

UWB BASED UAS POSITIONING

ESPEN HAUGEN & SIMON HØYDAL SÆTRE & HÅKON SKAU HØKSNES

1 SAMMENDRAG

2 FORORD

Dette er prosjektrapporten for vår bacheloroppgave i Droneteknologi hos Universitetet i Tromsø. Vi ønsket en bacheloroppgave som ga oss muligheten til å bygge en drone som kunne posisjonere seg ved hjelp av andre posisjonssystemer enn GNSS. Vi ønsker å rette en stor takk til Kongsberg Maritime, avd. VeRo som har gitt oss denne spennende oppgaven, og for å støtte med UWB-posisjonerings utstyret som skal brukes i prosjektet. Vi ønsker også å takke Bernt Inge Hansen og Kåre Edvardsen for god veiledning igjennom oppgaven.

INNHOLD

1	Sammendrag	2
2	Forord	3
3	Forkortelser og fagbegreper	6
4	Innledning	7
4.1	Bakgrunn for oppgaven	7
4.2	Problemstilling	7
4.3	Målformulering	8
5	Valg av løsning	9
5.1	Systemkrav	9
5.2	Forslag til andre posisjonsmetoder enn UWB	9
5.3	Kamera triangulering	9
5.4	Valg av UWB system	10
5.5	Valg av platform	12
5.6	Konsept	12
6	Teori	15
6.1	Hvordan finne posisjon	15
6.2	UWB	16
6.3	GNSS	17
6.4	Arduino	19
6.5	Elektronisk kompass/magnetometer	20
6.6	Akselerometer	21
6.7	gyroskop	22
6.8	Barometer	23
6.9	PID-regulator og posisjon hold	23
6.10	Extended kalman filter (EKF)	24
6.11	Multirotor	25
7	Sikkerhet og regelverk	27
7.1	Sikkerhetstiltak for operasjonene	27
7.2	Tiltak for drone på avveie	27
7.3	Regelverk for droneflyvning	28
7.4	Regelverk for sendinger	28
8	Gjennomføring	28
8.1	Oppsett av UWB-tags og ankere	28
8.2	Avlesning av posisjon med Arduino	29
8.3	Plotting med python	29
8.4	Presisjon og rekkevidde av UWB system	30
8.5	Analyse og konklusjon	33
8.6	Avlesning av posisjon og oppsett av drone	33
8.7	Test av posisjonshold i gang	34
8.8	Test av presisjon i krafthallen	34
8.9	Test av posisjonhold I Hamnahallen	34
8.10	Test av automatisk landing og preprogrammert rute	34

Innhold

FIGURER

Figur 1	Hull Skater	7
Figur 2	Sammenlikning av posisjoneringssystemer.	9
Figur 3	Droner for kameratriangulering.	10
Figur 4	Valg av system.	10
Figur 5	DWM100.	10
Figur 6	DWM1001.	11

Figur 7	Pozyx creator kit lite.	11
Figur 8	Valg av UWB system.	12
Figur 9	Drone ramme med 3D-printet beskyttelse.	12
Figur 10	Posisjoneringskonsept.	13
Figur 11	Stresstest i Fusion 360	13
Figur 12	Oppkobling av "Oversetter fra UWB-tag til flightkontroller.	14
Figur 13	Posisjonering ved ulike antall ankere.	15
Figur 14	Two way ranging.	16
Figur 15	Code-positioning. (Tallysman udatert)	18
Figur 16	GNSS-feilkilder.(Romsenter udatert)	18
Figur 17	Arduino nano.(Arduino udatert)	19
Figur 18	Hall-sensor. (Bosch udatert)	20
Figur 19	Posisjonen til den magnetiske Nordpolen de siste 100 årene. (Britannica udatert)	21
Figur 20	Aeromagnetisk anomalikart fra undersøkelsen utført av NGU i 2007. (Gellein 2007)	21
Figur 21	MEMS-akselerometer. (Developments udatert)	22
Figur 22	Piezoelektrisk akselerometer.	22
Figur 23	MEMS-gyroskop. (Bosch udatert)	23
Figur 24	BME280. (Bosch udatert)	23
Figur 25	PID-reguleringssløyfe.	23
Figur 26	EKF algoritme.	24
Figur 27	DJI Air 2	25
Figur 28	Pixhawk Cube	25
Figur 29	Børsteløs DC-motor	26
Figur 30	Aksler på et luftfartøy	26
Figur 31	Rotasjonsretning på multirotor	27
Figur 32	Krefter i fartretsningen på en multirotor	27
Figur 33	Ankerplassering	29
Figur 34	Pozyx creator controller	29
Figur 35	Oppkobling av pozyx tag og arduinor	29
Figur 36	Plot fra innendørstest. x-y planet i figuren til høyre, x-y plan- net i figuren til venstre.	30
Figur 37	RSSI måling over ulike distanser.	30
Figur 38	RSSI måling over ulike distanser.	31
Figur 39	Ulike anker oppsett.	31
Figur 40	Målinger med $2 \cdot 2\text{m}$ ankere.	31
Figur 41	Målinger med $2 \cdot 2\text{m}$ ankere uten de 50 første målepunkter.	32
Figur 42	Resultater fra måling 1.	32
Figur 43	Resultater fra måling 3.	32
Figur 44	Resultater fra måling 4.	32
Figur 45	Målinger med $5 \cdot 5\text{m}$ ankere.	32
Figur 46	Målinger med $5 \cdot 5\text{m}$ ankere uten de 50 første målepunkter.	32
Figur 47	Resultater fra måling 6.	32
Figur 48	Resultater fra måling 7.	33
Figur 49	Resultater fra måling 8.	33
Figur 50	MavLink melding for anker distanse.	33
Figur 51	Kommunikasjons vei for tag, arduino og flightcontroller.	34
Figurer		

3 FORKORTELSER OG FAGBEGREPER

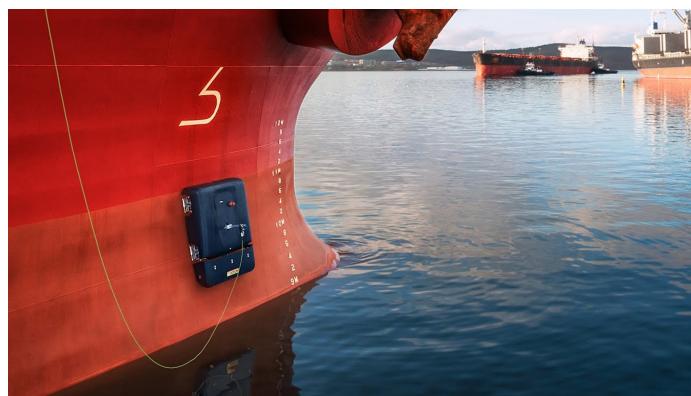
4 INNLEDNING

4.1 Bakgrunn for oppgaven

Sommeren 2021 hadde Håkon sommerjobb hos Kongsberg Maritime, avd. VeRo (Vessel Robotics). I samråd med firmaet kom vi frem til en oppgave som kan passe oss. Kongsberg Maritime ser på bruken av Ultra Wide Band (UWB) til posisjonering av en ROV på siden av skip. Og i den forbindelse ønsket de at vi skal se på bruk av UWB posisjonering på droner.

4.1.1 Kongsberg Maritime

Kongsberg Maritime er en teknologibedrift som hører til Kongsberg Gruppen. Kongsberg gruppen stammer opprinnelig fra Kongsberg våpen fabrikk, men i dag er det en høyteknologisk bedrift som driver med blant annet våpen, romfart og maritim teknologi. Kongsberg Maritime leverer blant annet systemer for posisjonering, overvåking, navigasjon og automasjon til skip og offshoreinstallasjoner.



Figur 1: Hull Skater.

Maritime har kontorer over hele verden, der hovedkontoret ligger i Norge. HSS, Hull Skating Solutions, er et samarbeidsprosjekt mellom Jotun og Kongsberg Maritime. Prosjektet utvikler en robot som kan brukes til å vaske skroget på store skip. Dette blir gjort for å hindre uønsket spredning av organismer rundt om i verden, samt minke drivstoffforbruket på båtene. Slik det gjøres i dag må båtene tas opp på land for å bli vasket, eller dykkere må dykke langs skroget for å rengjøre. (Keim 2015)

4.2 Problemstilling

VeRo ønsker å vite hvor på skipskroget HullSkater befinner seg. Skipskroget er et vertikalt plan som står normalt på vannoverflaten. GNSS vil her ikke gi tilstrekkelig presisjon for posisjonen. Noen grunner til dette er:

- GNSS gir ikke god høydepresisjon, og vil derfor bli for unøyaktig til dette bruksområdet.
- GNSS måler skipets og HullSkater sin posisjon i forhold til jorden.
- Skipet vil ikke alltid ligge like dypt i vannet.
- GNSS tar ikke høyde for endring i vannhøyden.

Med UWB kan man lage et lokalt posisjoneringssystem som vil fikse dette problemet og gi posisjonen til roboten på skroget uavhengig av posisjonen til skipet. Mangelen på lokal posisjonering er et problem som oppstår flere plasser hvor det

er viktigere å vite posisjonen i forhold til en gjenstand enn plassering på jordens overflate. Eksempler på dette i forbindelse med droneaktivitet er anleggsplasser, maritime operasjoner fra plattform eller skip og inspeksjon av konstruksjoner og bygninger. Tidligere er det kun brukt GNSS under disse operasjonstypene. Noen problemer med GNSS i disse situasjonene er:

- I forbindelse med operasjoner i urbane miljø er signalrefleksjon et stort problem. Dette gir dårligere nøyaktighet og mindre troverdig posisjon.
- GNSS kan lett forstyrres av andre signaler da signalet som treffer jorda er svakt i forhold til andre trådløse former for kommunikasjon.
- GNSS har en begrenset evne til å bestemme høyde. På grunn av plasseringen og avstanden til satellittene.

4.3 Målformulering

4.3.1 Formål

Gruppen skal se på bruken av UWB posisjonering av dronesystemer og se på fordele og ulemper med dette kontra GNSS.

4.3.2 Resultatmål

Dette prosjektet ønsker å se på bruken av et UWB system på en drone samt å sammenlikne presisjonen med GNSS. For å oppnå dette er det i samarbeid med VeRo satt flere delmål det skal jobbes mot. Disse er laget med økende kompleksitet, da det er vanskelig å si på forhånd hvor mye hvor mye tid det vil ta å oppnå de ulike målene. Målene med UWB er:

- Posisjonen til dronen blir loggført under en manuell innendørs flygning.
- Dronen holder automatisk posisjonen sin i luften.
- Dronen kan lande automatisk.
- Dronen kan fly et preprogrammert oppdrag med takeoff, planlagt rute og landing.

