dere finner hastigheten ut fra spekteret.
4. Vis estimat av signal-støy-forhold til radarmålingene ut fra frekvensspekteret.
5. Beregning av varians/standardavvik for hastighetsmålingene, og forklar hvordan dere laget estimatene.

Del IV

Optikk

Denne oppgaven har tre aspekter: (1) måle et signal på best mulig måte, ta høyde for at målesituasjonen din kan være dårlig/bra avhengig av skygge, lysforhold, hvor mye i ro objektet er (2) anvende signalprosessering for å dra puls ut av et signal (3) forstå resultatene ved hjelp av forenklede modeller og teori for fotontransport i hud, sette teorien i sammenheng.

Hovedutfordringen i denne oppgaven er ikke tilkoblingen eller selve opptaket, hovedutfordringen er å skrive kode for å prosessere signalene slik at du får ut puls.

10 Teori

10.1 Introduksjon

Måling av puls har vært en del av medisinske undersøkelser i svært lang tid. Hos intensivpasienter overvåkes pulsen og oksygenmetningen i blodet kontinuerlig. Pulsen kan enkelt måles manuelt med en finger mot et pulspunkt og ei klokke, og vi kan også logge den nokså enkelt med en relativ billig optisk sensor eller et kamera. Måling av puls i forbindelse med trening er blitt allemannseie etter pulsklokkenes inntog for noen år siden. I dag er det mest typiske å stroppe på seg en «heart rate monitor» (HRM) rundt brystet. Disse har en liten innvevd krets tilkoblet, og leser ut elektriske pulsdata via elektroder i pulsbeltet som blir sendt trådløst videre til klokke eller smarttelefon.

Det har etter hvert kommet diverse klokker på markedet som istedet måler pulsen optisk. Eksempler er Fitbit eller Apple Watch. Det finnes i tillegg flere apper til smarttelefoner som kan gjøre dette ved hjelp av kameraet på telefonen.

I dette labeksperimentet skal vi forsøke å gjøre det samme, nemlig å måle/estimere puls optisk ved hjelp av kameraet til Raspberry Pi.

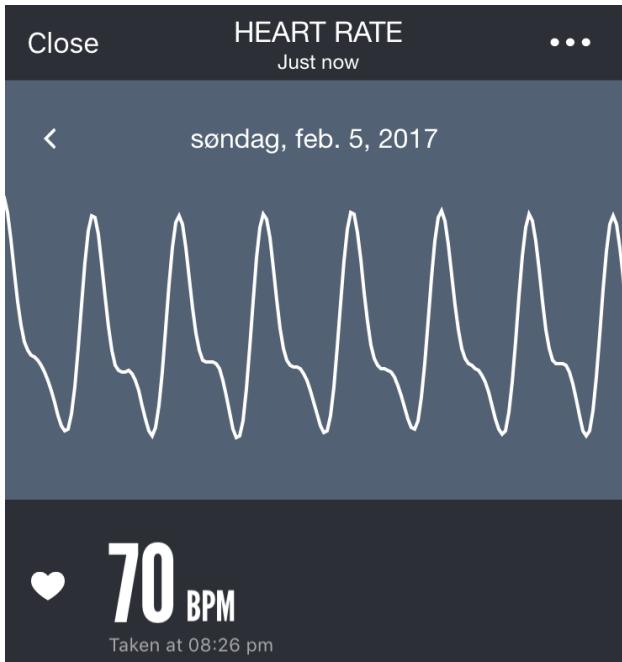
10.1.1 Måling av puls

Pulsen hos et menneske varierer med kjønn, alder og fysisk form. Dersom vi er syke eller utsatt for fysiske eller psykiske stressende situasjoner vil også pulsen kunne endre seg. Hos et vanlig friskt menneske (som ikke er toppidrettsutøver i en utholdenhetsidrett) ligger vanligvis pulsen mellom 40-50 og 230 slag per minutt. Hjertet pumper blod fra lungene og ut i kroppen via arteriene. Disse blodårene vil derfor pulsere i takt med hjerteslagene, men med en liten tidsforsinkelse/faseforskjell.

Hjerteslagene fører altså til at blodet beveger seg ut i vevet på en tidsavhengig måte. Hver gang hjertet pumper blod ut i kroppen kommer det en ny bølge av oksygenrikt blod ut i årene, som fører til attrukket og mengden blod i blodårene øker. Det er denne endringen i blodmengde vi skal forsøke å måle optisk.

Blod har en kraftig rød farge. Blodet som kommer fra lungene inneholder mer oksygen og har en annen farge enn blodet som har gitt fra seg oksygen til vevet. Fargen til blod skyldes hemoglobin, molekylet som binder oksygen til blodet. Hver eneste røde blodcelle inneholder hemoglobinmolekyler. Inne i et hemoglobinmolekyl finnes det igjen fire heme-grupper, som hver kan binde et oksygenmolekyl. Oksygenet frigis til vevet på vei gjennom blodbanen.

Heme ligner mye på andre stoffer som gjør tilsvarende jobb i andre arter. Midt inne i hememolekylet ligger et jernatom, som er viktig både for fargen til molekylet og for evnene til å binde oksygen (en



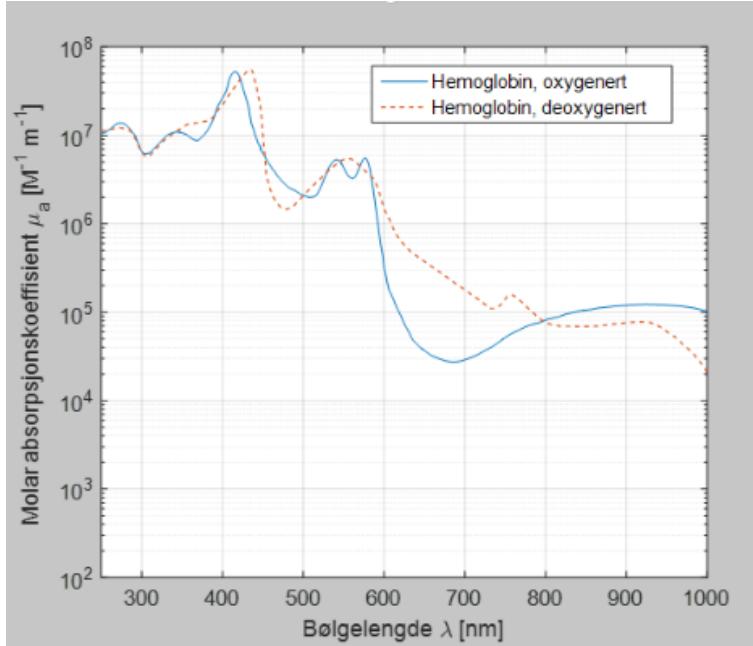
Figur IV.1: Eksempel på en pulskurve målt med mobiltelefon.

