

# EPP-basert protesemodell

**Ranveig Solberg**

Master i teknisk kybernetikk  
Oppgaven levert: Februar 2011  
Hovedveileder: Øyvind Stavadahl, ITK



# Oppgavetekst

Bakgrunn:

Det er tidligere gjennomført et prosjektarbeid der styringsprinsippet "Extended physiological proprioception" (EPP) ble studert og tilpasset en tenkt bruker med transradiell dysmeli og en intakt "finger". I denne oppgaven skal du videreutvikle resultatene til et demonstrerbart system.

1. Gjør et litteratursøk rundt teamet tilbakekoblede dynamiske systemer med mennesker(r) i tilbakekoblingsløyfen. Utred hvordan "menneskekomponenten" påvirker stabilitet og andre relevante egenskaper i slike systemer, og hvilke tiltak som kan treffes for å oppnå robusthet. Se dette i forhold til det foreliggende systemet, og påpek eventuelle fallgruber og tiltak som kan være relevant i dette tilfellet.
2. Beskriv en egnet programvarestruktur for det foreliggende protesesystemet.
3. Realiser programvaresystemet på en egnet plattform.
4. Test systemet mot en fysisk protesemodell. Modellen kan være et resultat av et tidligere studentarbeid og/eller en videreutvikling av denne.

Oppgaven gitt: 15. september 2010

Hovedveileder: Øyvind Stavadahl, ITK



## Sammendrag

Oppgaven er en fortsettelse av tidligere arbeid med å utvikle en EPP-basert protese med to frihetsgrader til en ung dysmelipasient som har transversell armdysmeli med en rudimenter finger. Denne fingeren skal brukes til å styre protesen. Det er gjort et litteratursøk for å kartlegge metoder og utfordringer for reguleringsteknisk analyse av epp- og telemanipulatorsystemer, og satt opp et sett kriterier som er viktige å ta hensyn til under realisering av systemet. Det er beskrevet et programstruktur, som siden er realisert på to AVR Butterfly, en for hver frihetsgrad. Disse er koblet til en protesemodell og en proksimal styringsmodul tilpasset en normal hånd for testing av konseptet.



# Innhold

<b>I</b>	<b>Innledning</b>	<b>7</b>
1	Formål med oppgaven	7
2	Bakgrunn	7
3	Motivasjon	8
<b>II</b>	<b>Litteratursøk</b>	<b>9</b>
4	Introduksjon av litteratursøket	9
4.1	Fokus for teksten	9
4.2	Presentasjon av systemet	9
5	Resultat av litteratursøket	11
5.1	Reguleringstekniske betraktninger	11
5.1.1	Selvbevissthet	11
5.1.2	Dynamikk i menneskelige lemmer	11
5.1.3	Systemer som er koblet med omgivelsene; fysisk interaksjon	11
5.1.4	Passivitet	13
5.1.5	Alternativer til passivitet	14
5.2	Hardware-kriterier	15
5.3	Fysiologiske krav fra hånden/mennesket	16
6	Diskusjon av litteratursøket	17
6.1	Reguleringstekniske betraktninger	17
6.2	Hardware-kriterier	19
6.3	Fysiologiske krav	20
7	Konklusjon av litteratursøket	21
<b>III</b>	<b>Realisering av systemet</b>	<b>23</b>
8	Programvarestruktur	23
9	Realisering av programvare	26
9.1	Introduksjon til realisering av programvare	26
9.2	Kommunikasjon med PC	26
9.3	ADC-driver (analog- digital converter)	26
9.4	FSR-driver	27
9.5	Potmeter-driver	27
9.6	PWM-/servodriver	28

9.7	Oppsummering av IO på AVR Butterfly . . . . .	29
9.8	Hovedprogrammet: EPP-controller . . . . .	31
<b>10</b>	<b>Beskrivelse av protesemodellen</b>	<b>32</b>
10.1	Distal modul / protesemodell . . . . .	32
10.2	Proksimal modul . . . . .	32
10.3	Tilkobling av servomotorer . . . . .	33
10.4	Tilkobling av kraftsensorer . . . . .	34
10.5	Sammenkobling av komponentene . . . . .	35
<b>11</b>	<b>Testing</b>	<b>36</b>
<b>12</b>	<b>Konklusjon av realiseringen</b>	<b>36</b>
<b>IV</b>	<b>Avslutning</b>	<b>37</b>
<b>V</b>	<b>Vedlegg</b>	<b>38</b>
<b>A</b>	<b>Oversikt over båndbreddekrav for menneskets fingre</b>	<b>39</b>
<b>B</b>	<b>Programkode</b>	<b>40</b>
B.1	main.c . . . . .	40
<b>VI</b>	<b>Kilder</b>	<b>46</b>



## Figurer

1	Blokkdiagram av systemstruktur . . . . .	9
2	Eksempel på telemanipulator, hentet fra Shimoga [16] . . . . .	10
3	SISO-system [5] . . . . .	12
4	Blokkdiagram av portfunksjoner [5] . . . . .	12
5	Eksempel på system bestående av sammenkoblede undersystemer [17] . . . . .	13
6	Håndens båndbreddekrav for input og output, hentet fra [16] . . . . .	16
7	Tidsforsinkelser i systemet . . . . .	19
8	Aktivitetsdiagram for systemets drift . . . . .	24
9	Klassediagram for systemet . . . . .	25
10	FSR-motstand som funksjon av kraft [10] . . . . .	28
11	Aktuelle registre for PWM-generering på timer1 [1] . . . . .	29
12	AVR Butterfly og dens inn- og utganger [13] . . . . .	30
13	Protesemodell . . . . .	32
14	Proksimal modul (ikke tilkoblet) . . . . .	33
15	PWM-styringssignal før og etter forsterkning . . . . .	33
16	Tilkobling av servomotorer til Butterfly . . . . .	34
17	Tilkobling av FSR . . . . .	35
18	Ferdig oppkoblet system . . . . .	36
19	Oversikt over båndbreddekrav for menneskefingre, hentet fra [16] . . . . .	39

## Tabeller

1	Systemets sensorer og aktuatorer . . . . .	23
2	ADC-tilkobling på mikrokontroller og Butterfly . . . . .	27
3	Oppsummering av systemets IO på AVR Butterfly og ATmega169 . . . . .	30
4	Tillatte posisjoner for servoer i modellen . . . . .	34



## Del I

# Innledning

## 1 Formål med oppgaven

Formålet med oppgaven er å videreføre arbeidet med å utvikle en EPP-basert armprotese med minst to frihetsgrader, som ble påbegynt i prosjektoppgaven “Demonstrator for EPP-basert protesestyring” [18]. Protesen er ment for en dysmelipasient som har transversell armdysmeli og en rudimentær finger, der fingeren skal kunne styre protesen. Et premiss bak utarbeidelsen av oppgaven var at den skulle være rettet mot praktisk utnyttelse, heller enn å være en ren akademisk øvelse; resultatet skulle kunne videreutvikles til en funksjonell protese til den konkrete pasienten i første omgang, og potensielt også andre pasienter senere.

## 2 Bakgrunn

Dysmeli er definert som en ”medfødt tilstand med mangelfullt utviklede armer eller ben”, og er delt inn i transversell dysmeli, der alt nedenfor et visst nivå mangler, og longitudinal dysmeli der stråle i arm eller ben er mangelfullt utviklet. I tillegg forekommer det redusert størrelse eller mangel på deler av det rammede lemmet. Omtrent 20 barn blir født med dysmeli i Norge hvert år [9]. Pasienten som diskuteres i denne oppgaven er en seks år gammel pike, som har transversell dysmeli på høyre arm. Armen stopper rett over albuen, og avsluttes med en liten og relativt svak, men følsom og svært bevegelig finger. Pasienten har en myoelektrisk protese som brukes veldig sjelden; den er vanskelig å styre nøyaktig og innkapsler armen med den anvendelige fingeren hun har. Foreldrene har uttrykt et ønske om at fingeren benyttes i styringen av en fremtidig protese for å utnytte og bevare bevegelighet og styrke i fingeren. Det er ønskelig at pasienten finner en protese hun kan trives med i dagliglivet, både fordi det kan være et nyttig redskap og fordi den manglende vekten av armen som skulle ha vært der potensielt kan føre til skjevheter i kropp og spesielt ryggrad. Det ble diskutert i prosjektoppgaven [18] at EPP-prinsippet er en god kandidat for protesestyring i dette tilfellet. EPP (extended physiological proprioception) utnytter kroppens naturlige proprioepsjon, evnen til å ubevisst kjenne posisjonen og tilstanden til alle deler av kroppen, og det faktumet at proprioepsjonen utvides når man bruker redskap hvis stilling er entydig bestemt av kroppens posisjon. Dermed vil for eksempel en tennisspiller ikke trenge å se på racketen for å vite hvor den er eller hvordan den beveger seg, den oppfattes nesten som en forlengelse av spillerens egen arm. På samme måte kan en bruker styre en EPP-protese, og kjenne hvordan protesen beveger seg og om den møter motstand uten å måtte se på den. Det er gjort flere vellykkede forsøk med EPP-baserte proteser med skulderbevegelse som input, men det er så langt forfatteren har funnet ikke publisert forsøk med finger-input.

### 3 Motivasjon

Et stort antall dysmelipasienter har rudimentære ledd som potensielt kan brukes til protesestyring, men som ikke blir utnyttet. Det kan virke som om de fleste av dagens proteser er utviklet med tanke på amputasjonspasienter, og at disse dysmelistene som brukergruppe til en viss grad er blitt glemt. Mange potensielle protesebrukere velger bort protesen av forskjellige grunner; de synes de klarer seg bra nok uten, protesen er for upresis eller vanskelig å bruke, eller hjelpen protesen kan bidra med er ikke nok til å veie opp for ubehaget og den reduserte følsomheten man får i det innkapslede lemmet eller den ekstra innsatsen protesen krever i vedlikehold og bruk [2]. En dansk undersøkelse viste at så mange som 46% av armprotesebrukere eller foreldre til brukere i alderen 0-15 år var i tvil om protesebruken, og at mange som bruker protese som barn slutter i ungdomsalderen [9]. Håpet bak denne oppgaven er at antallet dysmelister som får reell hjelp ved protesebruk kan økes ved utvikling av proteser som er mer presise og intuitive i bruk. En stor overvekt av dysmelier rammer armer og hender [9], og det er mye som tyder på at det finnes et behov for proteser som kan nyttegjøres i pasientenes hverdag og fortsatt opprettholder og utnytter førlighet i eventuelle rudimentære ledd.

## Del II

# Litteratursøk

## 4 Introduksjon av litteratursøket

### 4.1 Fokus for teksten

Mennesket inngår som komponenter i mange dynamiske tilbakekoblingsløyper, som for eksempel ved styring av biler og fly eller ved innstilling av temperaturen i springen. Faktisk er alle systemer med menneskelig operatør eksempler på dette. I tillegg har man systemer som vi i større eller mindre grad påvirker i felleskap og der påvirkningen til en viss grad skjer ubevisst, som for eksempel global oppvarming, aksjemarkedet eller rovdyrbestanden i et område. Det å diskutere alle mulige systemer med mennesker som en del av tilbakekoblingen blir et uoverkommelig bredt tema. Derfor vil denne teksten innskrenkes til å gjelde systemer som direkte kan sammenlignes med det aktuelle EPP-systemet som skal bygges.

