

Realisering og evaluering av armmontert eksoskjelett

Elling Ruud Øye

Master i teknisk kybernetikk og robotikk

Innlevert: juni 2014

Hovedveileder: Øyvind Stavdahl, ITK

Norges teknisk-naturvitenskapelige universitet
Institutt for teknisk kybernetikk



Masteroppgave

Studentens navn: Elling R. Øye
Fag: Teknisk kybernetikk
Tittel (norsk): Realisering og evaluering av armmontet exo-skjelett
Tittel (English): Realization and assessment of an arm mounted exoskeleton

Beskrivelse:

Telemanipulering brukes oftest for interaktiv styring av tradisjonelle robotmanipulatorer, og foregår ved at operatøren beveger en instrumentert robotarm (master) hvis bevegelser og/eller leddmomenter imiteres av hovedmanipulatoren (slaven). I systemer med krafttilbakekopling kan operatøren også kjenne en speiling av kreftene mellom slaven og omgivelsene.

Instituttet har i mange år drevet forskning og utvikling av slangeroboter. Så langt har fokus vært på å oppnå autonomitet mhp. fremdrift, slik at roboten selv kan ta seg frem i omgivelsene. Imidlertid krever de fleste realistiske anvendelser av slageroboter en viss grad av interaksjon med en operatør. I en tidligere studie har vi utredet et armmontert exo-skjelett som kan fungerer som master i et telemanipulator-konsept for styring av an en slangerobot. I denne oppgaven skal systemet realiseres og testes i form av en fysisk prototype.

1. Sett opp en komplett, konsis og realistisk funksjonsspesifikasjon som skal oppfylles av din første prototyp, basert på tidligere resultater.
2. Velg eller spesifiser mekaniske, elektroniske og programvarebaserte moduler for alle deler av systemet. Kommersielle komponenter bør brukes I størst mulig utstrekning, men spesiallagede moduler kan bennyttes der dette kan forsvares.
3. Koordiner anskaffelse og produksjon av moduler og sammenstillig av systemet. Bestilling av deler og mekanisk produksjon gjøres i samarbeid med instituttets verksted(er).
4. Så langt tiden tillater det, implementer systemets styringsprogramvare og evaluer systemets egenskaper i lys av dets spesifikasjoner. Diskuter systemets egnethet som operatørgrensesnitt for en slangerobot.

Veileder(e): Øyvind Stavadahl

Trondheim, desember 2013

Øyvind Stavadahl
Faglærer

Sammendrag

Denne oppgaven har hatt som mål å utarbeide en måte å styre et enkelt slangerobotledd ved hjelp av et ett-ledds eksoskjelett koblet til håndleddet til en operatør. Operatøren skal samtidig få tilbakemelding direkte fra eksoskjelettet på krefter som robotleddet utsettes for, slik at styringen blir bilateral. I forprosjektet til denne oppgaven ble teorien for styringen av et slikt system gjennomgått. To enkle styringsalgoritmer ble utviklet, og et forslag til design av en prototype ble utarbeidet.

Denne oppgaven fortsetter arbeidet fra forprosjektet. Det ble utarbeidet en funksjonsspesifikasjon for prototypen basert på forprosjektets resultater. Denne innebar at prototypen skulle kunne måle både dreiemoment og vinkelposisjon til operatørens håndledd, og samtidig ha en aktuator som kan yte en motkraft til operatørens bevegelse. Flere mulige design for eksoskjelettet ble utarbeidet og vurdert opp mot hverandre, før et endelig design ble bestemt.

Eksoskjelettet ble bygget i samarbeid med institutt for teknisk kybernetikk's mekaniske verksted. Komponenter ble hovedsakelig anskaffet fra verkstedets lager. Enkelte deler ble spesiallagd av verkstedet, eller bestilt fra butikk.

Akseptansekrav fra funksjonsspesifikasjonen ble gjennomgått for den ferdig konstruerte prototypen, og det ble konkludert med at eksoskjelettet oppfulgte de krav som ble satt ved starten av oppgaven. Det ble utført en test hvor et slangerobotledd og prototypen kobles sammen og styrer hverandre i henhold til målet med oppgaven. Av tidsmessige årsaker ble det ikke tid til å fullstendig teste styrealgortimene som ble utviklet i forprosjektet. I stedet ble en enkel posisjon-til-posisjonsstyring implementert for å vise at konseptet fungerer. Ved testing viste denne styringen å gi nokså tilfredsstillende resultater.

Det har blitt foreslått videre arbeid for prosjektet. Enten gjennom å utvide det eksisterende eksoskjelettet med flere ledd, eller å se på muligheten for å utvikle en ny og sterkere prototype med utgangspunkt i de erfaringer som har blitt gjort i løpet av dette prosjektet.

Forord

En stor takk til min veileder, Øyvind Stavdahl, for god veiledning, engasjement og humor gjennom hele semesteret. Det betyr mye å ha en veileder som engasjerer seg, og alltid finner tid til å hjelpe når komponenter eller programkode ikke lengre vil samarbeide. Takk skal du ha!

En stor takk også til Pål Liljebäck som alltid har engasjert seg når jeg har lurt på noe. Takk for lån av komponenter, og ikke minst takk for alle de gode innspillene. Takk også til Per Inge Snildal og ITK-verkstedet som greide å tolke skissene mine og konstruere et meget fungerende eksoskjelett.

Til sist, men ikke minst, takk til mamma for korrekturlesning og tilbakemelding på rapporten.

Trondheim, 13. juni 2014
Elling Ruud Øye

Innhold

1	Introduksjon	1
1.1	Rapportens oppbygning	1
1.2	Telemanipulering	1
1.3	Slangerobot	2
1.4	Styre slangerobot ved hjelp av telemanipulering	2
1.5	Spissing av oppgave	2
2	Teori og utforming av styresystem	4
2.1	Introduksjon	4
2.2	Metode	4
2.2.1	Litteratursøk	4
2.2.2	Generell modell	4
2.2.3	Valg av styringsalgoritme	6
2.2.4	Utfordringer med telemanipulering	7
2.3	Resultat	9
2.3.1	System uten tidsforsinkelse	10
2.3.2	System med tidsforsinkelse	11
2.4	Diskusjon	15
3	Design av prototype	17
3.1	Introduksjon	17
3.2	Funksjonalitet	17
3.2.1	Funksjonskrav/Sporbarhetsmatrise	18
3.3	Valg av ledd	19
3.4	Foreslåtte utforminger av prototypen	20
3.4.1	Feste av arm	20
3.4.2	Aktuering av ledd	20
3.4.3	Måling av vinkelposisjon	23
3.4.4	Måling av dreiemoment i håndledd	24
3.5	Endelig utforming av prototype	25
4	Realisering av eksoskjelett	28
4.1	Valg av komponenter	28
4.1.1	Mikrokontroller	28
4.1.2	Servo	29
4.1.3	Lastcelle	30
4.1.4	Potensiometer	34
4.2	Anskaffelse, konstruksjon og montering av komponenter	35
4.3	Programkode	36
4.3.1	Introduksjon	36
4.3.2	Overordnet funksjonalitet	36

4.3.3	Moduler	38
4.4	Oppkobling av komponenter	44
4.5	Bilder av prototypen	48
5	Funksjonstesting av eksoskjelett	49
5.1	Akseptansekrav	49
5.1.1	TPK-1: Måling av vinkelposisjonen	49
5.1.2	TPK-2: Måling av dreiemoment	50
5.1.3	TPK-3: Test av posisjonsregulering	51
5.1.4	TPK-4: Testing av sikkerhet	51
5.1.5	TPK-5: Testing av kommunikasjon	52
5.2	Posisjon-til-posisjonsregulator	53
6	Diskusjon og konklusjon	57
6.1	Utforming, realisering og funksjonstesting av prototypen	57
6.2	Testing av styresystem	58
6.3	Videre arbeid	59
7	Figurer	62
8	Digitalt vedlegg	64
9	Appendiks A - Dimensjoner, eksoskjelett	65
10	Appendiks B - Labview-applikasjon	67

1 Introduksjon

1.1 Rapportens oppbygning

Denne rapporten dokumenterer arbeidet som er gjort i løpet av våren 2014 samt forprosjektet høsten 2013 for å realisere en prototype for problemstillingen med å styre en slangerobot ved hjelp av et eksoskjelett.

I kapittel 1 introduseres problemstillingen, og det gis en kjapp introduksjon av telemanipulering, slangeroboter og hvordan disse kan tenkes å kombineres. I kapittel 2 går det grundigere gjennom teorien for telemanipulering og tilhørende matematiske modeller. Disse to kapitlene bygger på arbeid som ble gjort i forprosjektet høsten 2013, og er for det meste hentet fra [1].

I kapittel 3 går det så gjennom prosessen med å utvikle en funksjonsspesifikasjon og designe en prototype som tilfredsstillende disse kravene. I kapittel 4 beskrives de mest relevante komponentene og hvordan disse er koblet sammen. I tillegg beskrives programmet som styrer eksoskjelettet. I kapittel 5 presenteres forsøkene som ble utført med eksoskjelettet inkludert kontroll av at prototypen oppfylte akseptansekravene fra kapittel 3.

I kapittel 6 diskuteres resultatene som er oppnådd, både når det er snakk om konstruksjonen av prototypen og selve styresystemet. I tillegg legges det frem ideer om videre arbeid.

1.2 Telemanipulering

Telemanipulering går i hovedsak ut på å styre en maskin på avstand. Konseptet går ut på at operatøren styrer en master-robot som sender styresignal til en slave-robot som prøver å gjenskape bevegelsene til master-roboten. For at en slik styremekanisme skal fungere optimalt, behøves en form for tilbakekobling til operatøren som gjør det mulig å avgjøre om slave-roboten utfører oppgaven tilfredsstillende. En slik tilbakekobling kan for eksempel realiseres visuelt, gjennom instrument eller gjennom direkte tilbakekobling til master-roboten. Denne siste formen for tilbakekobling kalles bilateral telemanipulering.

De to store utfordringene for telemanipulering er stabilitet og transparens [2]. Med stabilitet mener man stabiliteten til tilbakekoblingssystemet som styrer slave-roboten samt master-roboten dersom telemanipuleringen er bilateral. Ved ideell transparens vil en operatør ikke merke forskjell på om han manipulerer slave-roboten på avstand gjennom master-roboten, eller om han manipulerer slave-roboten direkte [3]. Transparens er altså hvordan operatøren "føler" seg til stede hos slave-roboten. Grunnet hardware-begrensninger som sensorer og ikke minst kommunikasjonslinjer er det ennå ikke mulig å oppnå en så høy form for tilstedeværelse for operatøren. Utfordringen ligger derfor i å først finne rammene

for stabilitet, for å så oppnå høyeste mulig transparens innefor disse rammene [4].

1.3 Slangerobot

Markedet for mobile roboter er i dag hovedsaklig dominert av roboter som oppnår fremdrift gjennom hjulrotasjon. Noen av grunnene til dette er relativ god effektivitet, enkel styring samt muligheter for stor fart på flate underlag. Er derimot målet å kunne navigere i ulendt og ustabil terreng møter denne type mobilitet store problemer. En løsning på dette problemet kan være å ofre effektivitet mot større tilpasningsdyktighet ved å, for eksempel, bytte ut hjul med bein. Et annet alternativ, og kanskje det alternativet som gir den beste tilpasningsevnen, er en flerleddet robot med samme bevegelsesmønster som en slange [5].

Slangeroboter har et potensielt bredt bruksområde. Mye av dette potensialet ligger i tilpasningsdyktigheten til terrenget som nevnes over. Bevegelsesmønsteret og nærheten til bakken hos en slangerobot gjør dessuten faren for å velte eller og falle mye mindre. Bruksområder kan gå fra utforskning i raserte bygninger via inspisering av undervannskonstruksjoner til støtte under kikkhullskirurgi. Utformingen og realiseringen av en slangerobot har mange utfordringer, for eksempel design av ledd og elektronikkinfrastruktur. Likvel er kanskje det største problemet hvordan man skal kontrollere en enhet med så mange frihetsgrader [6].

1.4 Styre slangerobot ved hjelp av telemanipulering

Med denne oppgaven skal det utredes om kontroll av slangeroboter er mulig ved bruk av et eksoskjelett koblet til armen. Ved bruk av bilateral telemanipulering skal bevegelsene til armen styre bevegelsene til slangen, samtidig som slangeroboter måler kreftene den blir utsatt for fra omverdenen og sender disse kreftene tilbake til eksoskjelettet. Målet er at man ved hjelp av dette gjør det mulig for operatøren å føle seg fram med slangeroboter. Det er likevel noen store utfordringer som må overkommes. For det første har en slangerobot betydelig flere ledd enn en menneskearm. De to mest naturlige leddene å bruke for å gjøre en ålebevegelse med armen er albuen og håndleddet. Dersom konseptet skal testes grundig ut vil det dessuten være nødvendig å ta i bruk flere enn disse to leddene. Her kan man for eksempel legge til fingrene selv om dette mest sannsynlig vil kreve en bevegelse som ikke lengre føles like intuitiv.

1.5 Spissing av oppgave

Som i forprosjektet vil det fokuseres utelukkende på å realisere ett enkelt ledd i stedet for et fullstendig eksoskjelett for hele armen. Dette valget er tatt både utfra tidsbegrensinger og fordi det gir lite mening å utvikle flere ledd før det er testet og konkludert med at designet

for ett enkelt ledd fungerer tilfredsstillende. Målet for denne oppgaven blir altså å fortsette der hvor forprosjektet sluttet: Utforme, utvikle og realisere et ett-ledds eksoskjelett med en frihetsgrad, og dersom det er tilstrekkelig tid, teste ut styringssystemene sammen med et slangerobledd.

2 Teori og utforming av styresystem

2.1 Introduksjon

I dette kapitlet presenteres teorien som ble gjennomgått i forprosjektet høsten 2013. Stoffet er stort sett hentet fra [1], men med noen forbedringer. Grunnprinsippet for styring av en slangerobot ved hjelp av ett ekso-skjelett ligner mye på den vanlige reguleringstekniske oppgaven der et system prøver å følge en bane gitt av operatøren. Siden fokuset i denne oppgaven først og fremst blir å styre et enkelt ledd uten store påvirkninger fra resten av slangeroboten eller omgivelsene, kunne en slik utfordring løses av en enkel PID-regulator med ønsket posisjon som inngangsverdi. I forhold til telemanipulering kalles dette ofte et unilateralt system, siden kommunikasjonen bare går fra operatøren til systemet. Utfordringen blir større når man krever at slangerobotens ledd samtidig skal bestemme posisjonen til ekso-skjelettet. Dette kalles et bilateralt telemanipuleringssystem og er kritisk for at operatøren skal kunne oppleve de påvirkningene som slangeroboten opplever fra omgivelsene. Såkalt haptisk tilbakekobling.

2.2 Metode

2.2.1 Litteratursøk

Den første tiden av forprosjektet gikk med på å lese litteratur som hadde relevans til oppgaven. Hovedsakelig ble det fokusert på litteratur som omhandlet telemanipulering, hvordan et slik system kan realiseres, samt diverse teknikker for å sikre stabilitet og transparens. Siden omfanget av litteratur som er skrevet om dette temaet er svært stort, har det vært nødvendig å begrense utvalget til en håndterlig mengde. Derfor er teorigrunnlaget for dette prosjektet hovedsakelig bygget på artikler fra Dale A. Lawrence, Günther Niemeyer og Jean-Jacques E. Slotine. Disse er ofte referert til fra andre publikasjoner og gir til sammen et teorigrunnlag som passer godt til denne prosjektoppgaven. I tillegg er en doktorgradsavhandling av Edvard Nærum brukt som inspirasjon til en mer praktisk rettet tolkning av teorien.

2.2.2 Generell modell

Til å begynne med var det mest interessant å finne en modell som kunne beskrive et generelt telemanipuleringssystem. En slik generell modell ble lagt frem av Dale A. Lawrence i [4] og har i ettertid endt opp som et utgangspunkt for flere modifiserte generelle modeller. I figur 1 ser man en bearbeidet versjon av Lawrence-arkitekturen hentet fra [9]. På figuren ser man at systemet deles opp i fem deler:

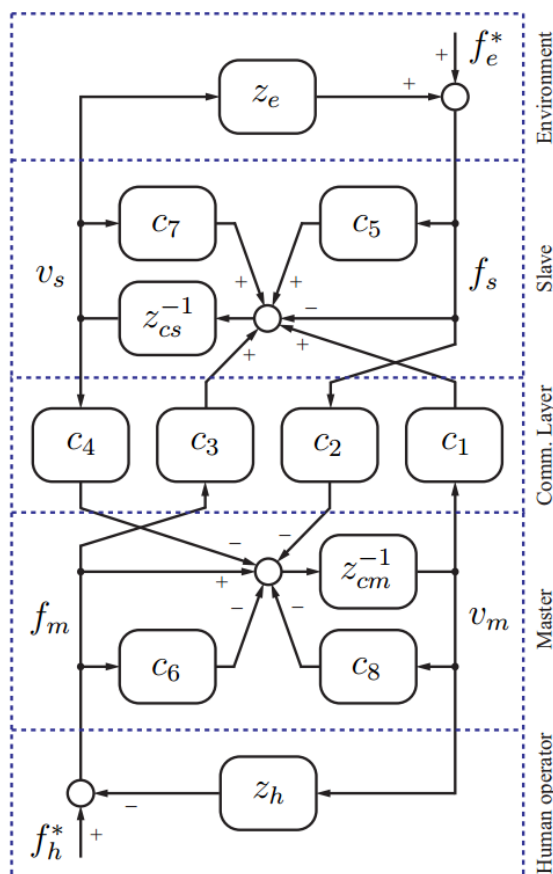
Human operator modellerer kraften (f_h) som operatøren påfører masteren, samt hvordan operatøren blir påvirket av farten (v_m) til masteren.

Master modellerer dynamikken til masteren, samt de lokale reguleringsløyferne.

Comm. Layer er kommunikasjonslinjen mellom master og slave. Her bestemmes hvilke målinger som sendes mellom master og slave. Dessuten modellerer man også eventuelle tidsforsinkelser som oppstår ved å sende signaler over kommunikasjonslinjen.

Slave modellerer slaven på samme måte som **Master** modellerer masteren.

Environment beskriver omgivelsene til slaven og hvordan disse påvirkes av slavens fart (v_s), samt hvordan kraft (f_e) fra omgivelsene påvirker slaven.



Figur 1: "The Extended Lawrence Architecture", hentet fra [9]

Ved å følge Lawrence-arkitekturen har man byggesteinene til å lage kompliserte styresystem som tar i bruk fart og kraftmålinger både fra slaven og masteren, samt egne lokale reguleringsløyfer som sikrer bedre styring og stabilitet. For dette prosjektet ble det likevel konkludert med at en enklere styringsalgoritme ville være god nok til testing av

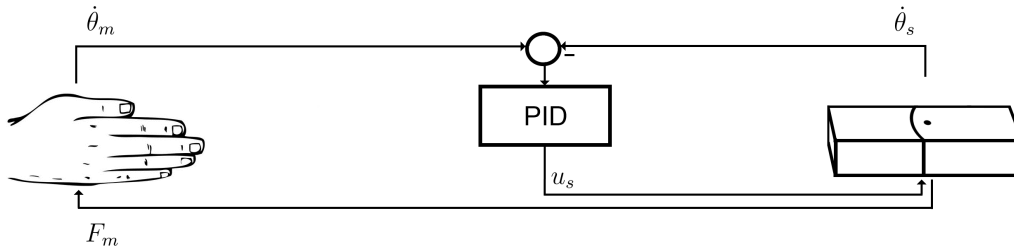
prototypen. Dette betyr at to kommunikasjonskanaler blir sløyfet og at man derfor bare sender en måling hver vei. Selv om modellen som presenteres videre er betydelig enklere enn Lawrence-arkitekturen, er prinsippene fremdeles de samme som modellen over. Fra 90-tallet og til nå har det eksistert delte oppfatninger om kravene for å oppnå best mulig transparens, altså at operatøren føler seg til stede i slavesystemet. Til å begynne med trodde man at det var nødvendig å bruke alle fire kommunikasjonslinjene [4], men det har i ettertid vist seg å være mulig å oppnå samme resultat med to linjer [9]. Derfor vil det, selv om systemet konstrueres med støtte bare for to kommunikasjonslinjer, være mulig å oppnå tilfredsstillende resultat dersom man bruker riktig styringsalgoritme.

2.2.3 Valg av styringsalgoritme

Etter at en generell arkitektur for styring var funnet, var det så mulig å komme frem til en forenklet styringsalgoritme som ville fungere tilfredsstillende nok til prototypetesting. Som nevnt over begrenset denne seg til to kommunikasjonslinjer. Det vil si at både master og slave bare har muligheten til å sende en måling hver. De aktuelle målingene, jamfør figur 1, er vinkelposisjon (eller vinkelfart) og dreiemoment. Det ble bestemt at en kombinasjon av disse målingene skulle prøves ut. Det vil si at det ene systemet styres med kraftmåling, og det andre systemet styres med posisjonsregulering. Siden man ønsket å teste om det oppleves noen forskjell dersom man bytter mellom hvilket system som styres av kraftmålinger og hvilket system som styres av posisjonsmålinger, innebar dette også at ekso-skjelettet måtte være i stand til å måle både vinkelposisjon og dreiemoment. Med disse målingene oppfyller man dessuten kriteriet for transparens gitt i [9]. I følgende to avsnitt beskrives den grunnleggende funksjonaliteten til de to valgte styresystemene.

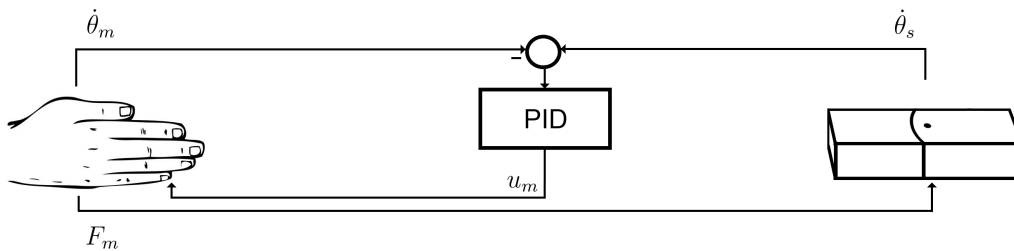
Kraftstyrt ekso-skjelett, posisjonsstyrt slangerobotledd I figur 2 ser man en oversikt over det første alternativet for styring mellom handleddet og slangeleddet. Fart fra både eksoskjelett og slangerobotledd (henholdsvis $\dot{\theta}_m$ og $\dot{\theta}_s$) måles, og differansen sendes som referanse til en PID-regulator som gir pådraget u_s til slangerobotleddet. I slangerobotleddet måles så dreiemomentet om rotasjonsaksen, og denne målingen F_s sendes til ekso-skjelettet som pådrag. Dersom man tenker seg en situasjon hvor operatøren beveger handleddet, vil avviket mellom vinkelfarten til operatøren og vinkelfarten til slangerobotleddet gi et pådrag til slangeroboten. Kraften som denne da opplever på grunn av bevegelsen vil så bli målt og sendt som pådrag tilbake til operatøren. Kommunikasjonslinjen i figur 2 vil i dette tilfellet ligge mellom eksoskjelettet og PID-regulatoren.

Posisjonsstyrt ekso-skjelett, kraftstyrt slangerobotledd Det andre alternativet til styring kan sees i figur 3. Det er ingen store forandringer i konseptet bortsett fra at ekso-skjelettet og slangeroboten har byttet plass. Dette vil si at en bevegelse utført av operatøren fører til et dreiemoment som blir målt og så sendt som pådrag F_m til



Figur 2: Reguleringsløyfe: Fart fra master, kraft fra slave

slangeroboten. Videre vil differansen mellom vinkelfarten til håndleddet til operatøren og vinkelfarten til slangerobotleddet gi et pådrag u_m tilbake til operatøren. I dette tilfellet ligger kommunikasjonslinjen mellom PID-regulatoren og slangeroboten.



Figur 3: Reguleringsløyfe: Kraft fra master, fart fra slave

2.2.4 utfordringer med telemanipulering

Etter at grunnleggende styreprinsipp var valgt, ble det aktuelt å se på diverse utfordringer relatert til et telemanipuleringssystem. Som nevnt før er stabilitet en av de store utfordringene med bilateral telemanipulering. Dette kommer hovedsakelig av tidsforsinkelsen mellom signalene som sendes mellom master og slave [10]. For prototypen sin del er det ikke urimelig å anta at tidsforsinkelsen mellom master og slave kommer til å bli marginal siden den fysiske avstanden vil bli liten og at informasjonen som skal sendes ikke er spesielt stor. Likevel er det vist i [10] at selv små tidsforsinkelser kan føre til ustabilitet. Derfor ble det anslått at det ville være uklokt å ikke ta høyde for at tidsforsinkelsene kunne bli et problem.