blekksprut har for eksempel et nokså likt molekyl som oss, men med kobber i stedet for jern. Blekkspruten har derfor blått blod). Når heme-molekylet binder oksygen forandrer formen på molekylet seg og dermed også lengden på bindingene mellom de ulike atomene. Dette gjør at blodet endrer farge og dermed også absorpsjonsspekter. Friskt oksygenmettet blod har en skarp, lys rød farge, mens blod fra venene har en mer leveraktig, mørk rødfarge. Absorpsjonsspektret til blod er vist i figur IV.2.

Oksygenert blod har tydelige absorpsjonstopper ved ca 410 nm, 542 nm og 576 nm, mens deoksygenert blod har tydelige toppe ved ca 420 nm, 555 nm og 760 nm. For å forvirre dere ytterligere kan det nevnes at det finnes også andre typer hemoglobin. For eksempel er det slik at hos røykere så vil blodet inneholde carboxyhemoglobin, som er hemoglobin med karbonmonooksyd bundet i stedet for oksygen. Blodet vil da se mer rosa ut på farge. Det skal dere imidlertid ikke ta hensyn til i denne oppgaven. Her skal dere kun ta hensyn til oxy- og deoxyhemoglobin.

Når vi skal måle puls optisk er det variasjonen i blodmengde vi skal måle, og det blir derfor et potensielt problem for oss at også oksygenmengden varierer siden absorpsjonkurvene for blod med og uten oksygen er nokså ulike. Dette kunne blitt løst ved å måle signalet på et punkt der kurven for oksygenert blod og ikke-oksigenert blod krysser hverandre. Dette kalles et isosbestisk punkt. Her vil absorpsjonen kun variere på grunn av blodmengden og ikke på grunn av endringer i oksygeneringen. Isosbestiske punkter kan finnes blant annet ved 548 nm, 586 nm og 805 nm. Denne muligheten har vi ikke for denne laboppgaven, da bølgelengdene i RPi-kameraet er faste.

Endringer i oksygennivå kan også måles ved hjelp av pulsoksymetri. Dette kan for eksempel gjøres ved å bruke et isosbestisk punkt og et punkt med maksimal avstand mellom absorpsjonskurvene. Se for eksempel følgende artikler:



Figur IV.2: Blodabsorpsjon for oksygenert og ikke-oksygenert blod.

- <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4574660/pdf/3320.pdf>
- <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5175543/pdf/4941.pdf>

10.1.2 Generell biooptikk

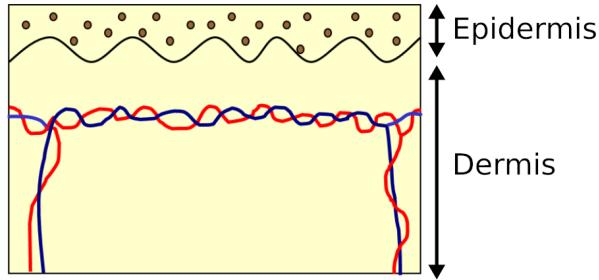
Når lys treffer huden kan det enten fortsette i samme retning, reflekteres, spres eller absorberes. Sannsynligheten for at lyset skal absorberes kan uttrykkes ved absorpsjonskoeffisienten. Dette er en kombinasjon av sannsynligheten for at lyset absorberes og den faktiske tverrsnittet av absorbatoren. Dersom vi har et materiale som bare absorberer lys kan vi beskrive hvordan lyset absorberes ved hjelp av Beer-Lamberts lov:

$$I(z) = I_0 \exp(-\mu_a z), \quad (\text{IV.1})$$

Her er I_0 den innfallende intensiteten på lyset, z er avstanden lyset har beveget seg og μ_a er absorpsjonskoeffisienten.

Denne formelen kan brukes dersom du har en hemolysert blodprøve i et glass og vil finne ut hvor mye den absorberer. Dessverre vil den ikke virke i vev. Beer-Lamberts lov virker bare for materialer med svært lite eller ingen spredning. Vevet vårt, og kanskje spesielt huden har høy spredning.

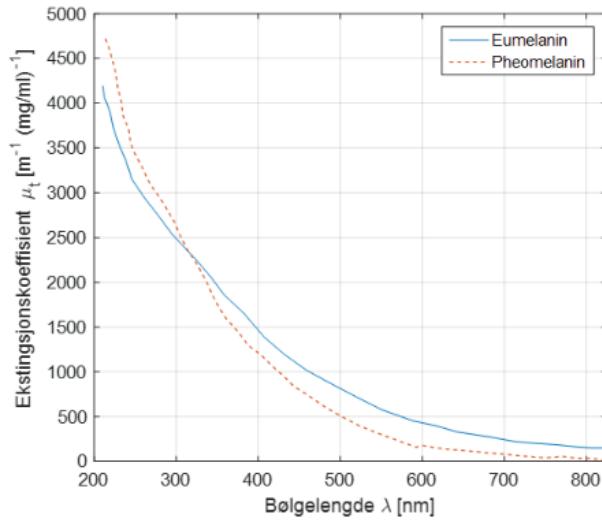
Huden er ikke gjennomsiktig, selv ikke hos de blekste av oss. Det skyldes at lyset spres av fett, bindevev og cellevegger. Spredningen i vev kan på samme måte som for absorpsjon beskrives som en eksponensiell sammenheng, bare med spredningskoeffisienten i stedet for absorpsjonskoeffisienten. Det finnes flere typer spredning i vev. Vanligvis beskriver vi det som en kombinasjon av Mie-spredning fra molekyler som er like store som eller større enn bølgelengden, og Rayleighspredning fra strukturer som er mindre enn bølgelengden.



Figur IV.3: Svært forenklet og stillisert tverrsnitt av den øverste delen av huden.