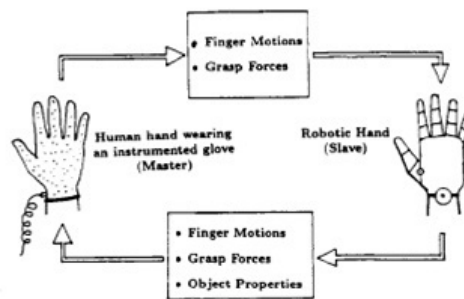
### 4.2 Presentasjon av systemet

EPP-systemet skal styres av brukerens finger ved hjelp av en proksimal del også omtalt som mastermodul eller master. I denne modulen måles positiv kraft fra fingeren i fire retninger (fleksjon/ekstensjon og adduksjon/abduksjon). Mastermodulen skal også sørge for at fingeren til enhver tid er i en posisjon som har er entydig bestemt av protesens stilling. På denne måten får brukeren alltid feedback om protesens posisjon. Protesen omtales som distal del, slave-modul eller slave. Denne skal reguleres slik at protesens hånd åpner seg når masteren mottar en kraft i ekstensjonsretning, lukker seg når masteren mottar en kraft i fleksjonsretning, og protesens håndledd vrir seg henholdsvis med eller mot klokka ved kraft i abduksjon- og adduksjonsretning. Kraften som protesen utøver på omgivelsene skal også være proposjonal med kraften som utøves på mastermodulen. Systemet er illustrert ved et blokkdiagram i figur 1; Fysiske interaksjoner er representert ved hule piler og signalinteraksjoner er representert ved fylte piler.  $F_{finger}$  og  $\hat{F}_{finger}$  representerer reell og målt kraft fra finger på proksimal del,  $x_{d\_protese}$  og  $\hat{x}_{protese}$  representerer ønsket og målt posisjon på protese,  $x_{finger}$  og  $x_{d\_finger}$  representerer reell og ønsket posisjon proksimal del skal holde brukerens finger i, og  $F_{protese}$  og  $F_{omgivelser}$  representerer henholdsvis kreftene protesen utøver på omgivelsene og kreftene omgivelsene utøver på protesen.



Figur 1: Blokkdiagram av systemstruktur

Tilsvarende systemer omtales i litteraturen under navn som telemanipulatorer, teleoperatorer eller haptiske enheter. Felles for disse er at man anvender en master-enhet til å måle brukerens bevegelser og gi feedback til brukeren. Denne brukes enten for å overføre hånd- eller fingerbevegelser til en robotisk slave-enhet og på denne måten håndtere objekter uten å berøre dem direkte, eller for å presentere brukeren med simulerte omgivelser og objekter i en virtuell virkelighet. Et eksempel på en teleoperator er vist i figur 2. Presentasjonen av berørings-feedback til brukeren omtales gjerne som haptisk fremstilling (haptic rendering), haptisk-, kinestetisk- eller taktil feedback, eller EPP. Utrykkene proprioepsjon og kinestesi brukes ofte om hverandre og betegner kroppens evne til å merke sin stilling og bevegelse, men kinestesi er muligens noe mer rettet mot bevegelse og proprioepsjon mot posisjon. Haptisk persepsjon er oppfattelsen av objekter ved berøring, og omfatter både proprioepsjon og taktil sansing. Taktil sansing er selve berøringssansen, der man oppfatter egenskaper som tekstur og temperatur.



Figur 2: Eksempel på telemanipulator, hentet fra Shimoga [16]

## 5 Resultat av litteratursøket

### 5.1 Regulerings tekniske betraktninger

#### 5.1.1 Selvbevissthet

En vesentlig, og ikke uproblematisk, egenskap ved systemer som inkluderer mennesker i tilbakekoblingsløyfen er at mennesket kan være bevisst på sin rolle som komponent i systemet. Denne selvbevisstheten kan føre til endringer i oppførsel som påvirker hele systemets dynamikk. Et eksempel på problemer på grunn av selvbevissthet er såkalte “pilot-induced oscillations”, der pilotens forsøk på å regulere flyet uten å ta tilstrekkelig hensyn til tregheter i systemet forårsaker ukontrollerte oscillasjoner som i værste fall kan medføre at flyet styrter. Hovedproblemet med selvbevissthet er at hver gang man endrer systemmodellen og regulatoren for å ta hensyn til menneskets nye dynamikk, kan mennesket igjen reagere med nye endringer i oppførsel.

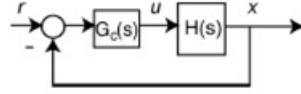
#### 5.1.2 Dynamikk i menneskelige lemmer

Dynamikken i en menneskearm er uforutsigbar, men variasjonen er bundet, blant annet i stivhet og i forholdet mellom kraft og akselerasjon. Det finnes også argumentasjon for at man selv om det ikke er bevist at menneskelemmer er passive, kan behandle dem som om de er det [5]. (Begrepet passivitet vil bli behandlet senere i teksten.) I Siciliano og Khatib [17] blir det påpekt at for impedans-styrte haptiske grensesnitt er det værste mulige scenarioet for stabilitet at operatøren slipper den haptiske enheten. Dermed kan vi velge å ignorere operatøren i analysen og anta at menneskekraften er lik null. Et system som da er stabilt vil også være stabilt hvis operatøren interagerer med enheten.

#### 5.1.3 Systemer som er koblet med omgivelsene; fysisk interaksjon

I Buerger og Hogan [5] er det definert at fysisk interaksjon oppstår når de statiske eller dynamiske egenskapene i et system dannet av sammenkobling av to systemer endres betydelig fra dynamikken i det regulerte systemet i isolasjon. Det vil si at et system som i utgangspunktet er stabilt og viser tilfredsstillende ytelse ikke nødvendigvis gjør dette når det blir koblet med et annet system. Fysisk interaksjon med omgivelser og med en menneskelig operatør gjør regulerings teknisk analyse av systemet vanskelig, fordi vanlige teknikker ofte er upassende. For eksempel [5] er det for et enkelt SISO-system (single input single output) fristende å fremstille systemet som et vanlig blokkdiagram (se figur 3) og bare legge til en blokk til høyre kalt ”omgivelser”, men denne enkle utvidelsen vil, etter definisjonen av fysisk interaksjon, gjøre at transferfunksjonen  $H(s)$  er ugyldig. Dermed må man heller inkorporere dynamikken fra omgivelsene inn i  $H(s)$ , noe som vil føre til gyldig regulering, men kun i ett, nøyaktig modellert tilfelle av omgivelser.

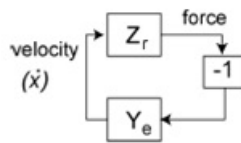
Det er to egenskaper ved SISO-eksempelet som gjør redskapene Bode- og Nyquistdiagram og rotkurver (root locus) nyttige; Selv om ytelse og stabilitet



Figur 3: SISO-system [5]

med lukket sløyfe avgjør systemets faktiske oppførsel, kan begge forutsees og karakteriseres ved å analysere åpen sløyfe-transferfunksjonen. Dermed vil manipulasjon av transferfunksjonen ha forutsigbare effekter på systemet med lukket tilbakekoblingsløyfe. Den andre egenskapen er at regulatoren opptrer som en direkte multiplikator til prosessen i transferfunksjonen, slik at endringer i regulatoren forutsigbart endrer åpen sløyfe –amplitude og fase (og derfor stabilitet og ytelse). Disse to egenskapene finnes ikke i interaksjonsproblemet.

Buerger og Hogan [5] presenterer også en modell basert på portfunksjoner (se figur4), som er mer lovende enn IO-oppsettet og er basert på ”conjugate power variables”. Systemet i figuren kan til en viss grad analyseres ved å bruke klassiske SISO-verktøy på transferfunksjonen  $Z_r Y_e$ , der  $Z_r$  er robotens impedans og  $Y_e$  er omgivelsenes admittans. Stabilitet kan avgjøres ved å se på polene til lukket sløyfe-systemet, eller røttene av  $(1 + Z_r Y_e)$ . Nyquist-plot av  $Z_r Y_e$  kan brukes til å evaluere stabilitet, og gi et visst inntrykk av stabilitetsmargin. Ytelse bør måles i forhold til port-oppførselen siden systemet ikke har relevante inputs og outputs, og kun vil relatere seg til  $Z_r$ ; uavhengig av miljøet. Robot-dynamikken spiller en kritisk rolle når man skal avgjøre interaksjonsstabilitet og ytelse, mens omgivelsenes dynamikk spiller en like stor rolle for stabilitet men er irrelevant for ytelse.



Figur 4: Blokkdiagram av portfunksjoner [5]

Selv om Buerger og Hogan [5] antyder at man kan bruke redskaper som Bode- og Nyquistdiagram, er det eksplisitt advart om at selv når analysen er begrenset til SISO-systemer og selv om både roboten og omgivelsene er modellert eksakt, så kan ikke designprosedyrer for servomekanismer overføres direkte til interaksjonsproblemet, men må brukes med forsiktighet. Griffiths og Gillespie [7] argumenterer også for at typiske frekvensdomene-teknikker for å vurdere stabilitet slik som fasemargin og forsterkningsmargin er upassende for haptisk fremstilling siden den fysiske interaksjonen med den menneskelige operatøren i



tilbakekoblingen introduserer betydelig ulineær, tidsuavhengig dynamikk.

#### 5.1.4 Passivitet



Figur 5: Eksempel på system bestående av sammenkoblede undersystemer [17]

En vanlig løsning for å sikre stabilitet i haptiske enheter er å bruke et passivitetskriterie [5, 7, 17, 21]. Den haptiske enheten med bruker og omgivelser blir sett på som et nettverk av undersystemer som er koblet sammen og interagerer gjennom porter (se eksempel i figur 5). Et system er passivt hvis og bare hvis det ikke kan produsere energi, dvs at output-energien fra systemet er begrenset av den initielle og akkumulerte energien i systemet. To passive systemer kan kombineres og danner et nytt passivt system. En tilbakekobling (feedback connection) av to passive systemer er også stabil. Noen eksempler på passivitetskriterier følger:

- “energetic passivity condition” [21]:

$$\int_0^t [F_1^T(\lambda)\dot{q}_1(\lambda) + F_2^T(\lambda)\dot{q}_2(\lambda)] d\lambda \leq c^2, \forall t \geq 0$$

der  $F_i$  er krefter fra menneske ( $i = 1$ ) og omgivelser ( $i = 2$ ), og  $\dot{q}_i$  er hastigheten av master ( $i = 1$ ) og slave ( $i = 2$ ) i “the tool frame”.

- “controller passivity” [21]:

$$\int_0^t [\tau_1^T(\lambda)\dot{q}_1(\lambda) + \tau_2^T(\lambda)\dot{q}_2(\lambda)] d\lambda \leq c^2, \forall t \geq 0$$

der  $\tau_i$  er kontroller-dreiemoment for master ( $i = 1$ ) og slave ( $i = 2$ ). Bevis av “controller passivity” impliserer “energetic passivity” og garanterer dermed stabilitet av telemanipulatoren.

- Et nødvendig og tilstrekkelig kriterie for at en lineært tids-invariant transferfunksjon mellom et par ”power”-variable, slik som kraft og hastighet, er passiv er at dens poler ligger i venstre halvplan og at Nyquist-plot ligger i det lukkede høyre halvplan [7].

Selv om passivitet er et tilstrekkelig men ikke nødvendig kriterie for stabilitet, og dermed kan være unødvendig restriktivt, så inkorporerer det usikkerheten i omgivelsene veldig godt [17]. Mange fysiske objekter er passive, bl.a. alle kombinasjoner av uvirksomme elementer, fjær og dempere, og det gjør at en stabil robot med passiv portfunksjon kan kobles til et stort og nyttig sett av omgivelser [5]. Generelt så antar man at slave-modulen kun interagerer med passive omgivelser uten aktive motorer eller lignende. Hvis man i tillegg antar det værste mulige scenariet for stabiliteten av systemet, nemlig at den menneskelige operatørens har sluppet master-modulen og hans påvirkning er lik

null, så er stabilitet sikret hvis det haptiske systemet er passivt. Dermed slipper man å tilføye en eksplisitt modell for omgivelsene [17]. Det finnes likevel en rekke utfordringer som kan ødelegge passivitet i reelle systemer. Blant annet gjelder dette diskret regulering, tidsforsinkelser, begrensninger i aktuatorer og sensorer, umodellert dynamikk, og hvis kraft-tilbakekobling er brukt til å redusere tilsynelatende endepunktstregghet (apparent endpoint inertia) til mer enn 50% under sin fysiske verdi [5]. I Griffiths og Gillespie [7] ble det vist at for et spesifikt telemanipulatorsystem finnes en klasse passive transferfunksjoner som ikke kan bli approksimert mens man samtidig utviser en passiv respons til den menneskelige operatøren. Denne klassen karakteriseres av kompensasjon av hardware-dynamikk.