Det er veldig mange problemer med GNSS-baserte droneoperasjoner som muligens kan løses med UWB-posisjonering. Det er derfor ønskelig å utvikle et system som bruker UWB-posisjonering og undersøke hvilke muligheter med også problemer som dette gir. Et siste mål er å sammenlikne presisjonen oppimot GNSS. Dette målet regnes som uavhengig i forhold til de andre målene.

4.3.3 Prosessmål

Gi studentene erfaring og kompetanse i å samarbeide i team, samt å planlegge og gjennomføre et større prosjekt.

5 VALG AV LØSNING

Dette kapittelet handler om valg av løsninger for oppgaven. Valg av posisjonssystem for prosjektet er i fokus.

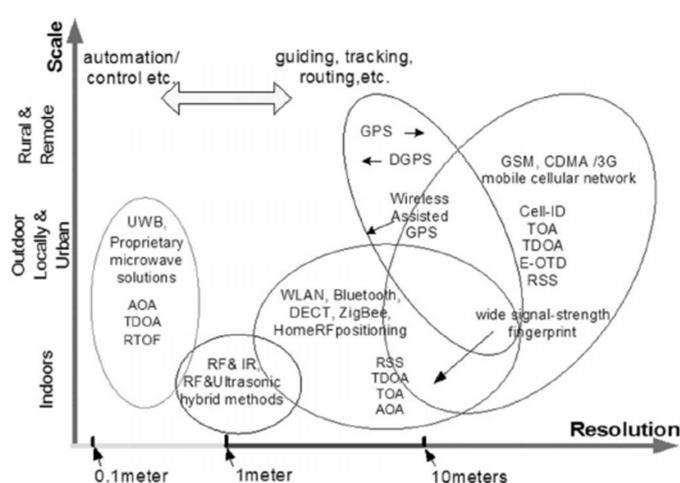
5.1 Systemkrav

Systemkrav for valg av posisjonssystem

- Systemet bør være motstandsdyktig mot interferens
- Systemet skal ikke干涉ere andre sendere/mottakere på dronen
- Posisjon i x/y planet skal være mer nøyaktig enn GNSS (uten RTK), sikter på system med nøyaktighet under 1m.
- Systemet skal være innenfor regelverk for sendestyrke og frekvenser
- Systemet skal kunne beregne lokal posisjon
- Rekkevidden til systemet bør være over 30meter

5.2 Forslag til andre posisjonsmetoder enn UWB

Bachelor oppgaven er i samarbeid med bedriften Kongsberg, bedriften har ønsket at prosjektet skal teste UWB som posisjonssystem. Likevel ble andre teknologier vurdert av gruppen for alternative løsninger. System som gir ut posisjon i koordinater ble vurdert og system som finner avstand til objekt slik som lidar og stereokamera ble ikke vurdert.



Figur 2: Sammenlikning av posisjoneringssystemer.

5.3 Kamera triangulering

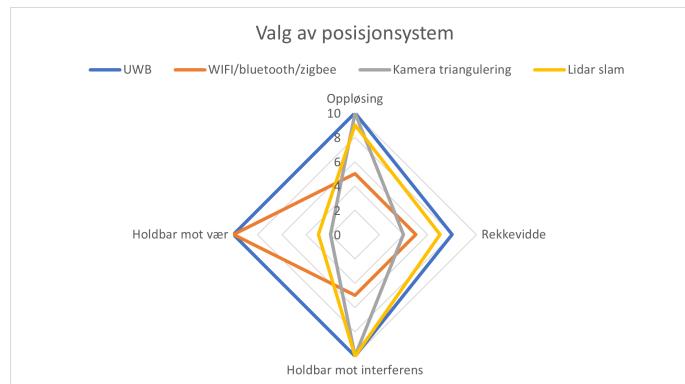
Kamera triangulering bruker flere kamera for å triangulere posisjonen til ett eller flere objekt. På objektet som skal spores plasseres flere markører. Slik kan man finne posisjonen og rotasjonen. Denne teknologien er mye brukt i filmproduksjoner og spillutvikling. Kamera triangulering kan være veldig nøyaktig. Systemet Optitrack beskriver opplösningsavviket på millimeternivå. Disse systemene kommer ofte med en høy pris, som er utenfor prosjektets budsjett. Rekkevidden til systemene er også litt kort til prosjektets anvendelsesområde, og utendørsbruk kan være krevende.



Figur 3: Droner for kameratriangulering.

5.3.1 Bluetooth, Zigbee og Wifi

Denne teknologien baserer seg på signalstyrke for å beregne posisjonen. Oppløsningen på posisjonen til vil ligge i meterområdet bluetooth (ca 1-5m) og Wifi(ca5-20), det vil være for unøyaktig for prosjektets anvendelsesområde siden det er mer unøyaktig enn GNSS kan være. Denne teknologien bruker frekvenser innenfor 2.4ghz og 5.8ghz De fleste droner bruker allerede disse frekvensene for kommunikasjon, her vil sendingen kunne forstyrre hverandre.



Figur 4: Valg av system.

5.3.2 UWB

I sammenligningen med de andre posisjonssystemene slår gruppen fast at UWB vil kunne fungere godt for anvendelsesområdet for oppgaven.

5.4 Valg av UWB system

For valg av UWB-system var det 3 forskjellige systemer som ble vurdert.

5.4.1 DWM100



Figur 5: DWM100.

Denne UWB modulen består av UWB sensoren DW1000 som er brukt i flertallet av UWB system. For prosjektets bruk sammen med en drone er det nødvendig å lodde modulen til en mikrokontroller. Det trengs også en løsning for å gi strøm til sensorene. For å bruke DWM1000 til å finne posisjon er det nødvendig med programvare som triangulerer signalene fra modulene, det finnes ingen tilgjengelig programvare for dette slik at mye arbeid i bacheloren ville gått til programmering.

5.4.2 DWM1001



Figur 6: DWM1001.

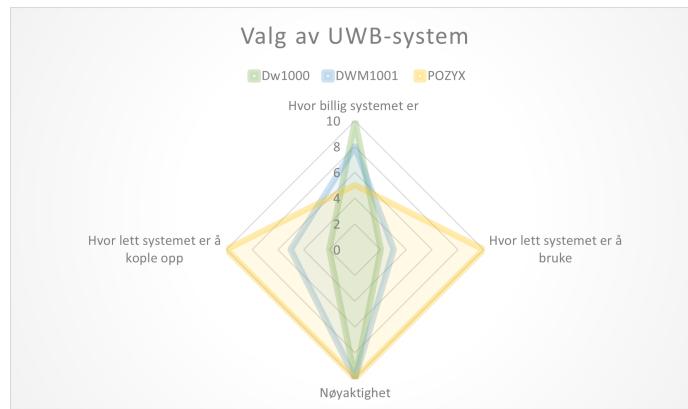
DWM1001 består av sensor/mottaker DW1000, i denne modulen er mikrokontroller inkludert, slik det er lite arbeid med oppkobling. Slik som DWM1000 vil det være mye arbeid med å programmere systemet til posisjonsbruk.

5.4.3 POZYX



Figur 7: Pozyx creator kit lite.

Pozyx systemet består av anker og tags som inneholder mikrobrikken Dw1000. Creator kit Lite kommer med 4 anker og 2 tags. Dette systemet kommer med programvare og har god dokumentasjon for bruk med egen programvare. For å velge hvilket system gruppen skulle bestille ble ulike fakturer vurdert. Figur 8 demonstrerer de ulike fakturene. Valget falt da på POZYX systemet da det passet godt til vår oppgave, det skal være lett å sette opp og har mye dokumentasjon. Det at POZYX systemet har ferdig programvarer og bibliotek passer godt til oppgaven, da oppgaven til prosjektet er ikke å programmere ett UWB-system fra grunnen, men det er å forske på hvordan UWB vil fungere i bruk med droner.



Figur 8: Valg av UWB system.

5.5 Valg av platform

For å knytte prosjektet opp mot droneteknologiutdanningen har gruppen bestemt seg for å bruke quadrotor (drone) for prosjektet. Om prosjektet fungerer godt med enn drone, vil det sannsynligvis fungere godt for roveren til Kongsberg. Erfaringer fra utendørsflyvninger i Tromsø der prosjektet utføres er at været er enn stor hindring, det ble derfor valgt å bruke en drone som er mulig å bruke innendørs. Valget ble å bruke en «racing» drone med propellerbeskyttere, som vist i figur 9. Denne dronen har løftekraften til å løfte sensorene som er nødvendig samtidig som den har liten størrelse.



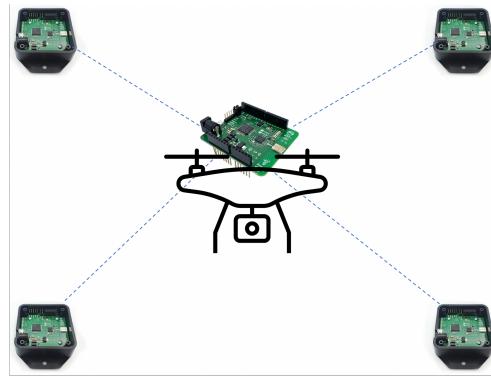
Figur 9: Drone ramme med 3D-printet beskyttelse.

5.6 Konsept

Systemet består av et dronesystem med påmontert UWB-tag som kommuniserer med 4 UWB-anker. UWB-tagen regner ut avstanden til ankerene og sender dataen til flightkontrolleren. Målet er å bruke UWB-systemet til å kunne få dronen til å flyve ved hjelp av UWB-posisjon. Figur 10 viser dette.

5.6.1 Design av drone

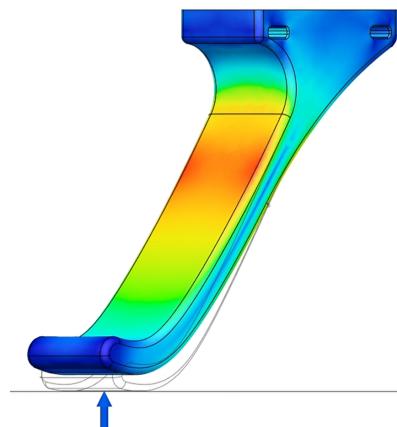
Det ble valgt å bruke en “racing” drone med 5 tommer propeller til prosjektet. For å få den sikker ble det vurdert at propellerbeskyttere måtte bli brukt.



Figur 10: Posisjoneringskonsept.

5.6.2 3D-printede deler

Det ble designet propellerbeskytter for å dekke både oversiden og undersiden av propellerbladet. Disse ble deretter festet på drone rammen. For å beskytte UWB-tagen ble det designet ett veltebur på oversiden av dronen. Batteriet ble plassert på undersiden av dronen på grunn av UWB-tagen sitter på toppen, for å beskytte batteriet mot harde landinger ble det designet landingsben for å ta imot kreftene. De 3d-printede delene ble printet i PLA plast, PLA vil kunne knekke i harde landinger eller krasj, men delene er designet for å kunne bli enkelt byttet ut. Delene ble designet i Fusion 360. For å finne ut hvor delen vil kunne få svikt ble stressanalyseverktøyet til Fusion 360 brukt, som vist i Figur 11.