Miespredning er den typen spredning vi ser i vanndråpene i skyer eller tåke. Den er ikke bølgelengdeavhengig og gir en hvit eller gråaktig farge. I vevet er det blant annet fett og cellevegger som gir Miespredning. Rayleigspreddning er svært bølgelengdeavhengig og gjør at blått lys vil spres fire ganger mer effektivt enn rødt lys. Dette, sammen med absorpsjonen, påvirker hvor langt inn i vevet de ulike bølgelengdene når. Blått lys spres veldig effektivt og når dermed ikke så langt inn i vevet, mens rødt lys spres sjeldnere og har større innstrekningsdybde.

I figur IV.3 ser man et eksempel på et tverrsnitt av hudens øverste del. Det øverste laget, epidermis, utgjør en beskyttende barriere, og består i stor grad av keratin. Dette laget inneholder pigmentet melanin. Epidermis innholder ikke blod og er fra ca 80-150 micrometer tykt. Den bølgende grensen mellom epidermis og dermis kalles basallaget. Her dannes cellene i laget over, inkludert pigment. Basallaget får næring fra blodkarene i dermis (nesten lag). Blodårene i dermis ligger med ulik tetthet i ulike dybder, her er arterier tegnet som røde, mens venene er blå. Her er kun det øverste laget vist. Dypere ned i huden finner man flere blodkar, og nederst ligger et lag med underhudsfett.



Figur IV.4: Figur av melaninabsorpsjon. Det finnes to typer melanin i huden, eumelanin som finnes i mørkt og enkelte typer blondt hår, og pheomelanin som finnes i rødt og enkelte typer blondt hår.

Hvor langt inn lyset når og hva som skjer med det er altså avhengig av hvordan vevet er sammensatt

og hva som sprer og hva som absorberer. I tillegg til blod inneholder huden en rekke andre stoffer som absorberer lys. Fett, vann og fargestoffer fra mat kan påvirke absorpsjonen, men innenfor den synlige delen av lysspektret er bidragene fra slike stoffer begrenset.

De viktigste fargestoffene i huden er hemoglobin i blod og pigmentet melanin. Melanin ligger ytterst i huden og har som hensikt å filtrere bort skadelig UV-stråling før det når kritiske strukturer som kan skades av strålingen. Det har derfor en absorpsjonskarakteristikk som et slags lavpassfilter i frekvensdomenet. Lyset vi skal måle med må gjennom laget med pigment i huden før det når ned til blodårene. Det at både absorpsjon og spredning er høy for korte bølgelengder gjør at kortbølget lys når svært mye kortere inn i vevet enn langbølget lys. Denne effekten kan man enkelt observere ved å holde en lommelykt under fingrene og lyse gjennom. Rødt lys når gjennom fingeren, men ikke blått.

10.2 Kræsjkurs i regning på fotontransport

10.2.1 Diffusjonslikningen

For å regne på fotontransport i hud, ignorerer vi vanligvis alt som har med bølgeegenskapene til lys (koherens, polarisering, interferens, ...), og ser heller på hvordan energien forplanter seg gjennom vevet. På toppen av hudoverflaten har vi en innkommende energi, og gjennom vevet vil energien enten bli absorbert eller spredt.

Med krav om bevaring av energi, kommer man frem til en likning for energiflyten i ulike retninger, i en gitt posisjon i vevet, uttrykt med ulike radiometriske størrelser (Boltzmanns transportlikning, ikke gjengitt her). Ved å anta at spredningen er mye høyere enn absorpsjonen og at avviket fra isotropien er lite, gir det en forenklet likning i kun fluensraten ϕ :

$$\frac{\partial \phi(\vec{r}, t)}{c \partial t} + \mu_a \phi(\vec{r}, t) - D \nabla^2 \phi(\vec{r}, t) = S(\vec{r}, t) \quad (\text{IV.2})$$

Diffusjonskonstanten D er gitt som $D = \frac{1}{3(\mu'_s + \mu_a)}$. Størrelsen μ'_s er redusert spredningskoeffisient.

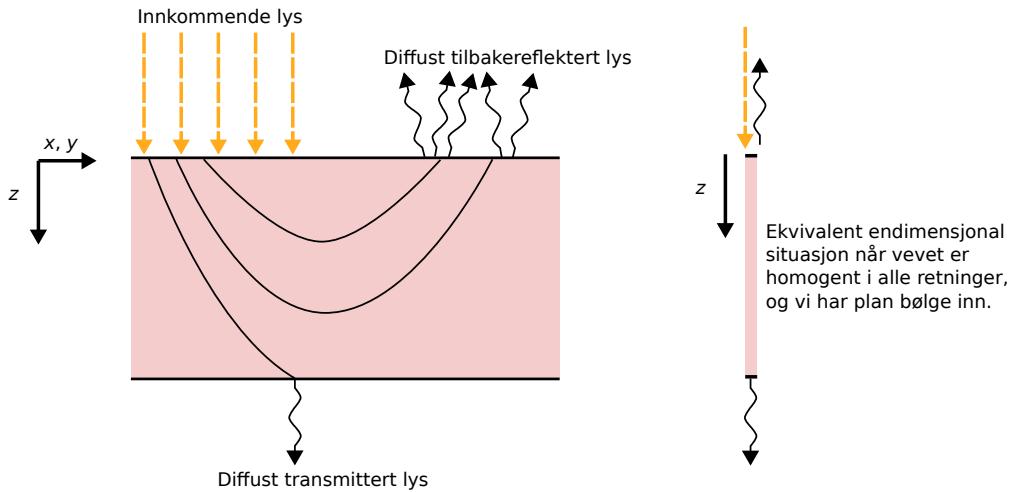
Dette er en partiell differensielllikning i $\vec{r} = [x, y, z]^T$ og t . I denne laboppgaven kan vi stort sett se bort ifra tidsavhengigheten. Vi kan også gjøre en antakelse om at vevet stort sett oppfører seg likt og at vi har en plan bølge inn, og redusere fluensraten ϕ til å være avhengig kun av z , dybden i vevet. Se figur IV.5.