### 5.1.5 Alternativer til passivitet

Mindre konservative koblede stabilitetskriterier enn passivitet kan være tilgjengelige gjennom robust stabilitetsanalyse med et antatt sett mulige brukerdynamikker [7]. Dette vil formilde konflikten mellom ytelse og stabilitet, men det kan likevel være andre trade-offs, for eksempel mellom ytelses-båndbredde og lukket sløyfe-båndbredde; for eksempel kan små økninger i ytelses-båndbredde forårsake stor økning i regulator-båndbredden.

Natural admittance control (NAC) tillater reduksjon av tilsynelatende friksjon uavhengig av tregghet, og tillater design av teoretisk passive friction-rejecting interaction controllers [5]. I teorien er passivitet (og koblet stabilitet) sikret hvis mål-treggheten holdes over halvparten av den fysiske verdien, og ytelse er manipulert ved å justere andre mål-impedans-parametere. NAC gir muligheten til å se til både stabilitet og ytelse i et nettverk, men potensielle trusler til stabiliteten slik som tidsforsinkelse etc kan ikke bli inkludert på en enkel måte; mål-tregghet er det eneste målet på passivitet eller stabilitet, og det eneste parameteret som kan designes for å oppnå en stabilitetsmargin.

I Buerger og Hogan [5] introduseres et stabilitetskriterie kalt komplementær stabilitet. Denne definisjonen forandrer koblet stabilitet til et robust stabilitetsproblem som kan løses vha "the small gain theorem". Artikkelen viser eksempel på at komplementær stabilitet anvendt på et koblet system med begrenset informasjon om menneskelig dynamikk er mindre restriktivt enn passivitet og tillater høyere ytelse. Fysisk implementasjon indikerer at modellen er mer konservativ enn den fysiske testen; altså at modellen forutser ustabilitet der systemet i praksis viser seg å være stabilt og ikke motsatt.

Passivitet kan som nevnt være overdrevent konservativt, og mange regulatorer blir overdempet hvis alle undersystemer er passive. I kontrast kan kombinasjonen av et aktivt og et passivt undersystem være passivt og stabilt og utvise mindre energitap. Fra nettverksteori spesifiserer Llewellyn-kriteriet når et muligvis aktivt to-port-element koblet med en hvilken som helst passiv en-port blir passiv. Denne aktive to-porten blir da kalt ubetinget stabil (unconditionally stable), ettersom den vil være stabil i kobling med hvilke som helst to en-porter[17].

## 5.2 Hardware-kriterier

I Shimoga [16] presenteres tre konstruksjonskrav til teleoperatører.

1. Både hånd-masteren og slave-robothånden må være kinematisk identiske til menneskehånden og dermed isomorfe [16].

Dette har sammenheng med at virkelighetsgjengivelse vil bli stekt endret om man endrer form eller størrelse på robothånden i forhold til menneskehånden. En liten robothånd kan for eksempel gi følelsen av at omgivelsene er urimelig store.

Asymmetri er også en utfordring. Det skilles mellom master sensor/aktuator asymmetri, der master-manipulatoren er ute av stand til å utøve ønskede krefter til operatøren i en eller flere av telemanipulatorens frihetsgrader, og slave sensor/aktuator asymmetri der slave-manipulatoren er ute av stand til å føle en eller flere av kreftene dens aktuatorer utøver på omgivelsene. Master-asymmetrier er vanligvis et resultat av færre aktuatorer enn posisjonssensorer på master-manipulatoren, eller av metning i motorer. Slave-asymmetrier er vanligvis et resultat av at det mangler en frihetsgrad i kraftmåling på slaven [21].

Telemanipulatorer med sensor/aktuator-asymmetri er i utgangspunktet ikke passive og kan også være ustabile. På grunn av designkriterier (kostnad, vekt osv) kan det likevel være uunngåelig. I slike tilfeller er det viktig at regulatoren gjør rede for asymmetrien.

2. Hånd-masteren må være designet slik at den har tilstrekkelig styrke, spesielt der hvor menneskefingeren oppfatter reaksjonskraften som resulterer fra finger-kraft-tilbakekoblingen [16].

Dette er et sentralt kriterie som man i EPP-artikler omtaler som "unbeatable servos" [3, 6, 14, 15, 19]. Sensorteknologi og datakapasitet forbedres stadig, men aktuator-teknologien sakker akterut og er den ledende begrensende faktoren innenfor terapeutiske robotmoduler ifølge Buerger og Hogan [5].

3. Enheten må tillate variasjon i menneskehånden, slik at flere operatører kan bruke den samme hånd-masteren [16].

I tillegg kan følgende konstruksjonskrav for proteser legges til fra andre kilder:

4. Statisk friksjon og slingring bør holdes på et minimum.

I Farrell et al.[20] ble det vist at statisk friksjon og slingring kan føre til oscillasjoner og rykkvise bevegelser i en armprotese.

5. Effektforbruk i styreenheten (microcontroller i vårt tilfelle) bør ikke overskride 5% av kildeeffekten.

Dette er en tommelfingerregel utviklet av Al-Angari et al. [4] fra tidligere erfaringer innen utvikling av elektriske proteser. Mikrokontrollerens effektforbruk er lineært relatert til krystallfrekvens og en lav frekvens er derfor ønskelig.

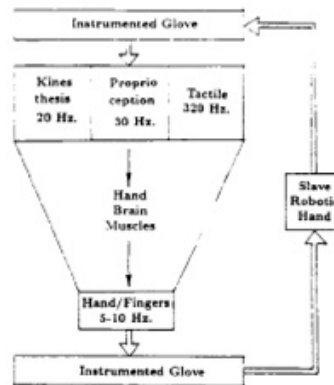
### 5.3 Fysiologiske krav fra hånden/mennesket

Shimoga [16] foreslår tre funksjonelle krav til teleoperatører.

1. Båndbredde: maksimal frekvens en typisk menneskehånd kan overføre bevegelseskommandoer til håndmasteren med er 5-10Hz, mens den krever at posisjons- og kraft- feedback blir presentert med en frekvens på minst 20-30Hz [16].

Feedback-løkken for kroppens individuelle kinestetiske sensorer er treg (mellom 0,5 og 1,7 Hz), men likevel kan mennesker produsere mye raskere komplekse bevegelser [8]. Minimum båndbredde for å gi kroppen en meningsfull oppfattelse av kraft er avhengig av gjøremål [5]. Krav til båndbredde og asymmetri i båndbreddekrav til input og output i hånden er gjengitt i fig 6 og vedlegg A I tillegg er proksimale leddrotasjoner (eks. skulder) mer følsomme, dvs har høyere båndbredde, enn distale ledd (eks. finger) [8].

For tilfredsstillende ytelse er det anbefalt å velge en båndbredde omtrent 10 ganger så stor som den minimale nødvendige båndbredden. Det vil si at masterens sensor-aktivitet, måling av vinkler, ratioer og krefter ikke må gjøres på mindre enn 100Hz mens kraft-tilbakekoblingen på fingrene ikke må være mindre enn 300Hz. Hvis operatøren ønsker å kjenne vibrasjoner på slaven, må kraft-tilbakekoblingen operere på minst 3kHz [16].



Figur 6: Håndens båndbreddekrav for input og output, hentet fra [16]

2. Operatørens tretthet og ubehag avhenger av gripekraft og arbeid/hvile-ratio. For en generell teleoperator vil operatørkomforten være innenfor trygge nivåer hvis gripekraften (eller refleksjons-/reaksjonskraften) er mindre enn omtrent 15% (dvs 30-50N) av den maksimale kraften operatøren kan utøve [16].

Menneskefinger kan utøve 30-50N over korte perioder mens de bare kan utøve 4-7N over lengre tid uten å påvirke evnen til å oppfatte størrelse og variasjon i kraft [16]. Childress et al. [4, 20] løser problemstillingen med operatørtretthet for en EPP-protese ved at kraft-spennings responsen er implementert slik at det for inputkrefter lavere enn en gitt terskel ikke vil bli noen output-spenning. Det vil si at protesen er låst og brukeren kan slappe av. I tillegg minker effektforbruket da servoene ikke er i drift.

3. Kraft-følsomhet i hånd-masteren bør tilsvare minst 10 ganger følsomheten i en menneskehånd, som kan kjenne kraftvariasjoner på 0,5N. Minimum trykk som en finger kan kjenne er  $0,2N/cm^2$  [16].

I tillegg kan man legge til følgende punkter fra andre kilder:

4. Hale og Stanney [8] angir posisjonsfølsomhet i noen av leddene i kroppen, gitt ved minste følbare forskjell (Just noticeable difference, JND):
  - skulder: 0,8 grader
  - albu/håndledd: 2 grader
  - finger: 2,5 grader

JND-verdiene varierer noe avhengig av bevegelsens retning (fleksjon/ekstensjon) og fart. Ofte merker en person at en bevegelse har skjedd før retningen merkes. Derfor må passiv kinestetisk stimulus inneholde nok bevegelse til å indikere retning.

5. Tidsforsinkelse vil i utgangspunktet føre til en ustabil respons fra brukeren, uavhengig av forsinkelsens størrelse [11].

Resultatene i Mendez-Inglesias og Ruiz-Sanchez [11] indikerer at en operatør som utfører en fri bevegelse- teleoperasjonsoppgave uten kraft-tilbakekobling i et tredimensjonalt arbeidsområde, med tidsforsinket visuell tilbakekobling, i utgangspunktet ikke har stabil oppførsel. Stabiliteten ble ikke forværret av økt tidsforsinkelse, men den er avhengig av mengden trening operatøren har fått i form av repetisjon av oppgaven.

## 6 Diskusjon av litteratursøket

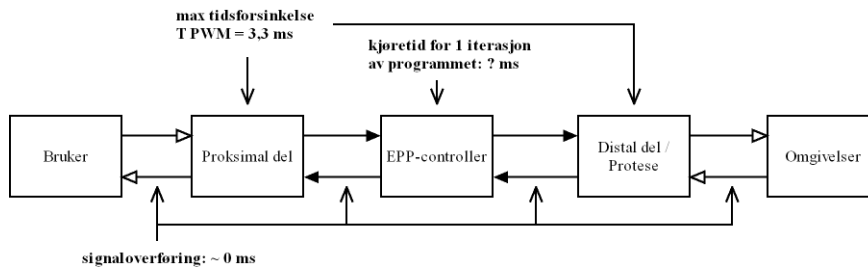
### 6.1 Reguleringstekniske betraktninger

Systemet i oppgaven er koblet med omgivelsene, og påvirkes av fysisk interaksjon både med brukerens finger i master-enden av systemet, og ved for eksempel objekter som gripes i slave-enden. Det er vanskelig å gjøre noen fullstendig reguleringsteknisk analyse, både av grunnene gitt i artikkelen og fordi mange av de konkrete parameterne i systemet ikke er kjent. Det er likevel mulig å

avgjøre om systemet er stabilt ved å se på passivitet. Både menneskearmen og omgivelsene kan antas å være passive. Servoene styres av en intern regulatorkrets som gjør at de også er passive. Kraft- og posisjonsmålerne er sensorer som ikke innfører dynamikk i systemet, og er derfor også passive. Mikrokontrolleren og programmet som skal styre servoene kan nødvendigvis innføre ustabil dynamikk, og man må sørge for at programmet styrer systemet passivt. Det er likevel nevnt noen utfordringer for passivitet som må tas i betraktning:

- Diskret regulering: Det tas målinger og posisjonen for servoene settes med diskrete intervaller, men det antas at frekvensen er høy nok til at prosessen kan sees på som kontinuerlig.
- Tidsforsinkelser: Tidsforsinkelse i de forskjellige delene av systemet er gjengitt i figur 7. Servoene styres av et PWM-signal som oppdaterer deres posisjon med en gitt PWM-frekvens. Dette gjør at man i verste fall må vente en hel PWM-periode for å få oppdatert servoens posisjon, og i beste fall skjer oppdateringen straks. Selve signaloverføringen mellom modulene skjer svært fort, og det antas at tidsforsinkelsen her er neglisjerbar. Maksimal forsinkning fra en servo i slave-modulen endrer posisjon og til tilsvarende posisjon settes i mastermodulen er dermed det dobbelte av perioden. I denne oppgaven benyttes digitale servoer slik at man kan man oppnå en maksimal tidsforsinkelse på så lite som  $6,7ms$ , som er lavt nok til at det ikke burde merkes av brukeren.
- Begrensninger i aktuatorer og sensorer: Det er valgt potensiometere som kan måle hele området protesen kan bevege seg i. FSR'er er mer følsomme jo mindre kraft de påtrykkes, og kan måle trykk over  $100N/cm^2$  [10]. Aktuatorene i mastermodulen har et dreiemoment på  $24N \cdot cm$ . En rask test med en vekt antyder at forfatteren klarer å trykke med en kraft på omtrent  $20N$  med pekefingeren, og dysmelipasienten som skal bruke protesen har betydelig mindre kraft i sin finger. Både kraftsensorer og aktuatorer er dermed passende til prosjektet. At servoene faktisk er "unbeatable" er et viktig moment i systemet. Om dette ikke er tilfellet vil systemet likevel ikke bli ustabil, men det vil ikke gi den nødvendige feedback'en til brukeren og konseptet bak systemet faller sammen.
- Umodellert dynamikk: Det er alltid en mulighet for at det finnes dynamikk som designeren ikke har tenkt på, og som kan sette stabiliteten til systemet på spill.