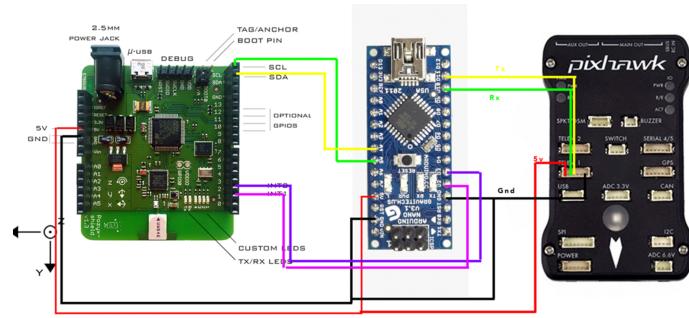
Figur 11: Stress test i Fusion 360

5.6.3 Oppkobling av dronen

Dronen består av flightkontroller, motorer, motorkontroller...

5.6.4 UWB-tag til flightcontroller

For å få data fra UWB-tagen til rett protokoll til ardupilot var det nødvendig med en mikrokontroller for å "oversette". I figur 12 vises oppkoblingen fra UWB-tag t.v til mikrokontroller til flightkontroller. På mikrokontrolleren kjøres ett datascript som setter opp UWB-tag for å kunne kommunisere med ankerene ved å sette rette adresser til anker, samt hvor ankerene er plassert. Deretter regnes det ut avstand til hvert anker og koordinat til tag-en. For at flightkontrolleren skal forstå dette må det sendes på riktig protokoll som arduinokoden ordner.



Figur 12: Oppkobling av "Oversetter fra UWB-tag til flightkontroller.

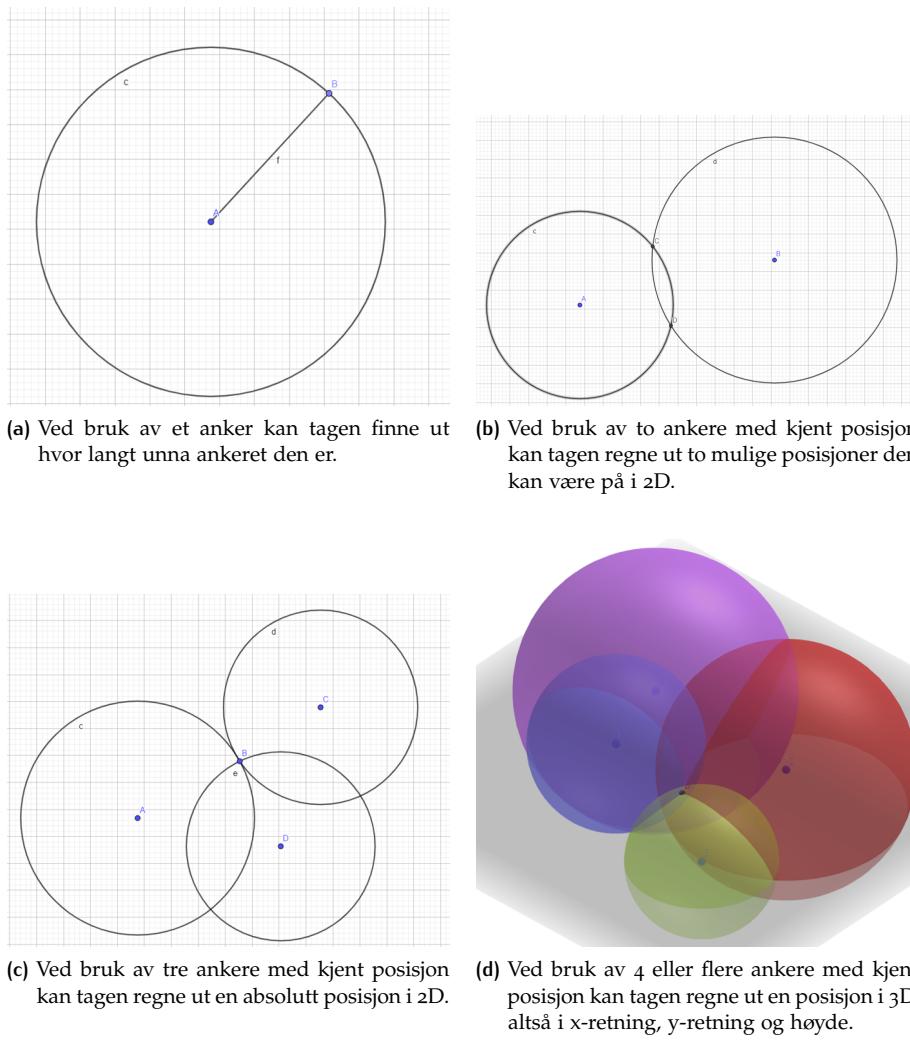
5.6.5 Oppsett av anker

For at UWB-tag-en skal kunne regne ut posisjonener den avhengig at ankerene er plassert riktig. Det er viktig at ankerene ikke beveger seg, har riktig orientasjon og riktig placering. Ankerene trenger også strøm for å kjøre siden de svarer tag-en sine meldinger. Løsningen her er å bruke borrelås for faste installasjoner, og tripodder for oppsett som er midlertidig. For å gi ankerene strøm blir små batteripakker brukt, dette bør fungere godt siden ankerene trekke svært lite strøm.

6 TEORI

6.1 Hvordan finne posisjon

All posisjonering handler om å finne posisjonen til et objekt i forhold til et annet objekt. Den vanligste teknologien for lokalisering på jorden er GNSS (Global Navigation Satellite System), ofte bare omtalt som GPS (Global Positioning System). Ved bruk av GNSS kan man finne en posisjon i forhold til jorden, altså hvor på jorden man befinner seg. Når UWB brukes til posisjonering opprettes det et LPS (lokalt posisjonerings system). Man finner da posisjonen i forhold til dette posisjonerings-systemet, som er nyttig dersom man ønsker å finne posisjonen innendørs der det ikke er GNSS, eller ved tilfeller hvor det er viktigere å vite posisjonen i forhold til en konstruksjon enn jorden. Et UWB LPS er satt opp ved bruk av anker som illustrert i figur 13. Et anker velges som origo, og posisjonen til de andre ankerene måles opp i forhold til origo. En tag som beveger seg rundt i dette rommet vil kunne bruke ankerene til å regne ut sin egen posisjon. Måten dette gjøres på er at tagen først bruker TWR til å finne avstanden til hvert anker. Både disse avstandene og de kjente posisjonene til ankrene blir deretter brukt til å regne ut tagens posisjon i forhold til origo.

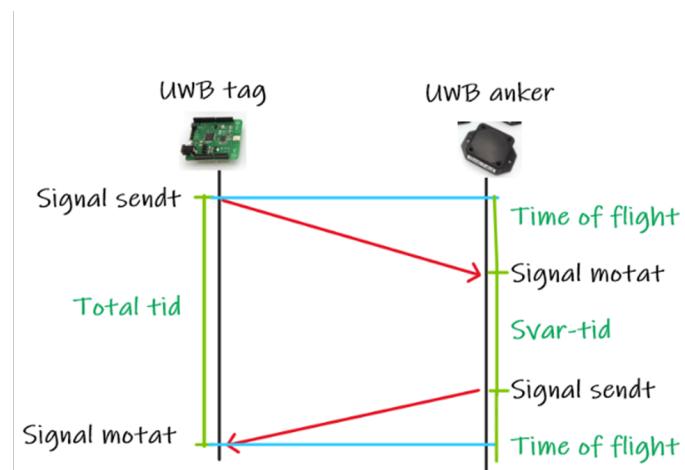


Figur 13: Posisjonering ved ulike antall ankere.

6.2 UWB

Ultrabredbånd, eller Ultrawideband (UWB), beskriver radiokommunikasjon med båndbredder høyere enn 500 MHz. På slutten av 1900 tallet ble UWB smått tatt i bruk, men ble da mest brukt til radar, sensorer og militært bruk. Dette endret seg etter 2002, da US Federal Communication Commission (FCC) vedtok at UWB kunne bli brukt til data kommunikasjon, radar og sikkerhets applikasjoner. UWB ble da tildelt frekvensene fra 3.1 til 10.6 GHz, som er den største båndbredden til noe kommersielt landbasert system. På grunn av at den store båndbredden kunne føre til mye interferens med andre kommunikasjonssystemer, ble det satt strenge regler for effekten til UWB. Disse reglene gjør f.eks. at ved bruk av hele UWB spekteret, 7.5GHz, vil man ikke kunne bruke en effekt høyere enn 0.5 mW. Grunnen til at UWB er blitt populært på mange områder er den store båndbredden. Denne gjør at det er mulig å sende mye data med høy frekvens. Den lave sende effekten gir både fordel og ulemper. På korte distanser kan man sende mye data samtidig som man bruker veldig lite strøm. For å sende data over lengre distanser derimot, trengs mer effekt enn det reglementet for UWB tillater. (Mearian 2019)

6.2.1 Two way ranging



Figur 14: Two way ranging.

For å finne avstanden mellom en tag og et anker brukes two way ranging (TWR) protokollen som vist i figur 14. Det man er ute etter å finne her er tiden signalet bruker fra ankeret til tagen, og bruke dette til å regne ut avstanden. Et signal blir først sendt fra tagen til ankeret. Dette signalet bruker tiden t_{of} (time of flight). Etter en viss tid, t_{svar} , sender ankeret et signal tilbake. Det antas at tiden fra tagen til ankeret er det samme som tiden fra ankeret til tagen, altså t_{of} . Når tagen mottar signalet tilbake fra ankeret regner den ut tiden fra den sendte ut signalet, til den fikk et signal tilbake (t_{total}). Dersom t_{svar} er kjent på forhånd, kan man regne ut t_{of} er dermed avstanden mellom tagen og ankeret:

$$\begin{aligned} t_{total} &= t_{of} + t_{svar} + t_{of} \\ t_{total} &= t_{svar} + 2 \cdot t_{of} \\ t_{of} &= \frac{t_{total} - t_{svar}}{2} \end{aligned}$$

UWB signalet reiser i lysets hastighet, altså: $v_{uwb} \approx 3 \cdot 10^8 \text{ m/s}$
Dermed kan distansen regnes ut ved:

$$d = t_{of} \cdot v_{uwb}$$

Grunnen til at TWR blir brukt er at det ikke krever synkronisering mellom tagen og ankeret. Så lenge $t_{svær}$ er kjent kan distansen regnes ut. Denne protokollen gjør også at det er tagen selv som regner ut distansen, som gjør det til en god protokoll for bruk på en drone som ønsker å vite sin egen posisjon. Det er også mulig å utvide TWR ved at tagen sender ut et ekstra signal etter den har fått signalet tilbake fra ankeret. På denne måten kan både tagen og ankeret regne ut avstanden mellom dem.