Forskningen på telemanipulering har fokusert mye på mulige løsninger av stabilitetsutfordringen som følger med tidsforsinkelse mellom master og slave. Denne forskningen innebærer løsninger både for konstante og varierende tidsforsinkelser. I dette prosjektet er det rimelig å anta en konstant tidsforsinkelse. En mye brukt løsning for dette problemet er bølgevariabler (eller på engelsk *Wave variables*) [10]. Dersom man anvender passivitetsteorien på et telemanipuleringssystem, kan man konstruere master- og slavesystemet slik at de begge hver for seg er passive. I følge passivitetsteoremet er da et tilbakekoblingsystem mellom master og slave også passivt og derfor stabilt [11]. Problemet oppstår når et element med tidsforsinkelse legges til i systemet. Dette elementet er ikke passivt, og betingelsene for passivitet og derfor stabilitet for hele systemet er ikke lengre til stede. Bølgevariabler er en modifisering til passivitetsteorien som sørger for at tidsforsinkelser blir passive element [7]. Dette oppnås ved å transformere fart- og kraftmålingene før de sendes mellom master og slave. Ved hjelp av denne transformasjonen oppnår man at tidsforsinkelseselementet er passivt uansett størrelse på tidsforsinkelsen. Bølgevariablene blir av Niemeyer og Slotine definert slik:

$$u = \frac{b\dot{x} + F}{\sqrt{2b}}, v = \frac{b\dot{x} - F}{\sqrt{2b}}, \quad (1)$$

Her er \dot{x} farten til systemet, F kraften systemet utsettes for og b er den karakteristiske bølgeimpedansen som kan brukes til å justere hvorvidt \dot{x} eller F påvirker bølgevariablen mest. Ved så å opprette et sett variabler for både master og slave får man disse ligningene:

$$u_m = \frac{b\dot{x}_m + F_m}{\sqrt{2b}}, v_m = \frac{b\dot{x}_m - F_m}{\sqrt{2b}}, \quad (2)$$

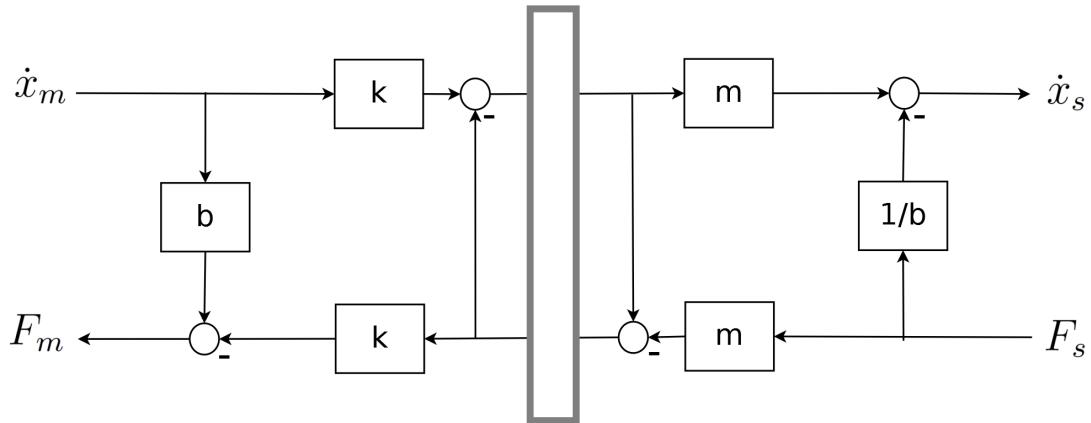
$$u_s = \frac{b\dot{x}_s + F_s}{\sqrt{2b}}, v_s = \frac{b\dot{x}_s - F_s}{\sqrt{2b}}, \quad (3)$$

For tilfellet der masteren sender målt vinkelfart til slaven og slaven sender målt dreiemoment til masteren, er det hensiktsmessig å skrive om (2) og (3) slik:

$$u_m = \sqrt{2b}\dot{x}_m - v_m, F_m = b\dot{x}_m - \sqrt{2b}v_m, \quad (4)$$

$$\dot{x}_s = \sqrt{\frac{2}{b}}u_s - \frac{1}{b}F_s, v_s = u_s - \sqrt{\frac{2}{b}}F_s, \quad (5)$$

I figur 4 kan man se (4) og (5) implementert i et blokkdiagram på samme måte som i [7]. Her kommer dessuten kommunikasjonsgangen for signalene noe tydeligere frem.



Figur 4: Blokkdiagram: Bølgevariabler

Her er F_m og F_s dreiemoment rundt henholdsvis master og slave. \dot{x}_m og \dot{x}_s er vinkelfart. I tillegg har man at $k = \sqrt{2b}$ og $m = \sqrt{\frac{2}{b}}$. Det grå rektangelet representerer kommunikasjonslinjen som naturlig nok har tidsforsinkelse i begge retninger.

For at systemet skal være kompatibel med bølgevariabler, er det nødvendig å benytte målinger av vinkelfart for styring av systemet i stedet for å velge målinger av vinkelposisjon. Dette medfører et ekstra problem med tanke på posisjonsdrifting. Siden man ikke kan garantere at målingene av vinkelfart blir absolutt nøyaktige på grunn av usikkerhet i sensorer samt feil i modelleringen, er det mulig at vinkelposisjonen vil drifte selv om styringssystemet tror det står i ro. Dette kommer av små avvik i målingen av vinkelfart. Det ble derfor foreslått å legge til et ekstra ledd i modellen som regulerer med tanke på posisjonen til enten slave eller master. Dette betyr at man vil få en regulator på utsiden av det allerede stabile systemet. Dersom regulatoren som utfører dette er en PI-regulator med svært små koeffisienter, vil dette leddet i praksis ikke ødelegge for stabiliteten som allerede er oppnådd siden regulatoren bare vil påvirke systemet når det står i ro over tid. Det er usikkert om dette leddet blir nødvendig, men siden regulatoren benytter seg av målinger som allerede er tilgjengelig, vil det være enkelt å aktivere og deaktivere dette leddet i programvare etter behov.

2.3 Resultat

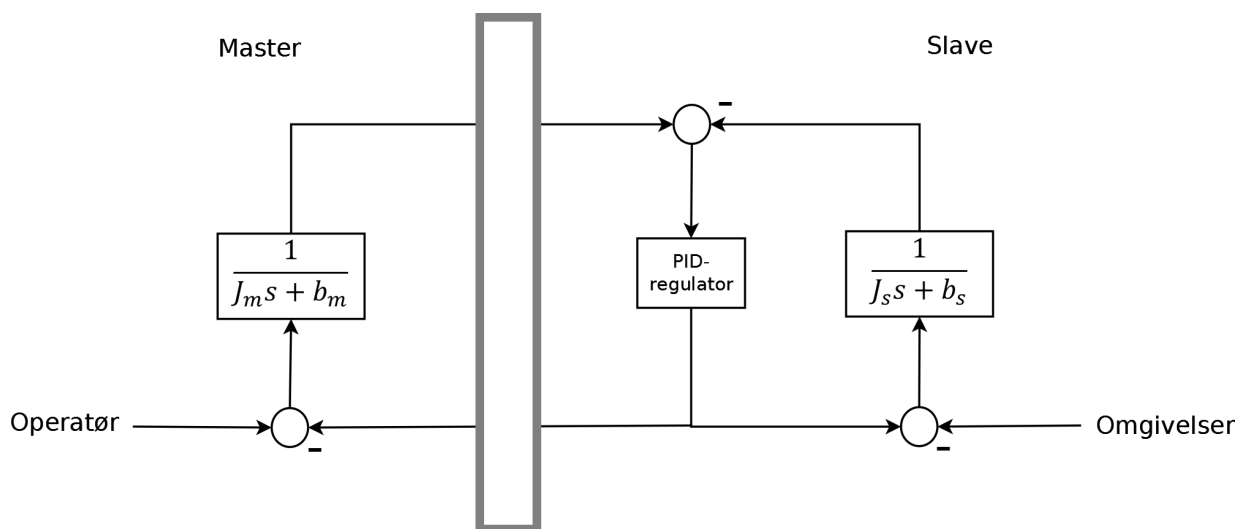
For å teste om antagelsene og konklusjonene fra litteratursøket er korrekte, ble det i første omgang fokusert på å simulere telemanipuleringssystemet i Simulink. Simuleringen deles inn i to faser. Den første går ut på å modellere masteren og slaven samt koble disse sammen til et ideelt system uten tidsforsinkelse. I den andre fasen tilføres så en tidsforsinkelse mellom systemene og bølgevariabler innføres for å sikre stabilitet.

2.3.1 System uten tidsforsinkelse

Både masteren og slaven, det vil si ekso-skjelettledet og slangerobotledet, modelleres som en vinkelbevegelse med dreiemoment J og demping b . Med vinkelposisjon θ og pådrag u får man da:

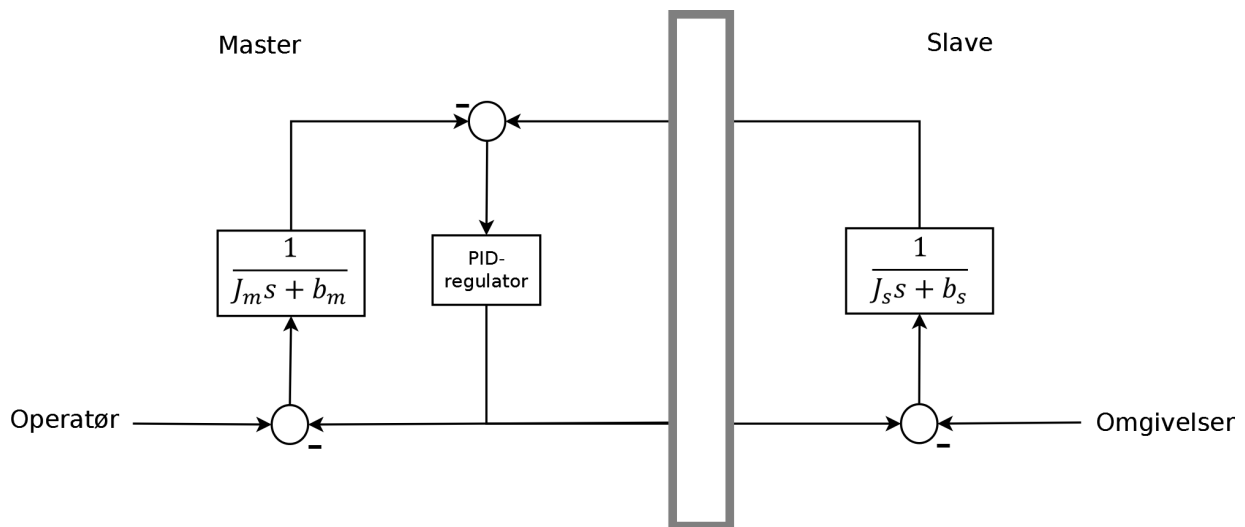
$$J\ddot{\theta} + b\dot{\theta} + u = 0 \quad (6)$$

Pådraget u vil bestå av flere komponenter avhengig av hvilken styremodell som brukes. For et system som styres ved hjelp av differanse i vinkelhastighet, vil u bestå av pådrag fra PID-regulatoren og operatøren/omgivelsene. For et system som styres av kreftene som oppleves av det andre systemet består u av et pådrag hentet direkte fra kommunikasjonslinjen samt krefter fra operatøren/omgivelsene. Som nevnt tidligere er de to aktuelle styresystemene *kraftstyrt ekso-skjelett*, *posisjonsstyrt slangerobotledd* og *posisjonsstyrt ekso-skjelett*, *kraftstyrt slangerobotledd*. Simulinkmodellene til disse to systemene kan sees i henholdsvis figur 5 og figur 6. Her er også kommunikasjonslinjen tegnet inn for å gi en bedre oversikt over systemet, men tidskonstantene er foreløpig satt til null.



Figur 5: Blokkdiagram: Fart fra master, kraft fra slave

Siden de to forskjellige styresystemene ikke skiller seg fra hverandre på andre måter enn at master og slave bytter posisjon i blokkdiagrammet, er det derfor tilstrekkelig å simulere ett av systemene så lenge man simulerer pådrag på både master og slave. I figur 7 ser man simuleringen av systemet i figur 5. Simuleringen foregår over 30 sekunder. Etter ett sekund gis en steppfunksjon som pådrag på masteren. Deretter, fjorten sekunder etterpå, gis et like stort pådrag på slaven i motsatt retning. Meningen er altså at disse pådragene skal utligne



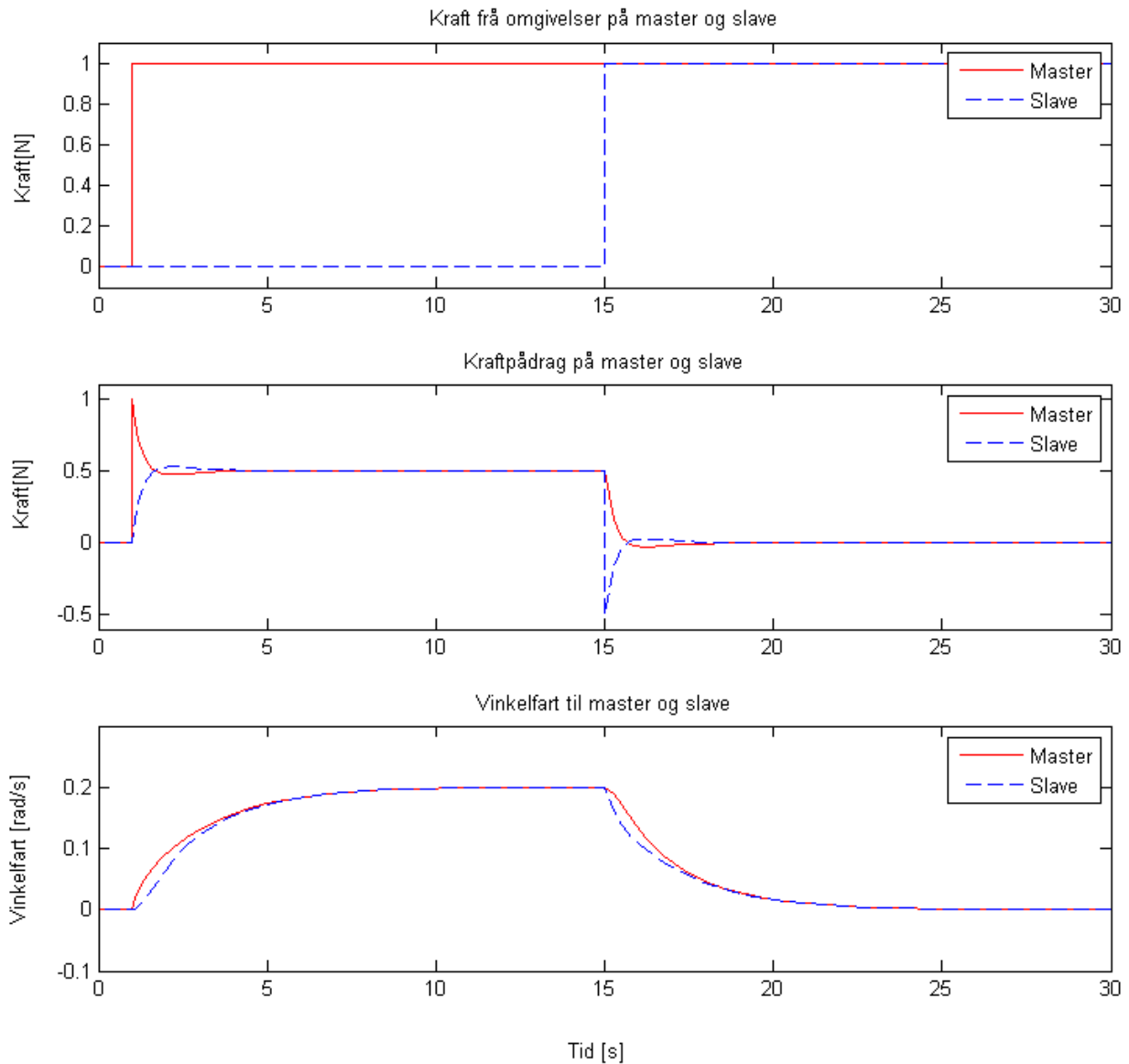
Figur 6: Blokkdiagram: Kraft fra master, fart fra slave

hverandre. I de to nederste grafene i figur 7 vises kraften som oppleves av hvert system og vinkelfarten. Som man kan se fra figuren opplever masteren en brå kraft fra operatøren etter ett sekund. Samtidig får man en økning i dreiemomentet hos slaven. Videre ser man at kraften på masteren synker og kraften på slaven øker til de ender opp på samme verdi. Dette kommer av at masteren og slaven nå har samme vinkelfart. Etter 15 sekunder ser man så at pådraget med samme størrelse påsettes slaven. Den samme dynamikken oppstår nå bortsett fra at rollene er byttet om. Til slutt når vinkelfarten til begge systemer null, og kreftene fra operatør og omgivelser nuller hverandre ut. Ut fra plottet er det vanskelig å se noen stor forskjell mellom pådrag fra masteren og pådrag fra slaven. Dette tyder på at antagelsen om at det er nok å simulere ett av styresystemene så langt holder.

2.3.2 System med tidsforsinkelse

Det innføres nå en konstant tidsforsinkelse mellom master og slave. I figur 8 ser man simulering av systemet i figur 5 med en tidsforsinkelse over kommunikasjonslinjene på 1000 ms. Med denne relativt store tidsforsinkelsen ser man at faseforstyrrelsen fra tidsforsinkelsen er nok til at systemet oscillerer ut av kontroll. Det er viktig å poengtere at dette er en svært stor tidsforsinkelse som ikke er realistisk med tanke på det fysiske oppsettet som skal implementeres senere. Dersom man senker tidsforsinkelsen til 100 ms får man, som vist i figur 9, en noenlunde stabil vinkelfart. Dette kan tale for at innføringen av bølgevariabler ikke er nødvendig. Samtidig er dette også en ganske enkel simulering som ikke tar hensyn til mange av støykildene som vi være til stede på den fysiske modellen.

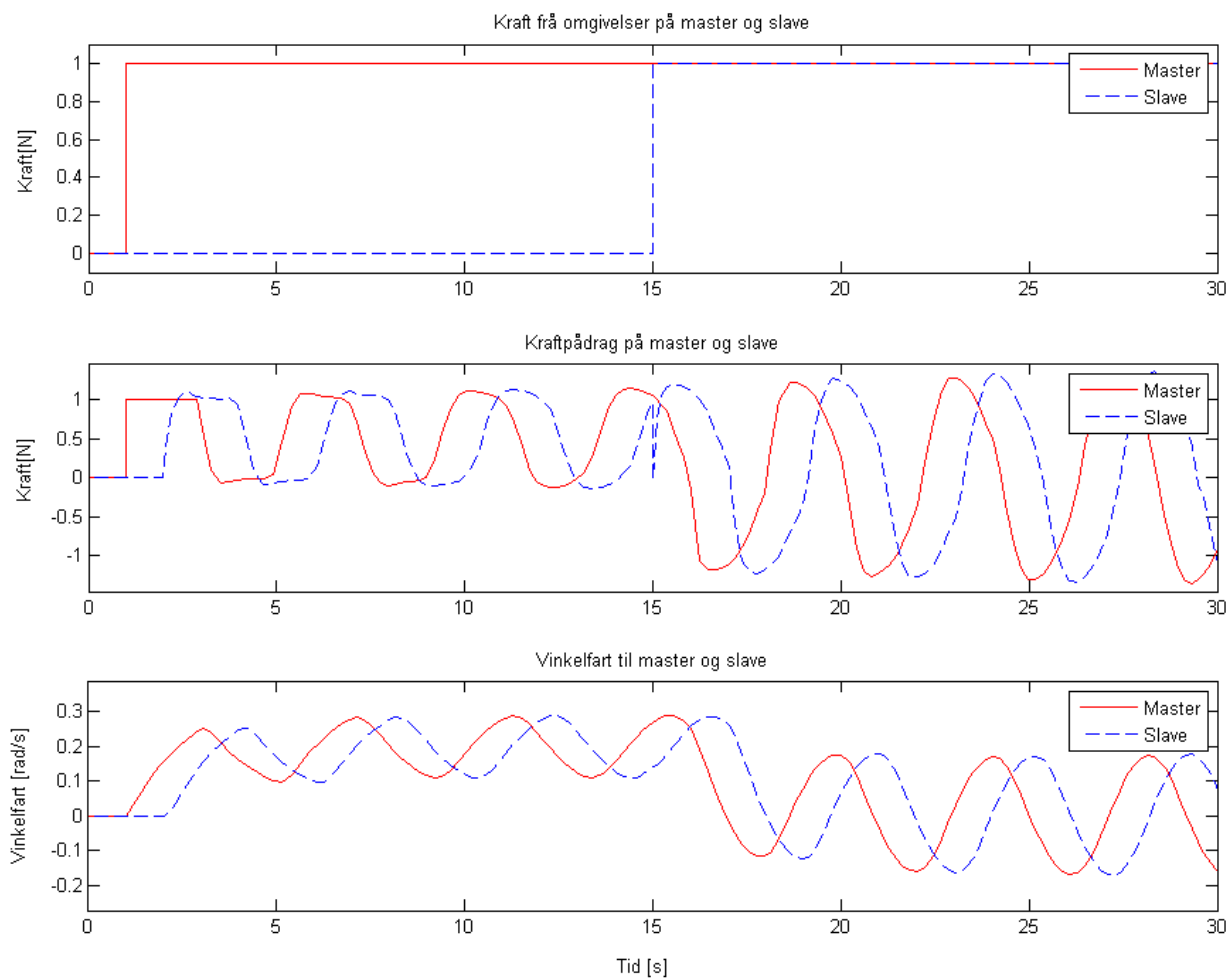
Bølgevariabler innføres så i modellen. Dette gjøres på samme måte som beskrives tidligere i rapporten. Systemet som simuleres er fremdeles det samme som kan sees i figur 5. Det vil si at masteren sender vinkelfartsmålinger og slaven kraftmålinger. Bølgevariablene kan



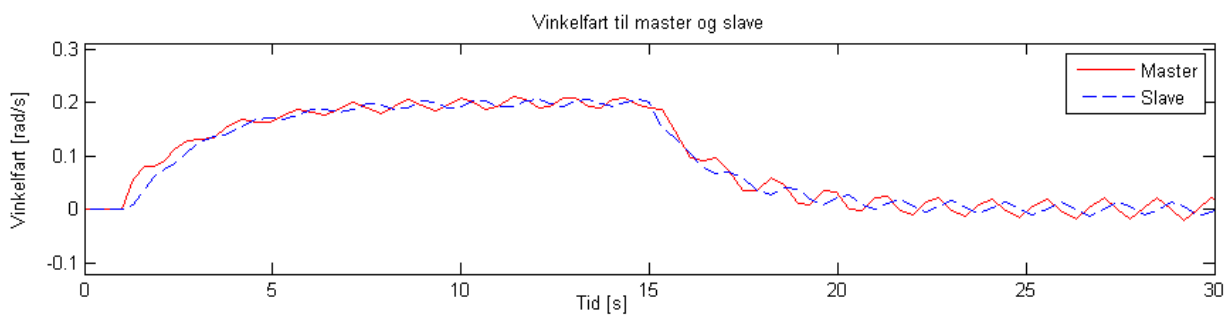
Figur 7: Simulering av system uten tidsforsinkelse

altså uttrykkes som i (4) og (5).

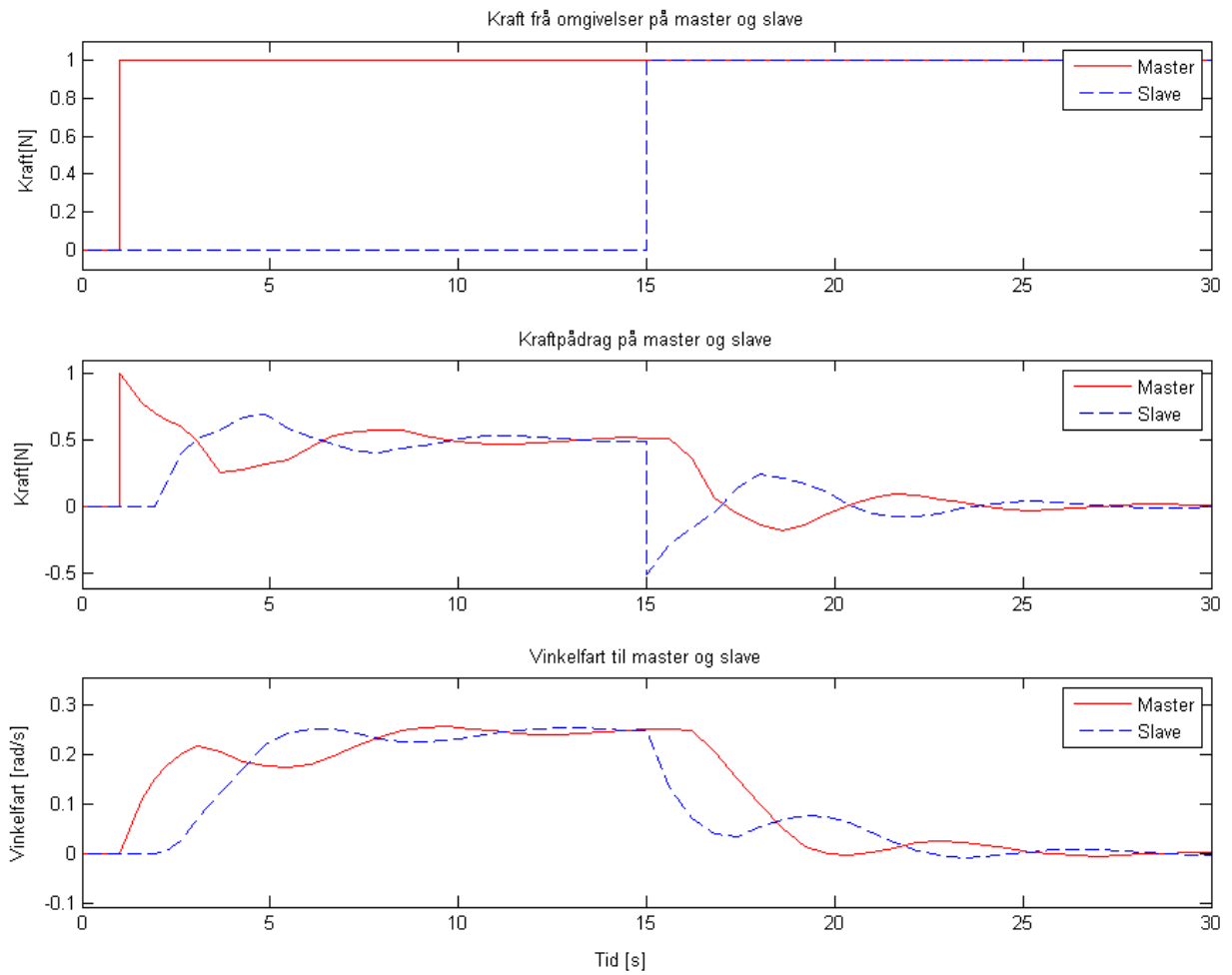
Systemet simuleres så med 1000 ms. Som man kan se i figur 10 fjerner ikke innføringen av bølgevariablene oscillasjonene til masteren og slaven. Dette er ikke til å unngå siden tidsforsinkelsen fører til at det ene systemet henger etter det andre. Derimot sørger bølgevariablene for at svingningene er stabile og minkende. Som man kan se av figuren roer variasjonene mellom master og slave seg ned etterhvert som tiden går etter siste pådrag.