I dette prosjektet antas servoene å være stabile, og det vil derfor ikke være behov for å kompensere for hardware-dynamikk. Det konkluderes derfor med at systemet som helhet vil være stabilt såfremt det ikke oppstår alvorlig uforutsett dynamikk eller tidsforsinkelsen viser seg å bli for stor.



Figur 7: Tidsforsinkelser i systemet

## 6.2 Hardware-kriterier

1. Både hånd-masteren og slave-robot hånden må være kinematisk identiske til menneskehånden og dermed isomorfe [16].

Dette vil ikke være mulig i vårt tilfelle da hele poenget med protesen er å gi pasienten en hånd der det ikke finnes en. I stedet fokuseres det på å gi brukeren en enkel og entydig sammenheng mellom stillingen på fingeren i forhold til protesen. Asymmetrier unngås i dette systemet. Masteren kan utøve ønskede krefter på operatøren tilsvarende alle telemanipulatorens frihetsgrader (master sensor/aktuator-symmetri). Slave-manipulatoren kan måle alle kreftene dens aktuatorer utøver på omgivelsene (slave sensor/aktuator-symmetri). For hver av kreftene som måles av operatøren på masteren gis et tilsvarende pådrag av slaven, og for hver måling på slaven gis et tilsvarende pådrag av masteren på operatøren.

2. Hånd-masteren må være designet slik at den har tilstrekkelig styrke, spesielt der hvor menneskefingeren oppfatter reaksjonskraften som resulterer fra finger-kraft-tilbakekoblingen [16].

Dette er et viktig krav i denne oppgaven. Hele konseptet er avhengig av at servoene i bruk må være sterkere enn brukerens finger (unbeatable).

3. Enheten må tillate variasjon i menneskehånden, slik at flere operatører kan bruke den samme hånd-masteren [16].

Dette punktet er ikke relevant når man snakker om proteser, da proteser tilpasses den enkelte bruker og ikke er ment å kunne brukes av flere operatører.

4. Statisk friksjon og slingring bør holdes på et minimum [20].

Punktet stemmer, men problemstillingen er nok i mindre grad relevant når det er snakk om en håndprotese enn i forbindelse med en hel armprotese fra skulderen og ned slik som det er snakk om i Farrel et al. [20].

5. Effektforbruk i styreenheten bør ikke overskride 5% av kildeeffekten [4].

Dette punktet er en tommelfingerregel og skal sikre at man holder effektforbruket lavt, noe som er viktig for en protese som skal brukes til daglig. I denne oppgaven lages derimot en demonstrator som vil være koblet til et strømuttak i veggen under testing, og effektforbruk vil ikke være et diskutert tema.

### 6.3 Fysiologiske krav

1. Båndbredde: maksimal frekvens en typisk menneskehånd kan overføre bevegelseskommandoer til håndmasteren med er 5-10Hz, mens den krever at posisjons- og kraft- feedback blir presentert med en frekvens på minst 20-30Hz [16].

Shimoga [16] foreslår en absolutt minimums-båndbredde for posisjons- og kraft-feedback på 20Hz, men anbefaler samtidig en båndbredde på minimum 300Hz for å oppnå meningsfull proprioepsjon. Aktuatorene i denne oppgaven vil styres av et PWM-signal på 300Hz. Dette følger anbefalingen fra Shimoga og er nær 320Hz-grensen mellom hva man oppfatter som enkeltkrefter og kontinuerlig påvirkning (se vedlegg A).

2. Operatørens tretthet og ubehag avhenger av gripekraft og arbeid/hvile-ratio. For en generell teleoperator vil operatørkomforten være innenfor trygge nivåer hvis gripekraften (eller refleksjons-/reaksjonskraften) er mindre enn omtrent 15% av den maksimale kraften operatøren kan utøve (dvs 30 – 50N) [16].

Brukeren av denne typen protese vil forhåpentligvis bruke og styre protesen på en måte og i en mengde som tilsvarer bruken av en vanlig hånd. Fingeren til brukeren er med all sannsynlighet svakere enn en normal finger, men dette kan kompenseres for med å innføre en forsterkning mellom brukerens kraftpådrag og gripekraften i protesen. For statiske posisjoner innføres det en nedre grense for pådraget som skiller mellom kraften fra fingeren når den holdes i ro og kraften når brukeren ønsker å bevege protesen. På denne måten vil operatørtretthet forhåpentligvis ikke være nevneværdig større enn normal tretthet når man beveger en naturlig hånd.

3. Kraft-følsomhet i hånd-masteren bør tilsvare minst 10 ganger følsomheten i en menneskehånd, som kan kjenne kraftvariasjoner på 0,5N. Minimum trykk som en finger kan kjenne er 0,2N/cm<sup>2</sup> [16].

Kraftsensorene brukt i dette prosjektet har en angitt minimums-trykkfølsomhet på < 1N/cm<sup>2</sup> [10], som i utgangspunktet ikke overholder anbefalingen i Shimoga [16], og vil være en naturlig feilkilde å sjekke ved eventuelle problemer med systemet. Tidlige tester antyder likevel at sensorene måler selv veldig lette berøringer, og forfatteren mener derfor likevel at sensorene burde fungere. Dette vil fort vise seg under testingen av systemet.

4. Posisjonsfølsomhet er angitt som 2 grader i albuen og 2,5 grader i fingre [8].



Det er naturlig å anta at pasientens finger, som er i albuehøyde, har en posisjonsfølsomhet et eller annet sted mellom 2 og 2,5 grader. For å spare systemet for små justeringer som ikke inneholder noe informasjon for brukeren kan det være hensiktsmessig å filtrere ut posisjonsendringer på mindre enn 2 grader innenfor en gitt tid.

5. Tidsforsinkelse vil i utgangspunktet føre til en ustabil respons fra brukeren, uavhengig av forsinkelsens størrelse [11].

Servoenes PMW-frekvens på 300Hz gir en maksimal forsinkelse fra bevegelse i protesen til tilsvarende bevegelse i masteren på omtrent 6,7 ms, som er helt på grensen av det som vil være mulig for brukeren å oppfatte som diskrete posisjons-inputs. Det antas derfor at brukeren vil være i stand til å gi en stabil respons.

## 7 Konklusjon av litteratursøket

Mennesket og de relevante omgivelser kan antas å være passive, det samme er de valgte komponenter i systemet, med forutsetning at programvaren ikke innfører ugunstig dynamikk. Det totale systemet er passivt, og dermed også stabilt. Det må tas hensyn til følgende utfordringer for passivitet:

1. Diskret regulering
2. Tidsforsinkelser
3. Begrensninger i aktuatorer og sensorer
4. Umodellert dynamikk

Følgende hardware-kriterier er relevante for systemet:

1. En enkel og entydig sammenheng mellom master- og slaveposisjon må presenteres til brukeren, og asymmetrier i systemet bør unngås.
2. Aktuatorene som brukes må være “unbeatable”, altså sterkere enn brukers finger.
3. Statisk friksjon og slingring bør holdes på et minimum.
4. Effektforbruk i styreenheten i den endelige protesen bør ikke overskride 5% av kildeeffekten.

Følgende fysiologiske forutsetninger må tas hensyn til:

1. Absolutt minimum båndbredde for kraft- og posisjonsfeedback er 20Hz, men en båndbredde på minst 300Hz er anbefalt for å oppnå meningsfylt proprioepsjon.

2. Man bør ta hensyn til operatørens tretthet, f.eks. ved innføring av dødsone for protesestyringen. For en generell teleoperator vil operatørkomforten være innenfor trygge nivåer hvis gripekraften er mindre enn omtrent 15% (dvs 30-50N) av den maksimale kraften operatøren kan utøve.
3. Det er anbefalt med en kraft-følsomhet for mastermodulen på minst 0,05N.
4. Posisjonsendringer på under 2 grader vil ikke kunne merkes av brukeren.
5. Tidsforsinkelser på over 50ms vil i utgangspunktet føre til en ustabil respons fra brukeren, uavhengig av i hvor stor grad denne grensen overskrides.

## Del III

# Realisering av systemet

## 8 Programvarestruktur

Proksimal del / master / styringsmodul	
sensorer (inputs)	formål
FSR1	Måler kraft oppover fra finger (ekstensjon)
FSR2	Måler kraft nedover fra finger (fleksjon)
FSR3	Måler kraft mot venstre fra finger (adduksjon)
FSR4	Måler kraft mot høyre fra finger (abduksjon)
aktuatorer (outputs)	formål
Servo1	Holder/styrer fingerens ekstensjon og fleksjon
Servo2	Holder/styrer fingerens adduksjon og abduksjon

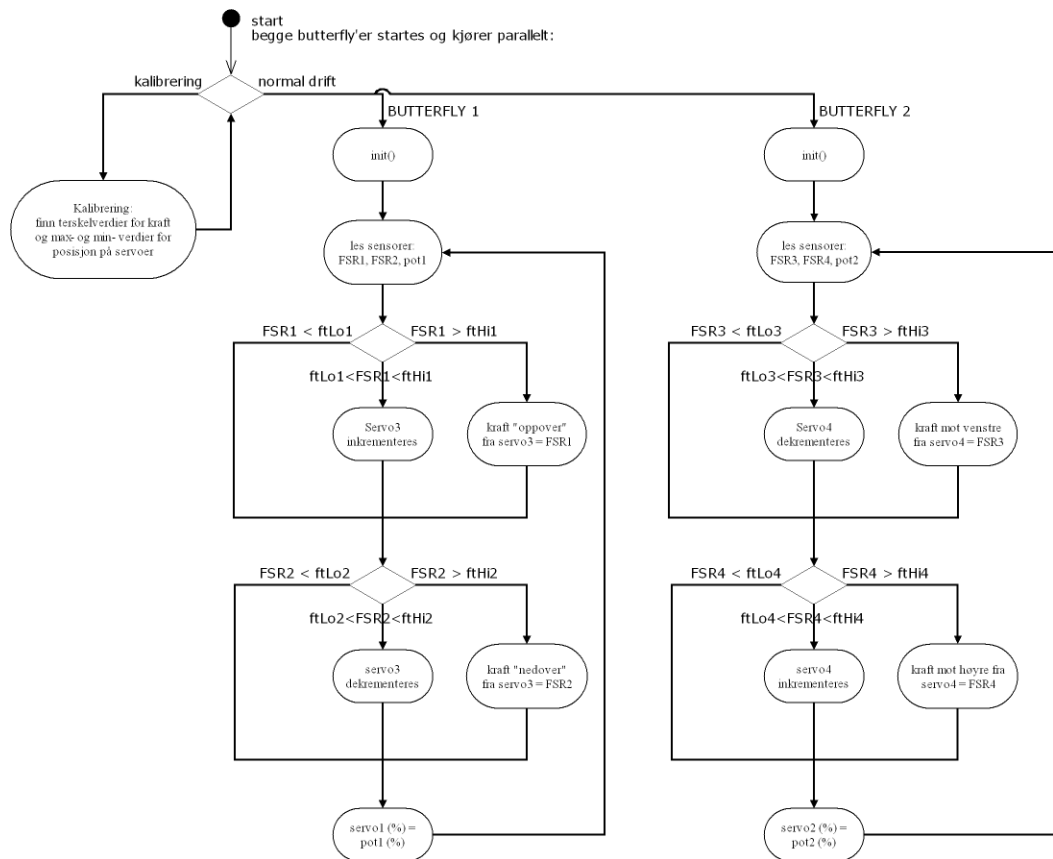
Distal del / slave / protese	
sensorer (inputs)	formål
Pot1	Måler protesens ekstensjon/fleksjon
Pot2	Måler protesens håndleddsvidning
aktuatorer (outputs)	formål
Servo3	Holder/styrer protesens ekstensjon/fleksjon
Servo4	Holder/styrer protesens håndleddsvidning

Tabell 1: Systemets sensorer og aktuatorer

Programmet må av praktiske grunner deles opp i to deler som skal implementeres på hver sin microkontroller på separate AVR Butterfly testkretser. Butterfly 1 relaterer fingerens fleksjon og ekstensjon mot protesens fleksjon og ekstensjon, og Butterfly 2 relaterer fingeren adduksjon og abduksjon mot protesens håndleddsvidning. Systemets sensorer og aktuatorer og deres formål er gjengitt i tabell 1. Kraftmåling på distal del er ikke nødvendig dersom det implementeres meningsfylt posisjonsmåling, slik at kraft kan regnes ut vha posisjonsmålingene. Posisjonsmåling på proksimal del er heller ikke nødvendig da servoene er posisjonsstyrt og antatt ”unbeatable”.