Om vi i tillegg antar at fotonkilden er en Dirac-delta-funksjon $S(z) = \delta(z)$, kan likningen for lystransport nede i vevet forenkles til

$$\mu_a \phi(z) - D \frac{d^2}{dz^2} \phi(z) = 0. \quad (\text{IV.3})$$

Vi har tilleggshensyn som grensebetingelser mellom hud og luft, spekulær refleksjon (Snells lov, ...), lagdeling av menneskehud, grensesjikt i huden og annet, men vi velger i denne laboppgaven å se bort ifra alt dette for å få en enklest mulig modell som skal brukes til å forstå resultatene på et kvalitativt nivå.

Løsningen på denne likningen er



Figur IV.5: Koordinatsystem for fotontransportmodellen.

$$\phi(z) = \phi(0) \exp(-Cz), \quad (\text{IV.4})$$

hvor $C = \sqrt{3\mu_a(\mu'_s + \mu_a)}$ og $\phi(0) = \frac{1}{2\delta\mu_a}$.

10.2.2 Penetrasjonsdybde

Optisk penetrasjonsdybde er et mål på hvor langt lyset når inn i vevet. Optisk penetrasjonsdybde δ i hud kan uttrykkes ved å se på hvilken z innkommende energi har blitt redusert til $1/e \approx 36.8\%$. Se figur IV.6.

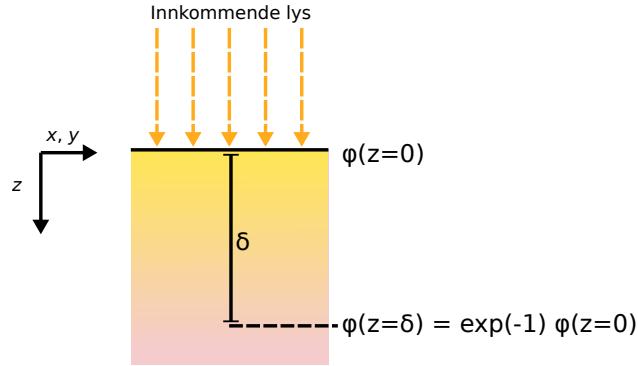
Om samme definisjon brukes på $\phi(z)$, må vi finne for hvilken dybde z at $\phi(z)$ er redusert til $\phi(0) \exp(-1)$. Løsningen på dette for (IV.4) er

$$\delta = \sqrt{\frac{1}{3(\mu'_s + \mu_a)\mu_a}}. \quad (\text{IV.5})$$

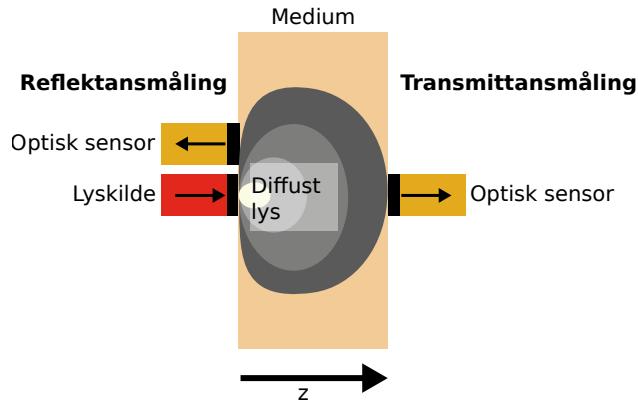
10.2.3 Måling av lys på overflaten

Måling av lyset vil skje på utsiden av huden. Det er derfor mest interessant å bruke modellene til å se på hva som kommer ut igjen av emittert lys, og hvor dypt lyset har probet. Det skiller mellom reflektans og transmittans, vist i figur IV.7.

Transmittans Anta en lyskilde inn på et medium (e.g. en finger). Transmittans er definert som energien som kommer ut på den andre siden, relativt til innkommende energi, og måler mengden lys som har gått gjennom hele mediet og passert ut på den andre siden.



Figur IV.6: Definisjon av penetrasjonsdybde δ .



Figur IV.7: Måling av reflektans og transmittans.

Vi ser på fluensraten ved dybden d relativt til fluensraten ved overflaten for å få et inntrykk av oppførselen til transmittansen:

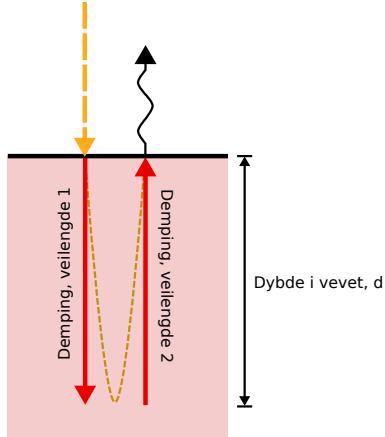
$$T = \frac{\phi(z = d)}{\phi(z = 0)} = \exp(-Cd). \quad (\text{IV.6})$$

Eksempel: Sette ei lommelykt på håndflaten din. Transmittansen blir mengden lys som kommer ut på oversiden.

Siden dette er energi du har målt på den andre siden, vet du at det har probet hele tykkelsen av mediet du mäter. Informasjonen du kan trekke ut herifra sier noe om hele fingeren.

Reflektans Anta en lyskilde inn på et medium. Reflektansen er definert som energien som kommer ut igjen på samme side, relativt til innkommende energi, og mäter mengden lys som har probet mediet og blitt spredt tilbake.

Betrakninger om probet dybde For å gi et inntrykk av probet dybde, anta at uttrykket $\exp(-Cz)$ fra (IV.4) kan måle hvor som er igjen av et lite energibidrag fra lyskilden over en avstand z i mediet. Skal et energibidrag som når ned til dybden d gå ut igjen som reflektans, må det totalt traversere over en lengde $2d$ (se figur IV.8). Av dette bidraget er det igjen $\exp(-2Cd)$.



Figur IV.8: Forenklet modell for reflektans.

Eksempel: Over den optiske penetrasjonsdybden δ og opp igjen vil bidraget være redusert til $\exp(-2) = 0.14$. Over to penetrasjonsdybder, 2δ , vil bidraget være redusert til $\exp(-4) = 0.02$. Med andre ord: Mesteparten av bidragene til reflektansen ut kommer fra mellom hudoverflaten og penetrasjonsdybden, bidragene som går lengre ned dempes fort. *

Betraktninger om relative reflektansnivå mellom forskjellige bølgelengder Likning (IV.3) uttrykker samlet lystransport, og bidragene i løsningen (IV.4) vil inkludere alt som er på vei inn i mediet for første gang i tillegg til det som blir spredt tilbake. Vi kan derfor bruke $\phi(z=0)$ som en pekepinn på hva vi får ut igjen ved ulike bølgelengder.