Figur 8: Simulering av system med 1000 ms tidsforsinkelse



Figur 9: Simulering av system med 100 ms tidsforsinkelse



Figur 10: Simulering av system med bølgevariabler

2.4 Diskusjon

Formålet med dette kapittelet var å få en oversikt over telemanipulering generelt samt å finne utkast til styresystem som skal kunne implementeres på prototypen når denne etterhvert bygges. Litteratursøket ble derfor fokusert på å gi en generell forståelse for telemanipulering samt gi et utgangspunkt for å kunne bestemme seg for et passende styresystem for oppgaven. Valget med å begrense seg til et lite utvalg forfattere har vist seg å fungere bra. Dette betyr selvsagt ikke at andre artikler kunne gitt en enda bedre og breiere forståelse, men ettersom forfatterne Dale A. Lawrence, Günther Niemeyer og Jean-Jacques E. Slotine er mye sitert og gjengitt av andre, virker det som om det teoretiske grunnlaget ble sterkt nok basert på disse forfatterne. Det har blitt tatt hensyn til de to hovedutfordringene med telemanipulering, stabilitet og transparens, når styringen for systemet skulle utarbeides. Riktignok har det blitt mer fokus på stabilitet, men dette er også det viktigste aspektet for å kunne teste prototypen. Først når prototypen er stabil og styringen virker som forventet, kan man gjøre endringer i programvare for å oppnå en bedre transparens. Det er dessuten bedre og enklere å finstille transparens etter at man har bygget det fysiske systemet siden det enda ikke er noen klare objektive krav for god transparens.

De to styresystemene som ble valgt er ikke spesielt kompliserte sammenlignet med de mer intrikate systemene som kan utledes fra Lawrence-arkitekturen. Det er heller ikke hensiktsmessig å bruke mye tid på et avansert styresystem inntil man faktisk har bygget et prototype og kan vise at de enkleste prinsippene fungerer. Ved å velge to styresystem som til sammen benytter seg av både vinkelfart og dreiemoment, har man dessuten oppnådd at den fysiske modellen tar de målingene som er nødvendige også for mer avanserte styresystem. Dette betyr at innføring av andre styresystem kan gjøres utelukkende i programvare uten noen endringer i maskinvare eller konstruksjonen til ekso-skjelettet.

Simuleringen som ble gjennomført er ikke primært tenkt som en nøyaktig modell av prototypen som skal bygges senere. Disse simuleringene gir derimot en bekreftelse på at tolkningen av litteraturen om telemanipulering er korrekt og at styresystemene som er valgt er gjennomførbare. Den første simuleringen uten tidsforsinkelser viser at systemet under ideelle forhold er stabilt og at de to systemene kan følge hverandre uavhengig av hvem som står for pådraget. Altså at endringer i både master og slave forplanter seg til det motsatte systemet uten store forskjeller. Dette er essensielt for et bilateralt telemanipuleringssystem. I samsvar med teorien ble det så observert stabilitetsproblemer når en tidsforsinkelse ble introdusert i kommunikasjonslinjene mellom master og slave. Det viser seg imidlertid at små tidsforskjeller ikke nødvendigvis fører til ustabilitet, men man kan med en gang registrere større oscillasjoner enn ved ingen forsinkelse. Dette viser at det er hensiktsmessig å utføre tester uten bølgevariabler for å se om dette også er tilfellet for prototypen. Det er ikke forventet at tidsforskjellen mellom master og slave skal være veldig stor.

Simuleringen av systemet med bølgevariabler og tidsforsinkelse gav også et resultat som samsvarer med teorien. Selv med store tidsforsinkelser var systemet stabilt. Riktignok vil store tidsforsinkelser gjøre systemet svært vanskelig å styre på grunn av oscillasjonene som oppstår av at et system henger etter det andre. Dette er tvilsomt et aktuelt problem for denne prototypen siden tidsforsinkelsene mest sannsynlig er så små at spørsmålet heller blir om innføring av bølgevariabler er nødvendig.

3 Design av prototype

3.1 Introduksjon

Denne oppgaven fortsetter der forprosjektet sluttet; design av en prototype. I forprosjektet ble det bare sett på én enkelt mulig utforming. Det ble derfor nødvendig å se på andre mulige løsninger for å unngå å låse seg til én enkelt ide. Noe av grunnlaget fra prosjektoppgaven beholdes likevel, både når det kommer til grunnleggende funksjonsspesifikasjon og valg av hvilket ledd på operatøren eksoskjelettet skal tilpasses til. Funksjonsspesifikasjonen er imidlertid mer detaljert og strukturert enn i forprosjektet.

I dette kapittelet går det først gjennom den utvidede funksjonsspesifikasjonen for eksoskjelettet samt en sporbarhetsmatrise som går gjennom de nødvendige testene eksoskjelettet skal kunne utføre etter at det er realisert. Videre presenteres ulike mulige realiseringer av eksoskjelettet som vil oppfylle kravene som stilles for funksjonalitet. Fordeler og ulemper med hvert design tas opp, før et endelig design bestemmes. Til slutt diskuteres valget og hvordan dette passer inn i videre arbeid i prosjektet.

3.2 Funksjonalitet

Som nevnt tidligere har man i denne oppgaven valgt å muliggjøre testingen av to forskjellige styresystem. Ett, der ekso-skjelettet kommuniserer vinkelposisjon, og ett, der ekso-skjelettet kommuniserer dreiemoment. I tillegg trenger man en metode for å styre vinkelutslaget til leddet. Disse kravene kan formuleres til tre hovedkrav for funksjonaliteten til prototypen:

Måling av vinkelposisjon For å kunne bestemme vinkelfarten til ekso-skjelettet, må leddet ha en form for måling som beskriver vinkelposisjonen. Denne målingen må så brukes til å regne ut vinkelfarten.

Dreiemoment om leddet For å kunne bestemme kraften som leddet utsettes for, må man ha en form for måling av dreiemomentet om leddet.

Styring av vinkel i handledd For å kunne realisere tilbakekoblingen fra slave, må man kunne styre vinkelen på ekso-skjelettet.

Eksoskjelettets overordnede funksjon er to-delt. For det første skal eksoskjelettet måle operatørens bevegelse i handleddet. Dette vil si å måle hvilken vinkel leddet står i samtidig som man måler dreiemomentet som operatørens handledd utsetter eksoskjelettet for. I tillegg til å måle handleddets bevegelse skal eksoskjelettet også være i stand til å begrense bevegelsen til operatøren. Dette vil si at eksoskjelettet skal være i stand til å motvirke endringer i handleddets leddvinkel.

For å realisere denne funksjonaliteten må eksoskjelettet oppfylle visse krav: For det første må eksoskjelettet kunne følge håndleddets bevegelse. Dette innebærer at systemet er festet på begge sider av leddet, slik at enhver endring i leddvinkel leder til en endring i eksoskjelettet. Videre må eksoskjelettet kunne utføre to målinger: Måling av vinkel, og en måling av dreiemoment eller kraft med påtrykk på en bestemt lengde fra leddet. Til slutt må en aktuator kobles til leddet slik at eksoskjelettet selv kan påvirke vinkelen på håndleddet. Under oppsummeres disse kravene i en sporbarhetsmatrise.

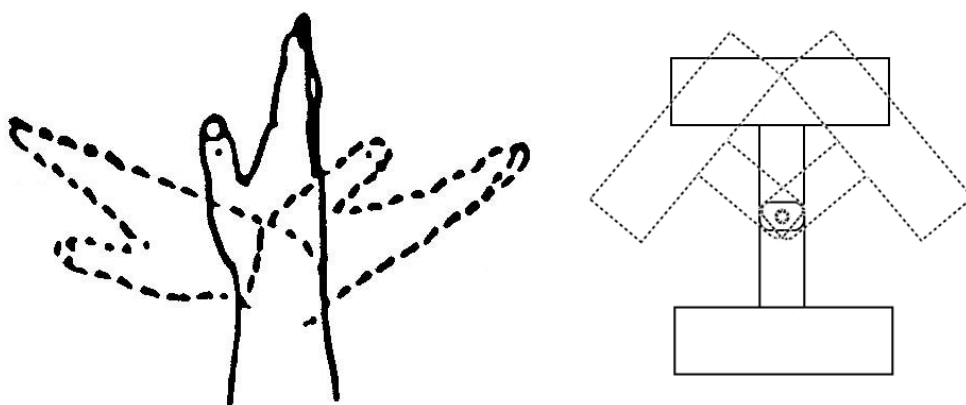
3.2.1 Funksjonskrav/Sporbarhetsmatrise

Krav ref	Produktkrav	Akseptansekrav	Test ref
PK-1	Eksoskjelettet skal kunne måle vinkelposisjon til operatørens håndledd.	Eksoskjelettet kjøres mellom ytterpunktene for leddutslaget og vinkelposisjon leses av og sammenlignes med den faktiske bevegelsen.	TPK-1
PK-2	Eksoskjelettet skal kunne måle dreiemomentet som håndleddet utsettes for. Dette vil si kreftene som oppstår mellom operatørens håndledd og eksoskjelettet.	En kjent kraft påføres leddet til eksoskjelettet mens det står i forskjellige vinkelposisjoner og måleverdien leses av. Denne sammenlignes så med den kjente kraften (dette kan f.eks. utføres med vekter som henges på håndleddet).	TPK-2
PK-3	Eksoskjelettet skal kunne styre operatørens håndledd til ønsket vinkelposisjon.	Eksoskjelettet kjøres til bestemte posisjoner og det sjekkes at disse oppnås.	TPK-3
PK-4	Eksoskjelettet må være trygt å bruke. Dette innebærer at håndleddet til operatøren ikke kan tvinges til en posisjon som medfører ubehag eller skade.	Eksoskjelettet kjøres med full kraft mot operatørens håndledd uten at dette gjør skade på brukeren.	TPK-4
PK-5	Eksoskjelettet skal kunne sende målinger av vinkelposisjon og dreiemoment til en ekstern modul. Dessuten skal eksoskjelettet kunne motta kommandoer fra en ekstern modul for styring av vinkelposisjonen til håndleddet.	Målingene skal kunne sees på en ekstern modul (f.eks en datamaskin). En annen modul (f. eks. en mikrokontroller) skal kunne styre eksoskjelettet til ønskede posisjoner.	TPK-5

3.3 Valg av ledd

Siden slangeroboten settes sammen av en rekke helt like ledd, er det for denne oppgaven nok å kunne styre hvilket som helst av disse leddene. Prototypen til ekso-skjelettet kan testes på et enkelt slangeledd som ikke trenger å være tilkoblet resten av slangen. For ekso-skjelettet er det derimot viktig at armleddet som skal benyttes er bestemt på forhånd siden leddene på armen skiller seg fra hverandre både i størrelse og leddutslag. Siden det i første omgang er snakk om konstruksjon av en prototype, er det ønskelig å gjøre konstruksjonen så enkel som mulig. Diverse fingerledd ble vurdert i starten. Fordelen med fingrene er at det ikke kreves noen stor konstruksjon for å realisere et fingerledds-ekso-skjelett. Ulempen er at en liten konstruksjon krever små komponenter samt at det er begrenset hvor mye vekt og kraft man kan utsette fingrene for med tanke på sikkerhet. En fordel var at det ble konstruert en fingerhylse med kraftsensorer og servostyring av ledd i en tidligere masteroppgave av Ranveig Solberg i 2011 [12]. Det viste seg imidlertid at forfatningen til fingerhylsen ikke var like god som for tre år siden. Dessuten var det usikkert om kraftsensorene brukt i hylsen ga en nøyaktig nok måling. Alternativet ble derfor å konstruere en ny fingerhylse selv. Tatt i betraktning begrensningene for en så liten konstruksjon ble det heller bestemt å fokusere på en av de større leddene i armen.

Valget stod derfor mellom albuen og håndleddet. Begge disse leddene krever en større konstruksjon, men kan derfor også tåle mer vekt, noe som gjør det enklere å utstyre ekso-skjelettet med komponenter. Siden leddene stiller forholdsvis likt med tanke på størrelse ble den avgjørende faktoren leddutslag. Håndleddet tilbyr en enkel og intuitiv mapping til et slangerobotledd som vist under i figur 11. For albuen må man derimot definere middelposisjonen til leddet til slangeroboten som et sted rundt 90° for albuen for å oppnå mulighet til å styre leddet begge veier. Dette er ikke et stort problem, men nok til at det virker hensiktsmessig å velge håndleddet som utgangspunkt for testing av konseptet.

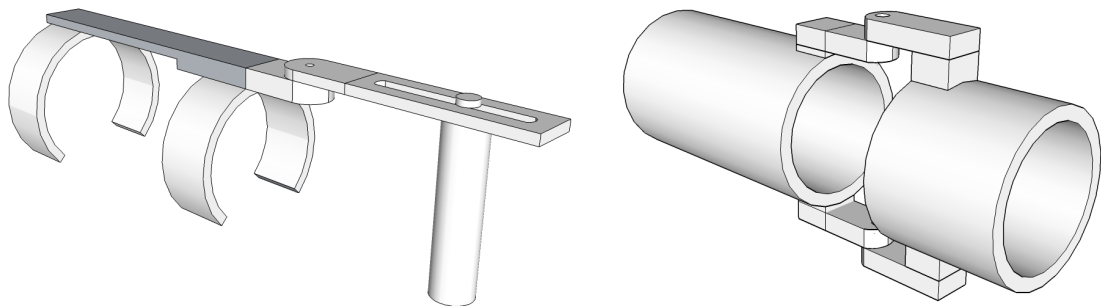


Figur 11: Håndledd og slangeledd

3.4 Foreslåtte utforminger av prototypen

3.4.1 Feste av arm

Festet mellom eksoskjelettet og underarm og hånd har ingen andre krav enn at systemet sitter fast på armen og følger håndens bevegelse. Det er derimot ganske viktig at rammen til eksoskjelettet er stort nok til at nødvendige komponenter kan festes på utsiden eller bygges inn i systemet. Forskjellige løsninger på aktuering og målinger vil variere kravene om hvor stort eksoskjelettet må være og dette vil vises når de forskjellige løsningene går gjennom senere i dette kapittelet. I figur 12 presenteres de to hovedidéene på hvordan rammen til eksoskjelettet skulle se ut.



Figur 12: De to forskjellige forslagene til eksoskjelett-rammer

Til høyre består rammen av to rør hvor det lengste omslutter underarmen og det korteste, hånden til operatøren, slik at håndleddet til operatøren ligger mellom leddene i eksoskjelettet. Fordelen med denne rammen er at den store overflaten gir god fleksibilitet til å feste komponenter til eksoskjelettet. Dette inkluderer gode muligheter til å koble til andre ledd både mot albue og mot fingerledd. Ulempen er liten fleksibilitet siden det er vanskelig å lage et rør som passer godt til flere operatører samtidig. Til venstre er et noe enklere design, hvor rammen består av to enkle metallskinner festet sammen med et ledd. Underarmen festes til to fleksible klemmer langs den lengste metallskinnen, og hånden festes ved at operatøren holder rundt håndtaket festet til den korteste metallskinnen. Dette gir en mer fleksibel og lettere ramme som til dels kan tilpasses flere operatører. Ulempen er at komponenter bare kan festes på de to metallskinnene.

3.4.2 Aktuering av ledd

I denne oppgaven har det i hovedsak blitt sett på elektrisk aktuering av eksoskjelettet. Et annet alternativ ville vært pneumatisk aktuering. Dette har blitt brukt suksessfullt i

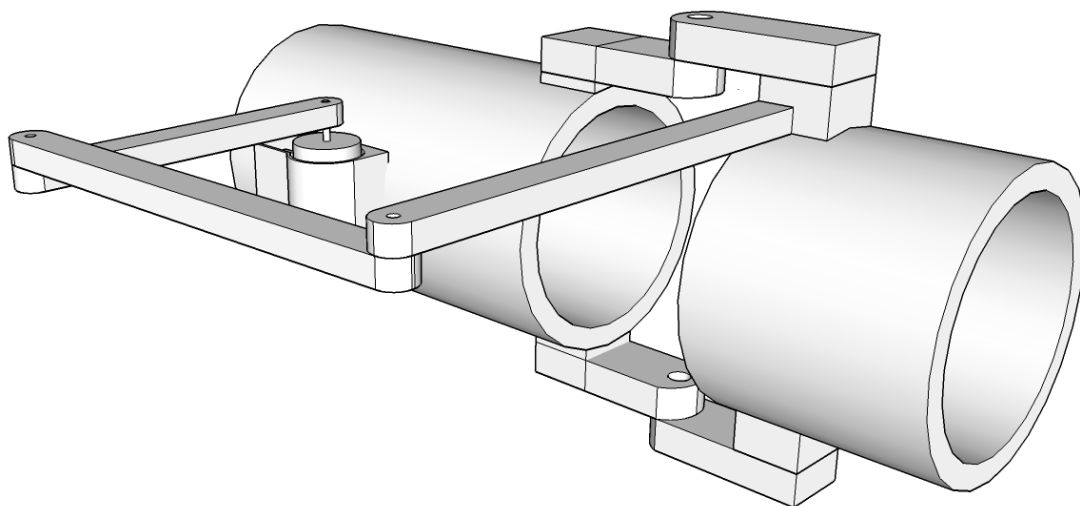
flere andre eksoskjelett før. En stor fordel med pneumatisk aktivering er muligheten for forholdsvis stor kraft uten at man trenger tunge komponenter festet direkte på eksoskjelettet. For en prototype er det for eksempel mulig å la en ekstern kompressor tilføre trykkluft som reguleres av elektrisk styrte ventiler på eksoskjelettet. For enkelhets skyld ble det likevel bestemt å benytte elektriske aktuatorer siden dette gir en mer mobil prototype som ikke er avhengig av andre tilkoblinger enn en strømforsyning. Det er dessuten mindre komplisert og mer forutsigbart å styre en elektrisk motor enn å åpne og lukke trykkventiler.

Hovedutfordringen for valg av elektrisk motor er å balansere størrelse og vekt på motoren med hvor mye dreiemoment den kan levere. En kjapp måling på eget håndledd viste at et dreiemoment på 3-5 Nm vil gi en forholdsvis grei motstand uten at motoren i seg selv er i stand til å gjøre mye skade på operatøren. Problemet er at elektriske motorer som kan levere dreiemoment i denne størrelsesordenen er for store og tunge til å bli plasserte på håndleddet til en operatør. Alternativene som ble sett på veide mellom 800 til 1200 gram og hadde et volum som gjorde det lite praktisk å montere motoren på utsiden av eksoskjelettet.

Det ble derfor bestemt å heller bruke en vanlig servo-motor til å drive leddet i eksoskjelettet. Dreiemomentet til en vanlig servo ligger godt under 3-5 N-m, så dette innebærer at motstanden operatøren vil oppleve i eksoskjelettet er såpass svak at det er fullt mulig å overstyre den med vanlig muskelkraft. Samtidig vil motstanden være såpass merkbar at det skal være mulig å demonstrere at systemet fungerer. Fordelen med å bruke en vanlig servo er at forholdet mellom størrelse og maksimalt dreiemoment er mer gunstig enn de fleste andre motorer. Dette kommer av at den elektriske motoren inne i servoen gires betraktelig ned. Ulempen er at motstand fra operatøren vil gi en ganske stor belastning på tannhjulene inne i servoen. For å minske sjansen for å ødelegge servoen, er det derfor ønskelig å lage en kobling mellom servo og håndledd som vil gi etter ved større påkjenninger slik at man unngår skade på tannhjulene.

Det har blitt sett på tre forskjellige metoder for aktivering av leddet til eksoskjelettet ved hjelp av en elektrisk motor. Alle tre innebærer bruk av en servo. Første alternativ ble tenkt ut i løpet av forprosjektet før jul. Dette forslaget går ut på å koble servoen til håndleddet via et to-leddet stag som vist på figur 13. På denne måten får man en mulighet til å endre girforholdet mellom servoen og håndleddet ved å endre lengden på stagen. For eksempel vil et kort stag fra servoen og et langt stag fra håndleddet føre til at en vinkelendring i motoren vil tilsvare en mindre vinkelendring i håndleddet og derfor gi opp dreiemomentet fra servoen. Som figuren viser er det nødvendig å bruke eksoskjelett-rammen som omslutter hele armen for å feste servoen. En annen fordel med dette oppsettet er at strekklapper kan festes på ett av stagen for en temmelig ukomplisert måling av hvilke krefter dette utsettes for. Ut fra denne målingen er det mulig å beregne dreiemomentet. Samtidig innfører stagen ekstra kompleksitet til systemet. For det første legges det til ekstra friksjon i leddene mellom stagen, i tillegg må det implementeres ekstra utregninger i styresystemet for å få en lineær måling og styring av håndleddet. Dette kommer av at forskjell mellom lengde på staget fra motor og staget fra håndleddet vil føre til at forholdet mellom farten til servoen og leddet

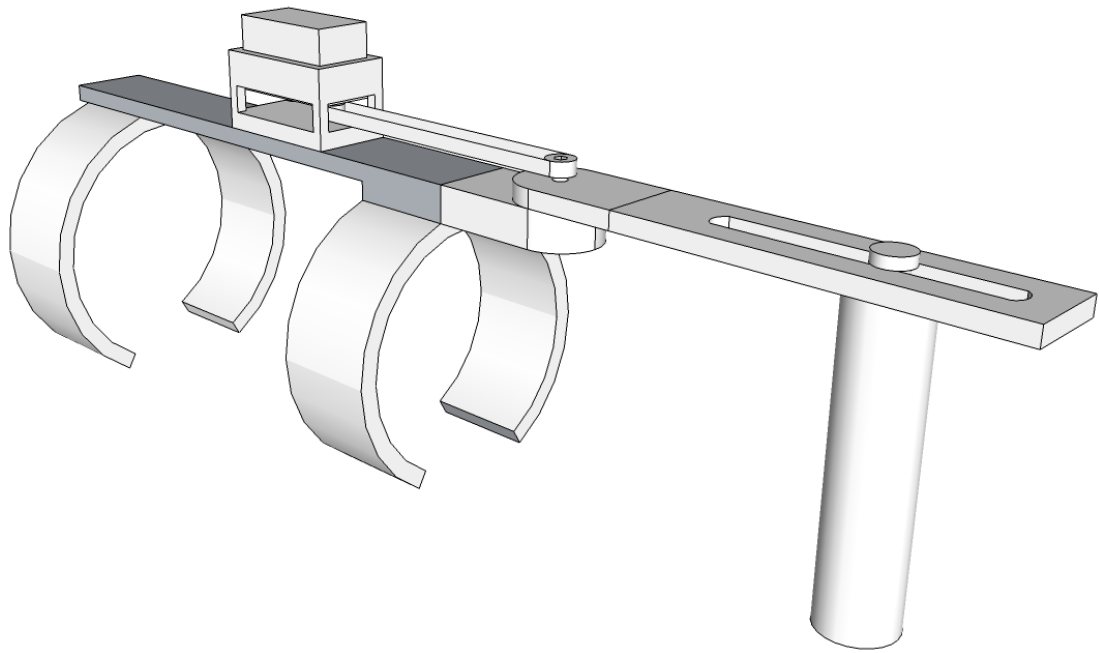
vil være en trigonometrisk funksjon. Dette kan unngås ved å ha like lengder på stagene, slik at de til sammen representerer tre sider i et parallelogram, men da forsvinner samtidig fordelene med å gire opp kreftene fra servoen.



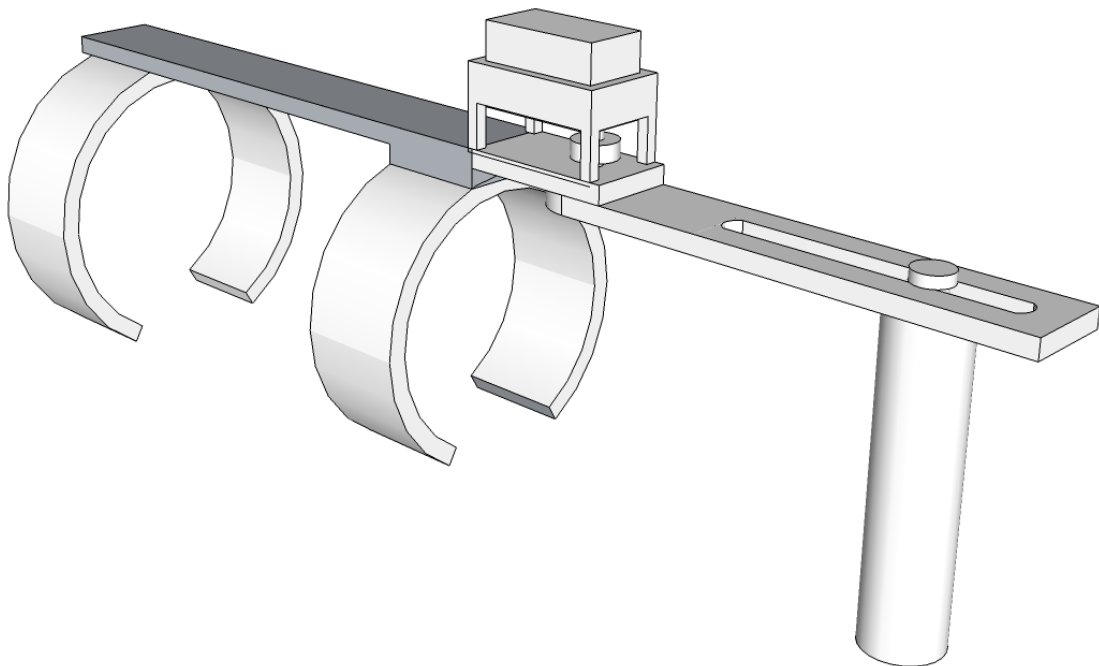
Figur 13: Prototype hvor leddet styres av stag

Den andre foreslåtte løsningen går ut på å koble servoen til håndleddet via en reim. Den mest åpenbare fordelen med denne løsningen er at servoen kan festes på en enklere ramme som den som kan sees i figur 14 siden servoen nå kan stå direkte bak leddet. I tillegg er det like gode muligheter til å skalere kreftene fra servoen til håndleddet ved å ha forskjellige størrelser på reimhjulene. Denne løsningen vil dessuten gi et proporsjonalt forhold mellom vinkelfarten til servoen og håndleddet. Reimdrift er i seg selv ikke den enkleste formen for tilkobling, men koblingen mellom reimhjul og reim kan fungere som en sikkerhet mot at servoen er svakere enn styrken i håndleddet til operatøren: Ved å stramme eller å slakke reimen kan man bestemme hvor mye kraft som skal til før denne slipper taket. Dette kan redusere risikoen for skade på servoen.