De to tilsvarende programmene kjøres parallelt på hver sin butterfly og går igjennom en evig løkke. Først leses verdier fra sensorene, og deretter vil programmet velge en passende respons i forhold til avlesningen for hver av kraftsensorene. Hvis verdien er under den nedre terskelverdien (ftLoX) indikerer dette at brukeren slapper av i fingeren, protesen skal være ”parkert” og den aktuelle aktuatorene vil forbli i samme posisjon. Hvis verdien er mellom nedre (ftLoX) og øvre (ftHiX) terskelverdi indikerer det bevegelse i gitt retning uten at protesen møter motstand, og aktuatorens posisjon vil henholdsvis inkrementeres eller dekrementeres avhengig av kraftens retning. Verdier over den øvre terskelen

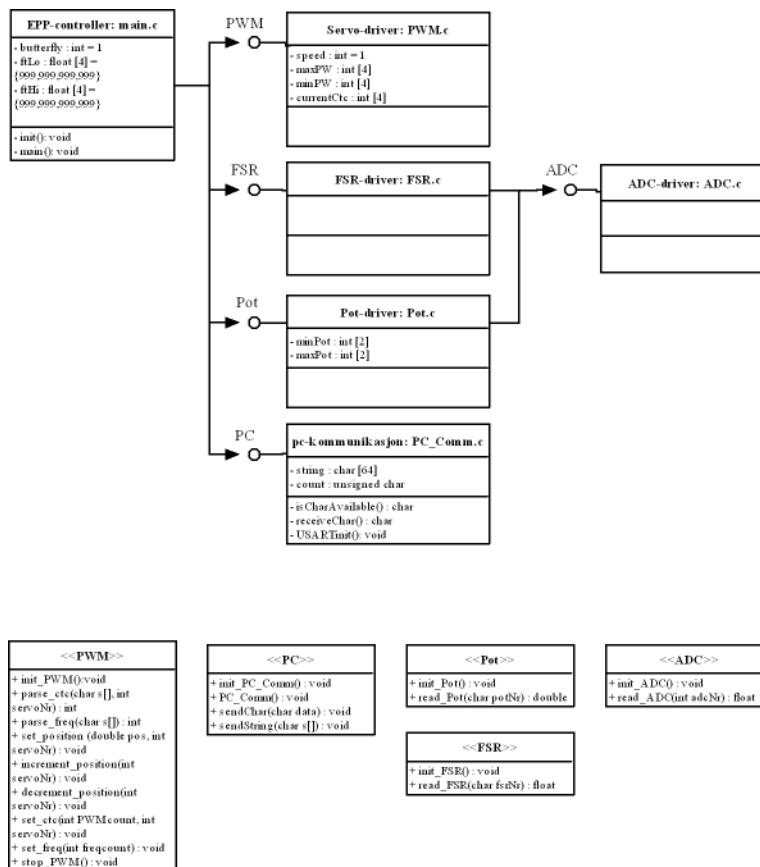
(ftHiX) indikerer at protesen møter motstand, for eksempel ved at den griper et objekt. I dette tilfellet skal kraften utøvet av aktuatoren reflektere kraften utøvet av brukeren. Til slutt skal master-modulens posisjon settes slik at den får brukerens finger til å følge entydig posisjonen av protesen. Posisjonene relateres til hverandre ved at de i stedet for å uttrykkes som vinkler vil uttrykkes som en prosentandel av maksimalt utslag. Prosedyren er illustrert ved et aktivitetsdiagram i figur 8.



Figur 8: Aktivitetsdiagram for systemets drift

Systemets to programmer skilles bare ved en enkelt variabel og vil benytte seg av felles drivere. Servo-driveren sørger for å sette rett posisjon til aktuatorene, og Pot-driveren og FSR-driveren leser av verdier fra sensorene. Siden begge disse i praksis vil være spenningsverdier benytter de en felles ADC-driver for å lese av analoge spenninger fra sine respektive innganger. Et klassediagram for systemet er vist i figur 9. Terskelverdiene ftLoX og ftHiX initieres til en

vilkårlig men høy verdi fordi det er bedre at verdiene ved en feil er for høye (protese og proksimalmodul står stille) enn at de er for lave (protese og proksimalmodul beveger seg uhemmet).



Figur 9: Klassediagram for systemet

## 9 Realisering av programvare

### 9.1 Introduksjon til realisering av programvare

I prosjektoppgaven [18] ble det foreslått å realisere programmet på en AVR Butterfly, hvilket ble fulgt opp i denne oppgaven. AVR Butterfly er en ferdigmontert testkrets som blant andre komponenter er utstyrt med en atMega169 mikrokontroller. Det ble etter hvert klart at kun 2 av mikrokontrollerens 4 PWM-kanaler kan generere et signal av ønsket frekvens. I tillegg er kun 5 av mikrokontrollerens 8 ADC-innganger tilgjengelige på Butterfly'en, noe som ville kreve en ekstern "sample-and-hold"-krets for å kunne bruke digitalisere signalet for å bruke en digital inngang til den siste analoge sensoren. I stedet for å skrive komplisert ekstra kode og legge til elektronikk eller bytte mikrokontroller midt i arbeidet vil det bli brukt to Bytterfly'er for å realisere systemet, da de to frihetsgradene i systemet kan styres uavhengig av hverandre.

For å realisere systemet er det brukt følgende programvare og utstyr:

- Bray's terminal
- AVR Studio 4
- Programmers Notepad
- 2 stk AVR Butterfly
- USB til RS232 adapter med FTDI chip.

Programvaren er skrevet i C, og drivere er basert på eksempler fra smiley micros Pardue [12] og demo-programvaren som følger med butterflyen. Tidvis er kun små endringer og tilpasninger gjort.

### 9.2 Kommunikasjon med PC

Kommunikasjon med PC er ikke nødvendig for bruk av protesen, men kan være et nyttig verktøy for kalibrering og debugging. Til formålet brukes et ferdig program fra Pardue [12] med små endringer, og finnes i vedlagt zip-fil under navnet PC\_Comm.c.

### 9.3 ADC-driver (analog- digital converter)

ATmega169 inneholder en analog-til-digital converter som er koblet til en 8-kanals analog multiplekser, og tillater dermed 8 input-spenninger på pin'ene på Port F (PF[0:7]). På Butterfly'en er imidlertid noen av ADC-inngangene brukt til andre formål; ADC0,2 og 3 benyttes til temperatur- og lysmåling, mens ADC1 har blitt tildelt en egen inngang på siden av butterfly'en ment for analog spenningsmåling. ADC[4:7] er koblet til JTAG-porten og kan benyttes hvis JTAG ikke er i bruk. ADC-pin'ene og deres tilsvarende kobling på Butterfly'en er oppsummert i tabell 2.

Pin på ATmega169	Pin på Butterfly
PF0 (ADC0)	Temperatursensor (ikke tilgjengelig)
PF1 (ADC1)	ADC (inkluderer spenningsdeler)
PF2 (ADC2)	Sensor_2 / lyssensor (ikke tilgjengelig)
PF3 (ADC3)	VCP (brukes internt på butterfly, ikke tilgjengelig)
PF4 (ADC4)	JTAG1
PF5 (ADC5)	JTAG5
PF6 (ADC6)	JTAG3
PF7 (ADC7)	JTAG9

Tabell 2: ADC-tilkobling på mikrokontroller og Butterfly

ADC-driveren er basert på ADC-programmer funnet i Pardue [12] og AT-MEL's demo-software. Disse er skrevet for avlesing av spenning på ADC-inngangen på Butterfly'en via en innebygget spenningsdeler. For å kunne bruke JTAG-inngangene i stedet så måtte det kompenseres for at disse ikke inkluderte noen spenningsdeler. ADC-driveren kan sees i vedlagt zip-fil under navnet ADC.c, og benyttes i drivere for både FSR og potensiometer.

## 9.4 FSR-driver

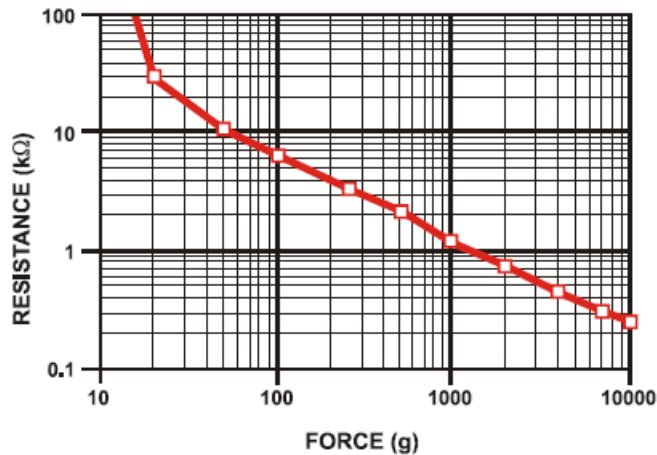
Signalet fra kraftsensorene er en spenning som er relatert til FSR-motstanden slik at spenningen blir større jo mer kraft FSR'en utsettes for. Ideelt sett ville read\_FSR-funksjonen returnert kraften på gjeldende FSR, men det viste seg å være vanskelig å finne noen nøyaktig formel for å regne om motstand i FSR'en til kraft. Det finnes imidlertid grafer basert på målinger som viser et omtrentlig forhold mellom kraft og motstand (fig 10), men det er vanskelig å lese av nøyaktige verdier. I stedet for å implementere en unøyaktig omregning mellom spenning og kraft, ansees det som en bedre løsning å bruke spenningsmålingene direkte, da formålet med sensorene uansett er å avgjøre om kreftene utøvet av brukeren befinner seg innenfor gitte intervaller.

Hadde man skullet implementere kraftmåling på protesen for å kunne gi nøyaktig feedback om denne til brukeren, kunne man løst problemet med en kalibrering. For eksempel kan dette gjøres ved å la protesehånden klemme på en FSR med forskjellige kraft-pådrag og sette opp en tabell som relaterer kraft-pådraget med spenningsmålinger fra FSR'en slik at kraften fingeren utøver på proksimalmodulen skal være lik kraften protesen utøver på omgivelsene.

Programkode for FSR-driveren finnes i vedlagt zip-fil under navnet FSR.c.