Ved overflaten er $\phi(z=0)$ proporsjonal med $\frac{1}{\delta\mu_a}$. Vi kan si at

$$R \propto \frac{1}{\delta\mu_a} = \sqrt{3(\mu'_s/\mu_a + 1)}. \quad (\text{IV.7})$$

Grovt sett: Lavere absorpsjon og/eller høyere spredning gir høyere reflektans, og kan brukes til å si noe om de relative forholdene mellom forventet reflektans fra ulike fargekanaler (om vi får høyere verdier fra rødt enn grønt, etc.).

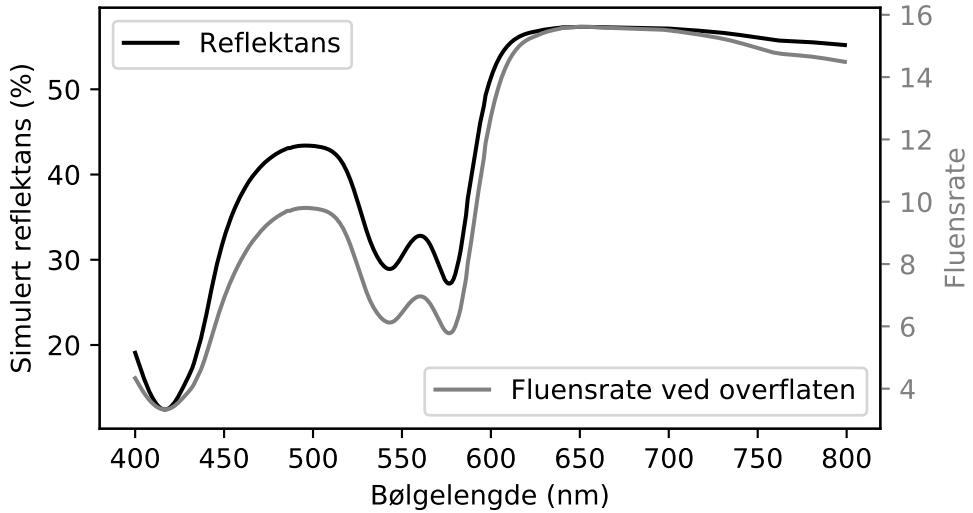
Mer korrekt simulering av reflektans fra en hudmodell over bølgelengdeområdet 400 til 800 nm er sammenlignet med resultatet fra likning (IV.7) i figur IV.9. Den forenkla modellen kan brukes til å si noe om relative nivå, selv om tallverdiene ikke er riktige.

10.2.4 Regning på pulsutslag/blodkontrast

Som forklart tidligere, vil hjerteslagene modulere blodvolumet i vevet. Dette gir endringer i lyset som kommer ut av menneskevevet, og gir oss til slutt et pulssignal.

Dersom vi antar et lite stykke medium og regner ut transmittansen gjennom det for ulike blodvolum kan vi se på hvor stort utslag vi ville fått i pulsmålingen vår når blodvolumet endret seg. Eksempel

*Skulle vi dratt dette et hakk lengre: Lyset reduseres til $\exp(-Cz)$ ned til dybden z , men bare andelen $0.5\mu_s dz$ blir spredt tilbake, og dempes så med $\exp(-Cz)$ på vei opp igjen til overflaten. Ytterligere spredning forover og tilbake av dette antar vi at allerede er inkludert i dempfingsfaktoren på vei tilbake.



Figur IV.9: Simulert reflektans (korrekt definisjon, med korrekte grensebetingelser mot luft og bedre valg av kildefunksjon) sammenlignet med fluensraten ved $z = 0$.

på kvantisering av dette utslaget:

$$\text{Kontrast} = K = (|T_{\text{Høyt blodvolum}} - T_{\text{Lavt/normalt blodvolum}}|)/T_{\text{Lavt/normalt blodvolum}} \quad (\text{IV.8})$$

Her vil en liten endring i T gi K nært 0, mens størst mulig endring (fra en eller annen verdi ned til 0) gir K nært 1.

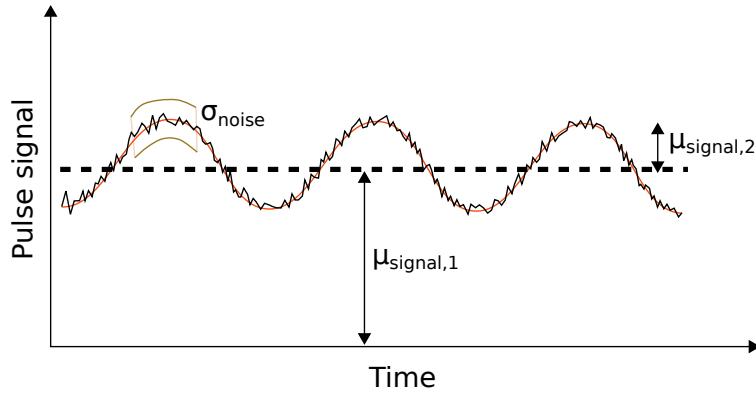
Blodet har ulik absorpsjon på ulike bølgelengder, som vist i figur IV.2. En bølgelengde med høy blodabsorpsjon betyr at T reduseres dramatisk når blodvolumet går opp. Vi får dermed stor pulsamplitude, og får stor kontrast mellom høy og lav blodmengde.

10.3 SNR i pulssammenheng

SNR (signal-to-noise-ratio) kan defineres på mange forskjellige måter. Felles for de fleste er at de bør måle størrelsen, mengden, amplituden, energien til interessesignalet relativt til mengden støy du har ellers i målingen. Den valgte SNR-definisjonen bør være nyttig for å kunne si noe om kvalitet på forskjellige målinger, og være relativt robust og gi sammenlignbare tall.

Eksempler på to definisjoner av SNR:

1. SNR i bildekvalitetsammenheng. Snittsignallnivå over bildet delt på standardavviket til støyen i bildet, $\mu_{\text{signal},1}/\sigma_{\text{image noise}}$. Uttrykker generell bildekvalitet. Interessesignalet er pikselverdien.
2. SNR i pulssammenheng. Maksamplitude i pulssignalet delt på standardavviket til støyen i pulssignalet, $\mu_{\text{signal},2}/\sigma_{\text{pulse noise}}$. Uttrykker kvaliteten på pulsutslaget, hvor begravd pulssignalet er i støy, hvor lett det er å estimere puls. Interessesignalet er pikselvariasjonen knyttet til puls.