Den tredje løsningen går ut på å koble servoen direkte på leddet som vist i figur 15. For å unngå skade på servoen, må denne koblingen konstrueres på en slik måte at den gir etter ved for store krefter. En slik kobling er en noe større utfordring enn å bare stramme og slakke reimer, men er til gjengjeld kompakt og tar desidert mindre plass enn de to andre løsningene.



Figur 14: Prototype hvor motor er plassert bak leddet



Figur 15: Prototype hvor motor er plassert direkte på leddet

3.4.3 Måling av vinkelposisjon

Som nevnt over vil det bli nødvendig med en kobling mellom motor og eksoskjelett som slipper taket dersom dreiemomentet blir for stort. Dette innebærer igjen at servoen som

skal styre eksoskjelettet ikke kan være posisjonsstyrt siden denne posisjonen blir feil med en gang koblingen gir etter. Servoen som skal brukes, må derfor modifiseres slik at den kan rotere kontinuerlig. Dette betyr at servoen skrues opp, potensiometeret fjernes og rotasjons-hindringer skjæres bort. Alternativt er det mulig å bestille servoer som er fartsstyrte direkte fra fabrikk. Utfordringen blir hvordan servoen så skal styres. Dersom PWM-signalet sendes som vanlig, får man en åpen sløyfe-regulering og i praksis en fartsregulering hvor pådraget går meget fort i metning (dette kommer av at regulatoren er tunet for posisjonsregulering og ikke fart). Det er derfor godt mulig at det hadde vært mer hensiktsmessig å koble seg direkte på DC-motoren i servoen og styre denne med en H-bro og på den måten holde den innebygde reguleringskretsen utenfor hele systemet. Det kan sies at med en slik løsning kan man like gjerne bruke en vanlig DC-motor. Likevel er fordelene, som nevnt tidligere, at ratioen mellom størrelse og dreiemoment til en servo er svært gunstig sammenlignet med de fleste vanlige DC-motorer.

Uansett betyr dette at posisjonsmåleren til servoen ikke kan brukes og at det må monteres en egen, ekstern posisjonsmåler. To mulige alternativer ble sett på. Det første alternativet var et vanlig potensiometer hvor posisjonen leses ut av endringen i motstanden. Det andre alternativet var en magnetisk posisjonsmåler. Med denne sensoren festes først en magnet på den roterende delen, mens et kretskort med en påmontert elektromagnet plasseres slik at den faste magneten kan rotere fritt inntil kortet. Ved å lese av elektromagneten, kan kretskortet så bestemme hvordan den faste magneten roterer. Siden det ikke var ferdige komponenter til den magnetiske posisjonsmåleren tilgjengelig, ble det til slutt bestemt å bruke et potensiometer.

Hvordan potensiometeret kan festes er avhengig av hvilken aktuatorløsning som blir brukt. For de to første er det mulig å feste potensiometeret direkte på leddet siden servoen ikke er direkte festet på leddet. For den siste løsningen blir det derimot vanskelig å få plass til både servo og potensiometer på samme plass. Løsningen som til slutt ble valgt var å plassere potensiometeret bak servoen deretter å strekke en reim fra leddet bak til posisjonsmåleren. Siden koblingen mellom servo og ledd kan slippe, er det da viktig at festet som reimen fra potensiometeret går til, roterer med hånden til operatøren og ikke servoen.

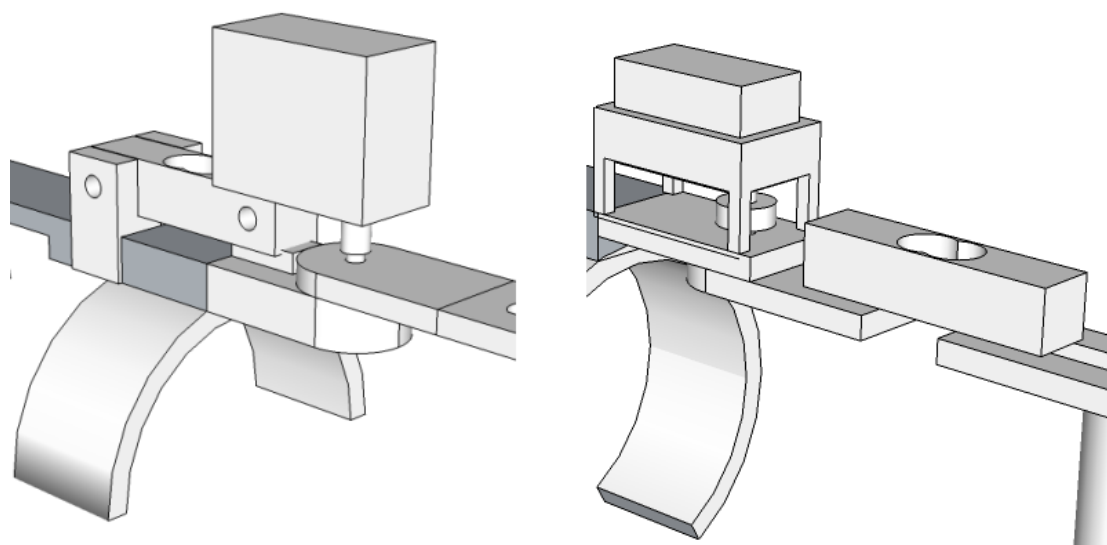
3.4.4 Måling av dreiemoment i håndledd

Det har blitt sett på flere forskjellige løsninger for hvordan man kan måle dreiemomentet i håndleddet. Fra forprosjektet ble det foreslått en løsning hvor motoren plasseres i en hylse hvor dreiemomentet motoren utsettes for måles. Dette konseptet var ikke særlig utprøvd og det ble bestemt å gå vekk fra dette for å unngå å bruke mye tid på noe som ikke nødvendigvis ville fungere. Det ble derfor sett nærmere på muligheten for å plassere strekkklapper på gunstige punkt på eksoskjelettet. For eksempel var en mulighet å plassere strekkklapper på ett av stagen i figur 13. Etterhvert ble det bestemt å heller gå for en kommersiell komponent: En lastcelle, nærmere beskrevet i avsnitt 4.1.3, med mulighet til

å måle krefter langs samme akse i begge retninger. Målingen utføres da ved å måle krefter og så regne ut dreiemomentet ut fra hvor langt påvirkningspunktet til denne kraften er fra leddet. Utfordringen ligger i hvor lastcellen skal plasseres.

De to mest aktuelle alternativene er å enten måle dreiemomentet mellom selve servoen og metallskinnen den er festet, eller å måle kraften mellom håndtaket som operatøren holder i og leddet som er festet til servoen. Skisser av begge konsepter kan sees i figur 16. Til venstre festes servoen direkte på lastcellen og er ellers ikke i kontakt med eksoskjelettet annet enn i koblingen til leddet. På denne måten vil rotasjonskraften som kommer av at servoen prøver å rotere leddet kun forplante seg til lastcellen som så måler denne kraften. Derfra kan så dreiemomentet regnes ut.

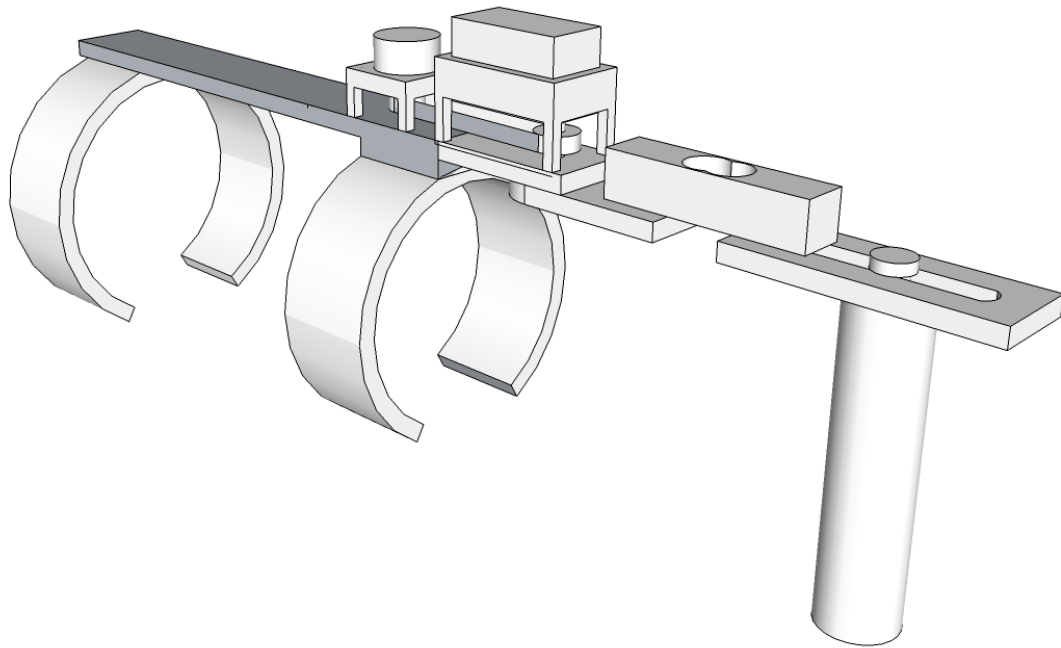
Til høyre i figur 16 er derimot lastcellen montert mellom leddet og håndtaket som operatøren holder i. Kraften som blir målt vil derfor være differansen mellom motorkraften og kraften som operatøren påsetter eksoskjelettet. Det ble etter hvert klart at det rent konstruksjonsmessig var mer hensiktsmessig å velge denne løsningen siden dette ga mer plass til å plassere både servo og potensiometer nærme leddet.



Figur 16: Forskjellige plasseringer av lastcelle

3.5 Endelig utforming av prototype

Det er ikke noe klart svar på hvilket alternativ som vil fungere best av de som er nevnt før i kapittelet. For denne oppgavens del er det viktigste at prototypen fungerer bra nok til at det vises at konseptet med å styre et slangerobotledd med et eksoskjelett er realiserbart. Siden dette er den første protypen uten at praktiske forsøk har blitt gjennomført tidligere,



Figur 17: Ferdig design av eksoskjelettet

er det dessuten realistisk å forvente at ikke alt vil fungere som det skal, og at det vil være nødvendig med justeringer av designet også etter at eksoskjelettet er ferdig konstruert. Dette taler for å styre unna de mer kompliserte alternativene, siden kompleksitet og fleksibilitet ikke nødvendigvis går hånd i hånd.

Ta for eksempel designet som ble utarbeidet i forprosjektet, hvor håndleddet styres av stag koblet til servoen (se figur 13). Selv om dette designet gir mulighet til å endre forholdet mellom dreiemomentet rundt motoren og dreiemomentet rundt håndleddet, tilfører stagene betydelig mer kompleksitet. Å tilpasse lengden på hvert stag, finne koblinger som gir tilfredsstillende lite friksjon, samt sørge for at konstruksjonen kan tåle påkjenningene, er ikke gunstig for et prosjekt med begrenset tid.

Derfor ble det tidlig bestemt å gå for en av de nye løsningene hvor koblingen mellom motor og ledd var betydelig enklere. Dette betydde at det stod mellom å koble motoren direkte på leddet eller plassere den lengre bak på metallskinnen og la den drive leddet med reimdrift. Samtidig var det også viktig å få plass til både potensiometer og lastcelle. Som nevnt før i kapitlet ble en ferdig lastcelle valgt fremfor å selv sette strekkklapper på konstruksjonen. Selv om en utprøvd, kommersiell komponent er sikrere, tar den samtidig mer plass enn strekkklapper som kunne festes direkte på eksoskjelettet. Som vist i figur 16 ble det vurdert to mulige plasseringer av lastcellen. Etter rådføring med ITK-verkstedet ble det klart at det enkleste alternativet var å feste lastcellen mellom leddet og håndtaket

på eksoskjelettet (som vist til høyre på figur 16). Dette ga mer frihet til konstruksjonen av festet til motoren enn dersom lastcellen skulle være eneste kontaktpunkt mellom motor og eksoskjelett. Ulempen er at lastcellen må bære vekten til ytterste del av eksoskjelettet.

Med lastcellen plassert var spørsmålet så hvordan motoren og potensimeter skulle festes. Etter ny rådføring med ITK-verkstedet ble det klart at den lettest gjennomførbare metoden var å feste motoren direkte på leddet. Dette innebar, som nevnt tidligere, å lage en kobling som ville gi etter dersom kreftene mellom leddet og motoren ble for store. Videre ble så potensimeteret plassert bak motoren med en reim festet til denne koblingen, siden det ikke var plass til å feste både motor og potensimeter på samme sted.

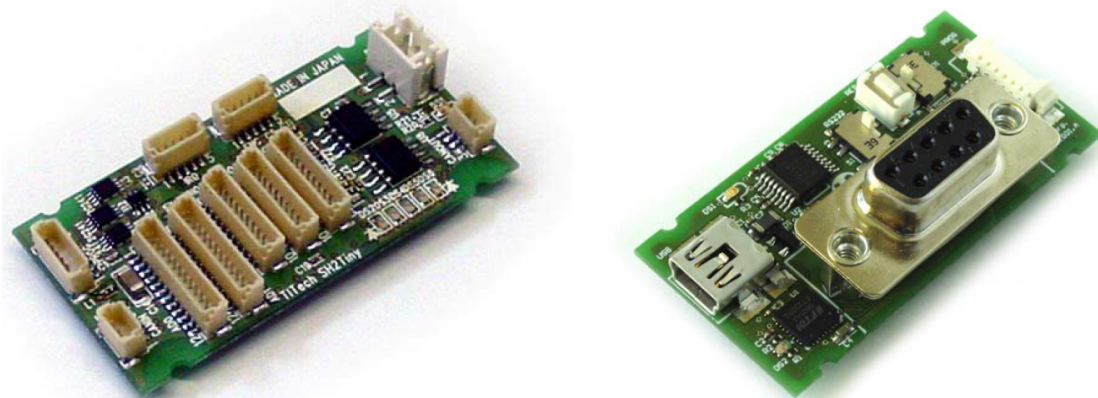
4 Realisering av eksoskjelett

I dette avsnittet gjennomgås valg av de viktigste komponentene i eksoskjelettet, og hvordan disse skal kobles sammen og samarbeide. I tillegg beskrives oppbygningen og funksjonaliteten til programmet som kjører på mikrokontrolleren.

4.1 Valg av komponenter

4.1.1 Mikrokontroller

Til å styre de elektroniske komponentene i eksoskjelettet benyttes en *TITechSH2 Tiny Controller* fra HiBot. Dette er den samme mikrokontrolleren som brukes i slangerobotten som skal styres. Prosessoren på brettet er en Renesas 32-bit RISC mikroprosessor, SH7047. Kortet har åtte ADC-kanaler samt en 22-bit digital I/O-port. I tillegg er det også tilkoblingsmuligheter for CAN-kommunikasjon. Mikroprosessoren programmeres via en seriell programmerer, *TITechSH2 Universal Serial Programmer (USP)*, levert av samme selskap som leverer mikrokontrolleren. Denne gir også muligheten for seriell kommunikasjon til datamaskinen, enten via RS-232 eller via virtuell COM-port gjennom USB. Selv om designet av eksoskjelettet ikke krever at data sendes til en ekstern datamaskin, er det likevel gunstig å ha muligheten for å sende feilmeldinger og måleverdier slik at disse kan leses av direkte på en skjerm. Dette er for eksempel helt nødvendig når lastcellen skal kalibreres. Når testfasen er avsluttet regnes det med at USP-en ikke lengre trenger å være en del av systemet.



Figur 18: Til venstre: TITechSH2 Tiny Controller. Til høyre: TITechSH2 Universal Serial Programmer (USP) (Bilde hentet fra produktets datablad)

SH7047 har en innebygd 10-bits A/D-omformer med 16 kanaler. En konvertering skjer på 6,7 mikrosekund per kanal. For eksoskjelettets formål er dette mer en nok til å få

jevn avlesning av lastcellen. A/D-modulen har støtte for både kontinuerlig skanning av hver kanal og enkeltskanninger styrt av programkoden. For styring av eksoskjelettet vil programmet selv starte konverteringen når det er behov for nye måleverdier. Dette vil gi bedre kontroll med hvor mye av mikrokontrollerens ressurser brukes på konverteringen.

4.1.2 Servo

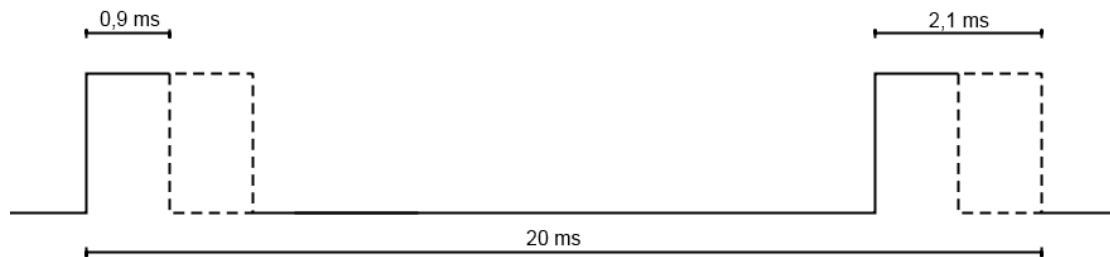
Servoen som benyttes til styring av eksoskjelett-leddet er en HITEC HS-5955TG – Ultra Torque Servo. Ved kjøring på 6 V har denne servoen et dreiemoment på 235 N-cm. I tillegg er gir-systemet av titan i stedet for plast, noe som gjør belastningsskader mindre sannsynlig sammenlignet med en svakere servo med plastgir. Fra fabrikk er denne servoen posisjonsstyrt, men er modifisert til å bli fartsstyrt.



Figur 19: HITEC HS-5955TG – Ultra Torque Servo (Bilde hentet fra produsentens hjemmeside)

En vanlig posisjonsstyrt servo styres ved hjelp av et puls-bredde modulasjon-signal (den engelske forkortningen *PWM-signal* benyttes videre i teksten) som kan sees i figur 20. PWM-signalet har en fast periode på 20 ms, og en pulsbredde som ligger mellom 0,9 ms og 2,1 ms. Mikrokontrolleren som ligger inne i servoen tolker dette signalet til hvilken vinkelposisjon brukeren ønsker, hvor 0.9 ms og 2,1 ms tilsvarer de to ytterste vinklene som servoen kan oppnå. Ut fra et potensiometer inne i servoen som gir posisjonstilbakekobling reguleres så servoen til ønsket vinkel.

Når servoen så blir modifisert til å styre fart, blir denne operasjonen litt annerledes. For det første fjernes posisjonstilbakekoblingen fra servoen, og resistansen fra denne blir i stedet konstant. I tillegg blir eventuelle fysiske barrierer som skal motvirke rotasjon utenfor det bestemte posisjonsområdet fjernet slik at servoen kan rotere fritt. PWM-signalet tolkes nå også annerledes av mikrokontrolleren inne i servoen. Siden potensiometeret nå har en konstant resistans vil servoen bare stå stille dersom et PWM-signal som samsvarer med denne verdien blir sendt til servoen.



Figur 20: PWM-signal for vanlig posisjonsstyring av servo

For å finne denne verdien ble det brukt en *HITEC Digital Servo Programmer og Servo Tester, HFP-10* (se figur 21). Denne gir en enkel mulighet til å variere PWM-signalet inn på servoen. Ved testing ble det funnet at servoen står stille dersom den mottar PWM-signaler med bredde fra $1509 \mu s$ til $1517 \mu s$. Utenfor dette området begynner servoen å bevege seg. Altså ligger ønsket midtpunkt for PWM-signalets pulsbredde på $1513 \mu s$. Maksimalfarten som servoen kan oppnå oppstår når pådraget til regulatoren inne i servoen når sitt metningspunkt. Dette vil skje lenge før PWM-signalet når en bredde på $0,9 \text{ ms}$ eller $2,1 \text{ ms}$. Ved måling ble det observert at servoen nådde maksimalfart ved pulsbredder som lå ca. $50 \mu s$ under eller over $1513 \mu s$.

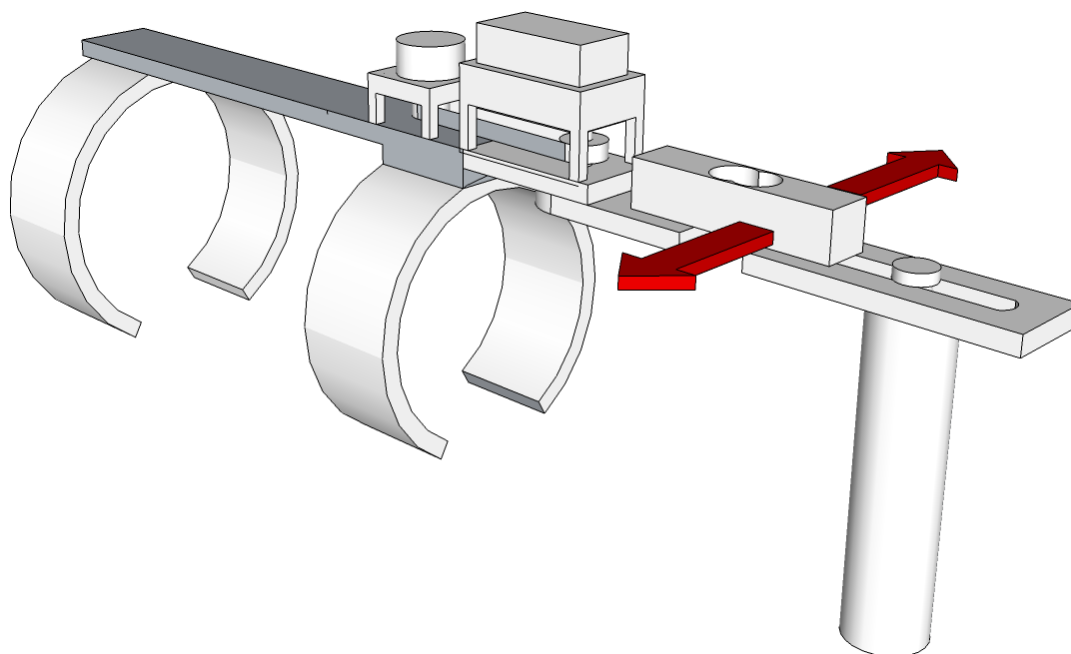


Figur 21: HITEC Digital Servo Programmer & Servo Tester, Model: HFP-10 (Bilde hentet fra produsentens hjemmeside)

4.1.3 Lastcelle

Som nevnt før ble det bestemt å benytte en kommersiell komponent for måling av kreftene som håndleddet til eksoskjelettet utsettes for. Alternativet var å selv plassere strekkklapper direkte på eksoskjelettet. Så lenge det er mulig å finne en kommersiell komponent som kan utføre samme måling, er det derimot lite poeng i å risikere å bruke noe selvprodusert. Fra det endelige designet ble det bestemt at dreiemomentet skulle måles ut fra kreftene som ytterste metallskinne på eksoskjelettet utsettes for. Altså skinnen som ligger mellom

leddet og håndtaket som operatøren holder i. For å kunne måle dreiemomentet trengs en kraftmåling av krefter som ligger vinkelrett på både metallskinnen og rotasjonsaksen til leddet (se figur 22).

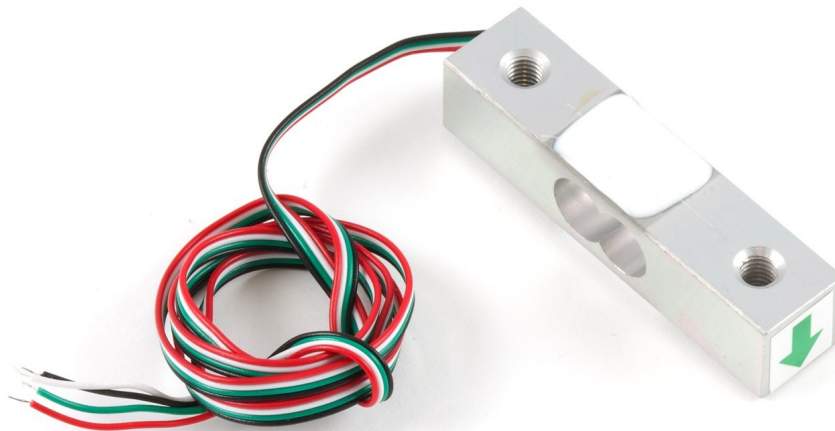


Figur 22: Retning på krefter som måles av lastcellen

Kreftene som skal måles er begrenset til hvor sterk operatøren er og hvor stor kraft koblingen mellom servo og ledd tåler før den gir etter. Ut fra målinger var det klart at en operatør som tar i, fint kan oppnå en kraft på 5 kg med en arm på 10 cm fra eget håndledd. En slik størrelse vil uansett være for stor til at servoen i eksoskjelettet kan holde igjen, og det er derfor ikke noe poeng å måle krefter større enn dette. En vanlig lastcelle som brukes i kjøkken- eller smykkevekter er egnet til en slik måling. I dette tilfellet ble det funnet en lastcelle av typen CZL635.

Denne kan måle en belastning opp til 5 kg i begge retninger. Som man ser på bildet består lastcellen av en aluminiumbjelke med to sammenhengende hull i midten. Disse to hullene svekker bjelken nok til at man kan måle strekket i de tynne sidene. Målingen skjer ved at en strekkklapp er festet til hver side av bjelken. I tillegg er to strekkklapper festet vinkelrett på strekkretningen. Disse fire strekkklappene kobles sammen i en *wheatstone bridge*.