## 9.5 Potmeter-driver

Potensiometer-driveren leser av en spenningsverdig fra potensiometeret ved hjelp av ADC-driveren. Denne regnes om til posisjon representert ved en prosent verdi av maksimalt utslag. Det er valgt å bruke prosent ved posisjonsberegninger fordi det virker naturlig at maksimalt og minimalt utslag i protesen skal stemme



Figur 10: FSR-motstand som funksjon av kraft [10]

overens med maksimalt og minimalt utslag for fingeren, og tilsvarende for alle posisjoner i mellom disse grensene. Programmet ligger i vedlagte zip-fil under navnet Pot.c

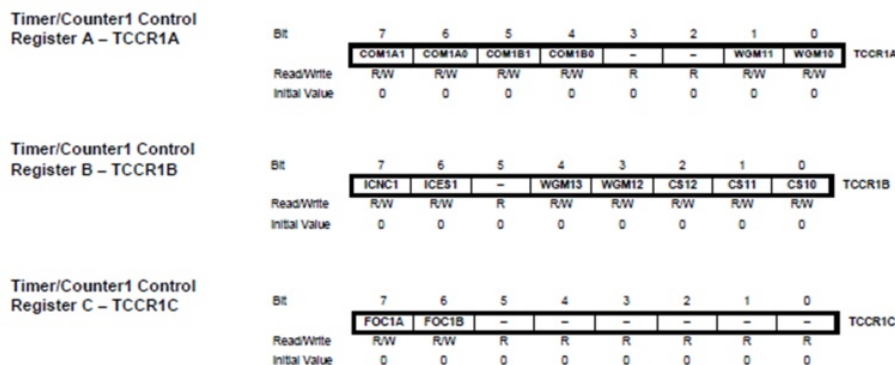
## 9.6 PWM-/servodriver

Som aktuatorer brukes digitale servoer. Deres posisjon styres ved hjelp av et pulsvidde-modulert (PWM-) signal på 300Hz. For å kunne velge PWM-frekvens fritt må timeren til mikrokontrolleren være i "phase and frequency correct PWM mode". På ATmega169 er denne modusen bare tilgjengelig på timer1, med to kanaler. Det vil derfor være nødvendig å bruke to butterfly'er for å implementere systemet vha mikrokontrollerens innebyggede PWM-kanaler. Alternativet er å generere PWM-signalet i software, men dette vil være komplisert, og mindre robust enn hardware-PWM.

For å kunne sette PWM-kanalen til rett type output så må man sette noen nøkkelvordier i kontrollregisterne til Timer1 (figur 11):

- WGM13:10 = 1000 setter pwm-modus phase and frequency correct, ICR1 top. (se datablad s 119)
- COM1A1/COM1B1: COM1x1:0 = 10 resetter OC1A/OC1B ved compare match på telling popover, og setter OC1A/OC1B ved compare match på nedtelling.
- CS12:10 = 001 setter klokkefrekvens/1, altså ingen prescaling.
- FOC1A/FOC1B må være 0 ved bruk av PWM.





Figur 11: Aktuelle registre for PWM-generering på timer1 [1]

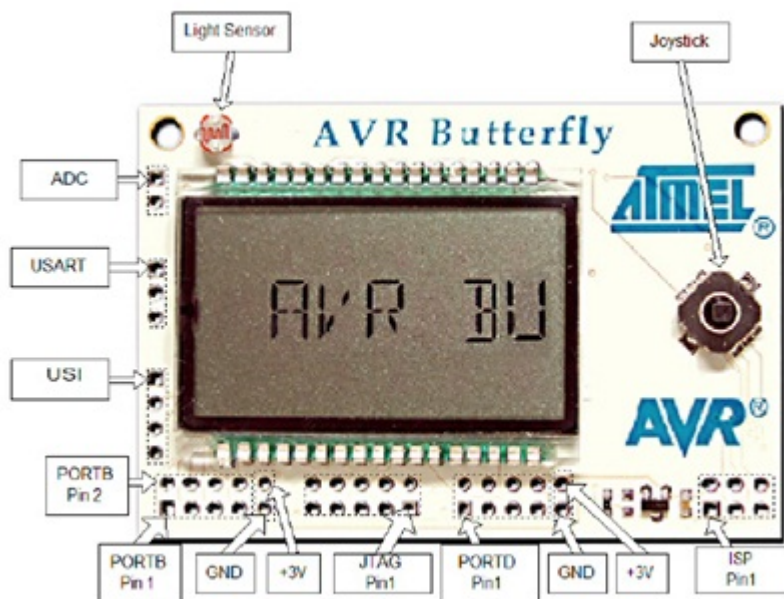
PWM-frekvensen settes ved å skrive en verdi til ICR1-registeret, bestemt av likning 1 [12]:

$$ICR1 = \frac{f_{clk}}{2 \cdot f_{PWM}} \quad (1)$$

Pulsbredden som avgjør posisjonen på servoen, settes ved å skrive en verdi til OCRA1- og OCRB1-registeret. OCRA1 er også koblet til et piezo-element på butterflyen, noe som vil føre til en bieffekt av PWM-signalet ved at butterflyen produserer en tone med samme frekvens som PWM-signalet. PWM-driveren ligger i vedlagt zip-fil under navnet PWM.c.

## 9.7 Oppsummering av IO på AVR Butterfly

Butterflyens IO er oppsummert i figur 12 og tabell 3.



Figur 12: AVR Butterfly og dens inn- og utganger [13]

<b>Butterfly 1</b>			
navn	funksjon	IO-pin på ATmega169	IO-pin på Butterfly
<b>Sensorer (inputs)</b>			
FSR1	M:Måler kraft oppover fra finger (ekstensjon)	BF1 PortF4	BF1 JTAG1
FSR2	M:Måler kraft nedover fra finger (fleksjon)	BF1 PortF6	BF1 JTAG3
Pot1	S:Måler protesens ekstensjon/fleksjon	BF1 PortF5	BF1 JTAG5
<b>Aktuatorer (outputs)</b>			
Servo1	M:Holder/styrer fingerens ekstensjon og fleksjon	BF1 OC1A/PB5	BF1 PortB6
Servo3	S:Holder/styrer protesens ekstensjon/fleksjon	BF1 OC1B/PB6	BF1 PortB7
<b>Butterfly 2</b>			
navn	funksjon	IO-pin på ATmega169	IO-pin på Butterfly
<b>Sensorer (inputs)</b>			
FSR3	M:Måler kraft mot venstre fra finger (adduksjon)	BF2 PortF4	BF2 JTAG1
FSR4	M:Måler kraft mot høyre fra finger (abduksjon)	BF2 PortF6	BF2 JTAG3
Pot2	S:Måler protesens håndleddsvidning	BF2 PortF5	BF2 JTAG5
<b>Aktuatorer (outputs)</b>			
Servo2	M:Holder/styrer fingerens adduksjon og abduksjon	BF2 OC1A/PB5	BF2 PortB6
Servo4	S:Holder/styrer protesens håndleddsvidning	BF2 OC1B/PB6	BF2 PortB7

Tabell 3: Oppsummering av systemets IO på AVR Butterfly og ATmega169

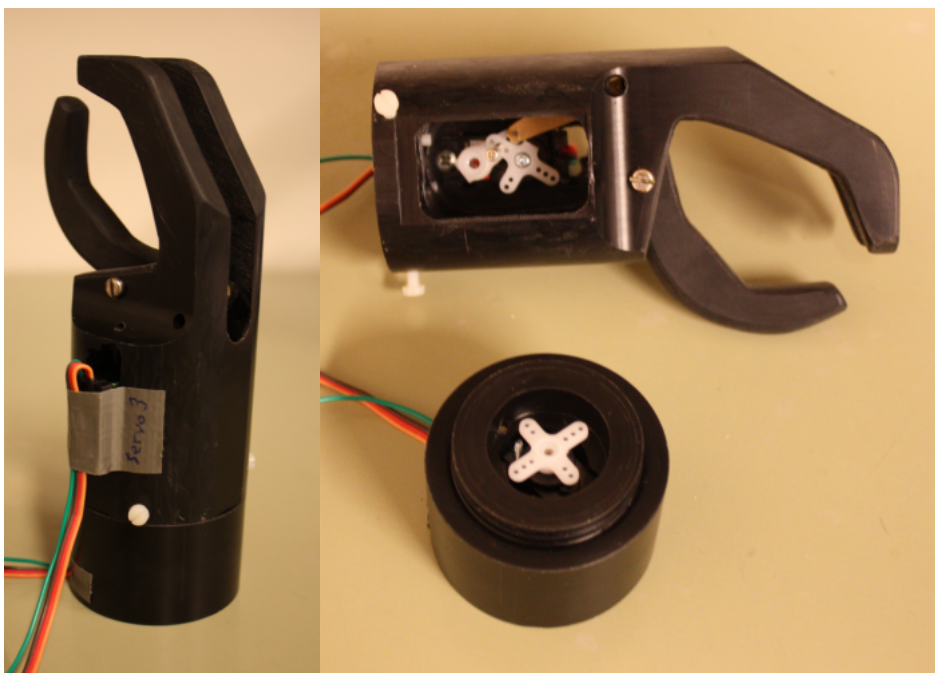
## 9.8 Hovedprogrammet: EPP-controller

Hovedprogrammet initierer alle driverne, og kjører deretter en evig løkke med pwm-algoritmen vist i aktivitetsdiagrammet (fig 8). Det tar seg også av kommunikasjon med PC når dette er aktuelt. I tillegg har kalibrering av systemets oscillatorfrekvens blitt plassert i hovedprogrammet. Programmet er gjengitt i vedlegg B.1 og er også vedlagt i zip-filen under navnet main.c.

## 10 Beskrivelse av protesemodellen

### 10.1 Distal modul / protesemodell

Den distale modulen ble bestilt hos mekanisk verksted ved institutt for teknisk kybernetikk. Modellen anvender to Huida HD-2216 digitale servoer som er modifisert med tilkoblinger for å kunne lese av målinger fra det innebygde potensiometeret. Modellen er vist i sin helhet, og demontert for å vise servomotorenes plassering i figur 13.



Figur 13: Protesemodell

### 10.2 Proksimal modul

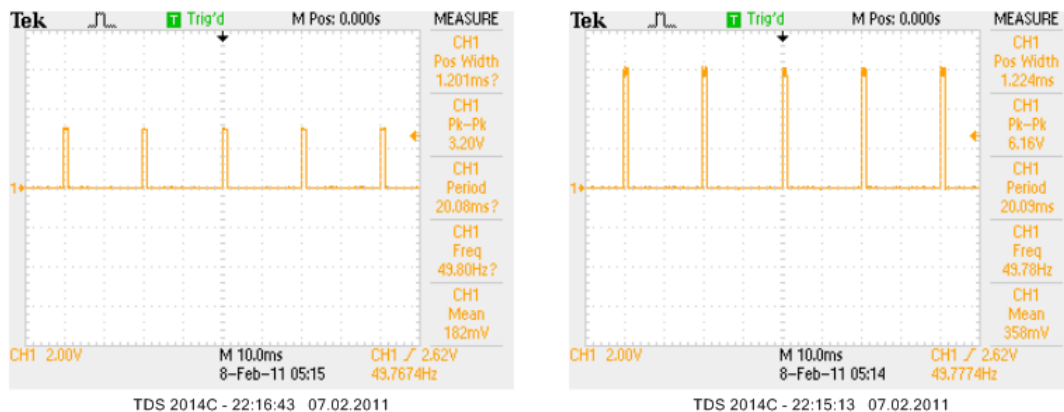
Den proksimale modulen ble laget med fimo, en modellermasse som vanligvis brukes til å lage dukker. Denne er enkel å tilpasse til hånden og komponentene og herdes enkelt ved en halvtime i vanlig stekeovn. Modulen anvender 2 stk Futaba S3156 servomotorer for å plassere brukerens finger i ønsket posisjon og 4 stk interlink FSR 402 for å måle krefter fra brukerens finger. Proksimal modul med plassering av sensorer og aktuatorer er vist i figur 14.