Figur IV.10: SNR-definisjoner.

Definisjon 1 ville vært nyttig om vi var ute etter å si noe om signalkvalitet i de ulike fargekanalene på et kamera, slik som generell lysstyrke, sensorkvalitet, ...

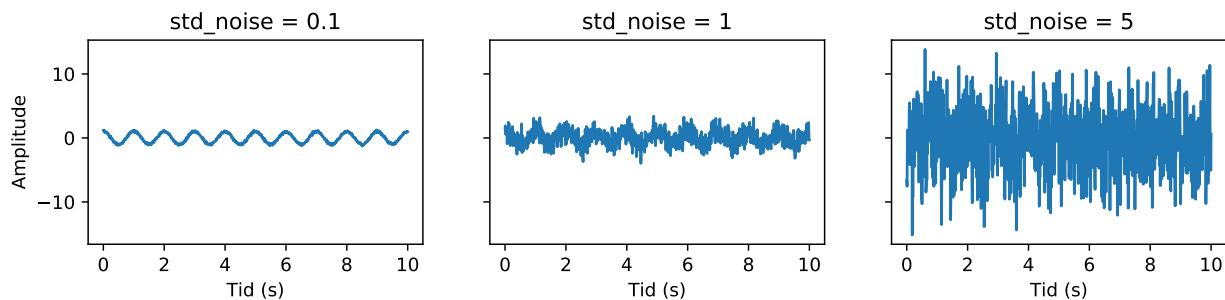
Definisjon 2 ville vært nyttig for å si noe om selve pulssignalet.

SNR-definisjonen over forutsetter et estimat på pulsutslagsamplitude og støystandardavvik. Dette vil gli over i hverandre i tidsdomenet for lave SNR-verdier.

10.3.1 SNR i frekvensdomenet

En mulig løsning på dette problemet er å ty til FFT, og mer klart definere hva som er signal og hva som er støy ved å se på ulike frekvensbøtter.

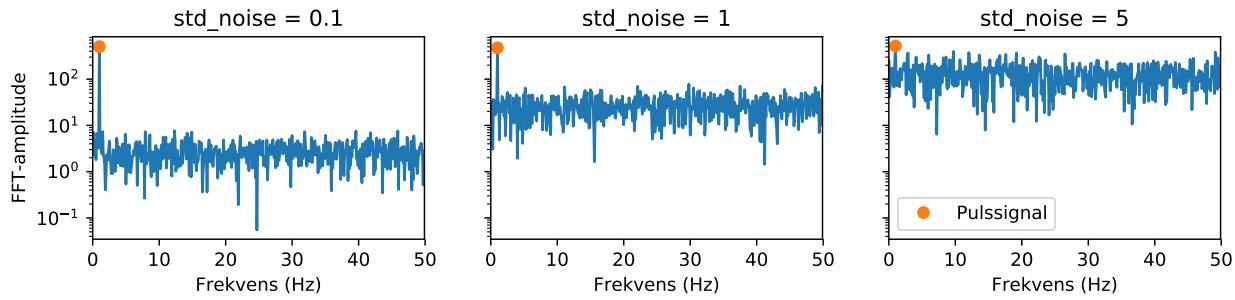
Eksempel: Simulerer en pulsmåling (sinussignal på 1 Hz (60 BPM), $f_s = 100$ Hz) med økende additiv, gaussisk støy, se figur IV.11.



Figur IV.11: Simulerte pulssignal.

Her er det lett å se at den første målingen har best kvalitet, og den siste dårligst. Signalene har kjent amplitude og støystandardavvik, og gir SNR-verdier på 10.0, 1.0 og 0.2 for økende grad av støy. SNR-verdiene kan brukes til å evaluere relativ kvalitet på målingene, men forutsatte at vi kjente pulsamplitude og støykarakteristikk.

FFT av disse signalene er plottet i figur IV.12.



Figur IV.12: FFT av signalene i figur IV.11.

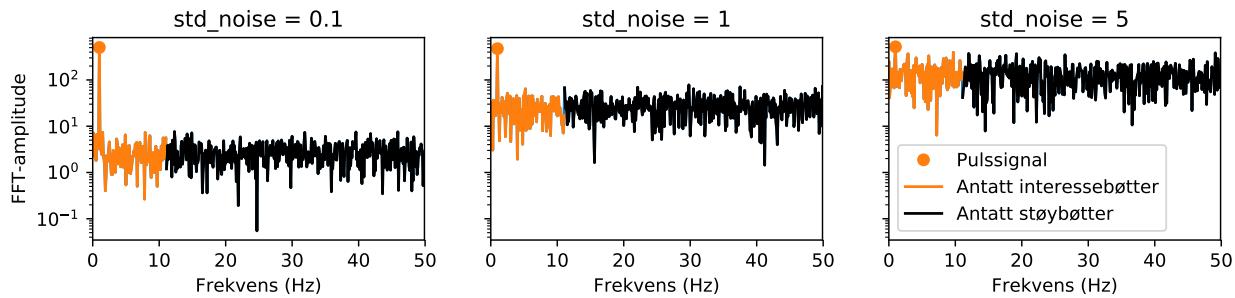
Anta at pulsenfrekvensen allerede er estimert ved hjelp av en eller annen metode. Vi kjenner altså hvilken frekvensbøtte pulssignalet burde ligge i etter at vi har tatt FFT. Denne er synlig som en smal topp i FFT-spekteret. Verdien her gir oss noe som er proporsjonalt med amplituden i signalet. I alle andre frekvensbøtter kan vi anta at vi bare har ulike komponenter av støyen, og estimerer derfor støyamplitude.

Om vi bruker amplituden fra interessefrekvensbøtten(e), og måler dette mot frekvensbøttene med støysignal vi ikke er interessert i, bør vi få et robust estimat på SNR. Eksempel:

$$\text{SNR} = \text{sum(frekvensbøtter med signal)} / \text{sum(frekvensbøtter med støy)} \quad (\text{snitt kan også brukes})$$

I tilfellet over gir en sum følgende verdier før økende støystandardavvik: 0.36, 0.035, 0.0079, og gir mening ut ifra kvaliteten på målingene.