6.3 GNSS

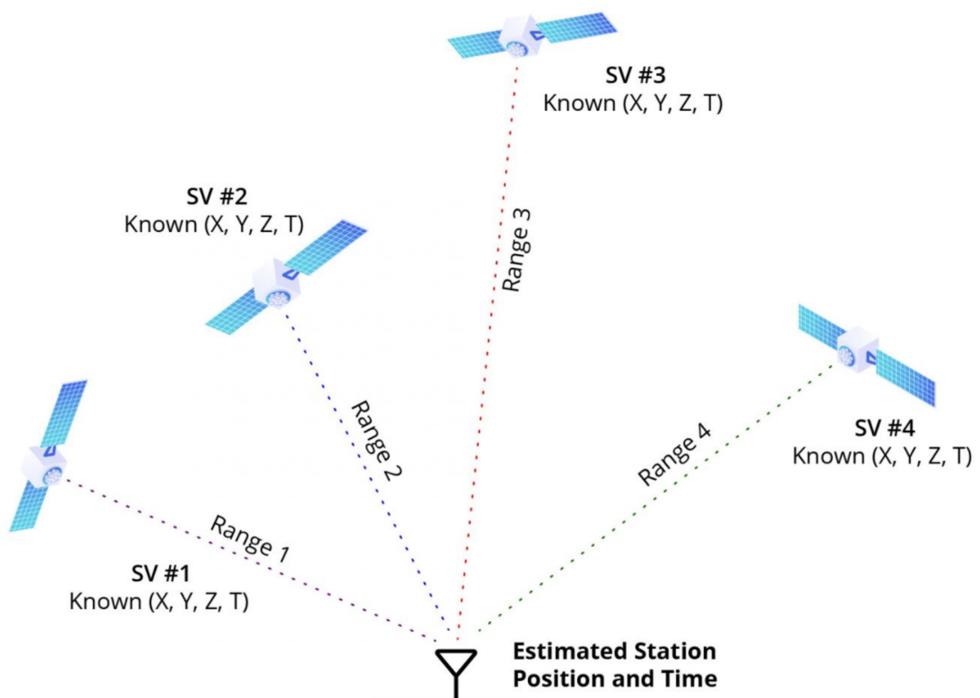
GNSS (Global Navigation Satellite System) er samlebetegnelsen for alle satellitt-baserte posisjoneringssystemer som brukes for finne posisjon på jordens overflate. Amerikanske NAVSTAR, som var det første operative systemet. Ble fullt utbygd på 1990-tallet og var i utgangspunktet tiltenkt militær bruk. Under utviklingen av systemet ble det i tillegg tilgjengelig for det sivile. Etter hvert har andre systemer kommet til, slik som GLONASS (Russisk), Galileo (Europeisk) og BeiDou (Kinesisk). (Forsell og Kjerstad **udatert**)

6.3.1 Funksjon

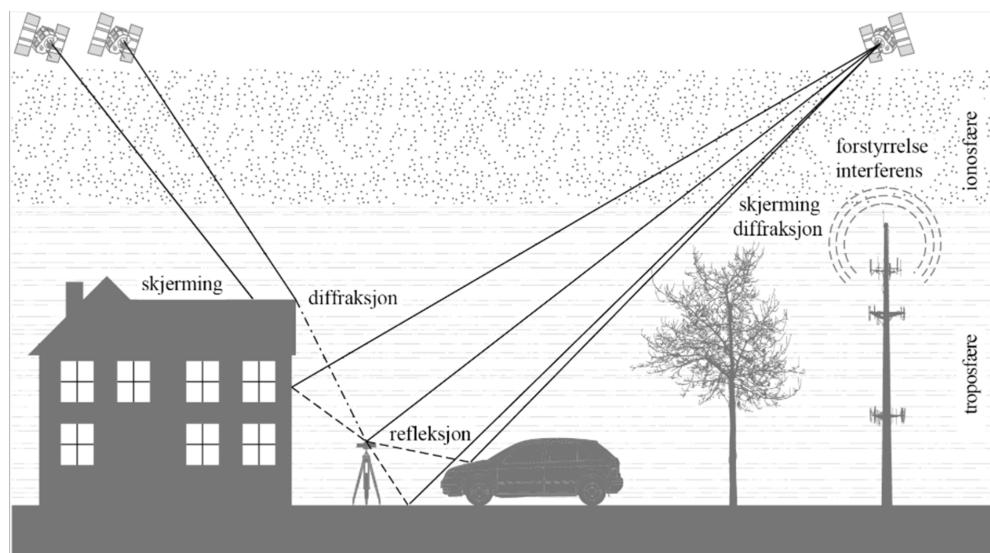
Felles for alle disse systemene er at avstanden fra en GNSS-mottaker til satellittene brukes til å beregne mottakerens posisjon ved hjelp av triangulering. For å triangulere en 2D-posisjon behøves kommunikasjon med tre satellitter, og fire for å kunne definere høyden til mottakeren. Jo flere satellittsignaler som benyttes i trianguleringen, jo bedre posisjonsnøyaktighet er det mulig å oppnå. (Romsenter **udatert**) Systemene fungerer ved at hver satellitt har et atomur som går svært nøyaktig med neglisjerbare avvik. En satellittmottaker har ikke tilsvarende nøyaktig klokke, men mottar jevnlig synkroniseringsdata fra satellittene. I tillegg går satellittene i kjente baner, slik at mottakeren på bakken til enhver tid vet hvor satellittene befinner seg. Ved å sammenligne tidspunktet signalet ble sendt med tidspunktet signalet ble mottatt, kan systemet beregne avstanden til mottakere ved hjelp av «time of flight»-beregnning vist i figur 15. Denne beregningen kan gjøres på flere måter, som igjen vil kunne gi forskjellig nøyaktighet i posisjoneringen. (Forsell og Kjerstad **udatert**) Teoretisk sett kan nøyaktigheten i avstandsbestemmelsen fra en GNSS-mottaker til en GNSS-satellitt være omkring 1 meter for den enkleste formen for avstands beregning (beregnet ut ifra det amerikanske systemet). Videre kan mer avanserte metoder som fasemåling gjøre at den teoretiske nøyaktigheten kommer ned mot millimeternøyaktighet da bølgelengden til bærefrekvensen L1 er på 19 cm. Fasene er periodiske og derfor kan ikke fasemåling brukes alene, da dette vil gi mange mulige avstander til satellittene. En tredje metode er differensiell måling, hvor en basestasjon ved et fastpunkt med kjent posisjon måler avvik mellom målt posisjon og faktisk posisjon. Dette avviket sendes så til en GNSS-sensor i nærheten som benytter avviket til å beregne en svært nøyaktig posisjon. Jo nærmere GNSS-sensoren er fastpunktet, jo mer korrekt er avviksdataen og dermed blir den målte posisjonen mer korrekt. (**sst.**)

6.3.2 Nøyaktighet

I bruk kan man ikke forvente å oppnå den teoretiske nøyaktigheten, da det er svært mange feilkilder som påvirker posisjoneringen. For å estimere hvor nøyaktig systemene faktisk er så benytter man begrepet «User Range Error» (URE) fra satellitten. I 1990 var URE omkring 4.6 meter, og i 2019 var gjennomsnittlig URE nede på cirka 0,5 meter. (**sst.**)



Figur 15: Code-positioning. (Tallysman [udatert](#))



Figur 16: GNSS-feilkilder.(Romsenter [udatert](#))

6.3.3 Feilkilder

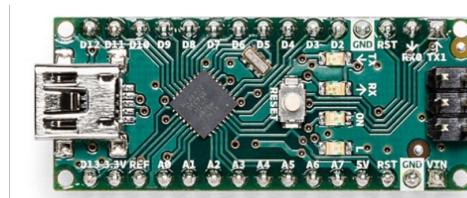
GNSS utsettes for en rekke feilkilder som gjør at det er vanskelig å oppnå den teoretisk mulige nøyaktigheten. Figur 16 illustrerer de forskjellige feilkildene GNSS er utsatt for.

- Det kan oppstå feil i banedataen til satellittene. Dette kan komme av teknisk feil, eller at det er lenge siden banedataen er oppdatert. Dette vil føre til at GNSS-mottakeren beregner sin posisjon basert på feil posisjon på satellittene.
- Signalene fra satellittene går i lysets hastighet, men de forsinkes noe gjennom atmosfæren, hvor blant annet luft bremser hastigheten. Hvis lokale forhold gjør at signalene forsinkes mer enn vanlig kan dette fremtvinge flere meter avvik i posisjon.
- Hvis mottakeren ikke har klar sikt i alle himmelretninger, kan dette dekke for satellitter. Dette kan gjøre at posisjonen beregnes med færre satellitter enn ønskelig. Noe som fører til lavere nøyaktighet.
- GNSS-signalene har svært lav effekt, og vil derfor lett kunne forstyrres av lokale signaler i nærheten av mottakeren som oppleves sterkere og som bruker omtrentlig lik frekvens som GNSS.
- GNSS-signalene har en liten, men ikke neglisjerbar evne til å böye seg rundt hindringer. Dette kan føre til at en mottaker som egentlig er skjult for en satellitt mottar signaler likevel. Disse signalene vil avvike fra den faktiske distansen til satellitten.
- Signalene har lett for å reflektere fra overflater og gi falske forsinkede signaler til mottakeren. Dette er vanlig i urbane områder hvor det er mange blanke og flate overflater som kan reflektere signalene. Det kan også forekomme i naturen i nærheten av fjell og vann som har reflekterende egenskaper. (Romsenter **udatert**)

I tillegg til disse feilkildene vil satellittenes posisjon i forhold til hverandre påvirke nøyaktigheten. Hvis en mottaker benytter satellitter som ligger svært tett eller på rekke, vil nøyaktigheten påvirkes negativt. Det er fordelaktig med satellitter med god spredning slik at vinklene mellom dem blir størst mulig. Dette gjør at krysningspunktene mellom radiene til hver satellitt blir minst mulig. (**sst.**)

6.4 Arduino

Arduino er en opensource plattform som inneholder både Arduino brett og Arduino kodespråk (basert på Wiring). Arduinoen brukt i dette prosjektet er en Arduino Nano (figur 17), som er basert på mikrokontrolleren ATmega328. Koden skrives i Arduino sin egen IDE, som også brukes til å laste opp kode og monitorere data fra Arduinoen. (Arduino 2016) I dette prosjektet blir Arduinoen brukt som en adapter mellom Pozyx tagen og Pixhawk flightcontrolleren. Den leser av avstandene til ankrene fra tagen, og sender denne informasjonen til pixhawk'en via MavLink.