Rød og svart kabel kobles så til henholdsvis 3,3V og jord. Dersom lastcellen ikke er belastet, vil motstandene i hver strekkklapp være omtrentlig lik, og spenningsforskjellen mellom hvit og grønn utgang vil være lik 0 V. Når så lastcellen belastes, vil strekkklappene som er festet langs bjelken, endre motstand, og man vil få en spenningsforskjell mellom hvitt og grønt målepunkt. Denne forskjellen vil være proporsjonal med kraften som påvirker lastcellen. Ut



Figur 23: Lastcelle, CZL635 (Bilde hentet fra produsentens hjemmeside)

fra databladet til lastcellen kan man lese at utgangsspenningen fra grønn og hvit ledning har en maksverdi på $1 \frac{mV}{V}$. Dette betyr at man kan forvente en utgangsspenning på $3,3 mV$ dersom man belaster cellen med $5 kg$ i en retning, og $-3,3 mV$ dersom belastningen skjer i den andre retningen. Produsenten spesifiserer at dette ikke nødvendigvis er nøyaktige verdier, og det vil eksistere forskjeller fra lastcelle til lastcelle.

For å sikre seg gode målinger ble det derfor laget et testoppsett hvor man kunne teste lastcellen nøyaktighet, samt å finne en forsterkning av signalet som gjorde det mulig å måle krefter i en tilfredsstillende oppløsning. Som vist på figur 24 ble lastcellen festet til en bordplate med en skrustikke, og laster på 1, 3, 4 og 5 kg ble påført i begge retninger. Signalet fra lastcellen ble forsterket opp av kretsen sett i bildet under.

Siden målebroen balanseres på 0 V når lastcellen ikke er utsatt for belastninger vil signalet ut kunne være både negativt eller positivt ladet. Siden operasjonsforsterkeren skal fungere på en strømforsyning på 3,3 V er det derfor nødvendig legge til en offset. Offset-spenningen tilpasses slik at utgangsspenningen til forsterkeren ligger midt mellom metningsverdiene når lastcellen ikke har noen belastning. På denne måten unngås det at operasjonsforsterkeren går i metning når spenningen fra wheatstone broen har negativ spenningsforskjell. Mikrokontrollerens ADC ble så brukt til å måle utgangsspenningen fra differensialforsterkeren og målte verdier ble så sendt til en LabVIEW-applikasjon hvor målingene ble sammenlignet med faktisk vekt på lastcellen.

Operasjonsforsterkeren *LM108A* ble brukt i oppsettet i figur 25. Med en forsyningspenning på 3,3 V, gir denne en utgangsspenning på mellom 0,55 V og 2,73 V. For den 10-bits analog-til-digital-omformerens tilsvarende dette en registerverdi på henholdsvis 168 og 845.



Figur 24: Oppsett for test av lastcelle

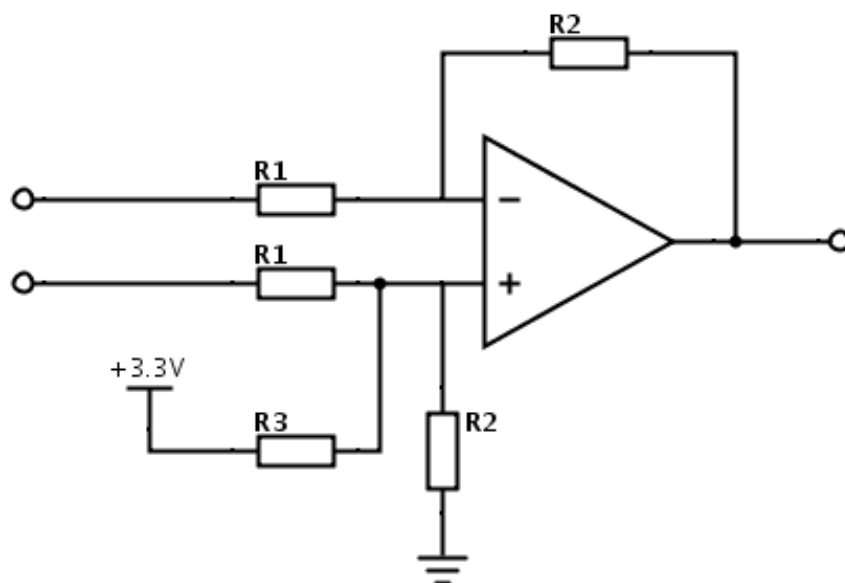
Forskjellige oppsett ble prøvd ut og til slutt ble det bestemt at en forsterkning på $\frac{1 M\Omega}{3,2 k\Omega}$ ga tilfredsstillende resultat. Komponentverdiene var da $R_1 = 3,2 k\Omega$, $R_2 = 1 M\Omega$ og $R_3 = 1,2 M\Omega$. Om man da kaller spenningsforskjellen mellom målepunktene i wheatstone-broen for V_{inn} blir forsterkningen V_{ut} slik:

$$V_{ut} = 240V_{inn} + \frac{3,3 V}{2,2} \quad (7)$$

Måleresultatene sees under:

Måleretning	0 kg	1 kg	3 kg	4 kg	5 kg
Negativ retning	494	436	317	261	207
Positiv retning	494	553	672	730	784

Ut fra denne tabellen ser man at en gjennomsnittlig endring i ADC-registeret er på 57,3 per påtrykt kg. Forholdet mellom verdien i ADC-registeret og faktisk vekt målt i kg blir derfor: $Vekt = \frac{registerverdi-494}{57,3}$. Med et slikt oppsett har man en oppløsning på 17 gram per verdiendring i ADC-registeret. En slik nøyaktighet er mer enn bra nok til formålet dette skal brukes til.



Figur 25: Oppkobling, differensialforsterker

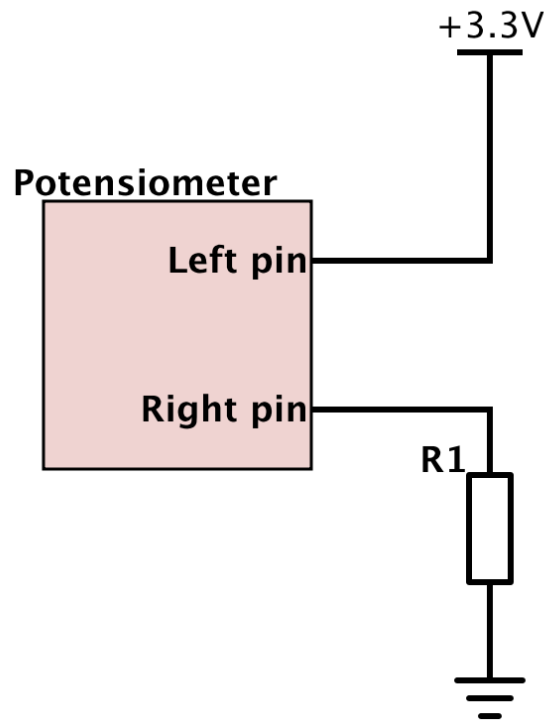
4.1.4 Potensiometer

For å måle vinkelen på håndleddet til eksoskjelettet ble det bestemt å bruke et vanlig potensiometer. Komponenten som ble valgt er Bourns 6537S-1-103. Dette lineære potensiometeret roterer fritt og har en variabel motstand som ligger mellom 0Ω og $10 k\Omega$. Endringen i motstand skjer med en oppløsning på $33 \Omega/grad$. For å lese av motstanden i potensiometeret kobles det sammen med en motstand i en spenningsdeler som sett i figur 26.

Ved å måle spenningen over motstanden R_1 kan man beregne hva resistansen i potensiometeret ligger på. Dersom man kaller motstanden i potensiometeret R_p , spenningen over kretsen for V og spenningen over R_1 for V_{R_1} , så har man at:

$$R_p = R_1 \left(\frac{V}{V_{R_1}} - 1 \right) \quad (8)$$

Motstanden R_1 er kjent og satt til 5630Ω . Det samme er V som er på $3,3 V$. V_{R_1} måles av A/D-konverteren til mikrokontrolleren. Ut fra dette kan altså resistansen i potensiometeret beregnes. I programmet vil det så være nødvendig å måle resistansen til potensiometeret når eksoskjelettet står i en kjent posisjon. Det mest naturlige for dette er å se hvilken verdi som tilsvarer en vinkel på null grader. Det vil si at håndleddet peker rett frem. Utfra dette referansepunktet kan så andre vinkler beregnes utfra oppløsningen på $33 \Omega/grader$.



Figur 26: Oppkobling for potensiometer

4.2 Anskaffelse, konstruksjon og montering av komponenter

Som nevnt i oppgaveteksten har det vært et mål å kunne realisere eksoskjelettet med flest mulige kommersielle komponenter og bare bruke spesiallagde moduler hvor dette er høyst nødvendig. I tillegg har det vært ønskelig å benytte seg av komponenter som allerede var tilgjengelig på ITK-verkstedet. Mikrokontroller og servo er lånt fra samme SINTEF-prosjekt som slangeroboten, og potensiometeret som benyttes ble tatt fra et eksisterende lager på ITK-verkstedet. Dette gjelder dessuten alle elektroniske komponenter som er brukt sammen med lastcellen og potensiometeret. Lastcellen er derimot bestilt fra USA gjennom nettbutikken *RobotShop.com*.

Rammen til eksoskjelettet (se venstre bilde i figur 12) og akslingskoblingen mellom servo og leddet er de eneste komponentene som er spesiallagde. Disse to komponentene er såpass spesifikke til denne oppgaven at det ikke var hensiktsmessig å finne kommersielle alternativer. All montering av komponenter ble utført av ITK-verkstedet.

4.3 Programkode

4.3.1 Introduksjon

Programmet som styrer eksoskjelettet er skrevet i C. Utviklingsprogrammet som er benyttet er *High-performance Embedded Workshop 4*. Koden er modulbasert for å gjøre koden mer oversiktlig og enklere å modifisere. I dette avsnittet presenteres hver moduls funksjonalitet samt hvordan modulene samarbeider.

4.3.2 Overordnet funksjonalitet

Mikrokontrolleren, og derfor programmet, som styrer eksoskjelettet har disse hovedoppgavene:

- Lese av målinger fra potensiometer og lastcelle.
- Styre servoen som er tilkoblet leddet til eksoskjelettet.
- Kommunisere med mikrokontrolleren i slangerobotleddet.
- Formidle relevant informasjon til brukeren ved å sende data til en ekstern datamaskin.

For den første oppgaven må programmet lese av de to A/D-konverter-kanalene hvor potensiometeret og lastcellen er tilkoblet. Dette gjøres med en samplingsfrekvens på 50 Hz. Verdien som leses av fra A/D-konverteren kommer fra et 10-bits register, og ligger derfor mellom 0 og 1023. Siden denne verdien er lineært proporsjonal med spenningen som måles, er det ikke nødvendig å regne ut hvilken spenning som kommer ut fra lastcellen. Dette kommer av at spenningen utfra lastcellen også er lineært proporsjonal til kraften som måles av sensoren. Derfor benyttes bare resultatet fra målingene i avsnitt 4.1.3 for å regne om fra registerverdi til kraft målt av lastcellen.

For potensiometeret er derimot registerverdien ikke lineært proporsjonal med vinkelposisjonen. Dette kommer av at spenningen over motstanden i figur 26 ikke er lineært avhengig av motstanden i potensiometeret (som vist i (8)). Altså må programmet først konvertere registerverdien til spenning før denne så settes inn i (8) for å få verdien på motstanden i potensiometeret. Denne verdien er lineært proporsjonal med vinkelen på eksoskjelettet og informasjon fra avsnitt 4.1.4 brukes for å konvertere fra resistans til grader.

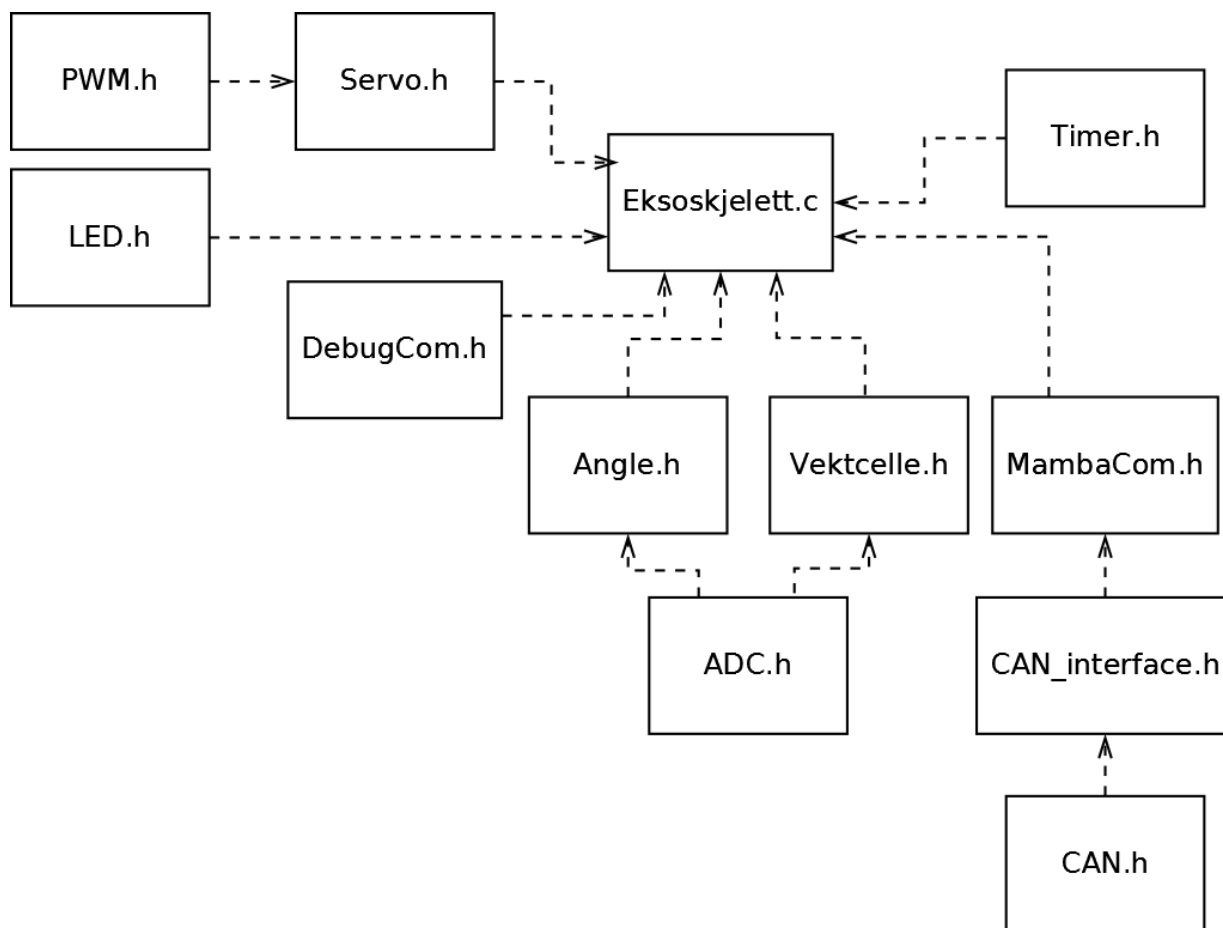
For å styre servoens pådrag brukes et vanlig PWM-signal. Signalet er tilpasset informasjonen fra avsnitt 4.1.2 og kan derfor bare justeres til en pulsbredde mellom 1463 μs og 1563 μs . Videre er det implementert en PID-regulator i programmet. Denne bruker tilbakekobling fra potensiometeret til å styre servoen til ønsket posisjon.

Kommunikasjonen med mikrokontrolleren i slangeroboten skjer over en CAN-buss. Programmet som styrer eksoskjelettet benytter samme programkode for CAN-kommunikasjon som slangeroboten. Kommunikasjonsprotokollen som slangeroboten bruker for å kommunisere mellom hver mikrokontroll gjør det mulig å enkelt be om og sende data og informasjon mellom hver slangerobotledd. For å enkelt kunne koble seg på slangeroboten ble det derfor bestemt å benytte samme protokoll mellom eksoskjelett og slangerobot. Programmet på eksoskjelettet benytter seg bare av et par funksjoner til denne protokollen: Lese av og sette ønsket vinkelposisjon til slangerobotleddet. 50 ganger i sekundet vil programmet etterspørre vinkelposisjonen til slangerobotleddet og sette denne verdien som referanse til PID-regulatoren i eksoskjelettet. Samtidig sendes også den målte vinkelposisjonen til eksoskjelettet som vinkelreferanse til regulatoren på slangeroboten.

Kommunikasjon med ekstern datamaskin skjer over en seriellport som beskrevet i avsnitt 4.1.1. Informasjon sendes 10 ganger i sekundet og har to forskjellige moduser. For den første er informasjonen tiltenkt å leses av direkte med for eksempel *PuTTY* eller *Google Terminal*. For den andre er modusen legges all informasjon i en string som er tilpasset til avlesning fra LabVIEW-applikasjonen.

4.3.3 Moduler

Som nevnt over er koden modulbasert hvor hver modul utfører hver sin oppgave og kommuniserer resultatet av denne oppgaven til andre moduler. I figur 27 vises kommunikasjonsflyten i koden. Under går hver modul gjennom i detalj.



Figur 27: Kommunikasjonsflyt mellom modulene i programmet

Timer.h Først og fremst er oppgaven til denne modulen å gi en tidsvariabel som resten av koden kan forholde seg til. Variablene *csPassed* og *sPassed* inkrementeres henholdsvis hundre ganger og én gang i sekundet. Dette skjer ved hjelp av en *Compare Match Timer (CMT)* som ligger på mikrokontrolleren. Denne består av en 16-bits teller som kan generere avbrudd i satte intervall. I dette tilfellet ble telleren satt til å gi avbrudd med 10 millisekunders mellomrom. Under avbruddet blir så *csPassed* inkrementert hver gang og *sPassed* inkrementert hver gang *csPassed* har økt med hundre. I tillegg blir en tredje variabel, *periodicLoopCounter*, inkrementert hver gang avbruddet kjører. Denne variabelen brukes til å time periodiske oppgaver som utføres av main-løkken (se beskrivelsen av *doPeriodicTask()*)

nedenfor).

Relevante funksjoner:

- void initTimerInt()* - Setter opp CMT på ønsket måte og nullstiller variablene som skal holde orden på tiden.
- void delay_cs(int cs)* - Kjører en while-løkke i antall centisekunder gitt av variabelen cs. Ble mest brukt i debugging av tidlig kode.
- int csSinceLastRunt()* - Returnerer antall centisekund som har passert siden sist funksjonen ble kallet. Brukt mest til debugging.

Vektcelle.h Denne modulen står for konvertering av den avleste spenningen som kommer ut fra lastcellen. Blokken har to konstanter: *VALPERKG* og *WEIGHTADCCHANNEL*. *VALPERKG* er verdien på endringen i spenning inn på analog-til-digitalconverteren som tilsvarer en belastning på 1 kg på lastcellen. Denne verdien er hentet fra forsøk nevnt i avsnitt 4.1.3. *WEIGHTADCCHANNEL* sier hvilken kanal på A/D-konverteren lastcellen er koblet til. I tillegg har blokken en lokal variabel *sensorMidPoint* som inneholder verdien fra A/D-konverteren som regnes som ingen last målt. Altså 0 kg.

Relevante funksjoner:

- int getWeight()* - Returnerer vekten som måles fra lastcellen gitt i gram.
- void setSensorMidPoint()* - Setter den målte vekten som nullpunkt. Altså 0 kg.

Angle.h Denne modulen står for konvertering av den avleste spenningen som kommer ut av potensiometerkretsen og ut fra denne beregne vinkelen på eksoskjelettet. Kretsen er beskrevet i avsnitt 4.1.4. Konstantene *CONSTRESISTOR* og *VOLTAGESUPPLY* har verdien på henholdsvis den faste motstanden i kretsen og spenningen som kretsen kjøres på. I tillegg forteller konstanten *DEGREEPERRESTANCE* hvor mange grader en endring på 1 Ω i potensiometeret tilsvarer og *ANGLEADCCHANNEL* sier hvilken kanal på A/D-konverteren som skal leses av. Variabelen *midAngle* har verdien på motstanden i potensiometeret som tilsvarer en vinkel på null grader.

Relevante funksjoner:

- int getResistorValue()* - Returnerer den beregnede verdien på motstanden i potensiometeret.
- int getAngle()* - Returnerer den beregnede vinkelen på håndleddet til eksoskjelettet. Denne vinkelen er gitt i grader og skal ligge mellom -90° og 90° .
- void setAngleMidPoint()* - Setter den målte posisjonen til null grader.

PWM.h Her initialiseres puls-bredde modulasjonen til mikrokontrolleren. Denne blokken utfører ikke egne beregninger, men utfører bare kommandoer fra andre blokker og fungerer

som et kommunikasjonsledd mellom resten av koden og puls-bredde modulasjonssignalet.

Relevante funksjoner:

- void initPWM()* - Initialiserer puls-bredde modulasjonen og setter PWM-signalet slik at det kan styre roboten. Det vil si med en periode på 20 ms. I tillegg settes bredden på signalet slik at motoren står i ro.
- void setPWMPulseWidth (int microseconds)* - Setter bredden på PWM-signalet til antall mikrosekund definert i variabelen *microseconds*. Funksjonen godtar ikke verdier under 9 ms eller over 2,1 ms og vil runde av verdier som ligger utenfor dette til nærmeste grense.

Servo.h Her styres pådraget og reguleringen av servoen. Siden servoen ikke er posisjonsstyrt består den grunnleggende styringen av å endre farten i valgt endring. Fra denne funksjonaliteten bygges en PID-regulator. Blokken inneholder diverse variabler som er nødvendige for å implementere en PID-regulator: Koeffisienter for hvert regulatorledd og variabler som holder kontroll på ønsket vinkel, reel vinkel og differanse mellom disse. Konstanten *ZEROSPEED_PULSEWIDTH* beskriver pulsbredden på PWM-signalet som sørger for at motoren står stille. *MAX_DEVIATION_FROM_ZEROSPEED* definerer den maksimale farten ved å sette en begrensning på hvor mye PWM-signalet kan avvike fra *ZEROSPEED_PULSEWIDTH*.

Relevante funksjoner:

- void setServoSpeed(int percent)* - Setter farten til 0-100% av maksimumfarten. Fortegnet på variabelen *percent* bestemmer fartsretning.
- void setDesiredAngle(int angle)* - Lagrer verdien i *angle* til variabelen som senere blir brukt av *updatePID()* som vinkelreferanse.
- int getDesiredAngle()* - Returnerer verdien på vinkelen som PID-regulatoren prøver å oppnå.
- int getAngleError()* - Returnerer differansen mellom den ønskede vinkelen og den reelle vinkelen til eksoskjelettet.
- int updatePID()* - Oppdaterer PID-regulatoren. Altså blir pådraget til motoren endret for å oppnå ønsket vinkel.

ADC.h Denne modulen tar seg av initialisering og kjøring av analog/digital-konverteren på mikrokontrolleren. Denne blokken gjør ingen andre beregninger enn å lese av registrene tilhørende hver enkelt A/D-konverterkanal og er derfor bare et kommunikasjonsledd til andre deler av koden (som *Vektcelle.h* og *Angle.h*). Initialiseringsfunksjonen setter opp A/D-konverteren slik at programmet selv må be om en ny skann av avleste spenninger. En lokal konstant *MVPERVALUE* har verdien på forholdet mellom registerverdien og spenningen (gitt i mV) inn på kanalen som leser denne verdien. Denne konstanten er

beregnet for når spenningen inn på ADCC pinnen til A/D-konverteren er 3,3 V.

Relevante funksjoner:

- void initADCmodule(int moduleNumber)* - initialiserer en av de to ADC.modulene.
- void startSingleScan(int moduleNumber)* - starter en konvertering av alle kanaler som tilhører modulen spesifisert av variabelen *moduleNumber*.
- unsigned int readChannel(int channel)* - returnerer verdien på registerverdien til kanalen definert variabelen *channel*.
- unsigned int getmVoltage(int channel)* - returnerer beregnet spenning gitt i mV på kanalen definert av variabelen *channel*.

DebugCom.h Denne modulen setter opp muligheten for å sende og motta informasjon gjennom seriellporten til datamaskinen. Hovedsakelig blir dette brukt til å sende beskjeder om målte verdier og debug-meldinger til datamaskinen samt å styre enkelte deler av koden fra datamaskinen (nullstille moment- posisjonsmålinger). Koden gjør det mulig å sende stringer fra mikrokontrolleren og chars fra datamaskinen. Dette vil si at alle variabler som skal sendes til datamaskinen må konverteres til stringer før de sendes. Som vist under er det implementert funksjoner som gjør dette.

Relevante funksjoner:

- void initDebugCom()* - initialiserer seriell kommunikasjon.
- void pStr(char* s)* - sender stringen *s* over seriellporten.
- void pStrWNum(char* s, int number)* - sender stringen *s* over seriellporten. Integren *number* legges inn i stringen hvor *%d* er plassert. (Samme funksjonalitet som *sprintf()*).
- void sendDataToLabview(int var1, int var2, int var3)* - sender tre variabler i en format som kan tolkes av Labviewapplikasjonen på datamaskinen.
- unsigned char newInputReady()* - Returnerer *TRUE* dersom mikrokontrolleren har mottatt en byte fra datamaskinen.
- unsigned char getInputFromComputer()* - Returnerer byten som er sendt fra datamaskinen.

LED.h Her styres de fem LED-lysene på mikrokontrolleren. Hovedsakelig har disse blitt brukt til debugging og til å indikere hvor i koden programmet befinner seg.

Relevante funksjoner:

- void initLEDs()* - Setter opp portene som går til hver enkel diode som utganger slik at det er mulig å variere hvilken spenning som ligger over hver enkelt LED ved å endre utgangsverdi på portene.
- void setLED(int LEDnumber, int state)* - Setter LED-lyset på lassen tilhørende verdien til *LEDnumber* til en gitt tilstand definert av *state*. De forskjellige tilstandene er *OFF*, *RED*, *GREEN*, *YELLOW*.