Fallgruver Dette kan bli mer utfordrende om signalet har en viss bredde i frekvensdomenet. Om vi tror signalet (i et ekstremt tilfelle) spenner seg fra 0 til 10 Hz, kunne vi bestemt oss for å sammenligne frekvensbøttene vist i figur IV.13:



Figur IV.13: Demonstrasjon av litt feilvalgte FFT-bøtter.

Men da har vi også inkludert frekvensbøtter med støy, og støyen har mye energi for std_noise = 5. Samme SNR-definisjon som over vil gi følgende tall: 1.43, 0.55, 0.57, men den siste målingen har åpenbart ikke høyere SNR enn den midterste målingen. Problemet her er at støyamplituder inkluderes i målet på interesse-signalet, og ender opp med å gi en sammenligning mellom epler og bananer. Vær obs på hvilke frekvensbøtter som plukkes ut for sammenligning.

Andre hensyn i en reell målesituasjon Pulssignalet er egentlig ikke et sinussignal, men kommer til å bestå av flere komponenter. Du mäter et biologisk system, og pulssignalet kan endre frekvens underveis i målingen. Interessesignalet kan derfor være litt bredere i frekvensplanet, og ikke bare bestå av én frekvenskomponent. Harmoniske kan også forekomme. Tilleggssignaler kan dukke opp, slik som pustefrekvens, bevegelse.

Det kan derfor være vel verdt å være obs på at antakelsen om at alle andre frekvensbøtter kun inneholder støy ikke nødvendigvis er helt riktig. De inneholder riktignok som regel signaler du ikke er interessert i, så kan gå som "støy". Men har du for eksempel bevegelse som modulerer signalet, kan dette ha kraftig utslag i FFT-spekteret ditt, og det kan bli rart å regne ut SNR fra dette. Husk å sjekke tallene, og sjekk at måten å regne SNR på gir mening.

Bruk en fast frekvens for pulsfrekvensen når du regner ut SNR. Pulsfrekvensen har du estimert med andre metoder, og du skal evaluere hvor mye signal denne frekvensen har, relativt til alle de andre signalene du ikke er interessert i i pulsmålingen.

Om det ikke er mulig å estimere pulsfrekvens fra en gitt fargekanal, bruk pulsfrekvensen estimert fra en av de andre kanalene for å kunne si noe om SNR i den "umulige" fargekanalen.

10.3.2 Avsluttende bemerkninger om SNR

Meningen med SNR-beregningene er å bruke det som et verktøy for å sammenligne målinger, og kunne kvantisere "dårlig pulsmåling" med et tall. Tallet må gi mening for at det skal være noen nytte i det. Viser det seg å være problemer med metoden, kan det diskuteres i sluttrapport.

For det meste kommer mengden støy til å endre seg lite fra måling til måling og fargekanal til fargekanal. Det som vil endre seg er amplituden på pulsutslaget, som påvirkes av biooptikken som er forklart tidligere. Om støyen kan verifiseres å være konstant, kan det være like greit å sammenligne basert på pulsamplitude alene.

10.4 Forskningslitteratur

10.4.1 Fjernmåling av puls med kamera

Enkelte forskningsgrupper jobber aktivt med det dere skal gjøre i laben. For de nysgjerrige som vil søke litt rundt, er typiske nøkkelord "photo-plethysmography", "plethysmography", "remote", "rgb". En fin startartikkel kan være <https://www.osapublishing.org/oe/abstract.cfm?uri=oe-16-26-21434>. Denne er sitert av mange andre artikler man kan rote seg videre bort i (https://scholar.google.no/scholar?cites=16511870114372664801&as_sdt=2005&sciodt=0,5&hl=no).

10.4.2 Fotontransport

Vi tok en del snarveier i modellutledningene for å kunne gjøre betraktninger om hva som egentlig skjer uten å innføre for mye kompleksitet. Nysgjerrige kan oppsøke følgende artikler for mer matematisk og fysisk korrekt behandling av grensebetingelser og størrelser som reflektans:

- Haskell et al., "Boundary conditions for the diffusion equation in radiative transfer", J. Opt. Soc. Am. A, Vol. 11(10), 1994.

- Svaasand et al., "Tissue parameters determining the visual appearance of normal skin and port-wine stains", *Laser. Med. Sci.*, vol. 10(55), 1995 (se appendix).

11 Forberedelsesoppgaver

11.1 Oppgave 1: Fototransportberegninger

- Regn ut optisk penetrasjonsdybde i typisk menneskevev for bølgelengdene RGB-kameraet er sensitiivt for*.
- Mål tykkelsen av fingeren din. Av lys som kommer inn i en slik finger, estimer med modellen hvor stor prosentandel som kommer ut igjen på den andre siden, for hver av de tre fargekanalene. Hva slags fingerdybde har du probet?
- Hvis du mäter reflektans, hvor dypt forventer du at signalet har probet fingeren på ulike bølgelengder?
- Anta en blodåre med diameter $300 \mu\text{m}$. Regn ut hvor mye lys som når ut på den andre siden av blodåren (blodvolumsfraksjon 100%), og sammenlign med hvor mye som når gjennom et tilsvarende stykke vev (blodvolumsfraksjon 1%). Regn ut kontrasten. Hva sier dette om forventet pulsamplitude?
- Hvilken fargekanal forventer du at vil fungere best til pulsmåling, og hvorfor?

11.2 Oppgave 2: SNR

Du har trukket ut et pulssignal fra en kameravideo. Du kjenner amplituden på pulssignalet og sannsynlighetsfordelingen til støyen.