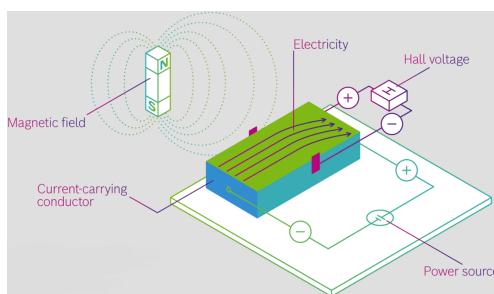
Figur 17: Arduino nano.(Arduino **udatert**)

6.5 Elektronisk kompass/magnetometer

Et magnetometer er et instrument som benyttes for å måle styrken og retningen på det magnetiske feltet som virker på instrumentet. Dette kan gjøres på flere måter. De to vanligste typene benytter Hall-effekt eller magnetoresistans for å måle magnetfeltet. (Skaar [udatert](#))

6.5.1 Hall-effekt

Hall-effekten går ut på at ladde partikler i en leder vil trekke mot en av sidene i lederen dersom de blir utsatt for et magnetfelt. På grunn av et overskudd av ladde partikler på den ene siden oppstår det en spenningsforskjell på tvers av lederen som kan brukes til å måle magnetfelt. Hall-effekten er en utvidelse av Lorentz kraften, som beskriver kraften som påføres en ladd partikkelen som går gjennom et magnetfelt. Hvis magnetfeltet er orientert vinkelrett på retningen av partikkelenes bevegelse, vil partikkelen oppleve en kraft som er vinkelrett på både bevegelsesretningen og orienteringen av det magnetiske feltet. (Keim [2015](#))



Figur 18: Hall-sensor. (Bosch [udatert](#))

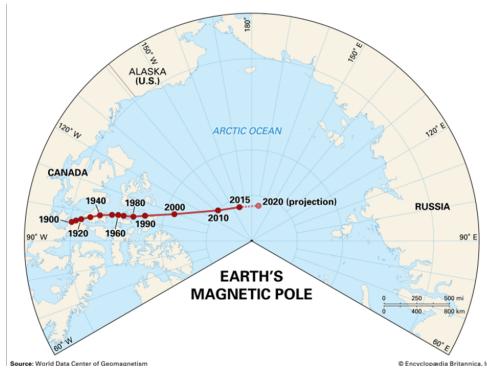
En Hall-sensor er i sin enkleste form et voltmeter som måler spenningen på tvers av en leder. Et magnetometer basert på Hall-effekten benytter typisk to eller flere Hall-sensorer for å kunne detektere magnetfelt i et plan eller tredimensjonalt. Dette gjøres ved at sensorene orienteres slik at hver sensor måler i hver sin akse, x-y-z.

6.5.2 Magnetoresistans

Magnetoresistans (MR) er en effekt som oppstår når enkelte ledere og halvledere utsettes for et magnetisk felt. Alle metaller har en form for MR, men styrken varierer fra metall til metall. Effekten går ut på at det oppstår en endring i et metalls elektriske-motstand når et magnetisk felt påvirker metallet. Et MR-kompass vil i likhet med Hall-kompass benytte flere sensorer for å nøyaktig måle retningen til jordens magnetiske felt. MR skiller seg fra Hall-effekt ved at den måler parallele magnetfelt i motsetning til vinkelrette magnetfelt. Dette gjør at en MR-sensor vil oppnå et større deteksjonsfelt. I tillegg har en MR-sensor høyere sensitivitet og lavere strømforbruk enn en Hall-sensor. (ROHM [udatert](#))

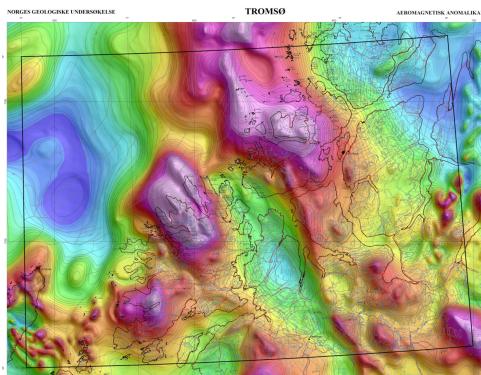
6.5.3 Magnetfelt

Det er viktig å forstå forskjellen mellom magnetisk nord og faktisk nord ved orientering basert på jordens magnetiske felt. Jordens magnetiske felt varierer fra plass til plass og derfor er det avvik mellom faktisk nord og magnetisk nord. Den magnetiske Nordpolen er ikke et fast punkt som den faktiske Nordpolen. Den magnetiske Nordpolen flytter på seg over tid, og avhenger blant annet av hvordan jordens masse forflytter seg. Dette vil si at magma under de tektoniske platene og vann/is vil påvirke hvordan den magnetiske polen forflytter seg. Figur 19 viser hvordan den magnetiske Nordpolen har forflyttet seg det siste århundret. I tillegg til avvik mellom polene og magnetfeltet er det også lokale variasjoner som gjør at kompass



Figur 19: Posisjonen til den magnetiske Nordpolen de siste 100 årene. (Britannica [udatert](#))

kan vise feil. Dette kan komme av lokale ansamlinger av metaller i jordskorpen eller bygninger i urbane miljøer. I 2007 utførte NGU en aeromagnetisk undersøkelse over Tromsø for å kartlegge variasjoner i de lokale magnetfeltet. Resultatet av denne undersøkelsen vises i figur 19. Undersøkelsen viser at blant annet områder på Kvaløya og Senja har økt magnetfelt i forhold til andre plasser. Derfor er det tenkelig at kompassmisvisningen vil være større i disse områdene. (Kartverket [udatert](#))



Figur 20: Aeromagnetisk anomalikart fra undersøkelsen utført av NGU i 2007. (Gellein [2007](#))

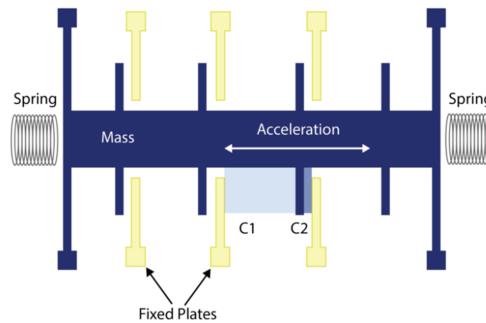
6.6 Akselerometer

Et akselerometer er en sensor som måler akselerasjon i en, to eller tre retninger. Akselerometeret benyttes ofte sammen med et gyroskop for orientering og styring av fly, droner, raketter og ubåter. I tillegg brukes de til skjermorientering i smarttelefoner, vibrasjonsmåling i bygninger og kollisjonssensorer i biler. (Balchen, Andersen og Foss 2000) Prinsippet for alle akselerometer er at det er en fast del og en løs del som typisk henger i en fjær. Dette fører til at den løse delen vil ha en treghet i forhold til den faste delen som vil bevege seg med det sensoren er festet til. ([sst.](#))

6.6.1 MEMS-akselerometer

Mikroelektromekaniske systemer er en samlebetegnelse på små systemer som utnytter både elektroniske og mekaniske prinsipper for å utføre en oppgave. Det er akselerometer bygget på denne teknologien som er mest vanlig i forbrukerelektronikk, blant annet fordi de kan lages så små. MEMS-akselerometer består av en masse som beveger seg langs en akse. Denne massen har små pinner som stikker ut til siden. Disse går imellom tilsvarende pinner som er statiske. Når det oppstår en akselerasjon langs aksen som sensoren måler vil avstanden mellom disse pinnene forandre seg. Når dette skjer vil det oppstå en endring i kapasitansen mellom de

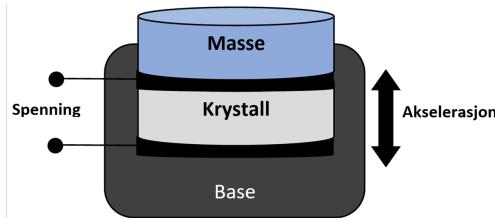
faste pinnene og de som forflytter seg. Det er denne endringen i kapasitans som benyttes for å beregne akselerasjonen.



Figur 21: MEMS-akselerometer. (Developments [udatert](#))

6.6.2 Piezoelektrisk akselerometer

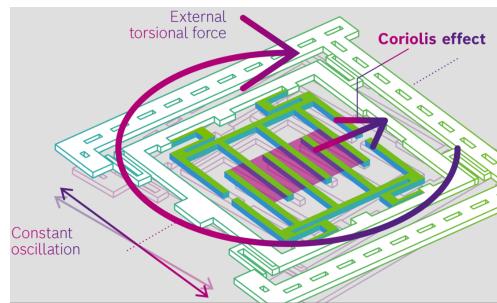
Den piezoelektriske effekten kan også brukes til å beregne akselerasjon. Effekten går ut på at enkelte krystaller skaper en spenning ved deformering. (Grøn og Sandstad 2021) Ved å koble til elektroder på hver side av et krystall kan kraften av en masse som presser mot krystallen beregnes ut ifra spenningen som oppstår over krystallen. Ut ifra kraften og massens vekt kan akselerasjonen beregnes. Prinsippet for dette akselerometret er beskrevet i figur 22. Basen vil være koblet statisk uelastisk til en gjenstand. Og når denne gjenstanden beveger seg i retningen som pilen viser vil en kunne beregne akselerasjonen ved hjelp av elektrodene på hver side av krystallen.



Figur 22: Piezoelektrisk akselerometer.