CAN.h og CAN_interface.h Disse to modulene består hovedsakelig av kode hentet direkte fra slangeroboten. Ved å bruke samme kommunikasjonsprotokoll over CAN-bussen som slangeroboten er det betydelig enklere å koble sammen eksoskjelettet og robotleddet. Dette betyr også at det ikke er nødvendig å endre koden som ligger på slangeroboten. Dette betyr også at koden som styrer CAN-bussen består av en del funksjoner som ikke er relevant for eksoskjelettet. Slik systemet er nå brukes CAN-kommunikasjonen kun til å sende vinkelposisjon. Nedenfor beskrives de funksjonene som er mest relevante for koden som kjører på eksoskjelettet.

Relevante funksjoner:

- void initCANinterface()* - initialiserer CAN-bussen og oppretter en modul-ID for mikrokontrolleren som andre tilkoblet CAN-bussen må bruke for å kommunisere.
- void CAN_processReceivedMessage(unsigned char priority, unsigned char function, unsigned char datatype, unsigned char sender, unsigned char* data)* - Går gjennom den mottatte meldingen fra CAN-bussen og bestemmer hvor den skal sendes videre utfra hva den inneholder. For eksoskjelettet forventes bare beskjeder som inneholder vinkelposisjonen til robotleddet. Denne funksjonen vil derfor ende opp med å kun sjekke om meldingen faktisk inneholder dette før den sendes videre til *CAN_processReceivedParameters()*.
- void CAN_processReceivedParameters(unsigned char datatype, unsigned char sender, unsigned char* data)* - Henter ut vinkelposisjonen fra CAN-meldingen, konverterer veriden til grader og sender den videre til *MambaCom.h*.

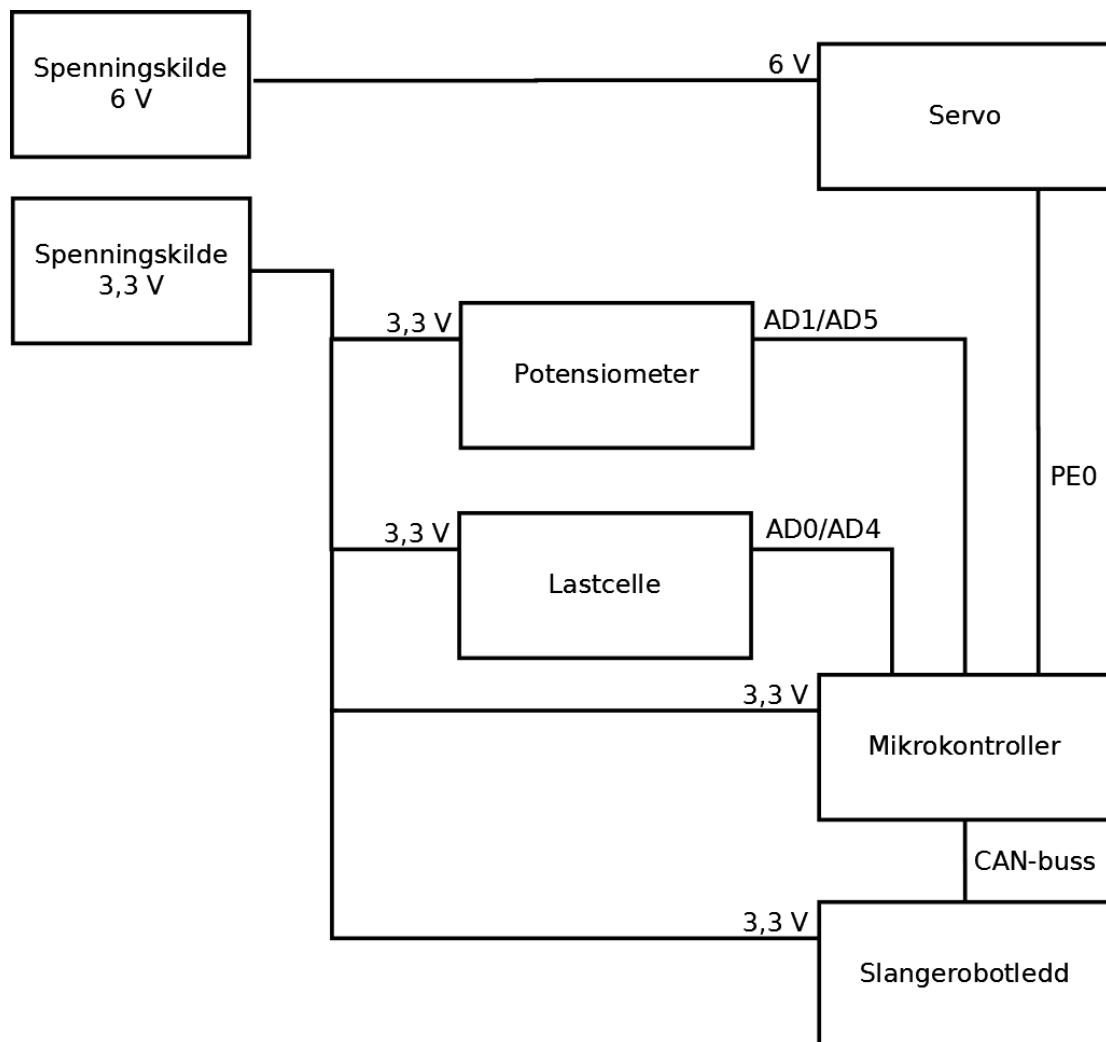
MambaCom.h I denne modulen styres kommunikasjonen med slangeroboten og mottatte målte vinkelposisjoner blir lagret og sendt videre til resten av koden.

Relevante funksjoner:

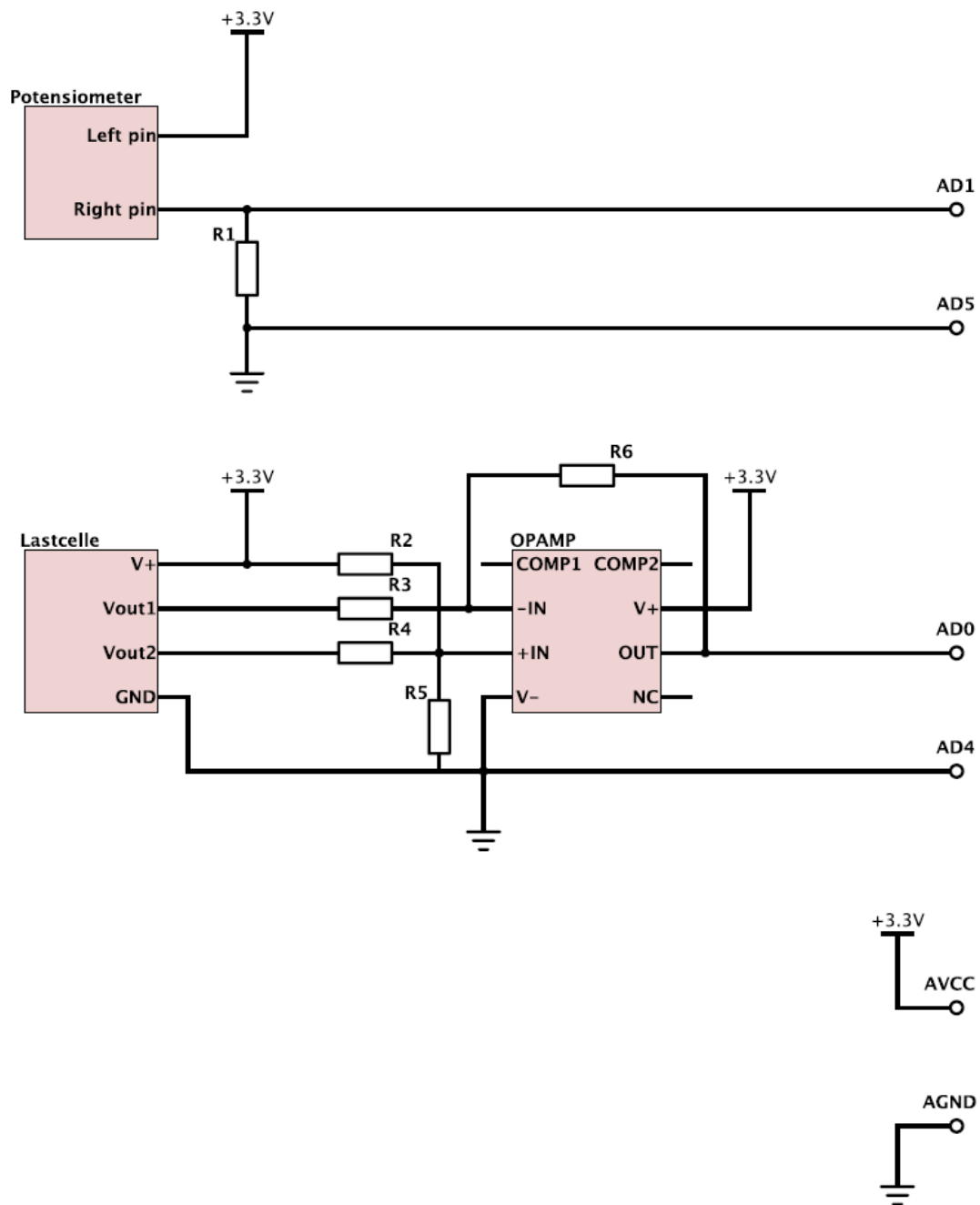
<code>void initMambaCom()</code>	- Nullstiller variabelen som lagrer vinkelposisjonen til slangerobotleddet.
<code>int MambaCom_setMambaAngle(int newAngle)</code>	- Komponerer en CAN-melding som følger kommunikasjonsprotokollen til slangeroboten. Denne meldingen gir servoen i robotleddet beskjed om å gå til vinkelposisjonen i <code>newAngle</code> .
<code>void MambaCom_saveObtainedMambaAngle(int newAngle)</code>	- Lagrer den nye vinkelen som mottas fra slangeroboten. Brukes av <code>CAN_processReceivedParameters()</code> .
<code>void MambaCom_askForAngle()</code>	- komponerer en CAN-melding som spør slangeroboten om sitt eget ledds vinkelposisjon. Slangroboten vil så svare med en egen CAN-melding.

Eksoskjelett.h Her ligger main-loopen til koden og hele programmet styres fra denne blokken. Programmet starter med å kjøre alle initialiseringsfunksjonene fra de andre blokkene før det så setter et LED-lys grønt for å indikere at initialiseringen er utført og at koden er inne i while-løkken som kjører hoveddelen av programmet. Inne i while-løkken polles `doPeriodicTasks()` fra `Timer.h`. Hver gang denne returnerer `TRUE`, altså 50 ganger i sekundet, kjøres funksjoner fra `ADC.h`, `MambaCom.h` og `Servo.h`. Disse funksjonene sørger for at A/D-konverteren oppdaterer verdiene sine, at vinkelen på slangeroboten avleses og at PID-regulatoren kjøres med en frekvens på 50 Hz. I tillegg kjøres funksjoner fra `DebugCom.h` 10 ganger i sekundet. Disse funksjonene sjekker om datamaskinen har sendt noen input over seriellporten samt sender informasjon tilbake til datamaskinen. Hvilken informasjon som sendes bestemmes av konstanten `DebugComMode` som settes før programstart. Denne har tre mulige verdier: `MODE_PUTTY`, `MODE_LABVIEW` og `MODE_SILENT`.

4.4 Oppkobling av komponenter

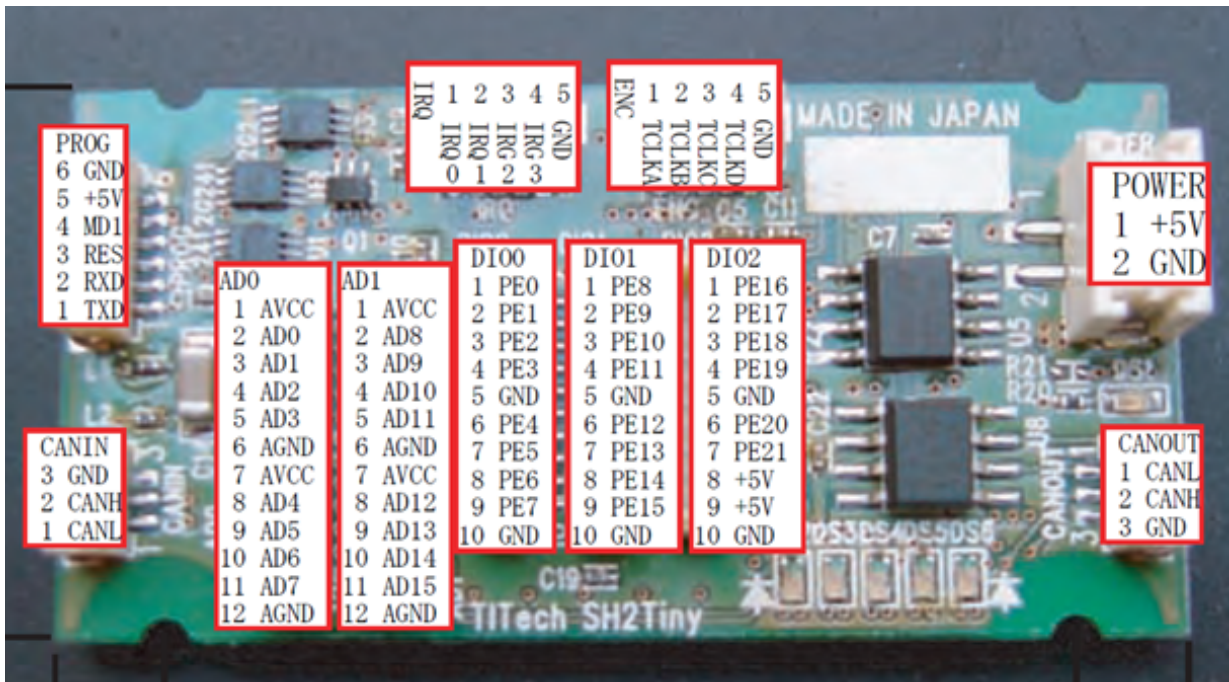


Figur 28: Oversikt over oppkobling av hele systemet



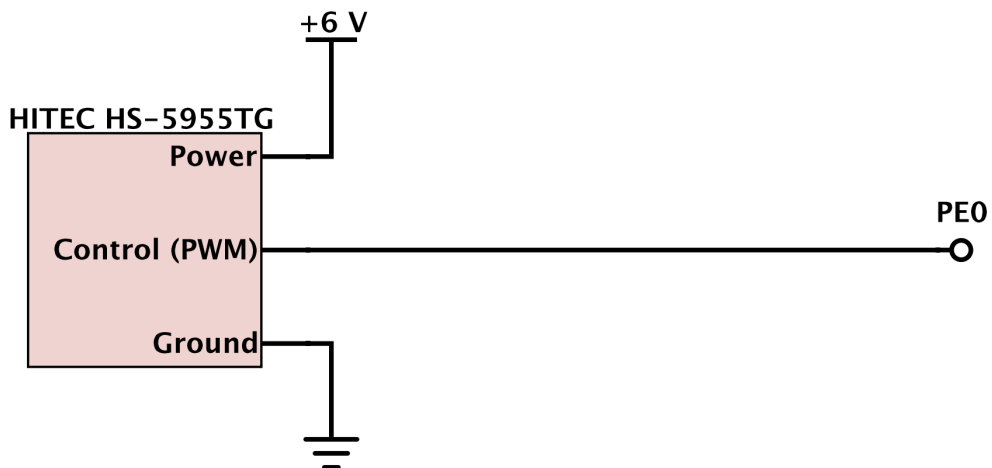
Figur 29: Oppkobling mellom komponenter og A/D-konverter

Avlesning av potensiometer og lastcelle skjer ved hjelp av analog/digital-konverteren på mikrokontrolleren. Lastcellen kobles opp som vist i avsnitt 4.1.3 hvor utgangen av differensialforsterkeren kobles til kanal 1 på A/D-konverteren. Potensiometeret kobles opp som vist i avsnitt 4.1.4 og kobles til kanal 2 på A/D-konverteren. Se figur 29 for oppkobling av komponenter og figur 30 for pin-oversikt på mikrokontrolleren.



Figur 30: Pin-oversikt på mikrokontroller

PWM-signalet som skal styre servoen kommer fra utgang *DIO0*, pin *PE0* på mikrokontrolleren (se figur 30). Servoen som skal styres opererer på en maksimalspenning på 6 V. Det er ønskelig å benytte denne spenningen for å oppnå størst mulig dreiemoment fra servoen. Derfor kobles en ekstern spenningskilde på 6 V til servoens + og - tilkoblinger mens PWM-inngangen kobles til mikrokontrolleren. Se figur 31.



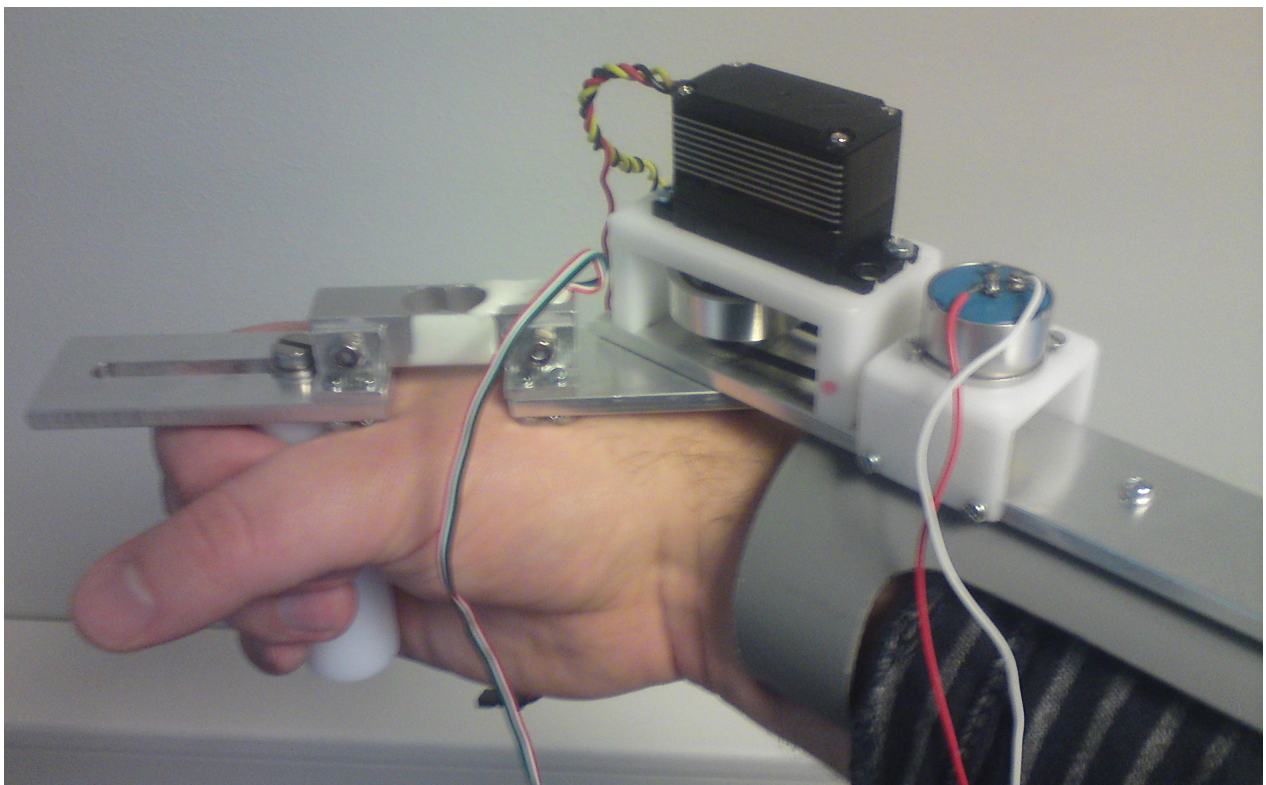
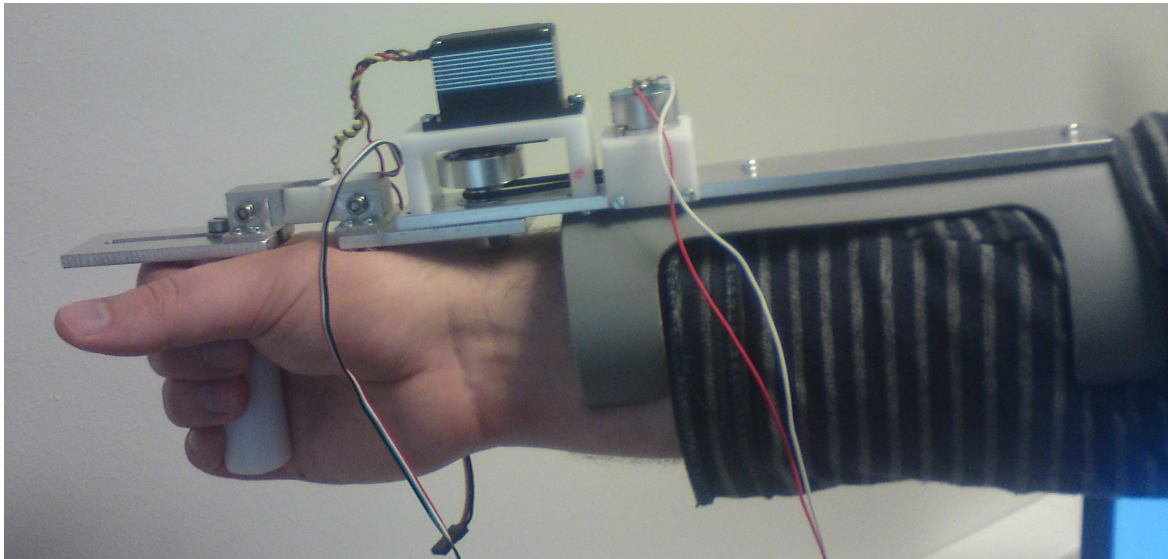
Figur 31: Oppkobling av servo

CAN-bussen mellom slangerobotleddet og eksoskjelettet kan kobles på hvilke som helst av

de to CAN-portene på mikrokontrolleren (se figur 30). Siden mikrokontrolleren på både eksoskjelettet og slangeroboten har to tilkoblingspunkt til CAN-bussen er det dessuten lurt å terminere inngangene som ikke brukes. Det har derimot ikke vært noe problem å kjøre systemet uten termineringsmotstander heller.

USP-en som brukes til å programmere mikrokontrolleren (se høyre bilde i figur 18), samt fungere som seriellport til datamaskinen kobles til inngang *PROG* på mikrokontrolleren (se figur 30). Tilkoblingen som ble brukt var allerede modifisert ved at strømtilførselledningen fra programmeren til mikrokontrolleren er fjernet. Denne strømtilførselen ville kjørt mikrokontrolleren på 5 V og derfor gjort det umulig å kjøre mikrokontrolleren på 3,3 V. Det har tidligere blitt observert at mikrokontrolleren blir svært varm ved 5 V. I tillegg er det en stor fordel å kjøre eksoskjelettet på samme spenning som slangeroboten siden kommunikasjonen skjer over en felles CAN-buss. Dette betyr at en ekstern spenningsforsyning må tilkobles mikrokontrolleren siden den ikke lengre kan dra strøm fra USP-en.

4.5 Bilder av prototypen



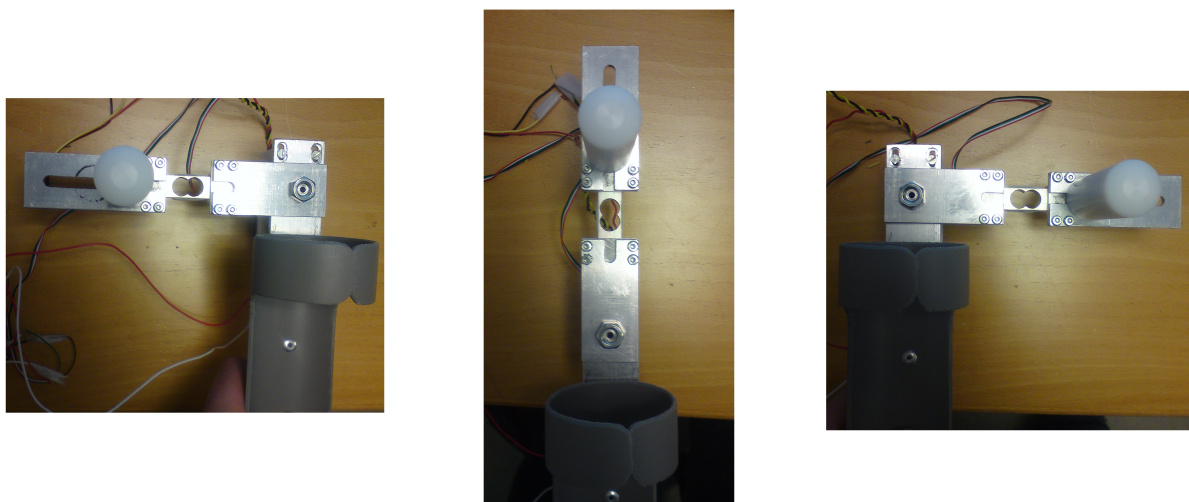
5 Funksjonstesting av eksoskjelett

5.1 Akseptansekrav

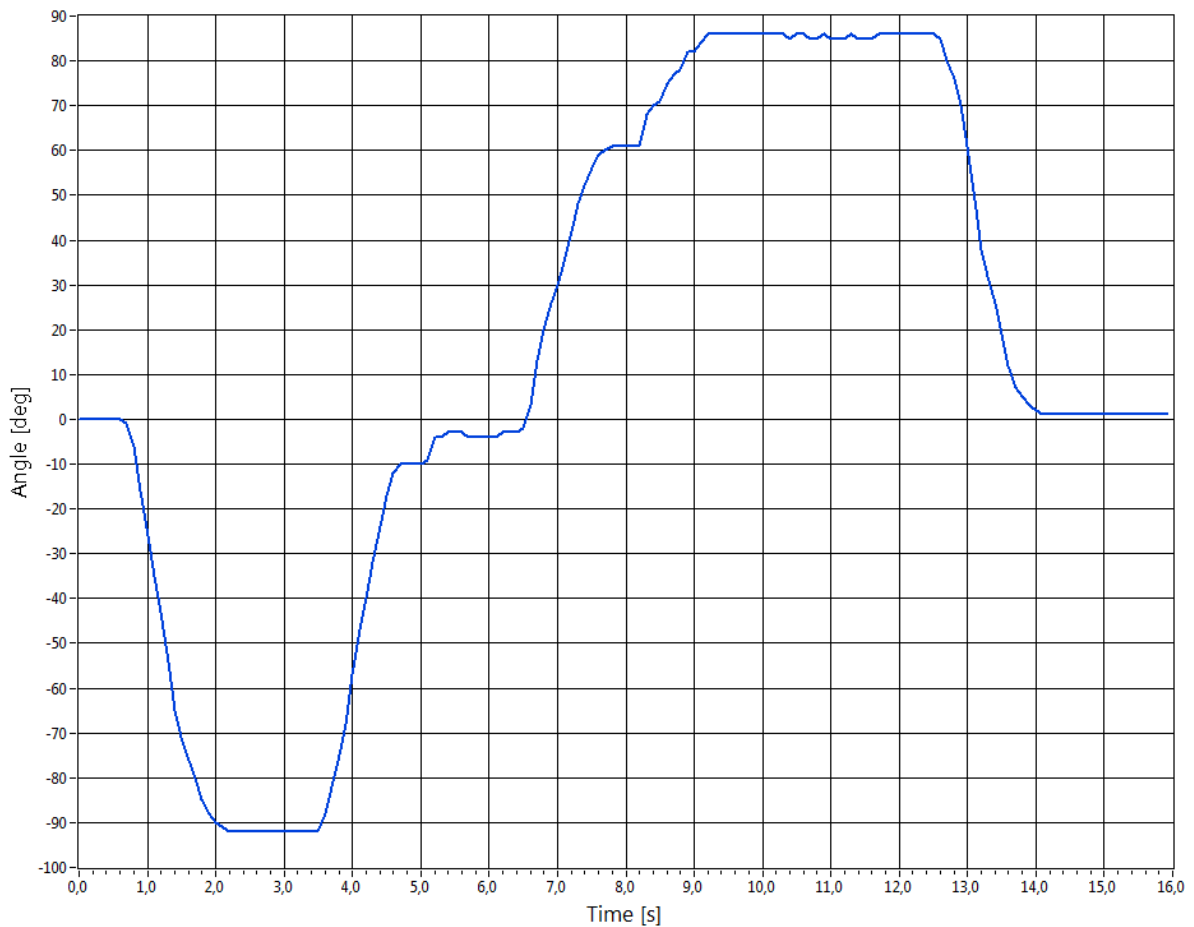
Ut fra sporbarhetsmatrisen i avsnitt 3.2.1 gås her gjennom hvert enkelt akseptansekrav.