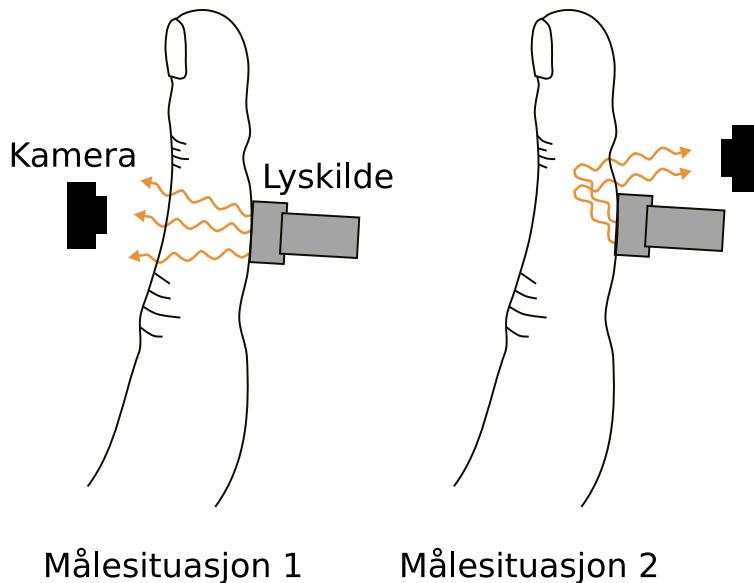
Hvordan kan du definere SNR dersom du er interessert i å se på hvor klart pulssignal du har?

Hvordan kan du gjøre dette med en FFT-basert tilnærming om du ikke har en god karakteristikk av støyen eller pulsamplituden?

12 Laboppgaver

Oppgaven har to implisitte deler: Samle data og implementere dataprosessering.

Det kan være dere blir nødt til å gjøre dette litt iterativt, da det ikke er sikkert datakvaliteten på de første dataene dere samler er god nok. Det anbefales at det først tas data av transmittans gjennom fingeren, før prosessering implementeres, og dere deretter tar mer data. Se tips til datafangs og prosesseringsssteg i eget notat.



Figur IV.14: Målesituasjoner: Fingertransmittans, fingerreflektans.

12.1 Samle og analysere data

12.1.1 Del 1: Basismålinger

For alle målesituasjonene som følger bør du ha:

- Tatt minst fem målinger av hvilepuls og regnet på snitt og standardavvik for puls estimert av hver av fargekanalene, og sammenlignet med en uavhengig måling tatt med e.g. en smartfonapp for å måle puls, pulsklokke, e.l.
- Regnet på SNR for hver av fargekanalene for en av de fem målingene *.

Målesituasjoner:

1. Sett ei lommelykt direkte mot fingeren, og mål det som kommer ut på den andre siden (transmittans)

Når du har gjort dette for én måling, er det egentlig lurt å bare begynne å implementere prosessering, ved å se på signalet dere får ut og prøve å trekke ut puls. Transmittanssignalet er som regel det enkleste signalet † å få en puls ut av, og som et første steg er dette veldig fint

*Informasjonen står i kameradatabladet på Blackboard. Hver fargekanal er sensitiv over et større bølgelengdeområde, forenkle dette ved å bruke bølgelengde som svarer til maksimum respons. Bruk skjelettscript ute på Blackboard for å gjøre beregningene (gir μ_a og μ'_s for typisk menneskehud). Anta en blodvolumsfraksjon (bfv) på 1% og en blodoksygenering (oxy) på 80% (satt i scriptet).

*Script på blackboard. Les teoridelen om dette, og sjekk at ting faktisk gir mening ved å kikke på FFT-spekteret. Siden det kan bli litt manuelt jobb av dette, trengs SNR bare for en av de fem målingene.

†Som regel... Fingeren kan ofte også være den delen av kroppen du har aller minst pulssignal fra. Plages du av kalde fingre og føtter? Da vil du også bli plaget med umulige målinger, siden kalde fingre har sammenheng med dårlig blodsirkulasjon. Dette er et kjent problem for pulsmålere på sykehuset. Varm opp fingrene dine tilstrekkelig (gni dem mot hverandre e.l.) før du gjør måling, så utvider blodårene seg. Vil også bli tilsvarende problem om det generelt bare er kaldt, siden blodårene trekker seg sammen når det er kaldt, og løses også av håndgning.

å se på. Gå deretter tilbake hit for å teste mer kompliserte målesituasjoner.

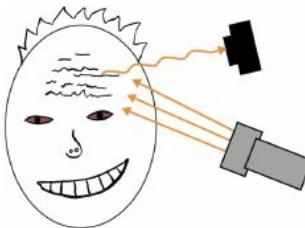
2. Sett ei lommelykt direkte mot fingeren og mål det du får tilbake (reflektans).

12.1.2 Del 2: Robusthetstester

Velg ut fargekanal (og målesituasjon, hvis relevant). Begrunn gjerne valget etter hvor lett det er å få ut puls, hvor praktisk det er å gjøre den målingen, hvor interessant dere finner den måten å måle på, eller andre hensyn.

Vi forutsetter at man gjør minst to forskjellige robusthetstester, hvor dere skal teste pulsmålemetoden deres i utrygge situasjoner. Hva disse testene blir, velger dere selv. Dersom det er vanskelig å holde et jevnt nivå mellom målinger i samme test, f.eks. ved høy puls, er det greit å lage bare én måling.

Tips til ting man kan teste:



Mål panna på et godt belyst

- ansikt. Dette er mulig, men er gjerne litt mer utfordrende.
- Varier pulsen ved å springe opp og ned trappa e.l., slik at dere får verifisert at riktig hvilepuls ikke bare var ren flaks.
- Varier belysningen, test ulike lyskilder. Med/uten taklys? Taklys på ulike dimmeinnstillinger? Med/uten skjermet målekammer (f. eks. en genser)?
- Iskalde vs. varme fingre
- Varier avstand mellom objekt og kamera, lyskilde og objekt.
- Sjekke om systemet kan måle puls fra flere mennesker samtidig.
- Sammenligne ulike prosesseringsmetoder.
- Sammenligne SNR før og etter ulike preprosesseringsssteg.
- Teste andre deler av kroppen.

Sammenlign pulsverdier, SNR-verdier, feilestimat og eksterne kilder som smartfonapp. Prøv gjerne å teste andre ting enn eksemplene som er listet opp over. Utforsk måleoppsettet. Et negativt resultat er også et resultat.

12.2 Krav for godkjent lab

For å få godkjent laben må dere, i tillegg til å få godkjent på forberedelsesoppgavene, gjøre følgende når dere demonstrerer laben.

1. Vis at dere har gjort basismålingene og regnet på SNR, gjennomsnitt og standardavvik

2. Plott rådataene og spektrumet til minst én måling, alle kanaler
3. Vis at dere har gjennomført robusthetstestene