Figur 14: Proksimal modul (ikke tilkoblet)

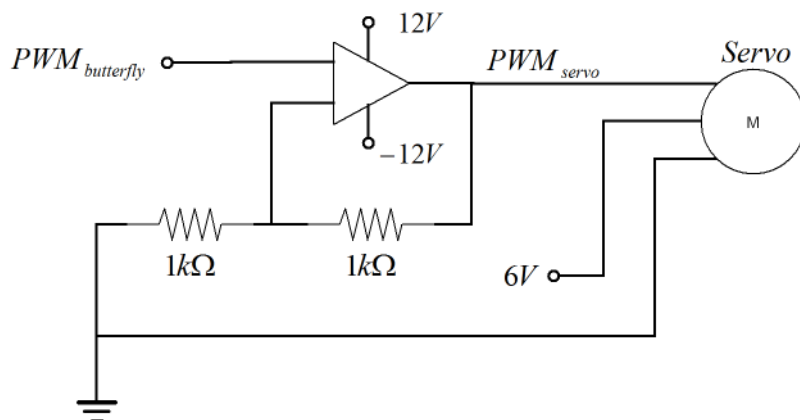
### 10.3 Tilkobling av servomotorer

Servomotorene krever et PWM styringssignal med en amplitude på  $6V$ , og en pulsbredde på  $0,9 - 2,1ms$ . Futaba-servoene godtar PWM-frekvenser opptil  $300Hz$ , men Hitec-servoene krever en frekvens på  $50Hz$  og dette blir derfor gjeldende for systemet. Butterfly'ene kan kun generere PWM med en amplitude opptil  $4,5V$  og styringssignalet til hver servo kobles derfor via en enkel forsterkerkrets (figur 16).



Figur 15: PWM-styringssignal før og etter forsterkning

På grunn av mekaniske begrensninger i protesemodellen og begrensninger



Figur 16: Tilkobling av servomotorer til Butterfly

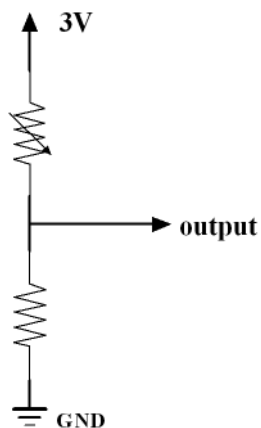
i bevegeligheten i brukerens finger føres det opp begrensninger for de tillatte posisjonene for hver servo, representert ved den tilsvarende pulsbredden, i tabell 4. For å lette utregningen av posisjon i programvaren er det også oppført tilsvarende målinger fra potensiometerne der disse finnes.

servo	funksjon	max PW	max pot	max stilling	min PW	min pot	min stilling
1	ekstensjon/ fleksjon av finger	1700 $\mu$ s	-	åpen hånd	900 $\mu$ s	-	lukket hånd
2	adduksjon/ abduksjon av finger	1200 $\mu$ s	-	venstre	1040 $\mu$ s	-	høyre
3	åpne/lukke protese	1900 $\mu$ s	3,5V	helt åpen	900 $\mu$ s	1,12V	helt lukket
4	håndledds vridning protese	2100 $\mu$ s	3,9V	max vridning mot klokka	900 $\mu$ s	1,10V	max vridning med klokka

Tabell 4: Tillatte posisjoner for servoer i modellen

#### 10.4 Tilkobling av kraftsensorer

Som kraftsensorer brukes kraftfølsomme motstander (FSR/force sensing resistors), som utviser varierende motstand etter hvor stor trykkraft de utsettes for. Disse kobles i en spenningsdeler (se figur 17) slik at spenning over R1 er relatert til FSR-motstanden ved  $V = V_{CC} \cdot \frac{R_1}{R_1 + FSR}$ .



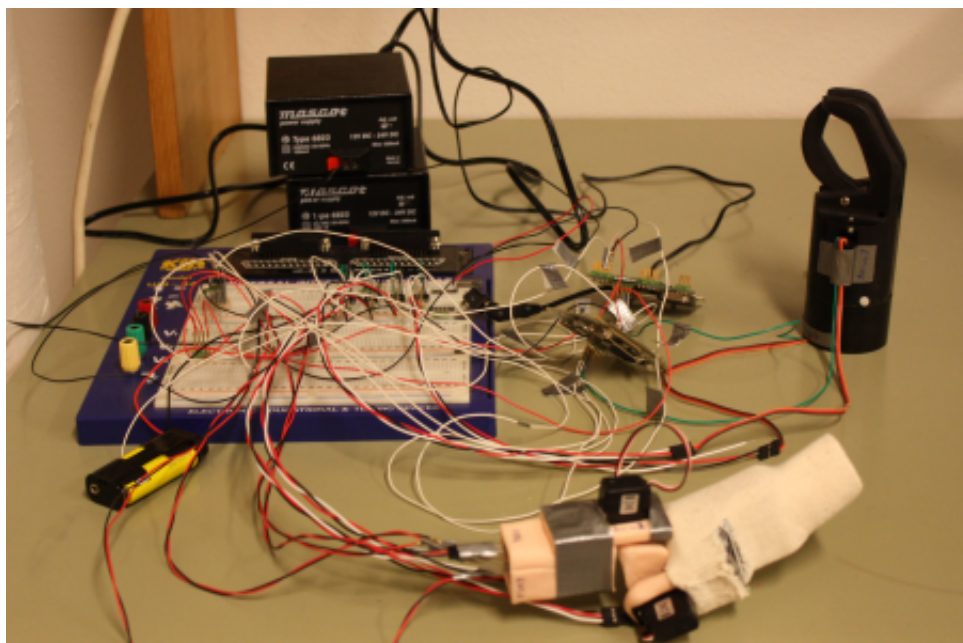
Figur 17: Tilkobling av FSR

## 10.5 Sammenkobling av komponentene

Hele systemet ble koblet sammen på et breadboard, med et utvalg av spenningsforsyninger. Utstyret brukt for hele systemet er som følger:

- 1 stk breadboard
- 2 stk AVR Butterfly
- 2 stk Futaba S3156 digitale servoer
- 2 stk Huida HD-2216 digitale servoer
- 4 stk interlink FSR 402
- 4 stk op-amper MC1741
- 8 stk  $1k\Omega$ -motstander
- 4 stk  $10k\Omega$ -motstander
- et utvalg ledninger
- 1 stk protesemodell, bestilt fra mekanisk verksted
- fimo-modellerleire, for utforming av proksimalmodul
- 2 stk Mascot powersupply type 6823
- 1 stk Vanson switching mode power supply
- 2 stk AAA batterier i batteriholder

Systemet ble koblet sammen på et breadboard etter spesifikasjonene angitt i teksten. Resultatet er vist i figur 18.



Figur 18: Ferdig oppkoblet system

## 11 Testing

Under driverutviklingen ble hver av komponentene testet hver for seg, og fungerte som forventet. Når det fullstendige systemet ble koblet sammen og kraftsensorene skulle leses av for å kalibrere terskelverdier, ble derimot avlesningene helt feil. Det ble raskt identifisert brudd i en ledning som feilkilde, og denne ble byttet ut, men problemet vedvarte. Det var desverre ikke tid til videre feilsøking.

## 12 Konklusjon av realiseringen

Protesemodellen ble levert bare en drøy uke før innlevering, og det ble derfor knapp tid til testing. En ukjent feil gjorde at det ikke var mulig å lese av korrekte verdier fra kraftsensorene, noe som førte til at systemet i sin helhet ikke virket. Protosemodellen manglet også kraftmåling, og bestilling av flere kraftsensorer var ikke mulig i tidsrommet mellom levering av modellen og innleveringsfristen av denne oppgaven. Dersom tiden hadde strukket til ville det vært nærliggende å gjøre en enkel modifikasjon av modellen; "Tommelen" ville blitt splittet på langs, og det ville blitt lagt imellom et materiale som fjærer litt (f.eks et tynt lag skumgummi) og en FSR. Siden hver enkelt komponent virker hver for seg antas det at med videre feilsøking vil systemet virke.



## Del IV

# Avslutning

Tidsmangel på slutten av oppgaven gjorde at det ikke var mulig å gjennomføre tilstrekkelig feilsøking, og det ble ikke oppnådd en funksjonell demonstrator. Likevel virket alle komponentene hver for seg, og det er såpass mange eksempler på lignende systemer som virker at det er nærliggende å tro det ikke gjenstår mye arbeid før forsøket vil være vellykket.

I videre arbeid anbefales det å finne alternativer til strømforsyning, da denne modellen var koblet til hele 4 forskjellige spenningsforsyninger, hvor av 3 var koblet i stikkontakter. Hvis systemet skal kunne videreutvikles til en funksjonell protese må det nødvendigvis utarbeides en holdbar batteriløsning.

Proksimalmodulen er unødvendig stor og klønete, og ved videreutvikling til en brukbar protese bør denne lages i et mer egnet materiale. Protesemodellen er også preget av å være en prototyp, og har forbedringspotensiale i forbindelse med utforming. Ved videreutvikling av systemet for bruk av pasienten anbefales å kontakte et ortopedisk verksted, da de muligens vil kunne være behjelpelig med støping av hylse for proksimalmodul og utforming av selve protesen. I tillegg er sensor- og aktuator-teknologien i stadig utvikling og det kan med fordel gjøres en ny vurdering av valg av komponenter før neste versjon av systemet igangsettes.

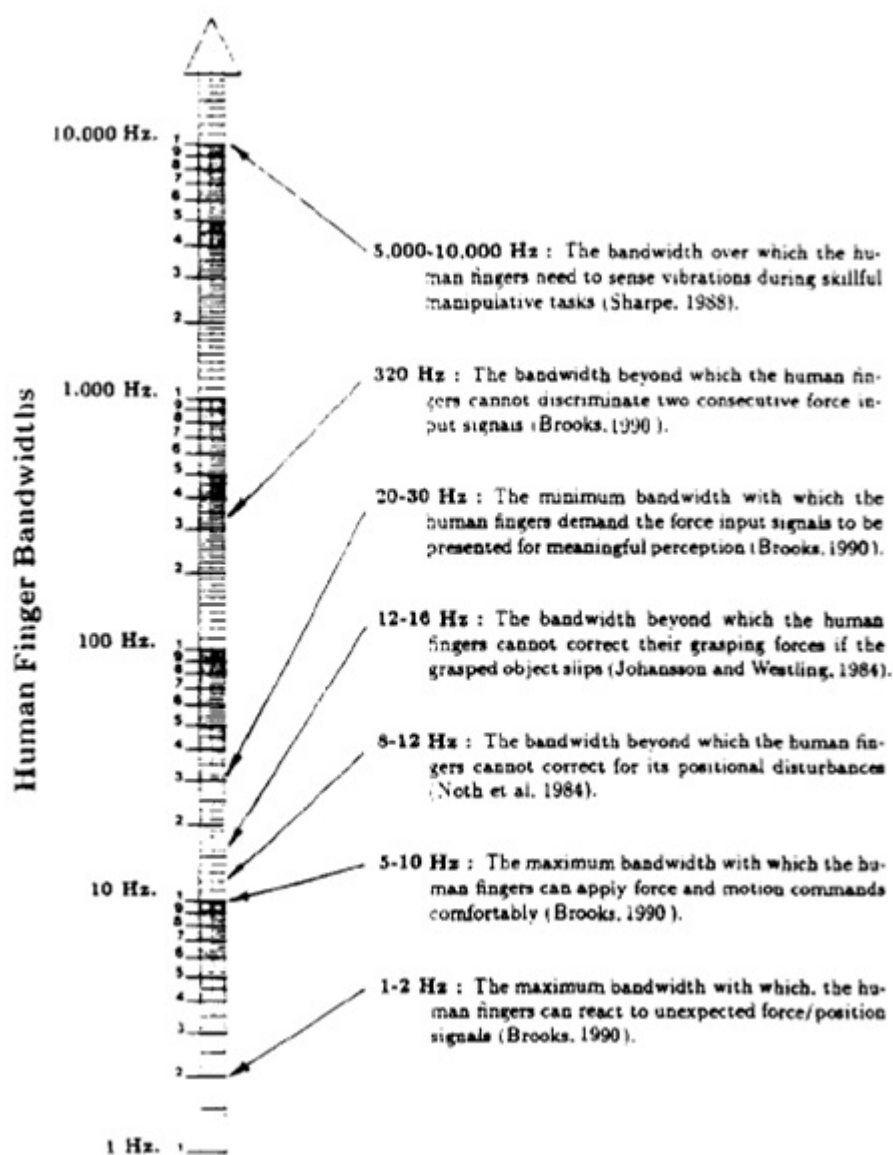
Det anbefales også å arbeide med klarere kartlegging av pasienten fysiologi og funksjonalitet i armen. Hvis det viser seg at musklene i overarmen kan beveges uavhengig av fingeren er det mulig å legge til en myoelektrisk sensor på overarmen og man oppnår enda en frihetsgrad i systemet, dog uten EPP. Med tre frihetsgrader kan det være aktuelt å legge til for eksempel albuebevegelse i protesen. I tillegg kan man vurdere muligheten for å legge til taktil tilbakekobling med gjengivelse av tekstur, temperatur og liknende.