6.7 gyroskop

Et gyroskop, eller en gyrosensor måler endring i rotasjon langs pitch, roll og yaw aksene. I likhet med akselerometer og magnetometer er gyroskop som benyttes i elektronikk bygget på MEMS-teknologi. MEMS-gyroskop benytter coriolisakselerasjonen som er en akselerasjon som oppstår på et legeme som beveger seg i en rett linje på et roterende referancesystem. Coriolisakselerasjonen vil da oppstå vinkelrett på bevegelsen. (Grøn 2021) Figur 23 viser et eksempel på et MEMS-gyroskop. Det består av en vibrerende struktur som holder en deteksjonsstruktur (grønn). Når sensoren er stabil, vil deteksjonsstrukturen bevege seg frem og tilbake med den vibrerende strukturen. Ved rotasjon vil deteksjonsstrukturen skyves ut til siden på grunn av coriolisakselerasjonen. For å beregne rotasjonen benyttes endringen i kapasitansen mellom kammene i deteksjonsstrukturen som oppstår ved forskjøvningen. Figur 23 viser en enkel rotasjonssensor som kun vil detektere rotasjon rundt aksen som står vinkelrett på deteksjonsstrukturen. For å detektere rotasjon i tre akser må flere slike rotasjonssensorer med forskjellig orientering kombineres. (Bosch [udatert](#))



Figur 23: MEMS-gyroskop. (Bosch [udatert](#))

6.8 Barometer

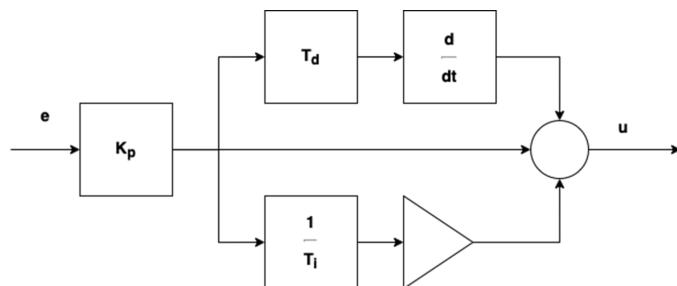
Barometer er ett instrument eller en elektrisk sensor som måler lufttrykk. Prosjektet tar for seg barometer for å beregne høyde, i prosjektet blir elektrisk sensor som sensoren i figur 24 brukt. Luftrykket på jorda minker ved økning i høyde, denne endringen i luftrykket kan bli brukt for å estimere høyde. Luftrykket kan endret av lokale trykkvariasjoner på jorda. Her har vær, klima, tidevann og temperatur noe å si. Ved bruk av barometer for høyde beregninger er det derfor nødvendig å kalibrere barometeret først. Bruk av barometer for høydeangivelse innendørs kan gi problemer med uønskede trykkendringer. Her kan åpning av dører eller bruk av ventilasjonssystemer skape store trykkforskjeller. (Grøn og Sandstad [2021](#))



Figur 24: BME280. (Bosch [udatert](#))

6.9 PID-regulator og posisjon hold

PID-regulatoren er en reguleringssløyfe som består av proporsjonalt, integrasjon og derivasjons-ledd. Reguleringssløyfen arbeider for å minke forskjellen mellom målt verdi og ønsket verdi.



Figur 25: PID-reguleringssløyfe.

- P: Proposjonalt ledd. $K_p e(t)$
- I: Integrat ledd. $K_i \int_0^t e(t) dt$
- D: Derivat ledd. $K_d \frac{de(t)}{dt}$

D-leddet gir pådrag gitt av akselerasjon. D-leddet øker båndbredden til regulatoren slik hurtig regulering er mulig. Ved høye frekvenser kan d-leddet gi uønsket støy. Filtrering er ofte nødvendig for å fjerne støyen. (Balchen, Andersen og Foss 2000)

6.9.1 PID-regulering for droner

For droner brukes PID regulatoren for å regulere rotasjon og posisjon til dronen. For å holde rotasjonen til dronen stabilt blir hver akse sin regulator justert for å minimere avvik. Dette kan være utfordrende å gjøre, analyse av logger og visuell inspeksjon av flyvninger kan hjelpe for å justere regulatorene.

Det er mulig å sette PID-regulator for å la dronen selv stabilisere seg. Her vil PID-regulatoren regulere hvor raskt og aggressiv dronen er for å rette opp rotasjonen. Mange droner har muligheten å sette dronen i Posisjon-hold modus. Denne funksjonen trenger PID-regulering for å bestemme hvor raskt og aggressivt dronen skal jobbe for å holde posisjonen.

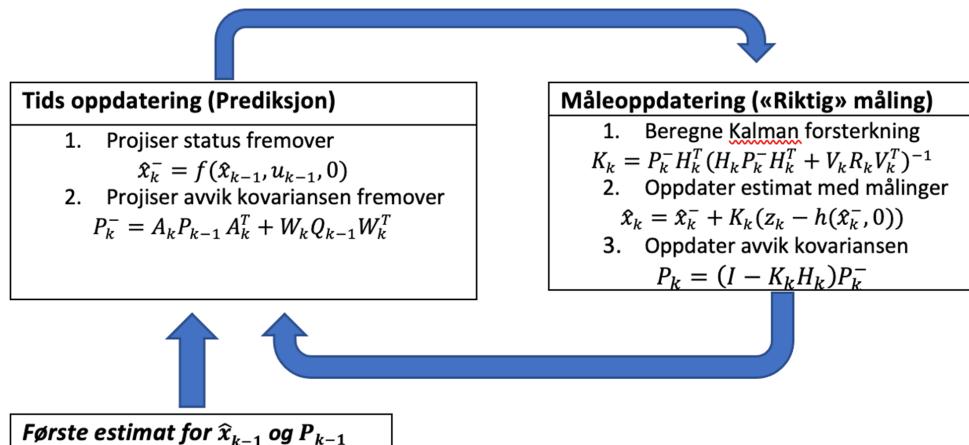
Transferfunksjon: 9.48

$$h_r(s) = K_p \left(1 + \frac{1}{T_i s} + T_d s \right) = K_p \frac{1 + T_i(s) + T_i T_d s^2}{T_i s} \approx K_p \frac{(1 + T_i s)(1 + T_d s)}{T_i s}$$

Filtrering Støy. Statisk filter Dynamisk filter.

6.10 Extended kalman filter (EKF)

Kalman Filter er en algoritme som kan predikterer neste status på ett system ved å bruke observert data fra sensorer og tidligere status til systemet. For system med u-lineære bevegelser slik som droner og roboter har, blir Extended Kalman Filter brukt for å finne posisjon, rotasjon, hastighet og akselerasjon til systemet. Bruk av EKF vil kunne gi bedre estimat på status av et system enn hva en enkel sensor kan. Dette fungerer ved at EKF kan bedre filtrere bort støy og avvik på en sensorer ved sammenligning av andre sensorer. Målinger fra sensorer slik som IMU-er kan være utsatt for mye støy, det kan EKF hjelpe å filtrere bort, eller avvise data ved for stort avvik.



Figur 26: EKF algoritme.

Figur 26 viser hvordan EKF oppdateres. EKF predikterer neste status på systemet fra forrige estimat og deretter korrigere dette estimatet med målinger i nåtid fra

sensorer. For en drone vil sensorer fra IMU og kompass gi rotasjon og akselerasjon. GNSS eller andre posisjonssystem og barometer vil kunne gi posisjonen og høyden til dronen. EKF samler dataen fra sensorene og gir estimat på posisjon, og rotasjon, fart og akselerasjon. Dette estimatet kan videre bli brukt til å regulere dronen sitt pådrag for å styre dronen, dette kommer rapporten mer inn på i PID-regulering avsnittet. (Team **udatert**)

6.11 Multirotor

Multirotor er en samlebetegnelse på alle typer droner med 3 eller flere motorer. Den vanligste typen er quadcopter, altså en drone med fire armer og en motor på hver arm. Figur 27 viser et populært quadcopter fra DJI.



Figur 27: DJI Air 2

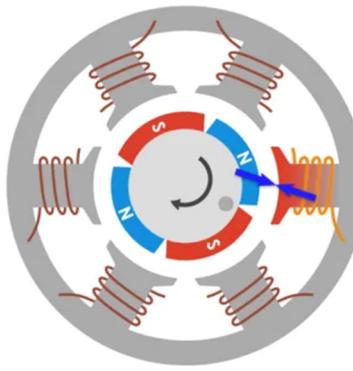
6.11.1 Komponenter

Et quadcopter består i sin enkleste grad av en ramme, fire motorer, fire propeller, en flight controller, en mottaker og fire ESC'er. Flightcontrolleren er hjernen i drona. Denne består av en liten datamaskin, IMU og porter. Oppgaven til flightcontrolleren er å ta imot signaler fra sensorer og radiokontroller, og gi signal videre til ESC for å styre hastigheten til de individuelle motorene.



Figur 28: Pixhawk Cube

Pixhawk, som vist i figur 28, er en populær flightcontroller til utvikling av droner. Denne kan kjøre ardupilot programvare, som er opensource programvare laget for ROV, fly og multirotor. Grunnen til at pixhawk og ardupilot er så mye brukt er at det enkelt kan kobles på sensorer og utstyr, og man kan enkelt gjøre egne endringer i programvaren. En mottaker kobles til flightcontrolleren for at man skulle kunne gjennomføre manuell flygning og kunne bytte mellom ulike moduser under automatisk flyging med en radio. ESC (Electronic Speed Controller) er fartskontrolleren til motoren. Denne får signal fra flightcontrolleren om hvor fort motorene skal spinne, og gir deretter tilsvarende strøm til motorene.

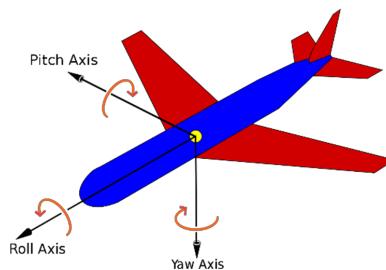


Figur 29: Børsteløs DC-motor

Børsteløse DC motorer blir oftest brukt på quadcopter. Disse består av to deler. En stator og en rotor. Statoren er bygget opp av flere viklinger med kobbertråd. Rotoren er laget av en magnet. Når det blir satt strøm på ulike kobbertråd viklinger vil det skape et magnetfelt som får magneten i rotoren til å spinne i ønsket retning, som vist i Figur 21. På denne måten kan man enkelt øke og minke hastigheten på motoren.

6.11.2 Styring

En multirotor styres som tradisjonelle fly med pitch, yaw og roll. Illustrert i figur 30



Figur 30: Akser på et luftfartøy

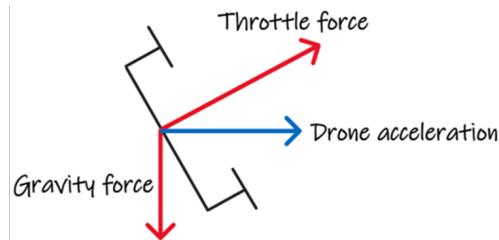
For å rotere i de ulike aksene, må motorene spinne i ulike hastigheter. Figur 31 viser det vanligste oppsettet for motorene sine rotasjoner:

- Pitch: for å få dronen til å bevege seg fremover må motor 3 og 4 spinne raskere enn motor 1 og 2.
- Roll: For å få dronen til å bevege seg til venstre må motor 2 og 3 spinne raskere.
- Yaw: For å dreie dronen mot høyre må motor 2 og 4 spinne fortare.

I tillegg til disse tre brukes throttle. Øking av throttle gjør at alle motorene spinner fortare, og dronen vil da få en akselerasjon normalt på planet til propellene. Altså rett opp dersom dronen står stille på bakken. Dronen styres da ved å bruke pitch, roll og yaw for å oppnå ønsket rotasjon, og throttle for å skape en akselerasjon i ønsket retning. Illustrert i figur 32.