5.1.1 TPK-1: Måling av vinkelposisjonen

Eksoskjelettet regner vinkelen 0° som posisjonen hvor leddet peker rett frem. Testen utføres ved å vri leddet 90° mot klokken, 180° med klokken og så tilbake til utgangsposisjonen. Dette kan sees i figur 33. Ut fra testen ser det ut til at eksoskjelettet måler håndleddets vinkel nokså godt, men med et lite avvik. Spesielt ved $+90^\circ$ er dette synlig. Noe av avviket kan komme av unøyaktighet ved selve forsøket, men som det vises i figur 32, er det mulig å oppnå ganske nøyaktige 90° -vinkler ved å matche rette sider på metallskinnene på leddet. Noe unøyaktighet kan også komme av unøyaktig måling av resistansen i potensiometeret. Det ble dessuten observert at reimen mellom akslingen og potensiometeret kan slippe litt ved større bevegelser, så dette er også en potensiell feilkilde. Ved lengre bruk er det tydelig at utgangsposisjonen ikke lengre blir målt til 0° . Det er likevel ikke snakk om større avvik enn 3° - 5° . Alt i alt er målingen nøyaktig nok for å teste resten av funksjonene.



Figur 32: Forskjellige vinkelposisjoner for eksoskjelettet



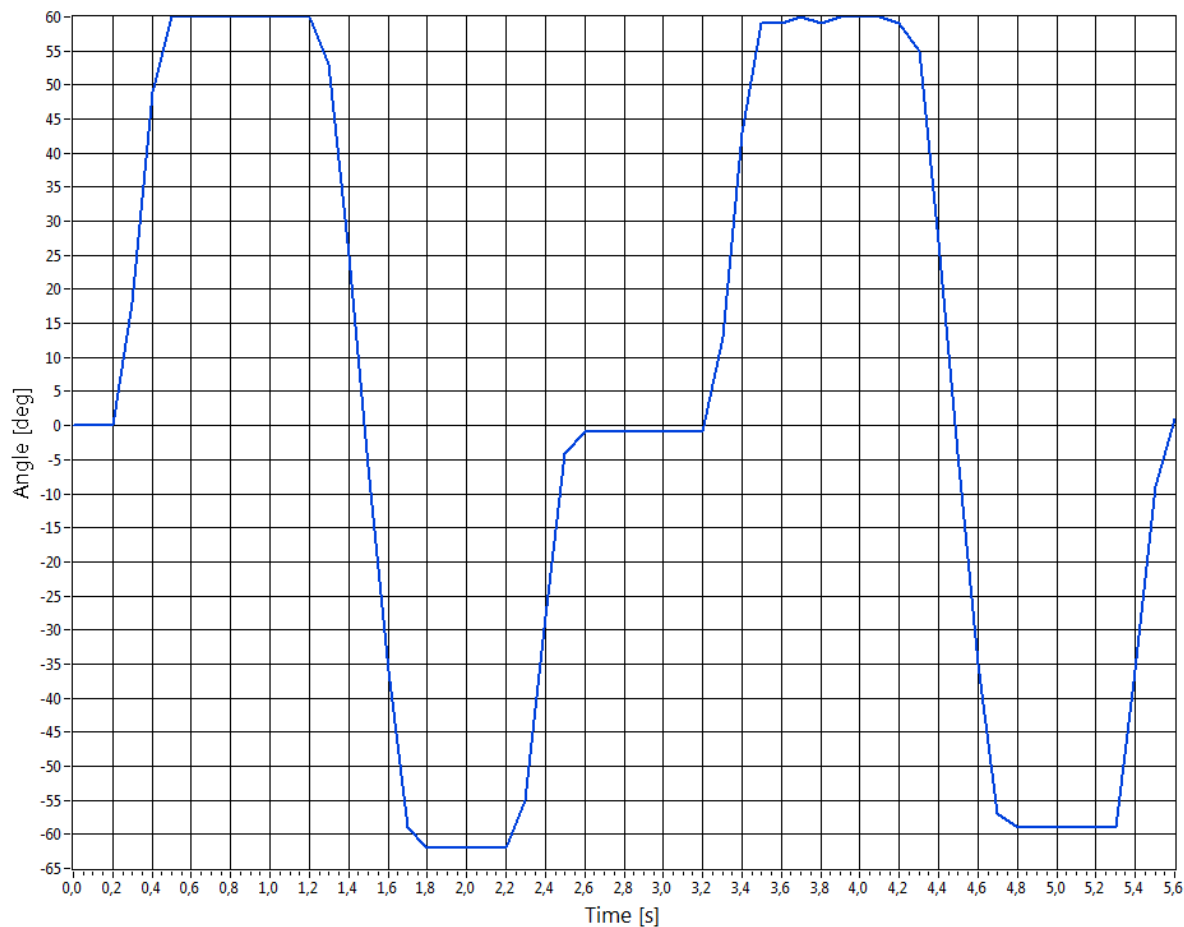
Figur 33: Måling av vinkelposisjon

5.1.2 TPK-2: Måling av dreiemoment

Samme test som ble utført i avsnitt 4.1.3 utførtes på nytt på eksoskjelettet. Det viste seg at akslingsfestet mellom servo og ledd gav etter ved krefter større enn 50 N-cm. For lastcellen tilsvarer dette en belastning på ca. 700 gram og er derfor godt innenfor det 5 kg-store måleområdet til lastcellen. Som forventet måles også deler av vekten til håndtaket alt etter hvilken vinkel eksoskjelettet holdes i. Som nevnt i avsnitt 3.5, er det en ulempe at lastcellen må bære vekten til ytterste del av leddet siden det ikke var kjent hvor mye dette ville påvirke målingene selv om vekten belastes vinkelrett på måleretningen til lastcellen. Det var derimot ikke noe tegn på at denne vekten forstyrret målingene nevneverdig når eksoskjelettet holdes med håndtaket vinkelrett ned som i figur 22. Dette betyr at lastcellen ser ut til å overse krefter som står vinkelrett på akse lastcellen måler krefter langs. Så lenge vekten til ytterste del faktisk bidrar til motvekten operatøren kjenner, er det ikke et problem at den måles.

5.1.3 TPK-3: Test av posisjonsregulering

For å teste posisjonsreguleringen ble eksoskjelettet satt til å periodisk veksle mellom posisjonene 0° , 60° og -60° . Resultatet kan sees i figur 34. Reguleringen er i hovedsak god, men også her kan det sees et visst avvik. Spesielt synlig er dette ved -60° . Feilmarginene er likevel innenfor målefeilene som ble observert i TPK-1 så det derfor ikke uventet at et slikt avvik inntreffer. Reguleringen er likevel nøyaktig nok for formålet i denne oppgaven.



Figur 34: Regulering av vinkelposisjon

5.1.4 TPK-4: Testing av sikkerhet

For ordens skyld ble det testet å gi fullt pådrag med servoen for å bekrefte at denne ikke var i stand til å forvolde noe skade på operatøren. Ikke overraskende var dette ikke noe problem, spesielt med tanke på at akslingskoblingen gir etter ved et dreiemoment på 50 N-cm. Det kan også gjentas at selve servoen ikke kan gi ut større moment enn 235 N-cm, som heller ikke er stort nok til å skade håndleddet til et voksent menneske. Til slutt kan

det nevnes at eksoskjelettet er ganske ubehagelig å ha på seg og at noen av skruene kan lage merker i huden på operatøren, men dette er ikke et prioritert problem med denne prototypen.

5.1.5 TPK-5: Testing av kommunikasjon

Som vist over er det fullt mulig å lese av verdier fra eksoskjelettet på en datamaskin, enten ved hjelp av PuTTY eller selvutviklede LabVIEW-applikasjonen. I tillegg er det mulig å sende korte meldinger til eksoskjelettet som f.eks gir beskjed om å resette vinkelmåling eller dreiemomentmåling. I tillegg kobles et slangerobotledd til CAN-bussen og eksoskjelettet settes til å følge slangerobotleddets bevegelse. Alt dette fungerer som forventet.

5.2 Posisjon-til-posisjonsregulator

På grunn av tidsbegrensinger ble det usikkert om det var tid til å implementere en kraft-til-posisjonsregulering. Derfor ble det sett på muligheten for å implementere en posisjon-til-posisjonsregulering. Dette er gjennomført før i både [14] og [15]. Fordelen med dette styresystemet er at det er enkelt å implementere, ulempen er at ingen av kraftmålingene benyttes, men heller posisjonsmålingen fra både slangerobotleddet og eksoskjelettet.

Det viktigste formålet med testen er å se om styresystemet blir stabilt og hvor godt slangerobotleddet følger eksoskjelettet og vice versa. Oppkoblingen for testoppsettet blir som vist i figur 28 for tolkning av testresultatet er det derimot mer interessant å se på hvordan måleresultater kommuniseres mellom modulene. Dette kan sees i figur 35.

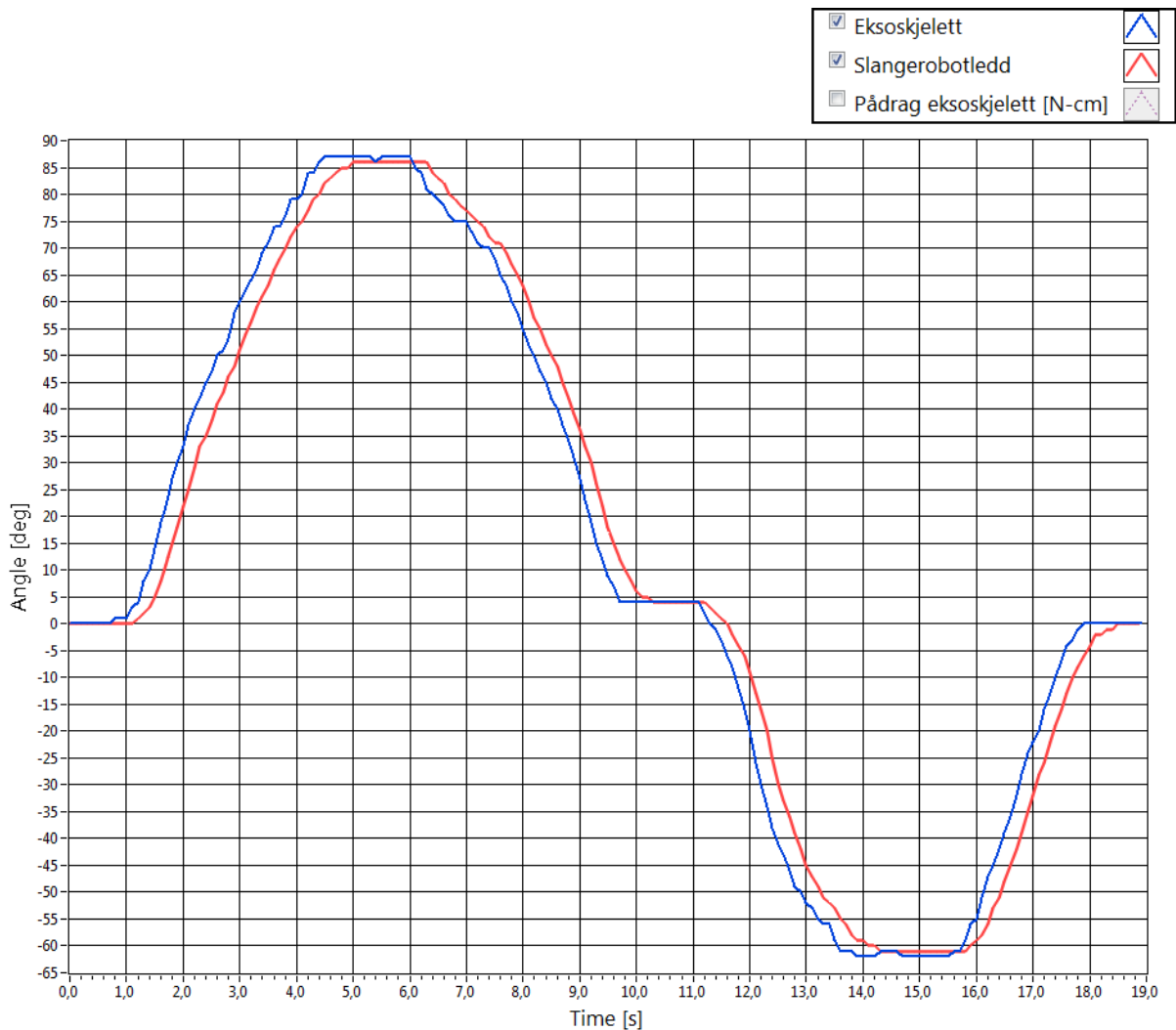


Figur 35: Kommunikasjonen mellom modulene i testoppsettet

Som figuren viser så går all kommunikasjon fra slangerobotleddet gjennom eksoskjelettet før det når datamaskinen. Et alternativ til dette oppsettet kunne være å også koble slangerobotleddet til datamaskinen via en seriell-buss. Dette ville imidlertid kreve en ekstra USP (se høyre bilde i figur 18) til å sende kommunikasjonen mellom slangerobotleddet og datamaskinen. I dette prosjektet var bare en USP tilgjengelig. Dette innebærer en viss tidsforsinkelse fra vinkelposisjonen måles i slangerobotleddet, til det mottas av eksoskjelettet og derfra sendes sammen med annen data til datamaskinen. Sammenlignet med andre kilder til tidsforsinkelse i systemet, som for eksempel tiden det tar fra pådraget settes på servoen til servoen reagerer, er det derimot lite trolig tidsforsinkelsen over CAN-bussen har mye å si for måleresultatet.

For å spare slangerobotleddet for unødvendig belastning ble det ikke påtrykt en ytre kraft for å holde igjen leddet. I stedet ble det satt en begrensning på farten som servoen på slangerobotleddet kunne oppnå. Dette er støttet av programmet som styrer slangeroboten. Tanken bak dette er at slangerobotleddet skal følge eksoskjelettet så lenge operatøren beveger håndleddet i en lavere fart enn grensen som er satt på servoen. Videre er forventes det at eksoskjelettet skal bremse operatørens bevegelse dersom denne er raskere enn det servoen på slangerobotleddet tillater. Dette simulerer altså en tenkt hindring ved raske bevegelser.

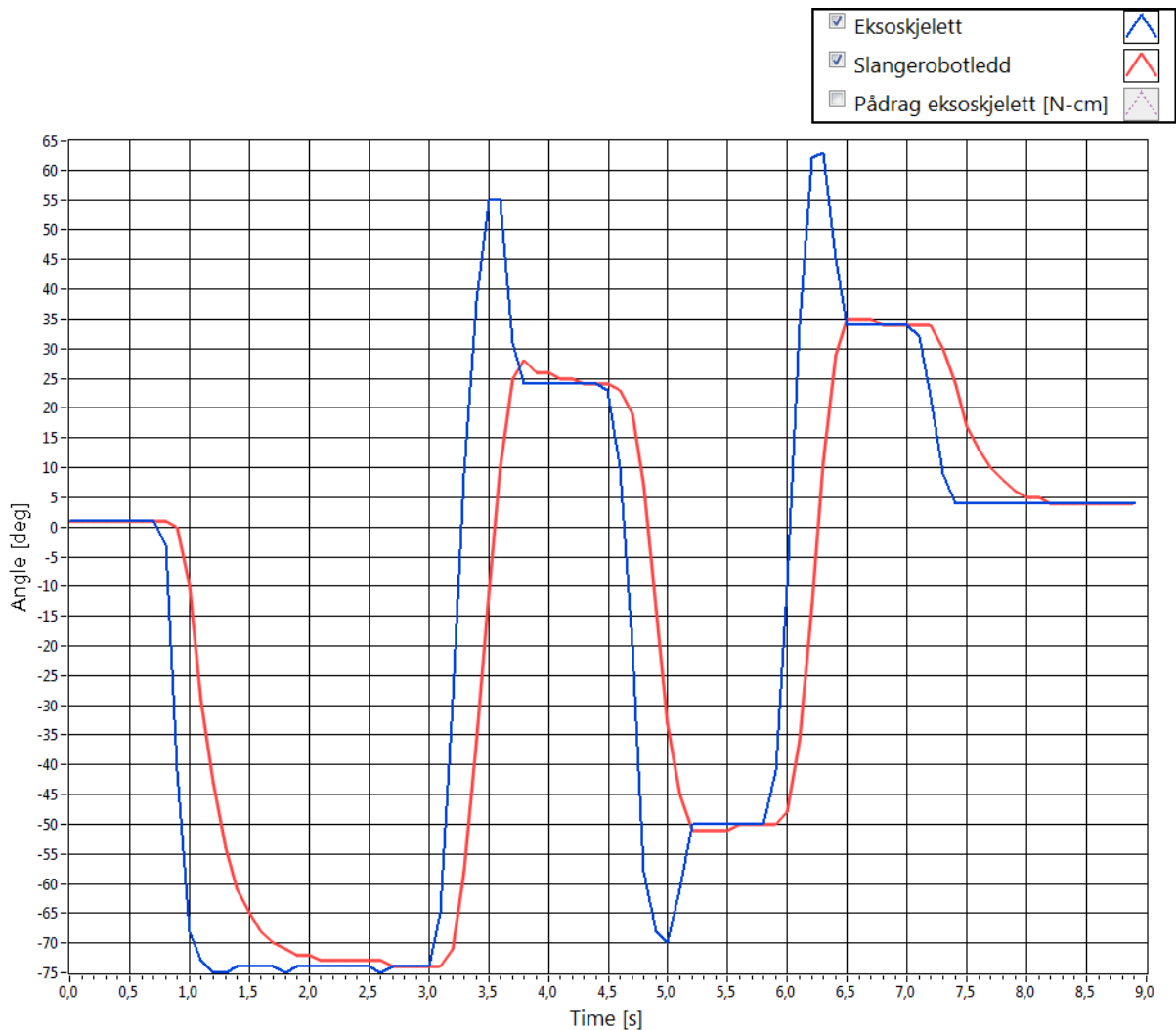
I første forsøk som kan sees i figur 36, ble det sett på hvor godt slangerobotleddet greier



Figur 36: Posisjonsfølging mellom eksoskjelett og slangerobotledd ved rolig bevegelse

å følge vinkelposisjonen til eksoskjelettet ved rolige bevegelser. Bevegelsen er her så rolig at fartsbegrensingen på slangerobotleddet ikke påvirker resultatet. Som figuren viser følger slangerobotleddet eksoskjelettets bevegelse med en tidsforsinkelse på mellom 200 ms - 300 ms. Ved flere målinger med høyere samplingsrate ser det som det tar slangeroboten et sted mellom 120 ms til 160 ms å reagere på bevegelser fra eksoskjelettet, men dette er altså fra stillestående til bevegelse, og det ser ut som denne tidsforsinkelsen blir noe større når begge system allerede er i bevegelse. Som figuren viser, fører ikke denne tidsforskjellen til ustabilitet.

Videre ble det så utført bråere bevegelser fra operatøren. Disse bevegelsene var såpass hurtige at fartsbegrensningen på slangerobotleddet senket hastigheten på servoen. Resultatet kan sees i figur 37. Figuren viser tre store toppler for vinkelposisjonen til eksoskjelettet ved

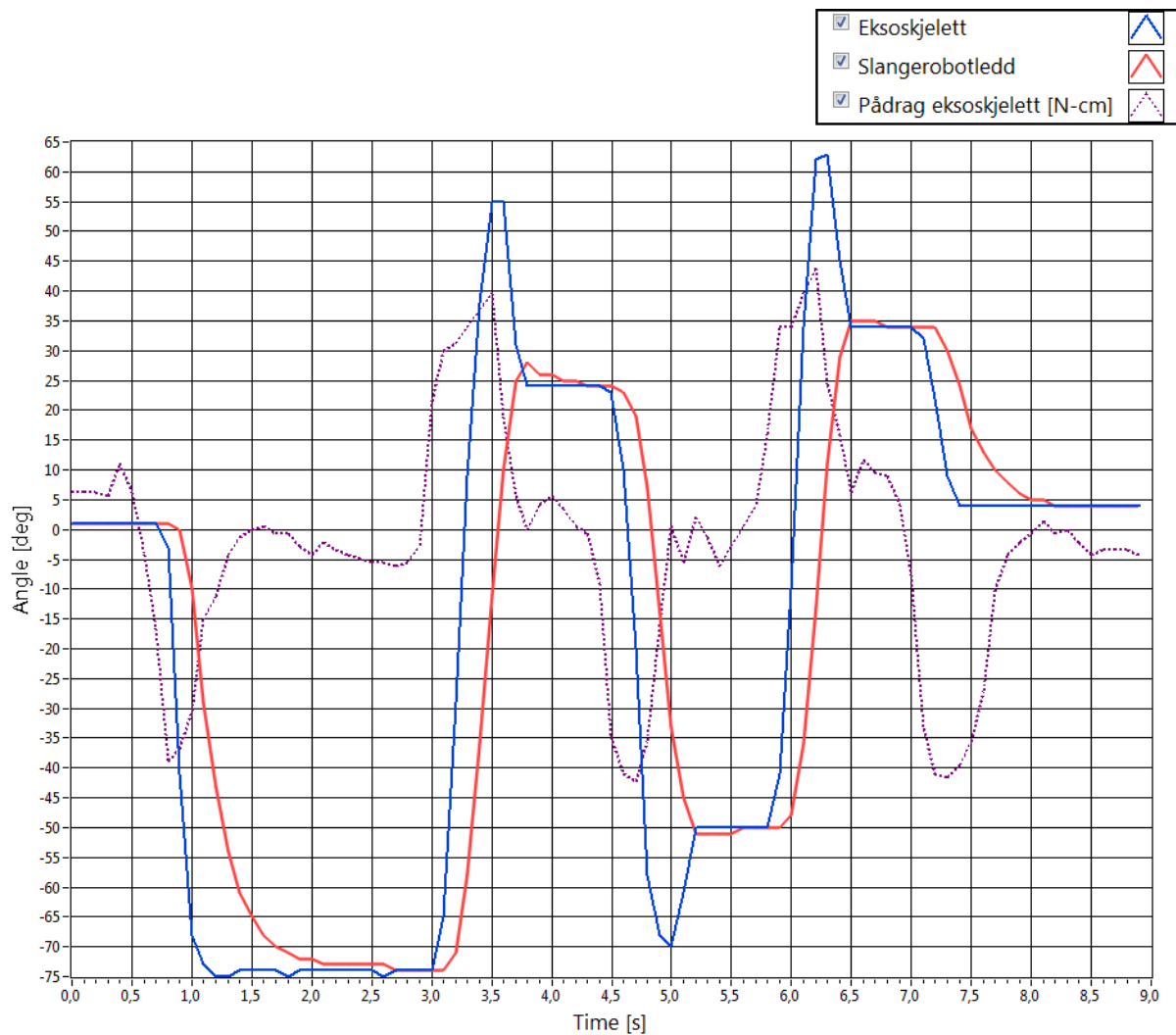


Figur 37: Posisjonsfølging mellom eksoskjelett og slangerobotledd ved hurtig bevegelse

3,5 s, 5,0 s, og 6,2 s. Disse kommer av at operatørens håndledd beveges i en fart større enn fartsbegrensningen til slangerobotleddet. Under dette forsøket ble begrensningene til eksoskjelettets kraft noe mer merkbart. Det er mulig å merke at kraften fra servoen øker i takt med differansen mellom vinkelposisjonene til de to systemene. Som operatør er det likevel en vanskelig balansegang mellom å bruke nok kraft til å kunne flytte leddet, og samtidig ikke ta i for mye slik at akslingskoblingen gir etter.

Dersom det i tillegg legges til momentmålingen i plottet, noe som er gjort i figur 38, er det enklere å se hvordan operatøren må ta i for å få ønsket resultat. Som figuren viser er det først når operatøren slapper av i håndleddet at eksoskjelettet greier å samkjøre vinkelposisjonen til de to systemene. Selv om posisjonsfølgingen er nokså god er det altså tydelig at styrkebegrensningen til eksoskjelettet er med på å gi et resultat som ikke er så

godt som håpet. Dette påvirker også den litt mer subjektive følelsen av transparens, som gjennomgås i avsnitt , på en negativ måte.