Systemets oppdeling i moduler gjør at fremtidig arbeid også kan involvere å legge til flere distale moduler. Noen eksempler kan være tilpassede proteser til forskjellige formål, radiostyrte leker, eller pc-spill med proksimal modul som input for å stimulere og trene opp barn (og voksne) til å bruke styringssystemet.



## Del V Vedlegg

### A Oversikt over båndbreddekrav for menneskets fingre



Figur 19: Oversikt over båndbreddekrav for menneskefingre, hentet fra [16]

## B Programkode

### B.1 main.c

```
#include "PC_Comm.h"
#include "ADC.h"
#include "BCD.h"
#include "FSR.h"
#include "Pot.h"
#include "PWM.h"
#include "main.h"

int butterfly = 2;

char servoNr_c;
int servoNr;
float ftLo[4] = {999,999,999,999};
float ftHi[4] = {999,999,999,999};

void init()
{
// initier drivere:
init_PC_Comm();
init_FSR();
sendString("FSR initialisert\r");
init_Pot();
sendString("Potmeter initialisert\r");
init_PWM();
sendString("PWM/Servo initialisert\r");

//sett pwm-frekvens 50 Hz
set_freq(50);

//fortell om tilgjengelige kommandoer
sendString("\r");
sendString("EPP starter epp\r");
sendString("freqXXX setter frekvens\r");
sendString("voltX leser spenning fra ADCX\r");
sendString("svXctcXXX setter servoposisjon for servo X\r");
sendString("s stopper PWM\r");
}

int main(void)
{
init();
```

```

//evig løkke
for(;;)
{
    //PC_Comm();
}

EPP();

return 0;
}

//her kommer selve EPP-funksjonen!
void EPP()
{

for(;;){//uendelig løkke
if (butterfly==1){
float FSR1 = read_FSR(1);
float FSR2 = read_FSR(2);

if (FSR1 > ftHi[0]){
//kraft oppover fra servo 3 skal matche fsr1
}else if (FSR1 < ftHi[0] && FSR1 > ftLo[0]){
//inkremitter servo 3
increment_position(3);
}else{
//gjør ikkeno
}

if (FSR2 > ftHi[1]){
//kraft nedover fra servo 3 skal matche fsr1
}else if (FSR1 < ftHi[1] && FSR1 > ftLo[1]){
//dekremitter servo 3
decrement_position(4);
}else{
//gjør ikkeno
}

float pot1 = read_Pot(1);
set_position(pot1,1); //sett posisjon på servo1 = pot1, altså samme prosent

} else if (butterfly == 2){
float FSR3 = read_FSR(3);
float FSR4 = read_FSR(4);

```

```

if (FSR3 > ftHi[2]){
//kraft oppover fra servo 4 skal matche fsr3
}else if (FSR3 < ftHi[2] && FSR3 > ftLo[2]){
//dekrements servo 4
decrement_position(3);
}else{
//gjør ikkeno
}

if (FSR4 > ftHi[3]){
//kraft oppover fra servo 4 skal matche fsr4
}else if (FSR4 < ftHi[3] && FSR4 > ftLo[3]){
//inkrements servo 4
increment_position(4);
}else{
//gjør ikkeno
}

float pot2 = read_Pot(2);
set_position(pot2,2); //sett posisjon på servo1 = pot1 i prosent
}

PC_Comm(); //sjekk for kommando fra PC (vil i de fleste tilfeller ikke være tilkoblet)
} //slutt på uendelig løkke

}

//definer hva som skjer når jeg skriver inn noe på pc'en (via PC_Comm.c)
//basert på programvare fra "c programming for microcontrollers" (Joe Pardue, 2005)
void parseInput(char s[])
{
// parse first character
switch (s[0])
{

case 'e':
if( (s[1] == 'p') && (s[2] == 'p') )
EPP();
break;
case 'f':
if( (s[1] == 'r') && (s[2] == 'e') && (s[3] == 'q') )
parse_freq(s);
}
}

```

```

break;
case 'v':
if( (s[1] == 'o') && (s[2] == '1')&& (s[3] == 't'))
read_FSR(s[4]);
break;
case 's':
if( (s[1] == 'v') ){
servoNr = s[2]-48; //ascii 48 = int 0
if( (s[3] == 'c') && (s[4] == 't') && (s[5] == 'c')){
parse_ctc(s, servoNr);
}
}
else{
sendString("stopper PWM.\r");
stop_PWM();
}
break;

default:
sendString("\rYou sent: ");
sendChar(s[0]);
sendString("' - I don't understand.\r");
break;

}
s[0] = '\0';
}

//hentet fra "c programming for microcontrollers" (Joe Pardue, 2005):
//Calibrate the internal OSCCAL byte, using the external
//32,768 kHz crystal as reference
void OSCCAL_calibration(void)
{
    unsigned char calibrate = 0;//FALSE;
    int temp;
    unsigned char tempL;

    CLKPR = (1<<CLKPCE);          // set Clock Prescaler Change Enable

    // set prescaler = 8, Inter RC 8Mhz / 8 = 1Mhz
    CLKPR = (1<<CLKPS1) | (1<<CLKPS0); //deler 8MHz-klokken på 8, og gir 1MHz.

    TIMSK2 = 0;                  //disable OCIE2A and TOIE2

```

```

ASSR = (1<<AS2);          //select asynchronous operation of timer2 (32,768kHz)

OCR2A = 200;              // set timer2 compare value

TIMSK0 = 0;              // delete any interrupt sources

TCCR1B = (1<<CS10);       // start timer1 with no prescaling
TCCR2A = (1<<CS20);       // start timer2 with no prescaling

while((ASSR & 0x01) | (ASSR & 0x04));          //wait for TCN2UB and TCR2UB to be cleared

// wait for external crystal to stabilise
for(int i = 0; i < 10; i++)
    _delay_loop_2(30000);

while(!calibrate)
{
    cli(); // mt __disable_interrupt(); // disable global interrupt

    TIFR1 = 0xFF; // delete TIFR1 flags
    TIFR2 = 0xFF; // delete TIFR2 flags

    TCNT1H = 0; // clear timer1 counter
    TCNT1L = 0;
    TCNT2 = 0; // clear timer2 counter

    while ( !(TIFR2 && (1<<OCF2A)) ); // wait for timer2 compareflag

    TCCR1B = 0; // stop timer1

    sei(); // __enable_interrupt(); // enable global interrupt

    if ( (TIFR1 && (1<<TOV1)) )
    {
        temp = 0xFFFF; // if timer1 overflows, set the temp to 0xFFFF
    }
    else
    { // read out the timer1 counter value
        tempL = TCNT1L;
        temp = TCNT1H;
        temp = (temp << 8);
        temp += tempL;
    }

    if (temp > 6250)
    {

```



```
        OSCCAL--; // the internRC oscillator runs to fast, decrease the OSCCAL
    }
    else if (temp < 6120)
    {
        OSCCAL++; // the internRC oscillator runs to slow, increase the OSCCAL
    }
    else
        calibrate = 1;//TRUE; // the interRC is correct

    TCCR1B = (1<<CS10); // start timer1
}
}
```

## Del VI

# Kilder

### Referanser

- [1] *ATMEL 8-bit AVR Microcontroller with 16K Bytes In-System Programmable Flash ATmega169V ATmega 169*. datasheet, [www.atmel.com](http://www.atmel.com).
- [2] Norsk teknisk ortopedis hjemmesider. [www.ortonor.no](http://www.ortonor.no), 2009.
- [3] Haitham M. Al-Angari, Richard F. Weir, Craig W. Heckathorne, and Dudley S. Childress. A two degree-of-freedom microprocessor based extended physiological proprioception (epp) controller for upper limb prostheses. *Technology and Disability*, 15(2):113–127, 2003.
- [4] Haitham M. Al-Angari, Richard F. Weir, Craig W. Heckathorne, and Dudley S. Childress. A two degree-of-freedom microprocessor based extended physiological proprioception (epp) controller for upper limb prostheses. *Technology and Disability*, 15(2):113–127, 2003.
- [5] Stephen P. Buerger and Neville Hogan. Complementary stability and loop shaping for improved human-robot interaction. *IEEE transactions on robotics*, 23(2):232–244, april 2007.
- [6] David T. Gibbons, Micheal D. O’Riain, and Sebastien Philippe-Auguste. An above-elbow prosthesis employing programmed linkages. *IEEE Transactions on Biomedical Engineering*, BME-34(7):493–498, 1987.
- [7] Paul G. Griffiths and R. Brent Gillespie. A fundamental conflict between performance and passivity in haptic rendering. In *2008 American Control Conference*, pages 183–188, Seattle, Washington, USA, june 2008.
- [8] K.S. Hale and K.M. Stanney. Deriving haptic design guidelines from human physiological, psychophysical, and neurological foundations. *IEEE Computer Graphics and Applications*, 24(2):33–39, 2004.
- [9] Grete Hjermstad. Kompetansesenter for dysmeli og armamputasjon. Technical report, Rikshospitalets Senter for Ortopedi, 1996.
- [10] Interlink Electronics. *FSR Force Sensing Resistor Integration Guide and Evaluation Parts Catalog*. [www.interlinkelectronics.com](http://www.interlinkelectronics.com).
- [11] Jorge A. Mendez-Inglesias and Fransisco J. Ruiz-Sanchez. Human stability analysis on time-delayed teleoperation tasks. In *2007 IEEE International Symposium on Industrial Electronics*, pages 2161–2196, Vigo, Spain, june 2007.
- [12] Joe Pardue. *C Programming for Microcontrollers*. Smiley Micros, 2005.

- [13] Joe Pardue. *Butterfly Alternate Pin Uses*, 2006. [www.smileymicros.com](http://www.smileymicros.com).
- [14] James A. Doubler Ph.D and Ph.D Dudley S. Childress. An analysis of extended physiological proprioception as a prosthesis-control technique. *Journal of Rehabilitation Research and Development*, 21(1):5–18, 1984.
- [15] James A. Doubler Ph.D and Ph.D Dudley S. Childress. Design and evaluation of a prosthesis control system based on the concept of extended physiological proprioception. *Journal of Rehabilitation Research and Development*, 21(1):19–31, 1984.
- [16] Karun B. Shimoga. A survey of perceptual feedback issues in dexterous telemanipulation: Part i. finger force feedback. In *1993 IEEE Virtual Reality Annual International Symposium*, pages 263–270, Seattle, WA, USA, september 1993.
- [17] Bruno Siciliano and Oussama Khatib, editors. *Springer handbook of robotics*, chapter 31. telerobotics. Springer, 2008.
- [18] Ranveig Solberg. Demonstrator for epp-basert protesestyring. prosjektoppgave i teknisk kybernetikk, NTNU, desember 2009.
- [19] MS Todd R. Farrell, PhD Richard F. Weir, MS Craig W. Heckathorne, and PhD Dudley S. Childress. The effects of static friction and backlash on extended physiological proprioception control of a powered prosthesis. *Journal of Rehabilitation Research and Development*, 42(3):327–342, 2005.
- [20] MS Todd R. Farrell, PhD Richard F. Weir, MS Craig W. Heckathorne, and PhD Dudley S. Childress. The effects of static friction and backlash on extended physiological proprioception control of a powered prosthesis. *Journal of Rehabilitation Research and Development*, 42(3):327–342, 2005.
- [21] Lawton N. Verner and Allison M. Okamura. Sensor/actuator asymmetries in telemanipulators: Implications of partial force feedback. In *2006 14th Symposium on Haptics Interfaces for Virtual Environment and Teleoperator Systems*, pages 2161–2196, Alexandria, Virginia, USA, march 2006.

## Takk til:

- Øyvind Stavadahl, for veiledning gjennom både prosjekt- og masteroppgave
- Trond Andresen og Maria Ottermo, for hjelp med å finne kilder
- Mekanisk verksted ved institutt for teknisk kybernetikk, for bygging av protesemodell
- Thomas Rygh, gutta på Omega Verksted og komponentlageret ved ITK, og forumdeltakerne på AVRfreaks.com for tips og hjelp underveis