Figur 31: Rotasjonsretning på multirotor



Figur 32: Krefter i fartsretningen på en multirotor

7 SIKKERHET OG REGELVERK

Dette kapittelet tar for seg sikkerhetstiltakene og reglene som gruppen har fulgt. Målene for gruppen var å: Ikke skade personer eller omgivelsene med dronen. Ikke ødelegge kritiske komponenter. Ikke interferere andre sendinger i nærheten, f.eks flytrafikk og radiosamband.

7.1 Sikkerhetstiltak for operasjonene

Det ble utført flere tiltak for å få dronen trygg for innendørs bruk. Gruppen vurderte at flyvninger med udekket propeller kunne få for store konsekvenser. Det ble derfor designet propellerbeskyttere til dronen. Gruppen har bare 2 UWB-tags tilgjengelig, og disse må ikke bli ødelagt. For å unngå å skade disse ble det designet ett ”rullebur” for å forhindre skade under krasj eller harde landinger. De 3d-printet delene ble designet for å deformere seg eller knekke for å minke nedslag til dronen sine kritiske komponenter. Dette ble valgt siden 3d-printet deler er enkelt og billig å bytte ut.

7.2 Tiltak for drone på avveie

For å alltid ha kontroll over dronen ble det brukt enn manuell-pilot under flyvningene. Den manuell-piloten sin oppgave var å ta over kontroll om drone kom på avveie på grunn av programfeil osv. Det ble også brukt basestasjon for å kontrollere og overvåke dronen. Personell på basestasjon overvåkte kritiske verdier som batteribruk og helsen til sensorene. Ved tap av manuell-kontroll link ble dronen programmert til å kutte motorene, dette ble valgt siden dronen kan utføre større skade med å prøve å lande ta seg selv. For å unngå brukerfeil under flyving ble sjekklister implementert.

7.3 Regelverk for droneflyvning

For innendørsflyvning gjelder ikke regelverket for droner. Slik at her kunne en flyve uten operasjonslisens. For utendørsflyvning har en av gruppens deltaker registrert seg i åpen kategori og har kommersiell forsikring.

7.4 Regelverk for sendinger

For sendingen i prosjektet er det ingen krav for tillatelser for bruk, men det er regler for hvilke frekvenser som kan brukes, hvilken sendestyrke som er lovlig og hvordan data blir sendt.

- Regelverket for kommunikasjon til dronen gjelder under forskrift om generelle tilatelser til bruk av frekvener (fribruksforskriften), kap 3.8 Diverse utstyr for kortdistansekommunikasjon. <https://lovdata.no/forskrift/2012-01-19-77/§33.8>
- For regelverket for UWB posisjon blir regelverket for generelle tilatelser til bruk av frekvener (fribruksforskriften), kap 35a følgt. <https://lovdata.no/forskrift/2012-01-19-77/§35a>

8 GJENNOMFØRING

8.1 Oppsett av UWB-tags og ankere

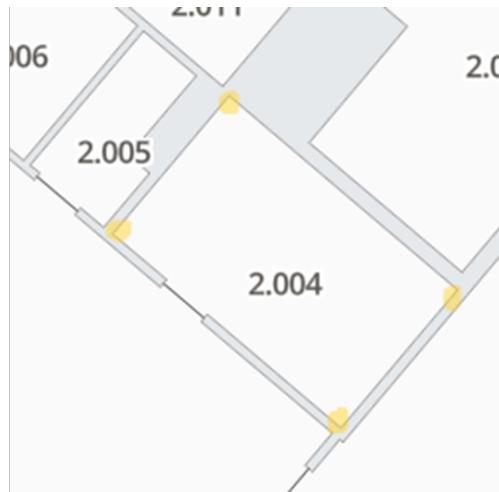
Det første oppsettet av UWB-tags og ankere ble gjort på et grupperom. Ankerene ble plassert i henhold til Pozyx sin guide. Guiden sier at ankerene skal bli plassert:

- Høyt og i synsrekkevidde for taggen
- Ikke i en rett linje, men rundt i rommet.
- 2-20 meter fra hverandre
- 20 cm vekk fra metall
- Vertikalt, slik at antennen peker opp eller ned
- I ulik høyde for å oppnå 3D posisjonering
- I synsrekkevidde til hverandre dersom man ønsker å bruke auto kalibrering

(Pozyx, n.d.) Ankerene ble plassert i vært hjørnet av rommet slik som figur 33 viser. Det ene ankeret ble plassert høyere enn de andre for å oppnå 3D posisjonering. Hjørnet nærmest det ene ankeret (ankeret til høyre på Figur 25) ble brukt som nullpunkt. Det ble brukt laser for å måle opp de andre ankerenes posisjon i forhold til dette punktet. Gulvet i rommet ble regnet som høyde 0. Ankerenes posisjoner i millimeter ble da:

Anker ID	x-retning	y-retning	z-retning
oxoD7A	27	44	1132
ox6834	27	2787	2415
ox684E	4725	494	1165
ox684F	4725	2795	1184

Når ankrene var målt opp ble oppmålingen lagt inn i Pozyx Creator Controller, vist i figur 34. Dette programmet ble brukt til å bekrefte at systemet fungerte, og at tagen greide å finne posisjonen sin.



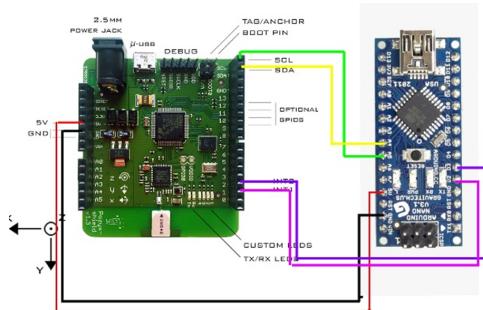
Figur 33: Ankerplassering



Figur 34: Pozyx creator controller

8.2 Avlesning av posisjon med Arduino

For å kunne bruke dataen fra Pozyx på flightcontrolleren til quadcopteret måtte dataen oversettes til MavLink. Til dette ble det brukt en Arduino Nano. Arduinoen ble loddet opp til Pozyx tagen som vist i figur 35. indoorLoiter (Mackay, 2016), et Arduino program skrevet av ArduPilot, ble brukt på Arduinoen. Dette programmet henter data fra Pozyx tagen, oversetter det til MavLink, og sender det deretter ut til Pixhawk.



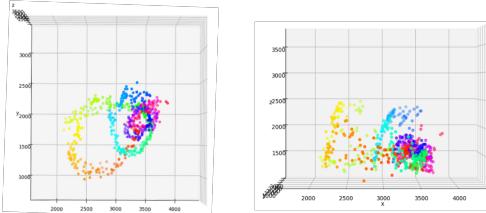
Figur 35: Oppkobling av pozyx tag og arduinor

I første omgang ble dette programmet brukt for å teste at Arduinoen faktisk greide å lese av posisjon fra Pozyx tagen, og se på nøyaktigheten til systemet.

8.3 Plotting med python

Når indoorLoiter programmet kjører på Arduinoen så sender Arduinoen ut posisjonen til tagen i x, y og z retning på USART. Dette ble brukt til å lese av posisjonen

på en datamaskin som kjørte python, og plotte posisjonen over ulike tidsrom. Figur 36 viser et ovenfra plot fra en test der tagen ble beveget rundt i grupperommet. Figur 36 viser et plot fra samme test, men fra siden. Plotene viser at posisjonen i x-y retning er mye mer presis enn posisjonen i z retning. Det ble ikke gjort mye arbeid for å forbedre dette, da høydeholdet på dronen uansett bruker barometer.



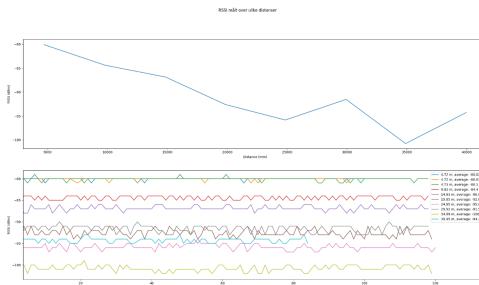
Figur 36: Plot fra innendørstest. x-y planet i figuren til høyre, x-y planet i figuren til venstre.

8.4 Presisjon og rekkevidde av UWB system

Hensikten med denne øvelsen var å teste presisjonen og rekkevidden til UWB systemet. Testene ble gjort ute, ved å enten gå rundt med en tag eller la en tag stå festet til en tripod.

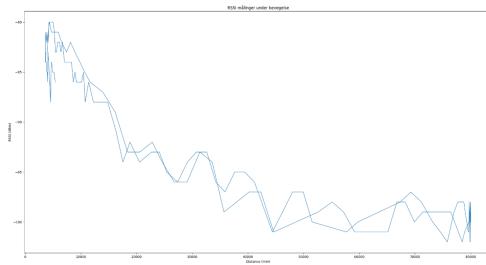
8.4.1 Rekkevidde

Først ble det utført en test for å se på rekkevidden til systemet. RSSI (Received Signal Strength Indicator) ble brukt som en indikator på hvor sterkt signal som ble mottatt, og dette ble sammenliknet med avstanden det ble målt over. Testen ble gjennomført ved å sette et anker på en tripod, og en tag på en annen tripod. Avstanden mellom disse ble målt opp med målebånd, da det var for sterkt lys til å måle opp med laser. De tre første loggingene ble gjennomført på 5 meter, og deretter ble tagen flyttet lengre og lengre vekk fra ankeret for vær logging, fem meter om gangen helt opp til 40 meter. En logging varte i 20 sekunder, og ga ca. 120 målinger. Figur 37 viser dataen fra loggene. Det øverste plotet viser gjennomsnitts verdien for hver avstand. Det viser en sterk trend på at signalet blir svakere når avstanden øker. Plotet under viser at på hver log holder RSSI målingene seg stabilt, med en nøyaktighet på $\pm 2\text{dBm}$.



Figur 37: RSSI måling over ulike distanser.

Det ble også gjennomført en test der tagen ble holdt i hånden, og flyttet fra ca. 5 meter til ca. 80 meter. Figur 38 viser resultatene fra denne testen. Bortsett fra et lite avvik helt i starten av testen, så viser dette plotet samme trend som tidligere. RSSI verdiene blir svakere når avstanden øker.



Figur 38: RSSI måling over ulike distanser.

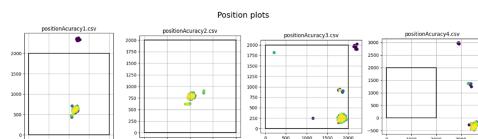
8.4.2 Presisjon

Det ble utført presisjonstester med to ulike anker oppsett. De fire første testene ble gjennomført som vist i figur 39, resterende syv tester ble gjennomført slik Figur 15 viser. Det ble skrevet et python program for logging av dataen fra Pozyx, og et python program for plotting og analyse.



Figur 39: Ulike anker oppsett.

Den svarte firkanten i Figur 40 og Figur 41 illustrerer posisjonen til ankerene i 2×2 meter oppsett, der ankrene er plassert i hvert sitt hjørne. Her er målingene plottet etter tid, der det første punktet er lilla og det siste er gult.