Figur 38: Posisjonsfølging mellom eksoskjelett og slangerobotledd samt momentmåling ved hurtig bevegelse

6 Diskusjon og konklusjon

6.1 Utforming, realisering og funksjonstesting av prototypen

Første mål med oppgaven var å komme frem til en funksjonsspesifikasjon som kunne oppfylles av prototypen. Fra forprosjektet var det allerede konkludert med at en prototyp med muligheten for å måle både posisjon og dreiemoment ville gi mest mulig fleksibilitet for hva som senere kunne implementeres med eksoskjelettet. Selve funksjonsspesifikasjonen ble derfor ikke spesielt tidkrevende å komme frem til. Derimot ble det brukt mye tid på å se på ulike mulige prototyper og det er derfor relevant å se på hvordan den valgte løsningen påvirket det endelige resultatet og sammenligne med hvordan andre design kunne gitt andre resultat.

Beslutningen om å bruke operatørens håndledd som utgangspunkt for eksoskjelettet ble beholdt fra forprosjektet. Idéen om at dette ville gi en passe stor prototyp med god fleksibilitet for ulike design, sammenlignet med en liten prototyp for fingerleddene eller en større og tyngre prototyp for albuen, viste seg å stemme bra med erfaringer gjort i ettertid. Det kanskje mest avgjørende valget ved utviklingen av prototypen var hvordan leddet skulle aktueres. Valget mellom å benytte en elektrisk motor i stedet for en pneumatisk aktuator ble gjort tidlig i oppgaven og begrunnet med at elektriske motorer er enklere å styre, samt at en prototyp bare avhengig av en strømkilde ville være mer mobil enn et eksoskjelett som i tillegg var avhengig av trykkluft.

For utviklingen av programmet som skulle styre eksoskjelettet var det absolutt en fordel å kunne kjøre prototypen utelukkende på elektrisk kraft. Dette gjorde det enkelt å teste ny kode uten å være avhengig av en luft-kompressor. Ulempen med elektrisk drift av eksoskjelettet var, som nevnt før i rapporten, manglende styrke i motorer som var små nok til å kunne monteres på prototypen. Hypotesen om at en elektrisk servo ville være sterk nok og derfor gi nok pådrag til at prototypens og styresystemets funksjon kunne testes ut, viste i og for seg å stemme. Dersom operatøren slapper av i håndleddet, greier eksoskjelettet uten problem å flytte hånden til en ønsket vinkelposisjon. Dersom operatøren derimot yter motstand, skal det ikke mye til før koblingen mellom servo og aksling gir etter. Det kan derfor diskuteres om den samlede kompleksiteten av PID-regulatoren som styrer servoen, servoens momentbegrensinger samt koblingen mellom servo og aksling, er like stor som kompleksiteten ville vært dersom pneumatiske aktuatorer hadde blitt benyttet. Ved bruk av pneumatiske aktuatorer hadde det vært mulig å gi operatøren større motkraft. Samtidig er det viktig å påpeke at slangeroboten selv styres av servoer. Disse har mange av de samme kraftbegrensingene som servoen på slangerobotleddet så det kan også argumenteres at det ville bli feil å gi eksoskjelettet større aktuator-kraft enn slangerobotleddet det skal styre. Det kunne for eksempel, ved posisjon-til-posisjonsregulering, gi operatøren en motkraft større enn den slangeroboten opplever.

Måleinstrumentene har også vist seg å gi tilfredsstillende resultat. Ut fra de kravene som ble satt i funksjonsspesifikasjonen tidlig i oppgaven presterer eksoskjelettet opp imot det som kan forventes. Størst avvik finnes i koblingen mellom potensiometeret og akslinger, hvor det ser ut til at reimen gir noe etter og posisjonen derfor kan bli noen grader feil etter lengre bruk. Dette kan tilsi at denne formen for kobling mellom potensiometer og ledd ikke var det mest ideelle. Samtidig har ikke eksoskjelettet plass til flere komponenter direkte på leddet når servoen allerede er tilkoblet. Det er heller ikke plass til å sette potensiometeret på undersiden av leddet, altså mellom operatørens håndledd og eksoskjelettets aksling.

For å konkludere samsvarer prototypen som er konstruert med de forventningene som ble satt ved starten av oppgaven. Et håndleddsstyrt eksoskjelett med en frihetsgrad, hvor både vinkelposisjon og dreiemoment kan måles og med en aktuator som påtrykker motkraft til operatøren.

6.2 Testing av styresystem

Som det kommer frem av teorikapittelet hentet fra forprosjektet, var det planlagt å teste et telemanipuleringssystem hvor den ene modulen styres av posisjon (eller fart) og den andre av kraft. Grunnet tidsbegrensninger på slutten av oppgaven ble det derimot for dårlig tid til å teste dette systemet. Et kompromiss ble utført hvor det heller ble benyttet en posisjon-til-posisjon styring da dette var mer tidsbesparende å implementere siden slangeroboten allerede hadde posisjonsreguleringsfunksjonen implementert.

Som resultatene viser fungerer posisjonsfølgningen mellom slangerobot og eksoskjelett nokså godt. De største forskjellene i posisjon ser ut til å komme av en kombinasjon av tidsforsinkelse og manglende styrke i servoen på prototypen. Stabiliteten til systemet er god, selv med den målte tidsforsinkelsen mellom eksoskjelettet og slangeroboten. Som nevnt i teori-kapittelet er tidsforsinkelser ofte en stor bidragsyter til at systemet blir ustabil. At dette likevel går bra kan komme av at friksjonen i leddet til prototypen er med på å dempe systemets oscillasjoner.

Som det nevnes i avsnitt 5.2, påvirker eksoskjelettets mangel på styrke transparensen til telemanipuleringssystemet på en negativ måte. Når operatøren hele tiden befinner seg på grensen for når akslingskoblingen gir etter, er dette med på at følelsen av å kjenne på samme motkraft som slaven blir dårligere. En mulig løsning på dette kan være å stramme opp koblingen mellom servo og aksling slik at denne tåler mer påkjenning før den gir etter. Som nevnt tidligere, ligger grensen for når koblingen gir etter ganske mye lavere enn maksimalpådraget som servoen kan gi. Det er også godt mulig at de opprinnelige styresystemene fra forprosjektet kunne gitt noe bedre resultat. Disse vil selvsagt ikke påvirke styrken til prototypen, men kanskje påtrykke operatøren en mer nyansert kraft enn det posisjon-til-posisjonsreguleringen gjør og derfor gi en bedre følelse av transparens.

Det er vanskelig å komme med en bastant konklusjon angående systemets egnethet som operatørgrensesnitt for en slangerobot ut fra dette ett-leddssystemet. Skal dette virkelig testes, bør det konstrueres et eksoskjelett med flere ledd slik at det er mulig å imitere en ordentlig slange-bevegelse. Det er likevel mulig å konkludere med hvor godt ett enkelt ledd kan styres og som resultatene viser: Posisjonsfølgingen fungerer nokså godt, men til gjengjeld oppleves ikke transparensen i telemanipuleringssystemet som like god. Dette betyr at systemet virker egnet til å styre direkte hvert slangerobledd. Når det kommer til prototypens evne til å gi en god kraft-tilbakekobling, er det derimot mer utvikling igjen før det kan slås fast at dette designet gir et ønsket resultat. Idéer rundt dette arbeidet går gjennom i neste avsnitt.

6.3 Videre arbeid

Ut fra de resultatene som er oppnådd i denne oppgaven peker det seg ut to mulige retninger for videre utvikling:

Den første vil være å ta utgangspunkt i eksoskjelettet som allerede er bygget og derfra videreutvikle oppgaven. Her vil det i første omgang være aktuelt å implementere et bedre telemanipuleringssystem basert på en kombinasjon av kraft- og posisjonsmålinger. Videre er det, for å virkelig kunne teste om styring av en slangerobot er oppnåelig, nødvendig å utvide eksoskjelettet med flere ledd. Her er kanskje fingerleddene og albuen de mest naturlige valgene. For albuen kan man gjenbruke samme design som for håndleddet, men for fingrene vil det mest sannsynlig være nødvendig å tenke nytt med tanke på hvor mye mindre plass man vil ha til komponenter.

Et annet mulig alternativ vil være å se på muligheten for å konstruere et nytt eksoskjelett med pneumatiske aktuatorer. Mange av erfaringene fra dette prosjektet kan brukes i utviklingen av et slikt eksoskjelett samtidig som det kan sees på eksisterende løsninger som for eksempel i [16]. Som diskutert tidligere i oppgaven vil dette kanskje resultere i et sterkere eksoskjelett som kan gi operatøren mer motstand og derfor gi et bedre telemanipuleringssystem.

Referanser

- [1] R. Øye, Elling. *Eksoskjelett for styring av slangerobot*, Prosjektoppgave, NTNU, 2013.
- [2] K. Hashtrudi-Zaad and S. E. Salcudean, "On the Use Of Local Force Feedback for Transparent Teleoperation," in *Proc. IEEE Int. Conf. Robotics and Automation*, 1999, pp. 1863-1869.
- [3] J.V. Draper, D.B. Kaber, and J. M. Usher, "Telepresence," *Human Factors*, vol.40, no.3, pp. 354-375, Sept. 1998.
- [4] D.A Lawrence, "Stability and Transparency in Bilateral Teleoperation," *IEEE Trans. Robot. Automat.*, vol.9, no. 5, pp. 624-637, Okt. 1993.
- [5] Sato, M., Masakazu Fukaya, and Tetsuya Iwasaki, "Serpentine locomotion with robotic snakes," *Control Systems, IEEE* 22.1 (2002): 64-81.
- [6] Dowling, Kevin J. *Limbless locomotion: learning to crawl with a snake robot*. Diss. NASA, 1996.
- [7] Niemeyer, Günter, and J-JE Slotine. "Using wave variables for system analysis and robot control." *Robotics and Automation, 1997. Proceedings., 1997 IEEE International Conference on*. Vol. 2. IEEE, 1997.
- [8] Spong, Mark W., Seth Hutchinson, and Mathukumalli Vidyasagar. *Robot modeling and control*. New York: John Wiley & Sons, 2006.
- [9] Naerum, Edvard, and Blake Hannaford. "Global transparency analysis of the lawrence teleoperator architecture." *Robotics and Automation, 2009. ICRA '09. IEEE International Conference on*. IEEE, 2009.
- [10] Niemeyer, Günter Dieter. *Using wave variables in time delayed force reflecting teleoperation*. Diss. Massachusetts Institute of Technology, 1996.
- [11] Khalil, Hassan K. *Nonlinear systems*. Vol. 3. Upper Saddle River: Prentice hall, 2002.
- [12] Solberg, Ranvieg. *EPP-basert protesemodell*, Masteroppgave, NTNU, 2011.
- [13] Cavallaro, Ettore, et al. "Hill-based model as a myoprocessor for a neural controlled powered exoskeleton arm-parameters optimization." *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*. IEEE, 2005.
- [14] Farkhatdinov, Ildar, and Jee-Hwan Ryu. "Hybrid position-position and position-speed command strategy for the bilateral teleoperation of a mobile robot." *Control, Automation and Systems, 2007. ICCAS'07. International Conference on*. IEEE, 2007.
- [15] Zhu, M., and S. E. Salcudean. "Achieving transparency for teleoperator systems under position and rate control." *Intelligent Robots and Systems 95. Human Robot Interaction*

and Cooperative Robots', Proceedings. 1995 IEEE/RSJ International Conference on.
Vol. 2. IEEE, 1995.

- [16] Tang, Zhichuan, et al. "An Upper-Limb Power-Assist Exoskeleton Using Proportional Myoelectric Control." *Sensors* 14.4 (2014): 6677-6694.

7 Figurer

Figurer

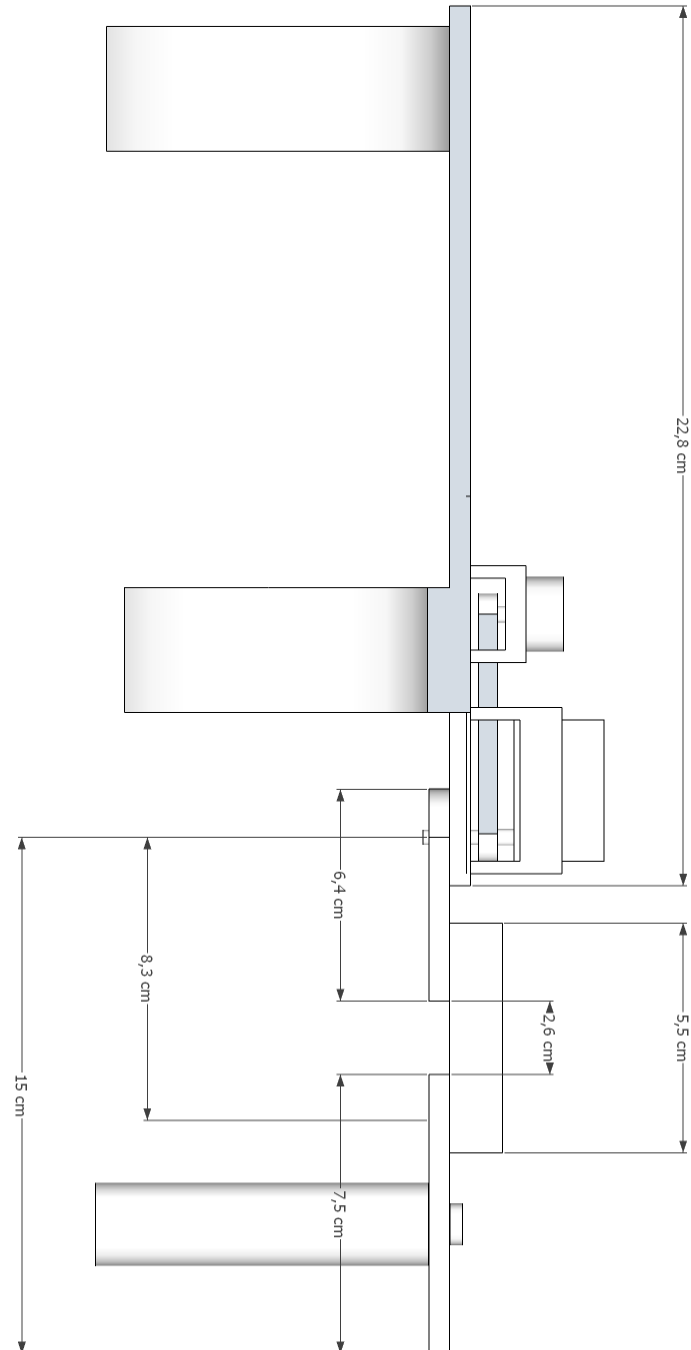
1	”The Extended Lawrence Architecture”, hentet fra [9]	5
2	Reguleringsløyfe: Fart fra master, kraft fra slave	7
3	Reguleringsløyfe: Kraft fra master, fart fra slave	7
4	Blokkdiagram: Bølgevariabler	9
5	Blokkdiagram: Fart fra master, kraft fra slave	10
6	Blokkdiagram: Kraft fra master, fart fra slave	11
7	Simulering av system uten tidsforsinkelse	12
8	Simulering av system med 1000 ms tidsforsinkelse	13
9	Simulering av system med 100 ms tidsforsinkelse	13
10	Simulering av system med bølgevariabler	14
11	Håndledd og slangeledd	19
12	De to forskjellige forslagene til eksjoskjelett-rammer	20
13	Prototype hvor leddet styres av stag	22
14	Prototype hvor motor er plassert bak leddet	23
15	Prototype hvor motor er plassert direkte på leddet	23
16	Forskjellige plasseringer av lastcelle	25
17	Ferdig design av eksjoskjelettet	26
18	Til venstre: TITechSH2 Tiny Controller. Til høyre: TITechSH2 Universal Serial Programmer (USP) (Bilde hentet fra produktets datablad)	28
19	HITEC HS-5955TG – Ultra Torque Servo (Bilde hentet fra produsentens hjemmeside)	29
20	PWM-signal for vanlig posisjonsstyring av servo	30
21	HITEC Digital Servo Programmer & Servo Tester, Model: HFP-10 (Bilde hentet fra produsentens hjemmeside)	30
22	Retning på krefter som måles av lastcellen	31
23	Lastcelle, CZL635 (Bilde hentet fra produsentens hjemmeside)	32
24	Oppsett for test av lastcelle	33
25	Oppkobling, differensialforsterker	34
26	Oppkobling for potensiometer	35
27	Kommunikasjonsflyt mellom modulene i programmet	38
28	Oversikt over oppkobling av hele systemet	44
29	Oppkobling mellom komponenter og A/D-konverter	45
30	Pin-oversikt på mikrokontroller	46
31	Oppkobling av servo	46
32	Forskjellige vinkelposisjoner for eksjoskjelettet	49
33	Måling av vinkelposisjon	50
34	Regulering av vinkelposisjon	51
35	Kommunikasjonen mellom modulene i testoppsettet	53

36	Posisjonsfølging mellom eksoskjelett og slangerobotledd ved rolig bevegelse	54
37	Posisjonsfølging mellom eksoskjelett og slangerobotledd ved hurtig bevegelse	55
38	Posisjonsfølging mellom eksoskjelett og slangerobotledd samt momentmåling ved hurtig bevegelse	56
39	Dimesjoner, eksoskjelett, 2D	65
40	Dimesjoner, eksoskjelett, 3D	66

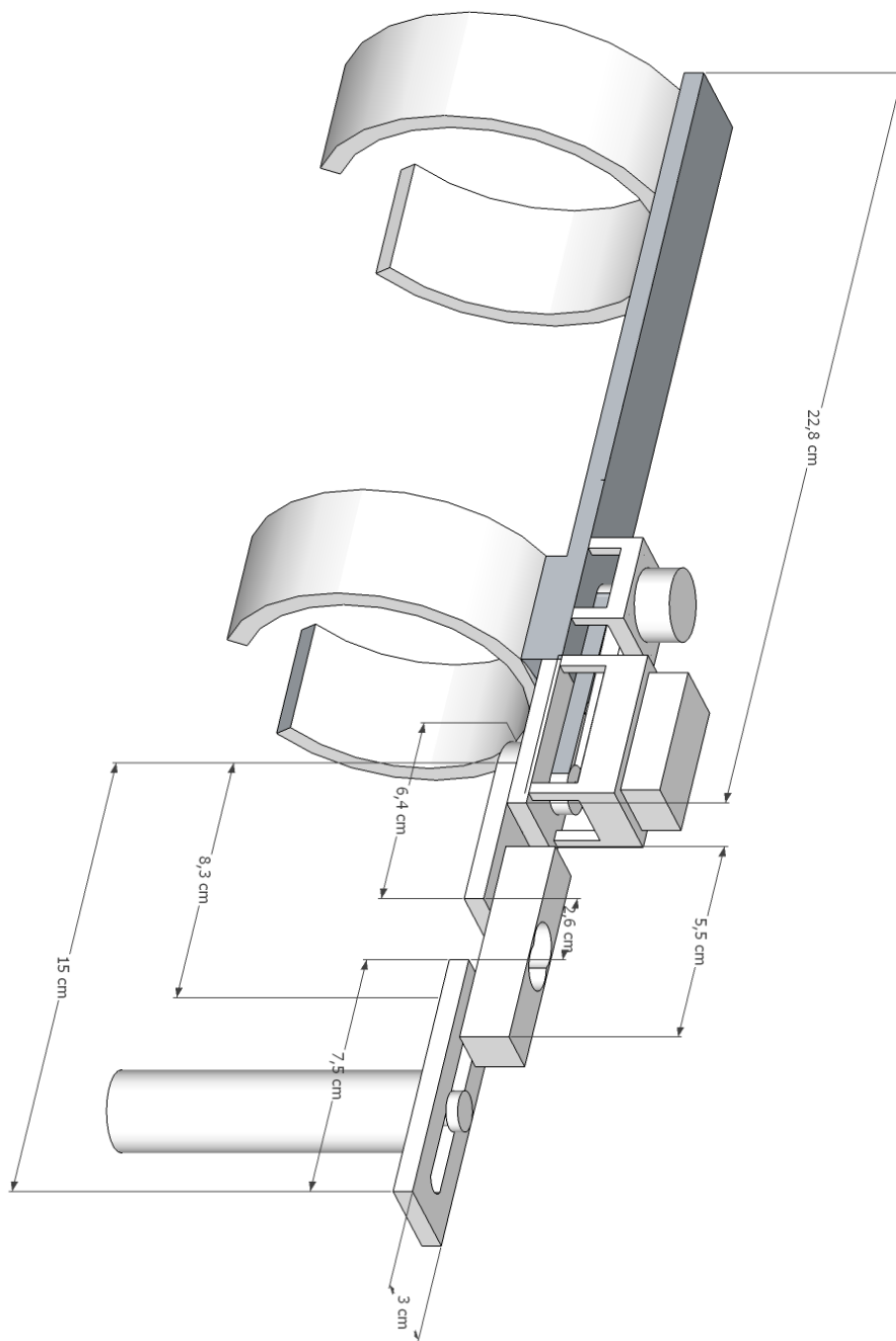
8 Digitalt vedlegg

I det digitale vedlegget ligger programkoden som skal kjøres på mikrokontrolleren til eksoskjelettet, samt LabVIEW-applikasjonen som leser av meldinger som sendes fra mikrokontrolleren til datamaskinen. I tillegg er databladet til mikrokontrolleren som ble brukt, samt et notat fra Pål Liljebäck om hvordan man programmerer mikrokontrolleren, lagt ved. For noen som eventuelt skal ta over denne oppgaven vil dette notatet være til god hjelp for å bruke mikrokontrolleren.

9 Appendiks A - Dimensjoner, eksoskjelett

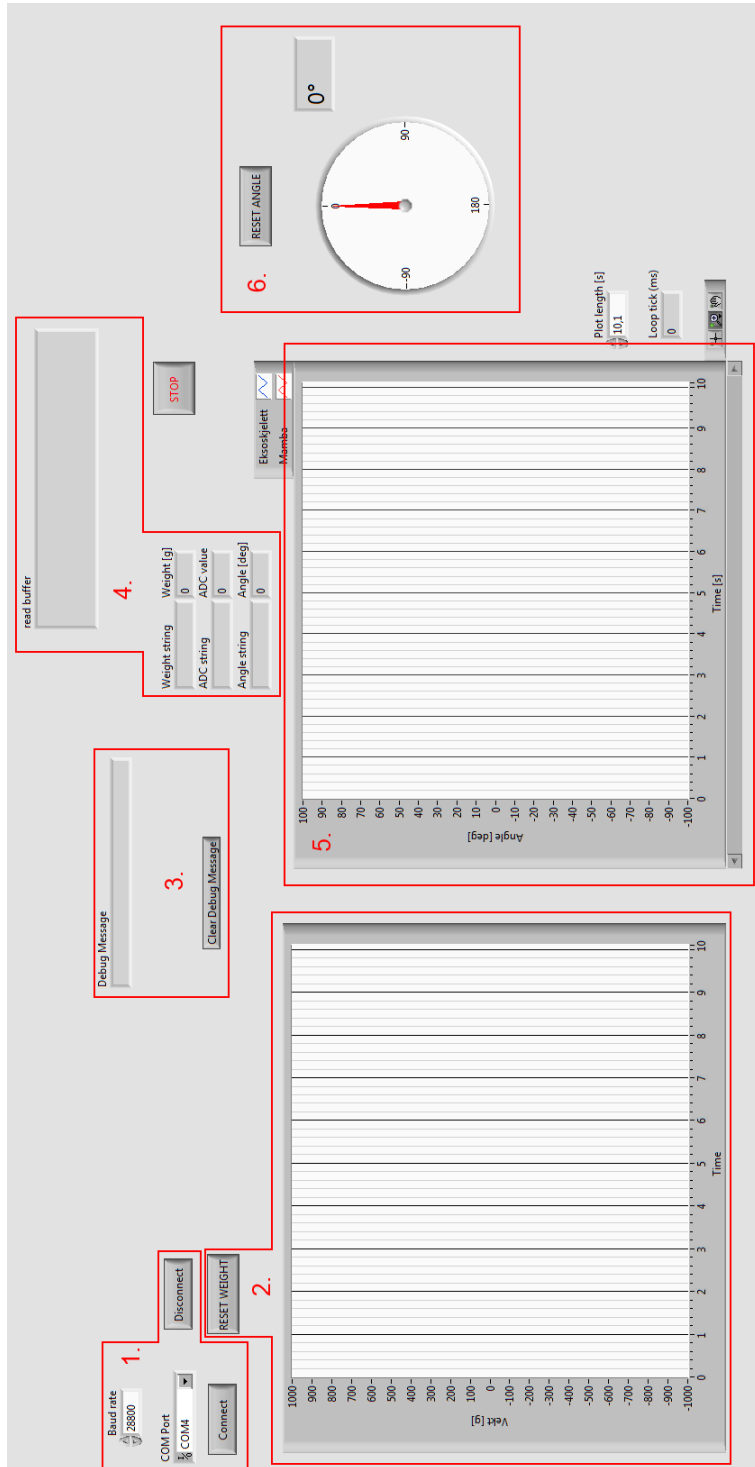


Figur 39: Dimesjoner, eksoskjelett, 2D



Figur 40: Dimesjoner, eksoskjelett, 3D

10 Appendiks B - Labview-applikasjon



1. Her styres oppkoblingen mot seriellporten til mikrokontrolleren. Baudrate og COM-port kan settes.
2. Plott av målingen til lastcellen. RESET WEIGHT forteller mikrokontrolleren at nåværende måling fra lastcellen skal regnes som 0.
3. Her mottas eventuelle debug-meldinger fra mikrokontrolleren.
4. Her vises først den rå stringen som leses fra mikrokontrolleren. Videre deles denne opp og konverteres til lesbare verdier.
5. Plott av vinkelposisjonen til både eksoskjelettet og slangerobotleddet.
6. Sanntidsvisning av vinkelposisjonen til eksoskjelettet med mulighet for å resette målingen.