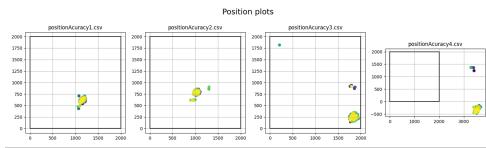


Figur 40: Målinger med $2 \cdot 2\text{m}$ ankere.

Som figur 40 viser er det mye feilmålinger helt i starten av loggingene. I figur 41 er derfor de 50 første målingene fjernet, som omrent tilsier de første 2 sekundene. Tabellene under viser et utdrag fra noen av målingene, med plot og resultater. I utregningene er også de 50 første målingene fjernet.

Figur 45 og figur 46 viser plot fra logg 6, 7 og 8. I figur 46 er de 50 første målingene fjernet for å få bort initialiserings feilmålingene.

Figur 47, fig:5x5res7 og fig:5x5res8 under viser de ulike målingene ved 5×5 meter anker oppsett, med plot og analyse data.



Figur 41: Målinger med $2 \cdot 2\text{m}$ ankere uten de 50 første målepunkter.

positionAccuracy1		
positionAccuracy1.csv		
Oppmålt punkt	1070 mm	725 mm
Gjennomsnitt	1158 mm	617 mm
Differanse	88 mm	108 mm
Standard avvik	21.68 mm	24.19 mm
Differanse på ca. 10 cm.		
Standard avvik på under 2.5 cm.		
Innenfor systemkravene.		

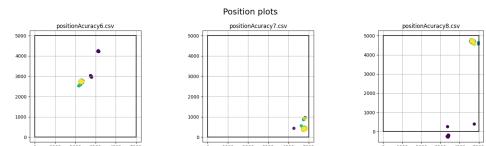
Figur 42: Resultater fra måling 1.

positionAccuracy3		
positionAccuracy3.csv		
Oppmålt punkt	2000 mm	0 mm
Gjennomsnitt	1840 mm	254 mm
Differanse	160 mm	254 mm
Standard avvik	60.68 mm	78.71 mm
Differanse på ca. 25 cm.		
Standard avvik på under 8 cm.		
Innenfor systemkravene.		

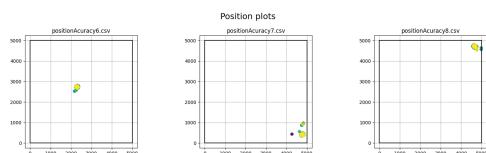
Figur 43: Resultater fra måling 3.

positionAccuracy4		
positionAccuracy4.csv		
Oppmålt punkt	4000 mm	0 mm
Gjennomsnitt	3535 mm	-320 mm
Differanse	465 mm	320 mm
Standard avvik	37.48 mm	107.54 mm
Differanse på ca. 47 cm.		
Standard avvik på under 11 cm.		
Innenfor systemkravene.		

Figur 44: Resultater fra måling 4.



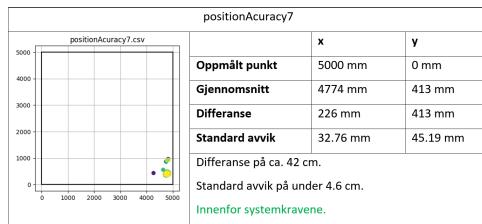
Figur 45: Målinger med $5 \cdot 5\text{m}$ ankere.



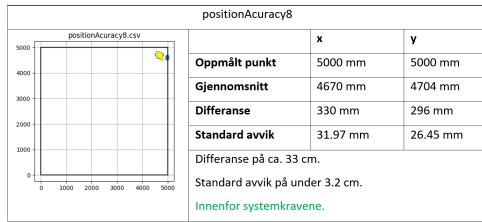
Figur 46: Målinger med $5 \cdot 5\text{m}$ ankere uten de 50 første målepunkter.

positionAccuracy6		
positionAccuracy6.csv		
Oppmålt punkt	2386 mm	2705 mm
Gjennomsnitt	2298 mm	2739 mm
Differanse	88 mm	34 mm
Standard avvik	25.35 mm	26.74 mm
Differanse på ca. 9 cm.		
Standard avvik på under 2.7 cm.		
Innenfor systemkravene.		

Figur 47: Resultater fra måling 6.



Figur 48: Resultater fra måling 7.



Figur 49: Resultater fra måling 8.

8.5 Analyse og konklusjon

Som tabellene for både 2x2 og 5x5 oppsettet viser er det ikke noe merkbar forskjell på presisjonene som blir målt. Nøyaktigheten derimot ser ut til å være bedre ved 5x5 oppsettet. Der er det største standardavviket på 4.6 cm, mens det for 2x2 er oppe i 11 cm. Oppmålingen av anker koordinatene som ble brukt til å regne ut posisjonene er en stor feilkilde i disse testene. Bakken der testene ble gjennomført var ikke helt rett, og det var vanskelig å måle nøyaktig med lasermåler på grunn av det sterke sollyset. Ved å fjerne de 50 første målingene fra testene ble resultantene mye bedre. Likevel viser plot 3 og 4 i figur 41 at enda flere målinger burde ha vært fjernet. Når systemet brukes for posisjonhold på en drone vil man aldri ta av i løpet av de første sekundene, så avviket i initieringsmålingene vil ikke ha noe å si for videre bruk. Målingene som ble gjort viser at systemet er innenfor kravet om 1m presisjon, og vil derfor kunne brukes videre i prosjektet.

8.6 Avlesning av posisjon og oppsett av drone

Dette kapittelet tar for seg hvordan dronen får posisjon ved hjelp av UWB systemet. En Pozyx tag, Arduino nano og flightcontroller ble koblet sammen og montert på dronen. Pozyx tagen måler avstanden mellom seg og ankerene, og sender disse til Arduinoen. Arduinoen laser av disse avstandene, og lager MavLink melinger som vist i figur 50 slik at Ardupilot kan bruke dataen videre. Flightcontrolleren mottar MavLink meldingene, og regner ut posisjonen utafrå avstandene til de ulike ankerene. Denne kommunikasjonen er illustrert i figur 51.

```
union beacon_distance_msg {
    struct {
        uint8_t beacon_id;
        uint32_t distance;
    } info;
    uint8_t buf[5];
};
```

Figur 50: MavLink melding for anker distanse.

Oppsett av drone: Det ble utført flere tester av flyvedyktigheten til dronen, etter å fått dronen til å fly var det forbedrigspotensiale i reguleringen. Drone fikk osilasjoner som burder bli fikset, men dette måtte bli utført i større lokaler eller utendør når været er bra.



Figur 51: Kommunikasjons vei for tag, arduino og flightcontroller.

8.7 Test av posisjonshold i gang

Etter å ha fått posisjon estimat i dronen under flyvning, var det neste steget å få dronen til å holde posisjon i luften.

8.8 Test av presisjon i krafthallen

For å få mindre magnetisk forstyrrelse og kunne tune drone bedre, ble det planlagt å fly i krafthallen. Autotune, for liten flytid.

8.9 Test av posisjonshold i Hamnahallen

Pid-regulering. Loiter test

8.10 Test av automatisk landing og preprogrammert rute

Fra tidligere tester hadde dronen suksessfullt klart å holde posisjonen i luften. Dronen fløy godt med nye PID-verdier. Vider i planen stod det å teste automatisk landing og flyging med preprogrammert rute. Testene ble utført i den eldre gymsalen på kraftsportsenter. Der var ankerene satt opp som hamnahallen med 12x12m oppsett i 150cm høyde. Før testing ble utført ble oppsettet testet etter sjekklisten. Når dronen var koblet opp mot basestasjonen ble dronen bevegd rundt i rommet for å se om posisjonen endret seg riktig. En testflight i posisjonhold ble utført for å se om det var noen problem med drone eller posisjonsystemet.

8.10.1 Landing

Dronen ble satt til å lande på en landingspad for å kunne gi indikasjon på treffsikkerheten til systemet. Landingspadden har en diameter på 100cm.

REFERENCES

- Arduino (2016). *Arduino*. URL: <https://store.arduino.cc/products/arduino-nano>.
- (udatert). *Arduino*. URL: <https://www.arduino.cc/en/Guide/Introduction>.
- Balchen, Jehns, Trond Andersen og Bjarne A Foss (2000). «Reguleringssteknikk». I. Bosch (udatert). *Data dwarfs make the world go round*. URL: <https://www.bosch.com/stories/bosch-mems-sensor-applications/>.
- Britannica, Inc Encyclopedia (udatert). *North Magnetic Pole*. URL: <https://www.britannica.com/science/North-Magnetic-Pole>.
- Developments, Level (udatert). *MEMS Accelerometer*. URL: <https://www.leveldevelopments.com/2020/10/what-are-inclinometers/mems-accelerometer/>.
- Forsell, Børje og Norvald Kjerstad (udatert). «Store norske leksikon GNSS». I: (). URL: <https://snl.no/GNSS>.
- Gellein, Jomar (2007). *Aeromagnetisk anomalikart, Tromsø*. URL: https://www.ngu.no/sokkelkart/Tromsoe_Magnetometri.pdf.
- Grøn, Øyvind (2021). *Coriolisakselerasjonen*. URL: <https://snl.no/coriolisakselerasjonen>.
- Grøn, Øyvind og Jakob Sandstad (2021). *Piezoelektrisitet*. URL: <https://snl.no/piezoelektrisitet>.
- Kartverket (udatert). *Magnetisk misvisning*. URL: <https://www.kartverket.no/til-lands/kart/turkart/magnetisk-misvisning>.
- Keim, Robert (2015). *Understanding and Applying the Hall Effect, All about circuits*. URL: <https://www.allaboutcircuits.com/technical-articles/understanding-and-applying-the-hall-effect/>.
- Mearian, Lucas (2019). *Computerworld*. URL: <https://www.computerworld.com/article/3490037/ultra-wideband-explained-and-why-its-in-the-iphone-11.html>.
- ROHM (udatert). *What is a geomagnetic sensor*. URL: <https://www.rohm.com/electronics-basics/sensor/geomagnetic-sensor>.
- Romsenter, Norsk (udatert). *Hva du bør vite om satellittnavigasjon*. URL: <https://www.romsenter.no/no/Fagområader/Satellittnavigasjon/Hva-du-boer-vite-om-satellittnavigasjon2>.
- Skaar, Johannes (udatert). *Magnetoresistans*. URL: <https://snl.no/magnetoresistans>.
- Tallysman (udatert). *GNSS Positioning Techniques*. URL: <https://www.tallysman.com/gnss-positioning-techniques/>.
- Team, Ardupilot Dev (udatert). *Ardupilot.org*. URL: <https://ardupilot.org/copter/docs/common-apm-navigation-extended-kalman-filter-overview.html>.